Ariadne: A common-sense thread for enabling provable safety in air mobility systems with unreliable components

Olatunde Sanni*, Mark Mote†

Georgia Institute of Technology, Atlanta, Georgia, 30332, USA

Daniel Delahaye‡

Ecole Nationale de l'Aviation Civile, Toulouse, France

Maxime Gariel§

Xwing

Thanakorn Khamvilai¶, Eric Feron||, Safa Saber**

King Abdullah University of Science and Technology, Thuwal, Saudi Arabia

Commercial air travel is by far the safest transportation modality available to humanity today. It has achieved this enviable status by deploying thousands of professionals, including pilots, dispatchers, and air traffic controllers to operate very reliable air vehicles, bringing them and their passengers safely from origin to destination while managing dangerous weather, other traffic and system failures for decades. The air transportation has been undergoing undeniable and continuous progress and modernization since its inception. Thanks to advances in navigation capabilities, such as satellite-based navigation systems, aircraft can fly increasingly complex trajectories, including final approaches. The same aircraft are envisioned to fly in formation relatively soon. More daring moves include the recent introduction of "Free Flight" operations. Despite all these impressive improvements, they remain largely incremental in nature and they hit a "wall of complexity" that makes it somewhat difficult to incorporate more automation, such as the elusive, and perhaps infeasible, goal of achieving fully automated air traffic control, and to design and insert autonomous vehicles, small and large, in cities and at high altitudes.

We introduce *Ariadne*, a thread to accelerate the productivity gains achieved by air traffic services providers around the globe. *Ariadne* is an engineered version of the common sense practice of always keeping a "Plan B", and possibly "plans C, D, E, and F" against unexpected events when any decision is made by pilots, air traffic controllers, dispatchers, and any other safety-critical actor of the air transportation system. The name "Ariadne" was chosen to honor the mythical character Ariadne, daughter of Minos the king of Crete, who conceived the "Plan B" mechanism that would allow her lover to exit Daedalus' Labyrinth after killing the Minotaur.

Ariadne and its informal definition as "Plan B engineering" offer surprising opportunities and properties, including not only provable operations safety with unproven components, but also a thread that can inherently be scaled up and quickly adapt to new air traffic scenarios, including the transition to free flight and accommodation of unmanned aviation. It also supports existing operations and therefore it does not conflict with current air traffic control practices. Modern computational capabilities and powerful AI algorithms make its implementation increasingly feasible to address more aspects of air traffic management.

*Graduate Research Assistant, School of Aerospace Engineering, Student Member AIAA, olasanni1[at]gatech.edu

†Graduate Research Assistant, Institute for Robotics and Intelligent, Student Member AIAA, marklmote[at]gmail.com

‡Professor and Head, Optimization and Machine Learning Group, delahaye[at]recherche.enac.fr

§Chief Technical Officer, maxime[at]xwing.com

¶Visiting Student, Electrical and Computer Engineering, thanakorn.khamvilai[at]kaust.edu.sa, tkhamvilai3[at]gatech.edu

||Professor, Electrical and Computer Engineering, eric.feron[at]kaust.edu.sa

**Graduate Research Assistant, Mechanical Engineering, safasaber[at]kaust.edu.sa

I. Defining the *Ariadne* thread

Ariadne : All air transportation operations should be planned, designed, and performed under the constraint that at any time, and for any known and reasonably likely system malfunction, and each significant combination thereof, at least one, but preferably more sequences of actions have been explicitly computed by hand, with a computer or a combination of both, that will bring the air transportation system back to absolute safety, where absolute safety is defined as a state where no person will be hurt or killed for the foreseeable future as a direct consequence of the behavior of the vehicles involved with the system's operation.

Ariadne defines a thread. It is therefore not an object, not an algorithm and not any specific, implementable solution. Its purpose, which is system safety, is probably not even achievable otherwise than by grounding all air vehicles, an extreme scenario that has happened on September 11, 2001. ***Ariadne***, however, offers a very simple paradigm to support the proper definition and design of new air transportation sub-systems and operations. ***Ariadne*** is sufficiently general to adapt to past, current, and future operations and equipment. It is sufficiently precise that its projection onto any specific, targeted development is easy to grasp and use to identify possible gaps. When used for design purposes it can support the engineer to quickly identify *and explain* the "pain points" that may need solving. Because it is defined at the level of the entire transportation system, it empowers the engineer or the analyst to think not only about the local issues raised by the subsystem under consideration, but also its impact on the air transportation system as a whole. Finally, it is practical, although it sometimes raises important challenges that require additional efforts to be undertaken.

