This paper is available online at www.praiseworthyprize.org
International Review of Aerospace Engineering (I.RE.AS.E), Vol. 12, N. 1
ISSN 1973-7459 February 2019

NtoM: a Concept of Operations for Pilots of Multiple Remotely Piloted Aircraft

Miguel-Ángel Fas-Millán, Enric Pastor

Abstract – *The concept of operations proposed here pursues the feasibility, from a human factors perspective, of having a single pilot/aircrew controlling several remotely piloted aircraft systems at once in non-segregated airspace. To meet such feasibility, this multitasking must be safe and not interfere with the job of the air traffic controllers due to delays or errors associated with parallel piloting. To that end, a set of measures at several levels is suggested, which includes workload prediction and balance, pilot activity monitoring, and a special emphasis on interface usability and the pilot's situational awareness. The concept relies greatly on the exploitation of the potential of Controller-Pilot Data Link Communications, anticipating future widespread implementation and full use. Experiments comparing the performance of the same pseudo-pilots before and after the implementation of part of the measures showed a decrease in the number of errors, oversights and subjective stress.* ***Copyright © 2019 The Authors.***

Keywords*: RPAS, UAS, CPDLC, Multi-RPAS, Multi-UAS*

I. Introduction

One of the principles behind the guidelines provided by the International Civil Aviation Organization (ICAO) is to guarantee equality in access to airspace. That was why in 2011 the agency set out the basis for the development and integration of Remotely Piloted Aircraft Systems (RPAS). As a response to that recommendation, in 2012, the European Commission created the European RPAS Steering Group of stakeholders, who developed and published the European RPAS Roadmap [1], a work plan defining the steps, responsibilities and milestones for the gradual insertion of RPAS in the Single European Sky (SES) [2]. Saying that this integration implies a completely new industry with a wide service sector and supply chain is not a prediction anymore, but a reality that is growing every day. However, even though the potential applications and promising market of the unmanned aerial systems (UAS) are well known, any facility for their commercial exploitation will mean an incentive to those investors needed to boost this emerging industry. That is one of the goals of this concept of operations (ConOps): to maximise the productivity of a fleet of RPAS and their remote pilots, allowing a pilot or aircrew to control several aircraft without losing sight of safety or the impact that this multitasking could cause on the work of the air traffic controllers (ATC). This impact could come from delays in the response time, errors or oversights.

These problems, prone to arise when piloting several aircraft in parallel, could also be found in a one-to-one pilot-RPAS relationship. So, the whole concept and its safety measures could also be leveraged for the one-to-one case. The supervision or control of multiple unmanned aerial vehicles (UAVs) is a growing topic of research still in its early days, but several approaches can already be found. Some of them are more appropriate for specific kinds of missions, others require almost full autonomy of the aircraft, and most of them were conceived for segregated airspace. Different paradigms can be found; the ones where the pilot is expected to pilot the aircraft or the ones where the operator is basically in charge of defining and guiding the mission [3]. In order to improve productivity and efficacy, much research, mainly addressed to military operations, focuses on improving the design of the Ground Control Station (GCS). Some of these improvements are based on a careful balance of cognitive resources and mental workload. Others try to enhance the human scope by teaming pilot and adaptive automation [4]. The search for productivity has also sometimes been based on task scheduling [5], [6]. NtoM combines some of both worlds: it tries to avoid unmanageable levels of workload by scheduling and task support, but also provides measures to back up the pilot if it appears. The paper begins with a list of assumptions regarding the terms used in the document and the conditions or scenarios considered for a possible implementation of the ConOps.

It also specifies which kind of operations would better leverage the NtoM concept. Next, there is a description of the architecture of the system, its inputs, parts and their relationships. Later in that section, its different features are explained grouped by type: usability, workload or monitoring. Then, there is a description of

Available online by February 28th, 2019
https://doi.org/10.15866/irease.v12i1.16153

the experiments conducted to check the performance of the pseudo-pilots with or without some of the features implemented and the results obtained. Finally, the conclusions about the potential benefits of the ConOps are given.

II. Assumptions

Throughout the rest of the paper, for simplicity, the Air Vehicle Operator (AVO) will be referred to as *the pilot*, but assuming that an aircrew could be behind the term. In principle, for the ConOps, there is no limit in the role architecture of the aircrew, as long as we could define the workload, interactions, and incompatibilities among their tasks. For instance, we could have a payload operator for several aircraft, while these are piloted by different AVOs. In a case like that, the system should handle the availability of the payload operator for the different flights besides the availability of those flights' AVOs if these are controlling more than one aircraft. The system could also allow a crew assignment on demand like the one suggested in NASA's SPO ConOps [7] for the hybrid ground dispatcher-first officer role in manned single-pilot operations. For clarity, this document will focus on how the concept manages the assignment of flights to the AVOs operating as a single pilot, but a greater degree of flexibility could be allowed in the quantity and type of roles involved. Current unsolved issues regarding the acceptance of the integration of RPAS by the general public, or the legal requirements of the States, like policies or the definition of responsibilities, are left out. In the worst case, the concept could still be useful in segregated airspace. This work aims to be a brick in the building of the RPAS integration but considering that the foundations of this integration are just a matter of time. Another question of time is the widespread use of Controller-Pilot Data Link (CPDLC) [8, Part IV]. Being NtoM a concept that would require a long-term implementation, it is assumed that CPDLC will already be the main air-ground means of communication. Then, some of the measures exploit its currently underused potential. It should be noted that the use of the term RPAS means that w a human pilot in command is considered. Anyway, the target aircraft will be so highly automated to allow the pilot to delegate a large part of the tasks, which is key to be able to control several aircraft concurrently. An example of this appears in [9], where a military pilot is expected to control several unmanned aircraft while piloting a manned aircraft. The GCS should allow the control of different aircraft models with heterogeneous levels of autonomy and equipment. [10] addresses the requirements and design for this kind of control station, a sample of which was part of the ambitious DARIUS project, with its Generic Ground Station able to control maritime, aerial or ground unmanned systems used in search and rescue operations [11]. In fact, the NtoM concept could be applied to any type of unmanned vehicle or to process supervision, as the basis of the system management relies on workload, task interaction constraints and procedure definitions, which are agnostic to the kind of activity performed. The kind of mission that would best profit from this ConOps would be one with a predetermined trajectory. Surveillance missions in which the path is decided on the go would not fit too well in the concept.

While the system can handle the consequences of a change of trajectory, like a direct to clearance, the recalculation of the workload forecast could force a rescheduling and negatively affect other pilots, especially if done often. The technological and computing infrastructure orchestrating the ConOps will be referred to, in short, as the system, or the server, without assuming any explicit implementation behind these terms, since its services could be distributed. The descriptions of the features of the concept appear in three groups, depending on the kind of measure intended to achieve the feasibility of the paradigm: workload, monitoring and usability. However, these aspects are dependent upon one another; usability affects workload as much as the support of the monitoring benefits the usability and the workload. Then, such categorisation is not exclusive and a feature can have some dependence; for better or worse, a change in one of them could affect the efficacy of another.

III. System Description

Fig. 1 shows the architecture of the NtoM server prototype. It takes as input flight definitions, which, besides the usual data in a flight plan, can contain preferences and constraints, further explained in the section about workload management. This input, besides the workload forecast provided by the Workload Module and the current status of the flight retrieved from the GCS Module, is used by the Scheduler to suggest or apply the flight-pilot assignments.

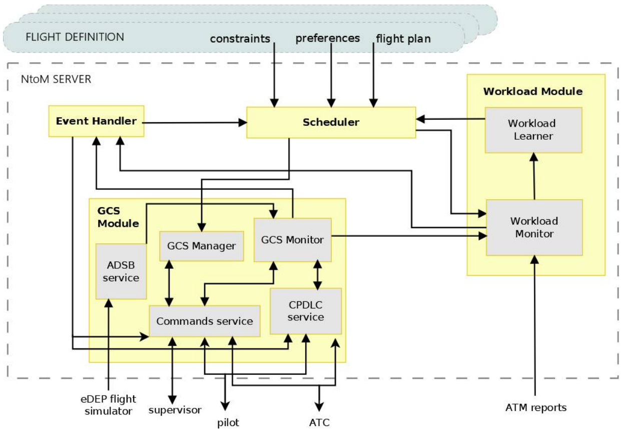


Fig. 1. NtoM server prototype

The Event Handler is in charge of attending requests to manage unexpected events. Acting somewhat like the so-called decision authority in adaptive automation, it will trigger notifications, requests for feedback or automated actions. The Workload Monitor or the GCS

Monitor sends those requests for management. Finally, the GCS Manager holds the flight-pilot relationships and the status of the aircraft, and it shares the module with the services attending to the communications and commands.

III.1. Usability

Although automation is essential to allow a pilot to control several aircraft, the usability of the GCS is key to allowing a quick response and appropriate awareness. A clear example is the work that led to the Noodle Mode [12] to draw the intent of each aircraft. While oriented to a different scenario, surveillance missions on segregated airspace, the lesson learned is that the type of information, the way of displaying it, and the kind of controls provided are basic pillars to safety and productivity. NtoM intends to be GCS-agnostic; we could have any GCS plugged into the system via the NtoM pilot interface (NPI), provided that it offered an application programming interface (API) for the NPI to access the flight instruments readings, send commands or monitor those executed by the pilot. Then, the design decisions described here are limited to the NPI, which is displayed on a dedicated monitor, even when it also follows guidelines addressed to the usability of the remote pilot stations [13]. While a final implementation of the NPI would require a process of refinement and survey pilots to determine which information and notifications they prefer, we can expect the usability of the GCS and the aircraft automation to be the main determinants of the level of workload associated to each task, and therefore the number of flights per pilot. The NPI (Fig. 2) shows a list of the assigned flights, each one given as a strip with the main readings and their associated timestamp. The relative position of the aircraft in the current segment can be seen at a glance, under the remaining distance and time to the next waypoint. The next segment shows the minimum link quality on it if the current link is kept. At the right end of the strip, a button allows the flight definition, previous pilots' sequence, and notes to be read. Reminders and warning icons will appear next to the button. The interface also contains the CPDLC display, and there is room left for notifications, confirmation dialogues, and a chatroom to communicate with the supervisor □ a role that will be described later □ or other pilots operating in the NtoM environment (the use of instant messages for coordination is already common in military operations [14]). Different aural indicators, depending on the urgency and type, sometimes accompany the printing of reminders or warnings.

When the message needs to be more attention-grabbing or descriptive, speech synthesis is used. The list of flights allows the pilot to choose which one to load in the GCS. As this loading could be relatively slow, the number of times it is required tries to be minimised by the summary of readings, the automation of tasks, and making it possible to send commands and communications to flights in the background (any flight not currently loaded in the GCS or selected in the NPI).

The GCS, which should be connected to the NPI, was simulated with an embedded set of commands, which was enough considering the kind of experiments and the fact that the participants were not professional pilots. At this point, an important question that might arise is who is keeping the connection to the rest of the RPAs alive while just one of them is loaded in the GCS. To NtoM this would be just a matter of implementation, and logistic or performance availability. Fig. 3 shows two possible configurations. One (Fig. 3, top), shows a GCS able to support multiple Command, Control and Communications (C3) links, as either an embedded capability in the GCS, or provided by an external switch.

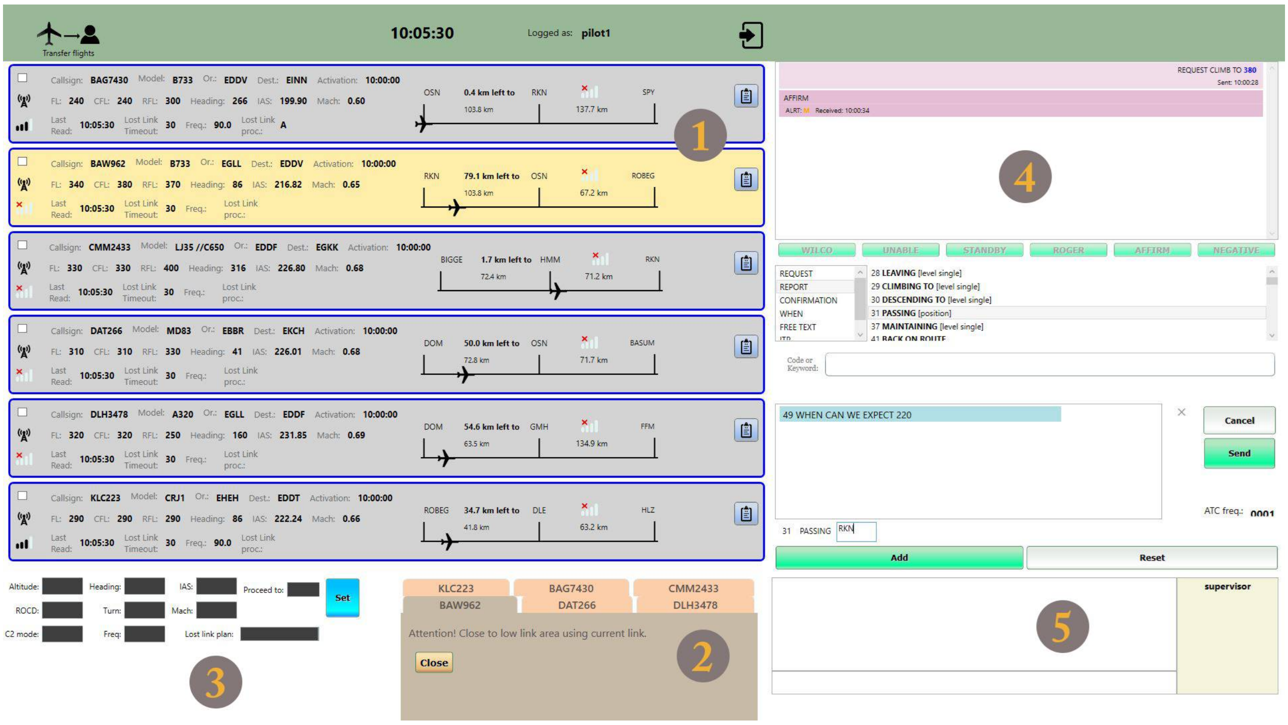
An example of this kind of GCS is General Atomics' Heresy multi-mission control, a software that a single pilot can use to control up to six large UASs like the MQ-9 from a common laptop including the possibility to transfer the control to other pilots in the network [15].

Depending on the option, the access of the NPI to the C3 would be via the GCS API, or directly connecting to the switch. The second option (Fig. 3, bottom) shows a GCS acting as a mere terminal, without any kind of connectivity with the aircraft, which is fully managed by the server. This seems an unlikely configuration for a real implementation, as it would add the network delay to the one of the C3 link, but it is the one representing the current NtoM prototype. In the prototype, the server reads the aircraft has simulated ADS-B readings from the socket of a single instance of RAISE. RAISE is a simulation environment mixing eDEP simulator [16] traffic and RPAS traffic, developed by ICARUS research group to evaluate the impact of the RPAS integration in the Air Traffic Management (ATM) [17]. The server forwards the readings to the pilot's client application, and vice versa to apply the commands of the pilot to the aircraft. The benefit of this configuration is in the use of the RTI Connext connectivity framework [18], compliant with the Data Distribution Service (DDS) standard, to communicate server and clients.

This allows a simulation of different scenarios of Quality of Service (QoS), providing a framework to test contingencies related to link quality to "*determine what compensatory behaviours, if any, air traffic controllers and UAV operators adopt in response to communications delays*" [19] or provide training in the associated procedures.

Following the usability, and contributing to reducing the total workload, there are task assistance features like the possibility of execution and reply of received instructions with a single click, enhanced CPDLC message composition (including voice composition) or automated reports with or without conditional events.

The CPDLC display was designed taking into consideration the possibility of piloting several aircraft, so it tries to reduce the head down to the minimum [8, Part I, Appendix to Ch. 2, 6-f]. Further details can be found in [20] and screencasts in [21].



Transfer flights 10:05:30 Logged as: pilot1

Callsign: BAG7430 Model: B733 Or.: EDDV Dest.: EINN Activation: 10:00:00
FL: 240 CFL: 240 RFL: 300 Heading: 266 IAS: 199.90 Mach: 0.60
Last Read: 10:05:30 Lost Link Timeout: 30 Freq.: 90.0 Lost Link proc.: A
OSN 0.4 km left to RKN SPY 103.8 km 137.7 km

Callsign: BAW962 Model: B733 Or.: EGLL Dest.: EDDV Activation: 10:00:00
FL: 340 CFL: 380 RFL: 370 Heading: 86 IAS: 216.82 Mach: 0.65
Last Read: 10:05:30 Lost Link Timeout: 30 Freq.: Lost Link proc.:
RKN 79.1 km left to OSN ROBEG 103.8 km 67.2 km

Callsign: CMM2433 Model: LJ35 //C650 Or.: EDDF Dest.: EGKK Activation: 10:00:00
FL: 330 CFL: 330 RFL: 400 Heading: 316 IAS: 226.80 Mach: 0.68
Last Read: 10:05:30 Lost Link Timeout: 30 Freq.: Lost Link proc.:
BIGGE 1.7 km left to HMM RKN 72.4 km 71.2 km

Callsign: DAT266 Model: MD83 Or.: EBBR Dest.: EKCH Activation: 10:00:00
FL: 310 CFL: 310 RFL: 330 Heading: 41 IAS: 226.01 Mach: 0.68
Last Read: 10:05:30 Lost Link Timeout: 30 Freq.: Lost Link proc.:
DOM 50.0 km left to OSN BASUM 72.8 km 71.7 km

Callsign: DLH3478 Model: A320 Or.: EGLL Dest.: EDDF Activation: 10:00:00
FL: 320 CFL: 320 RFL: 250 Heading: 160 IAS: 231.85 Mach: 0.69
Last Read: 10:05:30 Lost Link Timeout: 30 Freq.: Lost Link proc.:
DOM 54.6 km left to GMH FFM 63.5 km 134.9 km

Callsign: KLC223 Model: CRJ1 Or.: EHEH Dest.: EDDT Activation: 10:00:00
FL: 290 CFL: 290 RFL: 290 Heading: 86 IAS: 222.24 Mach: 0.66
Last Read: 10:05:30 Lost Link Timeout: 30 Freq.: 90.0 Lost Link proc.:
ROBEG 34.7 km left to DLE HLZ 41.8 km 63.2 km

Altitude: Heading: IAS: Proceed to: Set
ROCD: Turn: Mach:
C2 mode: Freq: Lost link plan:

KLC223 BAG7430 CMM2433
BAW962 DAT266 DLH3478

Attention! Close to low link area using current link.