Ariadne will probably appear to many as completely trivial and falling within the "common-sense" category, not the "research" category. The authors hope that the rest of the report, which explores ***Ariadne***, its relevance, and some of its practical consequences, will nevertheless be of value to all.

A. Basic principles

Ariadne can be simplified momentarily and be likened to always maintaining a safe "Plan B" during mission execution, with the purpose of executing "Plan B" as soon as the nominal activity appears to be dangerous. Considered from that perspective, maintaining a "Plan B" makes sense and hundreds of examples can be found in everyday life. For example, financial advisers always recommend their clients keep a few weeks' worth of salary in a savings account so as to be ready and face unexpected financial expenses, such as the medical costs following an accident, the replacement of a commuting vehicle following a collision, a few weeks at a hotel if the home is damaged by a hurricane, and so on. Defensive driving is another example, whereby a driver is trained to always be aware of her surroundings and be ready to take appropriate action in case of unexpected events. When faced with the prospect for difficulties procreating, such as when the presumptive father is about to undergo critical prostate surgery, "Plan B" is about storing the father's sperm by freezing it prior to surgery, with the possibility of using it to support alternative methods to natural procreation, should the latter become impossible.

Ariadne is formally, but sometimes erroneously defined in a feedback control framework as adopting an alternative, safe behavior, should the nominal behavior become impossible. Consider the block diagram shown in Fig. 1. The

Figure 1 General "Plan B" architecture

system is driven primarily by Plan A, which could originate from any system, including a human, an expert system, a neural network, or be a deterministic signal. The signal could be a sentence, a stream of discrete events, or anything that may be accepted as "Plan A". A secondary system produces Plan B. Unlike Plan A, Plan B is completely engineered and its sole role is to bring the system to a "safe haven", accounting for all known environmental variables (e.g. weather,

runway condition, traffic congestion, bird activity, etc.) that *may* yield serious trouble. Thus, Plan B is cautious and its only role is to "bring the system to a safe condition". In its most naive form, Plan B is activated when "Plan A" becomes infeasible. Several plan Bs (plans C, D, E, and F) might be present to account for the many different failures, sometimes elementary, sometimes combined, that may affect an airborne operation at any time.

B. Past research work related to *Ariadne*

"Plan B" is part of every form of life, and it is the result of a need for survival not only under nominal conditions, but also off-nominal. For example, it is now known that acacia trees, whose leaves are one common form of food for herbivorous fauna in Africa, are capable of exercising a kind of "Plan B" making these leaves lethal by raising tannin-C in their leaves to deadly levels in a matter of minutes if they are over-grazed [1]. Likewise, rodents very often create burrows with emergency exits or "bolt holes" to escape predators, flooding or other causes that may block their main entrances [2]. The sympathetic nervous system in mammals acts as a physiological "Plan B" by preparing the body for a "fight or flight" response. Upon detection of a threat, glucose is transmitted from storage sites, blood is shifted to the organs that are most essential for physical exertion, heart rate is increased, adrenaline is released, and cognition is sharpened [3].

Perhaps the first fully documented and engineered "Plan B" is the "Ariadne thread", where Ariadne is an important female character of the Greek mythology [4]. This myth gave the inspiration for the authors to name their paradigm after her. Other works centered on aeronautical and space applications chose the name "Ariadne" to describe their tools or findings, see [5–8] but neither do these works come close to matching Ariadne's mythical achievement nor does their meaning match so well that of "Ariadne's Thread" [9–11]. The original Ariadne Thread arose in ancient Greece, more specifically in Crete. Every nine years, Minos the king of Crete had to offer a human sacrifice to the Minotaur, a half-human, half-bull monster born from the illegitimate union of Pasiphaé, the wife of Minos, and a white bull sent by Poseidon to seduce her. The human sacrifice consisted of seven young men and seven young women that were thrown into the labyrinth, a convoluted maze designed by architect Daedalus, to be killed by the Minotaur. Theseus, a semi-god born from two successive unions by Princess Ethra with king Ege of Greece first and with Poseidon later that night, heard about the sacrifice and decided to bring it to an end by killing the Minotaur. He was helped in his task by Ariadne, the daughter of king Minos, who had fallen in love with him. She gave a spool of thread to Theseus for him to unwind as he was making progress through the labyrinth. Once he killed the monster, Theseus could find his way back to safety by following the thread back to the labyrinth's entrance. Ariadne had therefore defeated Daedalus' engineering genius. Later on, Daedalus managed to escape from Crete to Greece using a human powered flight system of his own making, a feat that was repeated no earlier than 1988 by a team of researchers at MIT [12–15]. His son, Icarus, was not so lucky and fell into the sea, where he drowned. Today, the "Ariadne Thread" designates a paradigm that, if it is adopted, will help a project reach a given goal. In the context of this paper, *Ariadne* is proposed as a supporting paradigm to maintain, modernize, and introduce novelty in air transportation operations while keeping the entire system safe.