Close

REQUEST CLIMB TO 380
Sent: 10:00:28

AFFIRM
ALRT: M Received: 10:00:34

WILCO UNABLE STANDBY ROGER AFFIRM NEGATIVE

REQUEST
REPORT
CONFIRMATION
WHEN
FREE TEXT
ITP

28 LEAVING [level single]
29 CLIMBING TO [level single]
30 DESCENDING TO [level single]
31 PASSING [position]
37 MAINTAINING [level single]
41 BACK ON ROUTE

Code or Keyword:

49 WHEN CAN WE EXPECT 220

Cancel
Send

ATC freq.: 0001

31 PASSING RKN

Add Reset

supervisor

Fig. 2. The NtoM Pilot Interface. 1) Assigned active flights. Each strip contains the main readings, reminders and access to the flight information. 2) Stack of notifications and dialogues used to request feedback or provide more informative warnings. 3) Embedded aircraft control commands. 4) CPDLC display. 5) NtoM network Chatroom

A delicate moment during the control of the RPAS is also addressed: the handover process. It has been a source of errors with dramatic consequences when the procedure required manual configuration or the GCS would not clearly show the status of the process [22].

Nevertheless, in NtoM, orchestrated by the server, it avoids human errors and guarantees an orderly procedure. A pilot that is selected to be assigned with a flight will see a confirmation dialogue to accept or reject it. If accepted, a new strip for the received flight appears, framed in a different colour to highlight that it still cannot be controlled. However, the readings are updated, and the pilot can access the flight plan and the history of CPDLC to form a view of the situation. While not implemented in the prototype, the interface could allow pilots to introduce annotations available to the following pilots. At the moment, it shows the list of previous pilots; that way the receiving pilot can check who is transferring the flight and use the chatroom to coordinate the transfer.

The receiver pilot, once prepared to control the flight, informs the transferring pilot, who applies the transfer.

The receiver can then control the new flight and the previous pilot sees the flight strip disappear from the NPI. The screencast in [23] illustrates the process. If considered useful by controllers, a free text message, sent automatically, could alert them about the process of transfer, as it could delay the communications or the execution of instructions. We could even consider alerting nearby traffic using an ADS-B broadcast.

Transfers could be scheduled, the result of a contingency measure, or requested by the pilot or the supervisor.

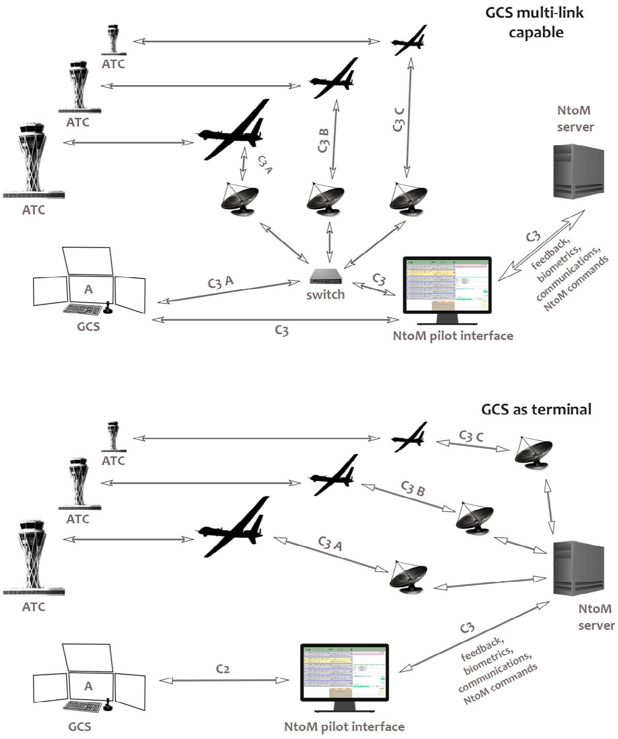


Fig. 3. Connectivity configurations

[24] describes an interface concept with common points with the NPI. The concept follows Endsley's [25] guidelines for any smart interface. A final NPI implementation should follow them, and the prototype already does this to some extent. First, it should provide a big picture of the situation, with high-level information,

which could be the definition of the flight strips. Second, the pilot should be able to decide which information is displayed. In this respect, in NtoM the flight loaded in the GCS would not necessarily be the one active in the NPI; you could have one flight loaded and reply to the ATC of a different flight, or apply the set of commands associated to the instruction received in a background flight. Flight strips can be rearranged, but in the original mockup, it was planned to place non-exclusive order preferences, like *closer to next waypoint first*, *pending actions first* or *urgent actions first*. This would be optional, as some pilots could prefer to find always the same flight in the same place. Third, Endsley stresses the importance of providing information about past events that could help pilots to form an idea of the current status. In NtoM, a pilot receiving a flight would have access to the CPDLC history, pending events, list of previous pilots, their annotations, or automated event logging (in the NPI all this information would appear, hiding the CPDLC display by using the button with the clip folder icon in the strip). Lastly, there is the importance of providing a personalised experience, to consider each user’s preferences and limitations. The preferences could be satisfied by providing some flexibility about how and which reminders and warnings are triggered but with some limitations, like critical warnings that could not be turned off. Regarding fitting pilot limitations, this conveniently leads us to the next section, which details how it would be pursued.

III.2. Workload

The platform needs to allow the pilot to comfortably operate more than one RPAS, which is an obvious safety measure and could determine its implementation permit; e.g. we find regulations that address the maximum workload on individual crew members to determine the airworthiness of the airplane in [26, FAR Title 14, Part 25.1523 Minimum Flight Crew]. This objective is possible because many current RPAs are highly automated and not all flight phases are equally demanding, sometimes reduced to long and monotonous supervision [27], which leads to disengagement and loss of awareness. In this respect, the multi-piloting induces a self-motivating attitude of awareness (Yerkes-Dodson law), as could be checked during the first set of experiments when participants’ performance and response time to ATC were better when they had four flights compared to having just one [28]. But the system must guarantee a viable load of tasks per pilot at all times. To that end, it needs to predict, evaluate, limit and distribute the workload. [28] described the requirements for the workload description of the tasks, exemplified by the process of measurement carried out during the first set of experiments. This workload, considered roughly as the level of attention and time required to perform the task, depends not only on the current task performed, but also on the concurrent ones, and is represented in a kind of pilot’s workload profile that would evolve over time as the pilot gains experience, a fact shown in [29] (the aforementioned pilot limitations). The workload measured while executing a task, or an overlap of tasks, is an input to the system. Such a measure would be GCS-specific, or procedural-specific, although, as discussed in [30], which dissects the factors contributing to the workload of the remote pilots, a trustworthy representation of workload would take into account all the environmental drivers, not only those related to the GCS or the specific procedures. NtoM accepts any kind of task workload measurement. It could be obtained using biometrics, performance evaluation, subjective assessments, or a mix of these techniques. In [31], the task workload and the total workload of task overlappings is estimated with analytically generated workload profiles (i.e. models prepared by an expert based on the experience of the pilots). The output of that work is a clear example of one of the main inputs for NtoM to be able to manage and predict the multi-UAS workload. During the first NtoM experiments, the measurement was done based on performance evaluation and subjective assessment, but it would be desirable to use more reliable descriptors, like biometrics able to reflect attention or stress. That will be the next step of development. Nevertheless, whatever the indicator(s) used, the representation should be consistent throughout the system; the Scheduler, the Workload Learner, the Workload Monitor and the initial profiles of the pilots should all evaluate the workload using the same parameters or method, or at least provide a normalisation of the representations. Once those values are obtained, they should be simplified in the pairs of time and average level of attention required that would use the Scheduler for the total workload aggregation at every moment. It should be noticed that there is no intention to represent how or when the tasks are performed exactly, especially when they overlap; that is left to the pilot’s will. The main interest is to determine the extent of availability of the pilot to handle more tasks. This respect for the pilot’s personal strategy for the workload management not only follows a guideline about RPAS interface usability for pilot acceptance, it also means an abstraction of the representation that decouples the system from any specific GCS or procedure, making it possible to easily adapt the concept to heterogeneous vehicles, mission-specific tasks, and levels of automation. A question that may arise is what exactly constitutes a task. Its definition is irrelevant to the Scheduler, but an appropriate level of granularity should be chosen: not too wide to be counter-productive, not being able to leverage the workload optimisation; and not too fine to be GCS-specific, to avoid an overfitting that could hypothesise too much about the pilot’s individual workload strategy. [32] shows an example of task analysis, taking the whole mission as the root of a tree that iteratively branches (e.g. mission, phase, segment, function, task) up to the sensory resources needs. [33] also exemplifies a thorough cognitive task analysis for a considerably more demanding military scenario and mission, to guide the design of the GCS team, finding the functional requirements and information required for appropriate awareness. In NtoM, a task definition does not need to fit a consistent conceptual fragmentation like these, although a similar approach could help to define them.

The reason is that while the previous examples seek to optimise the performance in highly demanding military scenarios, NtoM focuses on convenience for safer scheduling. Then, it would roughly define a task like a set or pattern of actions requiring a rather predictable amount of time, without too long waiting times in between that could justify breaking it into smaller tasks to leverage those idle times. These task definitions would be dependent on the mission, the GCS or the ATM's established procedures. A simple example of a task in the experiments was the change of link, which includes checking if the scheduled frequency keeps the link quality expected and then applying the change at an appropriate moment. Another task was the change of the lost link procedure (LLP) if the controller asked for it; otherwise, it was assumed automatically updated when necessary, following either the flight plan or assuming a dynamic contingency evaluation [34]. The procedure consisted of: replying with a WILCO, consulting an alternative plan, applying the change and reporting the new plan to the ATC with a free text message. A third example was the execution of a *direct to* clearance, assumed previously requested. The pilot had to check if there was link coverage available for the segment and, only if it existed, the pilot replied with a WILCO and then could send the commands to the aircraft. After that, if the LLP had to be changed for the new leg, the steps were similar to the change of the procedure by request. Each of these tasks had its own worst-case estimated time required and a workload level that would determine the concurrent tasks allowed. The assignment of the flights can be decided before the flight; here the system would play the role of a workforce scheduler, helping to organise the staff timetable. During the flight, when requested, the Scheduler acts as an online planner, trying to find the most appropriate pilot to receive a specific flight. This transfer could have been scheduled due to end of workday, preference, constraint or coverage issues. It could be the result of a request from the pilot, who is facing an unexpected contingency or subjectively considers that the current workload is too high. The prediction of the cognitive workload is not an easy task, and is at the mercy of the emotional or psychological situation; then the pilot has the option to select flights and ask for them to be transferred. Anticipating that, a transfer could be a decision of the Workload Monitor, which considers that the pilot should be released of the load. The supervisor can force transfers for whatever reason. Finally, a control migration could be a measure of the Event Manager if no activity and feedback from the pilot is received or the pilot is breaching instructions.

The Scheduler calculates the candidate assignments that best fit the constraints and preferences of the flight and also comply with the limit on the current and future workload of the pilot. These suggestions could be automatically applied or subject to confirmation by a supervisor. A similar dynamic crew allocation and supervisor role can be found in [4] when the dispatcher needs to hand off any flight to offer dedicated support as a first officer and the supervisor is in charge of that re-assignment; in NtoM, the best candidates would be automatically calculated and the supervisor, if present, just needs to select and apply a suggestion. A different and interesting approach, which is worth mentioning, appears in [35]. On it, like in NtoM, each aircraft is not permanently assigned to a team or pilot. Instead, the fleet is considered able to mainly proceed autonomously.

When an aircraft requires some interaction from the operator, its petition is queued until it can be served, and the aircraft waits for its turn, performing some holding pattern. While this kind of concept avoids any task overlap among flights □ the operator(s) attends only one at a time, a.k.a. vehicle-based control □ and therefore the workload is always limited, the fact that the aircraft could suddenly enter in a holding pattern to wait for the pilot's attention, something which may be unexpected by controllers, seems more acceptable in segregated airspace. The workload forecast is predicted based on the tasks scheduled in the flight plan and those added/deleted during the flight because of ATC's instructions (a screencast of this behaviour appears in [36]). But it would also take into consideration real-time information on external sources like weather reports and an internal map of the workload built by the Workload Learner. The idea behind this map is to find patterns of workload and place them in date-time-altitude-position coordinates.

This workload could be inferred from the aggregation of different sources like ATC interaction, weather, traffic density, link quality, subjective assessments or biometrics, in short, not task-related workload, but circumstantial workload, or the pilot's perception of it.

This map would be initialised from historic data which could provide some clues about the workload, such as that offered by Eurocontrol's Demand Data Repository (DDR) database. For instance, [37] uses machine learning to predict the level of workload from the controller-pilot voice communications. It would be gradually updated from the activity of the flights to get a workload estimation picture for the specific RPAS case and the pilot's perspective. The final contribution of this map to the workload prediction would depend on the probability of occurrence of the sources of workload registered at the points involved. For the current prototype and experiments, only the scheduled tasks' workload is taken into account for the assignments.

During scheduling, the maximum expected workload at any time is not the sole determinant to choosing a candidate. Constraints and preferences were mentioned as part of the flight definitions. A constraint could be related to the kind of mission or the aircraft model, like a required pilot's licence or certificate. Others could be associated to authorities' regulations, like working and rest hour distribution. Preferences correspond to the possibility to define the so-called stages (not to be confused with flight stages). These could be delimited by temporary or geographic criteria, the flight stage or at the pilot's will (when the limit is subjective or cannot be clearly calculated, like weather conditions) and specify which pilots or which conditions should hold those pilots to be assigned to the stage. For instance, a criterion could be to select pilots with greater expertise for some tasks (functional migration [22]) or just the opposite, select those who are less experienced to allow them to keep or improve their skills. Anyway, stages only represent preferences (i.e. soft constraints) adding points lead over other pilots that also comply with the first filter of workload restrictions and any specified constraints (i.e. hard constraints); but these can help the airline to more favourably organise the staff timetable. Also, they could be defined dynamically during the flight. More details about stages can be found in [28]. Once all the previous variables that contribute to the pilot's assignment have been considered, another significant factor should determine the final decision: to avoid handovers as much as possible, as this is a delicate procedure implying high cognitive resources. When receiving a new flight, the pilot needs to study the flight plan and create a picture of the flight status, pending and next tasks, and fit this new requirement of attention into the rest of the current flights. Consider now a candidate pilot still eligible after workload limits and constraints filters have been passed.

However, the workday of this pilot finishes in a period of time that is less than the remaining time of the flight, which would force another transfer. Therefore, if possible, the scheduler should choose a different pilot. A balance of workload among pilots would also contribute to the migration avoidance. If we do not push the pilots to their limit of workload, any unexpected increase in it will not easily imply a transfer. For that reason, the policy of preferably choosing those pilots with a lower total workload at any time should be followed. This whole idea of the evaluation of the current and future status of the aircraft and workload of the pilot to determine the response of the system can be found at a task level in [24], a suggestion for a pilot/vehicle adaptive cockpit interface focused on the rather more demanding needs of the combat pilot, but with a final aim in line with the requirements of NtoM concurrent piloting: "*improve the pilot's situation awareness and decision-making while alleviating workload*". On it, the workload evaluation will be used to select the kind of information provided to the pilot to perform the task and balance the *stimulus modality* (visual, auditory, haptic).

[24] takes into account the impact of the task switch in situation awareness, similarly as NtoM needs to consider this impact when the switch is made between flights too.

But it also advocates the need to reflect the individual preferences and limitations to cope with the workload, which could evolve as experience is gained, something that would also be reflected in the NtoM pilot workload profile, which defines each task's needs for a particular pilot. [5] provides a mathematical model for these operator-specific requirements of time and workload to perform a task, besides a thorough review of the literature on different aspects of the topic. The model is based on task workload levels that are subjectively rated and is suggested for the dynamic assignment of tasks to aircraft and pilots, optimising the distribution of tasks on the go. This concept schedules tasks, while NtoM schedules flights. But the way the total workload is calculated at a given moment also takes into account the *switching cost* or *cost of concurrence* between tasks, something that in NtoM will be separated into two different values: the cost for the in-flight task swap and the cost for the flight swap; this differentiation comes from the especially high cost of the flight swap. Another difference is that the previous paradigm relies on the scheduling of the tasks partly in the effectiveness both of the operator or the UAV, while NtoM, mainly focused on civil operations, prioritises other aspects: balance of the workload not only at a given moment for safety reasons or to avoid delays in communications and instruction execution, but also throughout the workday to avoid burnout; allowing task rotation to maintain skills, considering that by default a pilot is assigned a whole flight, not a task (scheduling based on performance will tend to specialisation); and avoiding handovers as much as possible. One issue to be determined is the way to update a pilot's workload profile. Recent experiments [38] found a decrease in the brain activity of UAV pseudo-pilots as they gained experience with the simulator. It seems clear that the full potential of a pilot would be underused if the improvement of skills, or the capacity to assume a higher workload, is not reflected in the workload profile. In the work mentioned, this improvement was detected by functional near-infrared spectroscopy. However, the Workload Learner should try to detect automatically such an improvement both in the time required and in the cognitive load. While this seems quite straightforward for the average time to perform the task, clues about the decrease in the mental workload required are not obvious. In any case, we could infer an increase in that load from a decrease in performance. If there is no other way to do it, pilots should periodically go through measurements to update their workload profile.