Another early and documented instance of "Plan B" dates back to the XVIIth century, when French writer Charles Perrault publishes "Le Petit Poucet" (Hop-o'-My-Thumb) in 1697 [16, 17]. The tale is essentially identical to the myth of Ariadne's Thread: The main character, Hop-o'-My-Thumb, keeps dropping small pieces of white gravel or breadcrumbs behind him when his parents take him and his brothers out to the forest and with the purpose of getting them lost. When the parents, out of despair and poverty finally abandon them in the deep of the forest, the children follow the trail of gravel to come back home safely. Variants of this theme are seen in other folk tales, such as the German "Hansel and Gretel".

More recent and formal work related to *Ariadne* include the "Simplex architecture" introduced by Lui Sha, who used this name to describe the architecture shown in Fig. 1 [18]. Innumerable references exist that refer to the Simplex architecture that will not be discussed here.

In 2003, Schouwenaars and co-authors [19–21] demonstrated an implementation of Plan B that allowed the successful and safe operation of a large drone (a retrofitted T33) from the cockpit of an F-15 using mixed-integer programming techniques.

A recent flurry of research activity has been formalizing *Ariadne* using control theory-oriented concepts, such as barrier functions, Lyapunov stability theory, and the computation of reachable sets, backwards reachable sets, viability kernels and related problems [22–26]. Though operating mostly in a continuous, deterministic setting, the concepts easily adapt to systems operating in discrete state-spaces, first and foremost in the spectacular application ACAS-X, whose entire state-space model is discrete, although the air vehicles it acts upon are usually considered to be continuous

time systems driven by ordinary differential equations [27–29].

C. *Ariadne* 's inspiration from existing implementations

The number of "Plan B" mechanisms that have been implemented in practice is enormous. While not all of them meet the level of rigor that model-based mathematical analyses afford, they all follow the architecture shown in Fig 1. For example, an elevator is equipped with emergency braking action that is ready to step in as soon as the elevator speed is excessive. A modern, electronic version of the elevator brake is the emergency braking action automatically triggered by road vehicles if a danger is perceived, for example an obstacle in front of the vehicle as shown in Fig. 2. Such automated emergency braking actions operate much faster than their manual equivalent because the time spent by

Figure 2 Emergency braking for a Volvo Truck - Illustration. *Source: Volvo Trucks*

automation from recognizing an obstacle to stopping the vehicle is far shorter, in fact negligible compared with the proven 0.2 second operation delay of human operators due to the speed of neural message transmission from scene reconnaissance to foot action. As a result, the minimum safe distance that must separate a vehicle from the next one ahead for a successful emergency braking action is smaller, with obvious and positive consequences on increasing road capacity safely. The same automated emergency braking actions do, however, suffer from sensing imperfections sometimes causing a vehicle to execute an emergency braking action to avoid colliding with irrelevant or inexistent objects [30], an unpleasant event often called "false alarm".

Inspiration for *Ariadne* is also prevalent in many aerospace applications [25, 31–33]. One well-known case is related to take-off procedures, whereby aircraft, runways, and aircraft take-off procedures are carefully arranged for a "Plan B" to always exist. In particular, a very precise set of airspeeds is identified for each aircraft so that, regardless of a number of adverse events, the aircraft always has a "Plan B" available. For example, past a certain speed V_1 , the aircraft should be in a position to take-off despite even in case of one engine failure, while below V_1 the aircraft should be able to break on the runway without exceeding the threshold, thus maintaining a "Plan B" available at anytime during take-off. The choice of such speeds is naturally planned assuming the runways have appropriate lengths to enable these "Plan B" to make sense, as will be discussed further below.