III.3. Monitoring

Having the system access the flight plan, status of the aircraft, ATC instructions and pilot's actions, the GCS Monitor will trigger warnings or emergency procedures in case an unexpected situation is detected. It can send reminders too, a safety measure when controlling several aircraft, which increases the possibility of forgetting a task, in particular conditional instructions that should be applied at a time or position ahead. Regarding compliance with the ATC instructions, the use of CPDLC instead of voice allows the system to easily understand the semantics of the message and monitor its execution, as recommended in [8, Part I, ch. 2, 5.9]:

"*Monitoring automation capable of questioning certain classes of operator actions that can potentially compromise safety must be designed into the system*".

Therefore, if the system detects that these are ignored or contradicted, after warning the pilot and supervisor, the control of the aircraft could be automatically transferred, with no need for approval from the current pilot. This kind of monitoring is important when pilots control several flights, as they could mix intentions, as was shown during the experiments. It also serves as a crosscheck in what will probably be single pilot operations. While CPDLC provides unambiguous and persistent communication, the cross-verification by both pilots before the execution of the clearances has been recommended in manned aircraft same as is done with voice communications [39]; in this sense, the NtoM monitoring would play the role of the first officer. An example of this need using CPDLC is reported in [40], when pilots do not realise the condition of the conditional clearances and execute the instruction immediately. There was also a problem with the terms "AT" and "BY", which resulted in different interpretations of the conditions [41]. Although these kinds of problems have led to a rewriting of some of the message elements, this illustrates that, with a condition that can be tracked, like position or time, the GCS Monitor can check if the pilot is executing when required and send a warning or confirmation request if not.

Anyway, when possible, the display tries to reduce possible input errors [8, Part I, 3.6] by automating replies or providing buttons that apply the whole sequence of commands associated to the instruction. Screencasts of monitoring and warnings can be found in [42]. Although the current implementation of the prototype relies on CPDLC, voice instructions could also be tracked as long as we could use reliable voice recognition for the system to parse the meaning behind the speech. [43] is a recent work that shows the feasibility of this option, and [44] is a proposal for detecting non-compliance actions to both voice and data link messages instructions. The system could keep an eye on the pilot's health. The simplest way would be to track the activity in the NPI or the GCS and ask for feedback, as a kind of dead man's pedal, after a certain period of inactivity. It could also use biometric devices, as suggested in SPO ConOps [4], which also considered the possibility of a video feed of the cockpit to check the pilot's behavioural state and clearly determine any possible pilot incapacitation and the need to transfer control to ground operators or the onboard automation. Warnings sent by the aircraft associated with failures or unexpected readings should also be highlighted, especially for flights in the background. Free text messages to ATC warning about these issues could be followed by an automatic dependent surveillance (ADS) report containing data like ground vector, air vector or short-term intent. An LLP activation could be followed by a notification to the current data authority if possible (if the link is lost with the GCS but not with the ATC) and a broadcast of the intent to the nearby traffic.

One of the roles of the supervisor, as could be inferred by now, is to provide an optional human confirmation or selection of the Event Handler suggestions. Supervisors could force assignments and transfers, overwriting the Scheduler decisions, or create stages and constraints. For a large fleet, the supervisor could be assumed as the operations manager of the airline operations centre. In smaller fleets, this role could be mixed with the one of the dispatcher, assuming other tasks like request for feedback (the supervisor has a chat room available with every pilot), clarifying actions and assisting pilots.

Besides the usual information that an operations manager has available, the NtoM supervisor interface would provide: the dynamic representation of workload forecast (screencast in [45]), the history of actions and communications, the sequence of pilots for a flight, and the log of any relevant event like warnings, lost links or emergency procedures loaded. Modes or levels of automation would determine the tasks of the supervisor; that way, a supervisor who is busy attending some emergency could delegate to the Event Handler the decisions to apply assignments, avoiding confirmation requests. Regarding the process of confirmation of an assignment suggestion, the Scheduler would provide an ordered list of candidate pilots (those left after considering the hard constraints) and the values of the variables that determined the ranking. In this way, in the event that the best option calculated could not be applied for any reason, the supervisor could take an informed decision, sorting by variables like current workload, worktime left, current number of aircraft or preferences met by the pilot for the stage. The monitoring can also be applied to the link quality. When detecting a dangerous link degradation that could affect the service, the system could warn, and suggest or assign, if possible, an alternative pilot with better coverage. Consider that for NtoM, pilots could be scattered around the world.

Finally, monitoring and workload aspects are combined in the Workload Monitor. This part of the system constantly evaluates the current workload of the pilot, which could be done using several sources: biometrics (physiological parameters like heart rate or pupil dilatation [46]; in any case a non-invasive method that could hamper the piloting), the match of the current task(s) performance with the workload profile, or the response time to reply and execute ATC instructions. If it becomes higher than that allowed by the thresholds observed, different measures could be applied. These should be agreed with pilots. A transfer of control when a high peak of workload is detected could be more annoying to receiving and transferring pilots than just dealing with this high workload if it is only expected to last a short time. The data obtained by this module would be used to update the pilot-independent map of expected workload.

Besides, pairing the record of activity with performance-based evaluations and/or physiological measurements would allow the pilot's workload profile to be updated.

IV. Method

Early experiments, described in [28], were addressed to outlining a basic description of the averaged cognitive workload for each task and time required to carry them out when using the minimalistic GCS embedded in the NPI. That way, we could estimate the limit of workload to be allowed by the Scheduler, and discover, from the usual problems found by the participants, any other aid that could be implemented apart from those already projected. These experiments tried to show the procedure to evaluate and describe the workload in a way manageable by the Scheduler, although it lacked an objective measurement of the cognitive workload, like biometrics, which was substituted by subjective values based on the observation of the participants' performance. That procedure should be extrapolated to each specific GCS and procedures involved. During those first experiments, guidelines like reminders and monitoring were not available. The reason is that the scenarios were designed to identify problematic levels of workload and task overlapping. In this way, the workload threshold could be determined and incompatibilities in task overlapping detected. Evaluating the workload without the previous aids follows the policy that pilots should not rely on reminders and warnings. The system, acting just as a helper to avoid any dangerous consequences, does not try to avoid proactive awareness.

In the stage of development described here, the same initial experiments were carried out, with the same participants, but with some of the aids activated. These were expected to obviously contribute to reducing the number of errors, but it was an opportunity to check to what extent, and evaluate how appropriate was their how and when. Several months had passed from the first tests and participants requested the chance to refresh the concepts, procedures and usage of the interface, and could not remember the events of the scenarios. It did not seem likely that a possible improvement in performance could be just due to an increase in accumulated experience. During the first round of experiments, the workload threshold was intuitively set to 10. Having a flight with no tasks at a given moment contributed with one unit of workload to represent the periodic checking of readings. Nevertheless, the threshold would be easily and greatly exceeded when ATC instructions arrived while doing scheduled tasks; a request for an LLP update added three units and a *direct to* clearance, four. Some errors began to appear when the workload was of 12, so the threshold was reduced to 6 in the second set of experiments. Anyway, the scenarios were designed to provoke overwhelming situations, like demanding controller requests arriving while a couple of scheduled tasks in different flights were concurrent on time, which generated several peaks in a range between 14 and 20.

These situations could be avoided in practice thanks to the workload map learned by the Workload Module. If these could not be avoided, there is another threshold to be considered: the one handled by the Workload Monitor, which would request the Event Handler to act when an excessive workload is detected for a pilot. Reducing the Scheduler threshold also reduces the possibility to require that intervention. Anyway, in this case, the decrease in the threshold did not reduce the number of flights, which was four in both scenarios, and the Workload Monitor was not enabled. In summary, the results should be observed as related to a worst case scenario, with peaks of workload that would probably not be allowed by the system in practice, but able to reveal the potential or the weaknesses of the features and design decision. The aids included in this set of experiments were the following:

A) *Link update reminders*. The link changes were scheduled at specific waypoints representing the close end of coverage for the current link, or the starting of an area with better quality using a different frequency. The most delicate case of coverage transition would be that represented in Fig. 4. If the pilot updates the link in 1, where there is no coverage for the next frequency yet, there will be a link loss, and after the period of time specified by the lost link timeout value, the onboard computer will execute the LLP loaded, which could negatively affect the traffic and the task of the controllers. The link would also be lost if the pilot exhausts the buffer area without changing the link. To avoid these mishaps a couple of warnings were implemented. One of them writes the words "LINK CHANGE" in blue in the strip when a scheduled link change is 30 s ahead. Participants were told that this was a reminder, not necessarily an appropriate moment to change the link, as it depends on the lost link timeout (Fig. 4). They could use the time remaining to the next waypoint to decide the moment. The second warning is triggered some seconds after passing the waypoint if the change has not been registered yet. The words "IN BUFFER AREA" are printed in red, and a synthesised voice reads them, preceded by the callsign. A screencast can be found in [47].

B) *Value checking*. When receiving a clearance, if the pilot introduces a command with a value that does not match that specified in the clearance, such as incorrect waypoint, speed or altitude, a request for confirmation showing the expected value is received before applying the command. If the pilot confirms anyway, the supervisor receives a notification and the controller could also receive a free text message notifying about this. A screencast is shown in [39]. (This kind of problem disappears when the instruction can be applied from the same message in the CPDLC display.)

C) *Unexpected actions*. When an action that should be preceded by a clearance is detected, as before, a confirmation is requested, and both the controller and supervisor are notified if confirmed. A screencast is shown in [39].

D) *Monitoring of instructions*. The system monitors the execution of the ATC requests only if these have been replied with a WILCO. If these must be immediately executed and they are not, the pilot receives a notification after a while. If it is still ignored, the supervisor is notified or the Event Manager requested. For conditional clearances, the pilot is warned with a voice notification and a printed message in the strip when the condition is met; then the procedure is the same as the clearances of immediate application. A screencast is shown in [39].

E) *Airsystem timeout warning*. In CPDLC, messages have a limited time to be answered, after which pilots must ignore the content of the message and the controller reverts to voice to check the problem. To avoid overburdening the ATC, a synthesized voice warns the pilot when there are 30 s left to answer. A screencast is shown in [47].

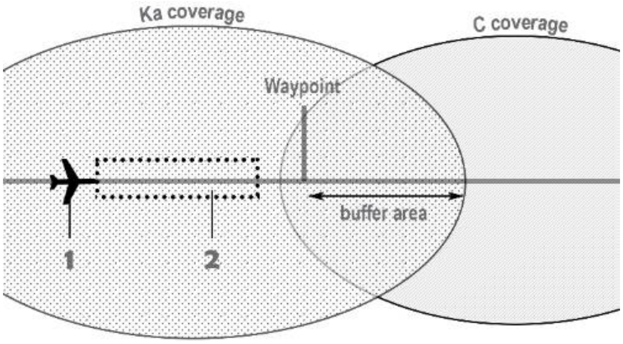


Fig. 4. The dotted rectangle represents the distance ahead given the current speed and lost link timeout. If the pilot changes the link to C in 1, the lost link procedure would be executed when reaching the right end of the box. If they need to change the link before the scheduled waypoint to attend concurrent tasks, they should do it in a position like 2, from where the distance to timeout would overlap the C coverage and the link would be recovered before the timeout

V. Results

Below there are the observations on the result of putting to the test some of the measures of usability and safety conceived for the system, and the comparison of the pilots' performance with the one they showed when the control of the flights only depended on their ability to keep in mind a picture of the situation in every flight.

The voice reading the messages received by background flights was considered useful by a couple of participants, who paid attention to the content of the message and the callsign. The other two found the voice useful as an additional highlight of an unseen message (an envelope icon is displayed too), but felt too stressed and focused on their current tasks to pay attention to what the voice was saying. The "LINK CHANGE" print, conceived to be useful in cases of loss of awareness due to low activity, was instead considered especially useful in moments of high activity and overlapping of tasks. A couple of participants suggested, however, that it should be accompanied by a voice warning, a sound, or blinking. Suggestions like this need to be considered carefully. The aim behind the quantity and kind of warnings is to find a middle ground where these aids help avoid problems but do not encourage a loss of awareness. Too much guiding could induce pilots to let themselves be carried along by reminders instead of actively keeping in mind the situation of the flight, pending actions or close tasks. Even worse, an annoying excess of notifications is an easy way to get people to ignore them. However, the "IN BUFFER AREA" print was considered redundant, as they said that the previous warning already put them on alert. The aim of the "LINK CHANGE" reminder was to be a way to avoid the change of link just in time, placing it ideally some seconds before/after overflying the waypoint. Table I compares when the link update was done depending on whether they had the reminder or not. On it, the change of link is considered done past the scheduled waypoint if it is forgotten or done still inside the buffer area but after at least ten seconds once the waypoint is passed. The changes previous or over the waypoint include the incorrect ones, as this only evaluates the awareness of the update. The results show in parentheses the total link changes of both scenarios, and show no difference with respect to the moment of the change, which has the reminder, while there is a clear decrease in the number of errors when the link change is applied. Maybe the reminder motivated them to check sooner which was the new frequency and its availability, or it pressed them to finish their current task, or just helped them to better organise their time and this allowed them to check their steps more carefully.

TABLE I
LINK UPDATE BEFORE/ON THE WAYPOINT
(TOTAL CHANGES) - ERRORS

Participant	Without reminder	With reminder
1	7 (9) - 1	9 (11) - 0
2	8 (9) - 4	10 (11) - 1
3	8 (9) - 3	10 (12) - 0
4	9 (9) - 0	9 (11) - 1

TABLE II
SEVERITY OF ERRORS WITHOUT AND WITH MONITORING AND GUIDING

Pilot	1st tests	2nd tests
P_1	2 × forgets LLPlan change report 1 × CPDLC timeout 3 × lost link	1 × wrong LLPlan letter in report 2 × CPDLC timeout 1 × lost link
P_2	3 × lost link	No errors
P_3	1 × applies instruction without WILCO 4 × CPDLC timeout 2 × lost link	1 × wrong LLPlan letter in report 1 × forgets LLPlan set but not the report
P_4	1 × forgets LLPlan change report 1 × forgets LLPlan change	2 × wrong LLPlan letter in report 1 × LLPlan change to wrong aircraft

Another interesting decrease was observed for the seriousness and quantity of the errors. Table II details the kind of errors before and after the implemented aids.

The CPDLC air system timeout aural warning [47] was not considered useful by a couple of participants, but in this respect it is important to stress that both of them failed to comply with the priority rules, so they probably did not understand their importance very well.

Participants were told that the highest priority was to keep the link alive, followed by answering the controller to avoid the timeout. Then, there was the execution of instructions that should be applied as soon as possible, but, among them, the one with less urgency would be the change of LLP. However, while carrying out an LLP change request, they received a message in the background and preferred to finish what they were doing instead of answering the other controller; they preferred to get rid of the current task first. One of them arrived on time to reply, the other one exhausted the timeout. In one of the cases, the synthesised voice read a request for an LLP change in the background. Participants were told to always answer with a WILCO, there would always be an alternative plan available. Selecting the other flight, then the message received in the CPDLC display, pressing the WILCO button, and then coming back to the previous flight, is all a matter of four clicks and can be solved in two seconds without involving any decision making; but this participant preferred to try to finish the current task first, failing to reply on time. The second participant received a direct to clearance while doing a task. To reply, first they need to check if there is link coverage available for the new segment, which is slightly more demanding, and could discourage the pilot from stopping what they are doing to reply. While they still have the option to send a STANDBY, which is also a matter of two seconds, to double the time allowed to reply, this pilot still preferred to take a risk and finish her current task, and ended up replying in extremis. The act of postponing an incoming activity while busy on one involving a high cognitive process seems to be a default in human behaviour, as shown in [48], while a task switch can be expected if the current task does not involve a high workload. In the couple of cases mentioned the task they were performing was the most deferrable of all, but it seems that what made them reluctant to answer was facing the flight swap; having to load in your mind the situation of the other flight and then come back to the current one. Therefore, in this respect the interface should address a couple of issues; first, trying to reduce the involvement of cognitive resources in the loading of the second flight situation. Maybe the possibility to reply to background messages using voice recognition (partially implemented but not used during the experiments) to avoid the overload of visual resources, could alleviate some of these situations, a *task interference* solution as shown in [49]-[51] and with a promising performance in [52]. Second, despite their natural inclinations, tasks with higher priority should not be deferred in favour of those with lower priority, whatever the mental workload each of them involves. It would be interesting to try to highlight this fact in the NPI implementation, warning pilots to switch and respect the priority rules if they let themselves get carried away by the inclination to clear a task from their minds. The notifications of wrong values introduced and the request for confirmation of unexpected actions came into play a few times, as sometimes participants forgot to select the correct flight before applying a command, or because they forgot to accept the instruction before applying, so they found it especially useful. Two participants suggested an added measure: a warning before applying a change of link frequency that would mean a loss of link. This measure was initially considered but left out because in the exercises they were allowed to do it if the aircraft reached the area of coverage before the lost link timeout was exhausted (Fig. 4) and, being a common and voluntary action in the scenarios, it was thought that a warning could be annoying. However, one participant emphasised that she preferred to assume that cost and be sure that she would never lose the link by mistake. Anyway, the request for confirmation seems an appropriate middle ground to deal with these errors. [24] describes two main ways to cope with unexpected values at a GCS level: warn or ban. The problem with the first option is that during high levels of workload, the warnings were unnoticed; pilots rejected the second option. NPI asks for confirmation; the action is not forbidden, but the pilot is forced to acknowledge the warning to go on. One particular mistake appeared in every tester related to the request for an LLP change, reinforcing the importance of automating the reports as much as possible to avoid input errors. They were told that there was always an alternative plan, so the reply should always be a WILCO. After that, they would check the substitute procedure on a paper chart given the position, altitude and remaining fuel. Once set, they should find time to inform the controller of the new procedure (identified as LLPA, LLPB…) using a free text message. Nevertheless, all of them sent, at least once, the wrong letter of the new plan, although correctly set in the aircraft. One participant sent the report without previously setting the plan, nor doing it later. The inputs monitored up to now rely on the syntax of the message elements to parse and compare the values. In a free text message element, the report could come like “LLPlan set to X”, “LLPY set” or “X changed to Y”, which makes it difficult to compare the content with the plan just set. Of course, this report could have been considered automated from the beginning, as others are already for ATC report requests. The alternative procedure could also be suggested by the interface given the current status and readings, instead of using a printed chart. But the procedure, as designed, tried to represent a sample of demanding tasks in time and cognitive resources, where the pilot needs to take her/his eyes off the current screen for a moment. A real RPAS pilot even reported that this could take them up to five minutes. In practice, it seems appropriate to automate the report to the ATC once the LLP is changed. In one of these report errors, and in favour of the CPDLC display usability, the user noticed the error and sent the free text message again, this time with the correct plan letter. The time elapsed between the first and the second message was impressive: 2 s. The lukewarm reception of the aural warnings, especially the synthesised voice, should be considered in perspective. In these experiments, the GCS was embedded in the NPI. This means that they are constantly looking at a unique screen, with all the information at a glance, so they can rely mainly on visual resources. However, in a real installation, pilots could be focused on the GCS consisting of several monitors when an event appears in a background flight, and the fact of having a voice reading the message or describing the event, with no need to take their eyes from the current monitor, could be appreciated a lot more as this lets them know the urgency of the event and quickly decide their workload management strategy [51] to attend the new task. This would correspond to the findings of Wickens et al. [49] about the benefits of the cross-modal displays, also tested in flight simulation tasks, for better management of mental resources while attending multiple tasks. Concerning the notifications (Fig. 2, 2), like those received when a pending action is delayed, users found it useful to remember which aircraft had the pending task, as these dialogues include the associated callsign in the tab. If they remember having a pending action but cannot remember which aircraft it was, they are forced to click on each strip and check the CPDLC history or the printed scheduled tasks to find it out, which takes time. This revealed the need to print some kind of pending action icon in the strip. It should be different depending on whether this comes from an ATC request or a scheduled task, so they could know beforehand where to check the details of the action. The same need was observed for the open messages. By now, an open message icon is printed on the CPDLC display next to the message that has arrived to remark that it has not been answered.

However, it is needed to have that aircraft selected to see the icon. When a new message in a background flight arrives, an envelope icon, differently coloured depending on the severity, is added to the strip, meaning that there is an unread message. This icon disappears when that aircraft is selected. If the pilot does not answer the message at that moment, and moves to another flight, there are no cues in the strip to remember which was the callsign with the open message, and pilots looked for it by trial and error.

VI. Conclusion

The ConOps proposed pursues the feasibility of having a pilot/flightcrew controlling more than one aircraft concurrently in non-segregated airspace by providing tools to manage the workload, information, feedback and safety required for such a scenario. A first set of experiments was used to exemplify the process of measurement and description of the workload of the pilots and try to determine a threshold. Non-professional pseudo-pilots, with a minimalistic GCS, performed those experiments and an estimation of the workload values based on observation. While more realistic testing with objective metrics would be desirable, the previous tests were intended to illustrate the input requirements of the Scheduler, i.e. a workload model describing the cognitive load and time required for each pilot while performing each task or any task overlap. This will be the focus of the next step of development: the description of a methodology to build the personalised cognitive workload model from objective data and how to estimate the thresholds. With this model being dependent on the GCS usability, the procedures behind the tasks and the pilot's expertise, the methodology will address the propagation of the variation of any of those variables on the rest of the model. In that way, the system will be able to update automatically the model as pilots improve their skills or any task procedure changes or are automated in a particular aircraft. If all the features suggested in this ConOps are finally implemented and put to the test, and pilots conclude that they do not feel like they are able to safely pilot more than one aircraft, their suggestions and opinions could be introduced in a new iteration of the design and development process, eventually leading to a version that could satisfy their needs. However, some collateral benefits could be leveraged. First, the implementation of the standardised CPDLC display used during the simulations could allow future researchers and students to perform faithful simulations of procedures that will be common for manned and unmanned pilots in the future. This CPDLC display was proposed in [20] as a potential training tool [8, Part I, ch. 3, App. B, 2.4 and 2.5], and an optimised display for RPAS pilots. Second, the system is able to compare the pilot's actions with the flight plan and the controller's instructions, which is useful as a contingency detection measure even if we are considering a one-to-one pilot-aircraft relationship.

Third, in a transfer of control handled by the system, any error related to the misconfiguration of the receiving GCS disappears. The whole process and the information involved, like GCS modes, aircraft status, pending actions and communications, are kept and transferred by the server, so the role of the transferring pilot is minimum or even inexistent. This is especially interesting when the transfer was requested because of an excessive workload, due to GCS malfunctioning or pilot incapacitation. To make it possible, we need a GCS providing an API for the NPI to be able to apply commands, load/unload flights and listen to the activity and flight instruments readings. If monitoring or piloting the aircraft is considered to deserve a dedicated AVO per flight, the concept, or at least the workload scheduling, could still be applied to other roles, like dispatchers, first officers or payload operators. In the event that the experiments show that the measures successfully address the requirements of the pilots, some extra benefits could be added to those aforementioned. The most straightforward is that it would allow the air carrier to optimise its human resources. In addition, a map of expected workload would be learnt to inform the Scheduler workload forecast; this kind of information could be useful to different organisms and researchers to analyse the consequences of the traffic management from the RPAS point of view, and apply corrective measures, if necessary, in the design of airways or procedures, conducive to better integration. The balance of the workload could also help to avoid the burnout usually suffered by RPAS pilots [53], avoiding fatigue as an online planner and seeking a healthier shift distribution when acting as a workforce scheduler.

Acknowledgements

This work was part of a thesis funded by the grant with reference BES-2014-071096 belonging to the Research Personnel Training sub-programme from the Ministry of Economy and Competitiveness of Spain, and partially by the Ministry of Economy and Enterprise of Spain under contract TRA2016-77012-R. Special thanks to Javier Negre, Joan Pere Sánchez and Laura López for their collaboration in the experiments and to Dr. Alan Hobbs and Jerónimo Vicente for their kind provision of expert advice in the RPAS area.

References

[1] European RPAS Steering Group, *European RPAS Roadmap*, 2013, http://ec.europa.eu/DocsRoom/documents/10484/attachments/1/translations/en/renditions/native.

[2] *Single European Sky*. https://www.sesarju.eu/approach/objectives

[3] M. L. Cummings et al, Task Versus Vehicle-Based Control Paradigms in Multiple Unmanned Vehicle Supervision by a Single Operator, *IEEE Transactions on Human-Machine Systems*, 2014.
doi: 10.1109/THMS.2014.2304962

[4] A. Schulte, D. Donath, F. Honecker, Human-System Interaction Analysis for Military *Pilot Activity and Mental Workload Determination, IEEE International Conference on Systems, Man, and Cybernetics*, 2015.

[5] C. C. Murray et al, Incorporating Human Factor Considerations in Unmanned Aerial Vehicle Routing, *IEEE Transactions on Systems, Man, and Cybernetics*: Systems, Vol. 43, No. 4, 2013.

[6] A. E. Ortiz, C. Langbort, On Multi-UAV Scheduling for Human Operator Target Identification, *American Control Conference*, 2011.

[7] K. D. Bilimoria, W. W. Johnson, P. C. Schutte, Conceptual Framework for Single Pilot Operations, *International Conference on Human-Computer Interaction in Aerospace (HCI-Aero 2014)*

[8] *Manual of Air Traffic Services Data Link Applications*, First Edition, Doc 9694-AN/955, International Civil Aviation Organization (ICAO), 1999.

[9] A. D. Churchman, W. F. Havens, C. A. Pell, S. C. Pilet, Controls Automation and Task Allocation Program, *Proceedings of National Aerospace and Electronics Conference (NAECON '94).*
doi: 10.1109/NAECON.1994.332852

[10] T. Chen, *Management of multiple heterogeneous unmanned aerial vehicules through transparency capability*, doctoral thesis, Human-Computer Interaction [cs.HC]. Télécom Bretagne; Université de Bretagne Occidentale, 2017. https://hal.archives-ouvertes.fr/tel-01577924.

[11] DARIUS (Deployable SAR Integrated Chain with Unmanned Systems) Project,
https://cordis.europa.eu/project/rcn/102362/factsheet/en
http://www.spacetecpartners.eu/sites/default/files/attachments/08-day1-darius_project_overview.pdf

[12] G. Calhoun, H. Ruff, C. Breeden, J. Hamell, M. Draper, and C. Miller, Multiple Remotely Piloted Aircraft Control: Visualization and Control of Future Path, *Virtual, Augmented and Mixed Reality. Systems and Applications, Proceedings, Part II*, 2013.
doi: 10.1007/978-3-642-39420-1_25

[13] A. Hobbs, B. Lyall, Human factors guidelines for remotely piloted aircraft system (RPAS) remote pilot stations (RPS), *technical report for NASA UAS in the NAS Project*, 2016.
doi: 10.13140/RG.2.2.12562.45768

[14] M. L. Cummings, The Need for Command and Control Instant Message Adaptive Interfaces: Lessons Learned from Tactical Tomahawk Human-in-the-Loop Simulations, *CyberPsychology and Behavior* vol. 7, 2004.
doi: 10.1089/cpb.2004.7.653.

[15] *Control Six MQ-9 Reapers From Your Laptop With GA-ASI's Heresy Software*, Avionics International, October 2018, https://www.aviationtoday.com/2018/10/19/gaasimmc/.

[16] Early Demonstration & Evaluation Platform (EDEP) http://www.eurocontrol.fr/projects/edep/

[17] M. Pérez-Batlle, R. Cuadrado, C. Barrado, P. Royo, E. Pastor, Real-time Simulations to Evaluate RPAS Contingencies in Shared Airspace, *Fifth SESAR Innovation Days*, 2015.

[18] RTI Connext DDS https://www.rti.com/products

[19] J. S. McCarley, C. D. Wickens, *Human Factors Implications of UAVs in the National Airspace*, Technical Report AHFD-05–05/FAA-05–01, Atlantic City, NJ: Federal Aviation Administration, 2005.

[20] M.-A. Fas-Millán et al., Controller-Pilot Data Link Communications Display oriented to multiple Remotely Piloted Aircraft Systems pilots, (2019) *International Review of Aerospace Engineering (IREASE)*, Accepted for publication.

[21] NtoM website: http://recerca.ac.upc.edu/ntom

[22] A. P. Tvaryanas, *Human Factors Considerations in Migration of Unmanned Aircraft Systems (UAS) Operator Control*, Report No. HSW-PE-BR-TE-2006–0002, Brooks City, United States Air Force, Performance Enhancement Research Division, 2006.

[23] http://recerca.ac.upc.edu/ntom/videos/NtoM_handover.mp4

[24] S. S. Mulgund, G. L. Zacharias, A Situation-Driven Adaptive pilot/Vehicle Interface, *Proceedings of the Third Annual Symposium on Human Interaction with Complex Systems - HICS '96*, IEEE Computer Society Press.

[25] M. R. Endsley, C. Bolstad, Human Capabilities and Limitations in Situation Awareness, AGARD CP-520, *Combat Automation for Airborne Weapons Systems: Man/Machine Interface Trends & Technologies*, Neuilly Sur Seine, France.

[26] Electronic Code of Federal Regulations, https://www.ecfr.gov

[27] R. J. Shively, A. Hobbs, B. Lyall, C. Rorie, Human performance considerations for Remotely Piloted Aircraft Systems (RPAS), *Remotely Piloted Aircraft Systems Panel Second Meeting (RPASP/2)*, Montreal, 2015.

[28] M.-A. Fas-Millán et al., Dynamic Workload Management for Multi-RPAS Pilots, (2019) *International Review of Aerospace Engineering (IREASE)*, Accepted for publication.

[29] P. G. Raeth, J. M. Reising, A Model of Pilot Trust & Dynamic Workload Allocation, *Proceedings of the IEEE 1997 National Aerospace and Electronics Conference. NAECON.*
doi: 10.1109/NAECON.1997.617760

[30] B. L. Hooey et al, The Underpinnings of Workload in Unmanned Vehicle Systems, *IEEE Transactions on Human-Machine Systems* Vol. 48, Issue: 5, Oct. 2018.
DOI: 10.1109/THMS.2017.2759758

[31] C. F. Rusnock et al, Workload profiles: A continuous measure of mental workload, *International Journal of Industrial Ergonomics*, Vol. 63, January 2018.
DOI: 10.1016/j.ergon.2016.09.003

[32] C. Bierbaum, S. Szabo, T. Aldrich, *Task analysis of the UH-60 mission and decision rules for developing a UH-60 workload prediction model*, Vol. I: Summary report, U.S. Army Res. Inst. Behav. Social Sci. Belvoir, VA, Tech. Rep. AD-A210763, 1989.

[33] F. M. Almirao, F. B. da Silva, S. D. Scott, M. L. Cummings, Designing Decision and Collaboration Support technology for Operators in Multi-UAV Operations Teams. *MIT report, Humans and Automation Laboratory, HAL*, 2007.

[34] E. Pastor et al, In-Flight Contingency Management for Unmanned Aerial Vehicles, *Journal of Aerospace Computing, Information, and Communication*, Vol. 9, No. 4, 2012.
DOI: 10.2514/1.55109

[35] L. A. M. Bush, Shared UAV Enterprise Operator Pooling Framework (SUAVE), *2015 IEEE International Inter-Disciplinary Conference on Cognitive Methods in Situation Awareness and Decision Support (CogSIMA).*
doi: 10.1109/COGSIMA.2015.7107970

[36] http://recerca.ac.upc.edu/ntom/videos/NtoM_workload_demo_DirectTo.avi

[37] H. Wang, D. Gong, R. Wen, Air traffic controllers workload forecasting method based on neural network, *27th Chinese Control and Decision Conference*, 2015.
DOI: 10.1109/CCDC.2015.7162334

[38] D. Richards, K. Izzetoglu, UAV Operator mental workload - A neurophysiological comparison of mental workload and vigilance, *AIAA Modeling and Simulation Technologies Conference*, 2017.
doi: 10.2514/6.2017-3670

[39] K. Cardosi, T. Lennertz, *Flight Deck Human Factors Issues for National Airspace System (NAS) En Route Controller Pilot Data Link Communications (CPDLC)*, report for the Federal Aviation Administration Human Factors Division (ANG-C1), 2017.

[40] IFALPA, *Navigational errors on the North Atlantic*, IFALPA Safety Bulletin 30 Sept 2011, https://www.ifalpa.org/media/2214/12sab013-navigation-errors-on-the-north-atlantic.pdf.

[41] T. Lennertz and K. Cardosi, *Flightcrew Procedures for Controller Pilot Data Link Communications (CPDLC)*, report prepared for the Federal Aviation Administration, DOT/FAA/TC-15-12, 2015.

[42] http://recerca.ac.upc.edu/ntom/videos/NtoM_monitoring_examples.mp4

[43] V. N. Nguyen, H. Holone, N-best List Re-ranking Using Syntactic Score: A Solution for Improving Speech Recognition Accuracy in Air Traffic Control, *16th International Conference on Control, Automation and Systems (ICCAS)*, 2016.
doi: 10.1109/ICCAS.2016.7832482

[44] Lacko et al., *Aircraft Systems and Methods for Detecting Non-Compliant Pilot Action*, U.S. Patent No.: US 9,446, 852 B2, issued Sep. 20, 2016.

[45] http://recerca.ac.upc.edu/ntom/videos/NtoM_workload_demo_speed_change.mp4

[46] Othman, N., Romli, F., Mental Workload Evaluation of Pilots Using Pupil Dilation, (2016) *International Review of Aerospace Engineering (IREASE)*, 9 (3), pp. 80-84.
doi: https://doi.org/10.15866/irease.v9i3.9541

[47] http://recerca.ac.upc.edu/ntom/videos/NtoM_reminders.mp4

[48] D. D. Salvucci, P. Bogunovich, Multitasking and Monotasking: The Effects of Mental Workload on Deferred Task Interruptions, *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, 2010.

[49] C. D. Wickens et al, Compatibility and Resource Competition between Modalities of Input, Central Processing and Output, *Human Factors*, Vol. 25, No. 2, 1983.
doi: 10.1177/001872088302500209

[50] S. Dixon and C. Wickens, Control of multiple-UAVs: A workload analysis, *Proc. 12th Int. Symp. Aviation Psychol.*, 2003.

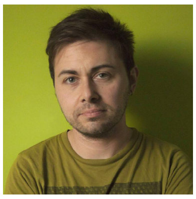
[51] C. Wickens, S. Dixon, D. Chang, *Using interference models to predict performance in multiple-task UAV environment–2 UAVs*, technical report for Micro Analysis and Design, 2003.

[52] M. Draper, G. Calhoun, H. Ruff, D. Williamson, T. Barry, Manual versus speech input for unmanned aerial vehicle control station operations. *Proceedings of the Human Factors and Ergonomics Society 47th Annual Meeting*, 109-113, 2003.
doi: 10.1177/154193120304700123

[53] W. Chappelle, A. Salinas, K. McDonald, *Psychological Health Screening of Remotely Piloted Aircraft Operators and Supporting Units*, Defense Technical Information Center, 2011**,** http://www.dtic.mil/dtic/tr/fulltext/u2/a582856.pdf.

Authors' information

Polytechnic University of Catalonia.



Miguel-Ángel Fas-Millán is a computer engineer from the Universitat Jaume I (UJI) of Castelló, master in Computer Vision and Artificial Intelligence from the Universitat Autònoma de Barcelona (UAB), and currently Computer Architecture Ph.D. candidate at the Polytechnic University of Catalonia (UPC).

Enric Pastor is a computer engineer and Ph. D. in Computer Architecture from the Polytechnic University of Catalonia (UPC). Dr. Pastor has been working in the UPC since 1992 and is currently an associate professor in UPC's School of Telecommunications and Aerospace Engineering in Castelldefels. He is co-founder and head of Intelligent Communications and Avionics for the Robust Unmanned Aerial Systems (ICARUS) research group.