Other *Ariadne* instantiations in Aerospace systems include the Traffic Collision and Avoidance System, TCAS. In essence, TCAS provides a Plan B (a collision avoidance maneuver) every time an aircraft enters a collision course with another aircraft. The avoidance maneuvers (Plan B) are chosen from a simple and finite library (typically altitude change maneuvers) so as to simplify their implementation by human pilots and maximize the efficiency of collision evasion maneuvers. In this case, it is worthy to note that the execution of the Plan B, in this case, not only must account for the relative speeds and maneuvering characteristics of the aircraft involved, but also the uncertainties in pilot actions, such as their reaction time, the amplitude of the maneuvers as implemented by the pilots, and other factors. It must be noted, however, that uncertainties as dramatic as those leading to the Überlingen accident [34] can only be accounted for once they are found to happen: Until that accident, TCAS was rooted in high confidence that TCAS-equipped aircraft/pilot tandems would follow TCAS recommendation. The Überlingen collision showed that was not the case: the TCAS

systems worked appropriately and cooperatively, and the accident was due to a difference of priority given to TCAS vs. Air Traffic Control resolution advisories by the pilots of the aircraft that collided. Once this accident happened and the possibility of confusion of priority between air traffic control and TCAS became known, updating TCAS resolution advisories accordingly was done as published in [35] and illustrated in Fig 3.

Figure 3 TCAS before and after Überlingen accident [35]

Further, other collision avoidance systems have been developed, most notably Auto-GCAS, an automatic ground collision avoidance system initially developed for the F-16 fighter aircraft. The system constantly computes the intersection between a hypothetical recovery trajectory (a 5-g pull-up maneuver) and the ground. If the intersection is

EMERGENCY BRAKES

A crash alert system would automatically take control of an F-16 when it is 1.5 seconds from crashing. The F-16 would then make a 5-G climb before leveling out at a safe altitude.

Source: Air Force Research Laboratory

JOHN BRETSCHNEIDER/STAFF

Figure 4 Auto-GCAS illustration. *Source: Air Force Research Laboratory*

non-empty, then the run-time assurance triggers the pull-up maneuver to avoid the ground. The success of Auto-GCAS has been so great that the US Air Force has posted numerous videos of actual recovery maneuvers on YouTube [36] and the engineering team in charge of developing Auto-GCAS won the Collier Award.

Redundant avionics systems, especially as implemented in Fly-by-Wire commercial aircraft, also form an embodiment of *Ariadne*, whereby a backup computer is always ready to take over the control activities left by the primary flight computer, should that computer fail.

II. Recent and ongoing research

Recent examples of *Ariadne*-inspired/inspiring research of direct relevance to the Aerospace enterprise include ACAS-X, on the one hand, and a set of mechanisms that aim at verifying "Plan B" implementations.

A. Collision avoidance Systems: ACAS-X

ACAS-X and its variant ACAS-X_u are systems preoccupied with collision avoidance [37]. ACAS-X is thought of as a potential successor to today's standard for "last-minute collision avoidance", known as TCAS and used world-wide. At the time when this report is written, ACAS-X is still under active development and evaluation, supported by the Federal Aviation Administration and executed by a number of partners, including MIT - Lincoln Laboratory, Johns Hopkins - Applied Physics Laboratory, and other university participants, such as Carnegie Mellon University. The key novelty of ACAS-X is that it was developed using techniques falling under the broad umbrella known as Artificial Intelligence, including modelling via Markov Decision Processes, and decision-making using Reinforcement Learning and Machine Learning techniques. Among the surprising facts related to ACAS-X, there is the basic aircraft behavioral model, which is a model operating over a *discrete state-space*, whereas aircraft behaviors are usually captured by differential equations [38, 39] - thus bringing the relevance of these traditional models into question for the development of future safety-critical functions for aerospace systems. Another surprise is the expression of high-level requirements on ACAS-X in the form of objectives to be optimized and constraints to be respected. The more traditional approach usually directly specifies behaviors, regardless of whether they are optimal, or not. The net result is that, owing to the fundamental nature of reinforcement learning, the collision avoidance strategy structure is extremely simple, except for its reliance on a very large lookup table that challenges the information storage capabilities of several aircraft currently in use. The high assurance required from the software that supports ACAS-X requires such software to meet the "level B" qualification, only a notch below "level A" qualification, which addresses essential software necessary to operate the basic airplane functions, including flight, take-off and landings, safely. Indeed, "Level A" qualification addresses all flight control software [40]. Verification and validation are essential elements to validate ACAS-X and other complex, safety-critical systems. On the one hand, there is the need to verify that the software functions appropriately (absence of bugs etc). The practice of proper software development and its verification are very much discussed in DO-178C and its extensions and this practice is considered to be standard, even for a relatively new system like ACAS-X. On the other hand, there is the validation issue: Does ACAS-X perform its function as intended? For that purpose, there have been numerous efforts leading to a number of corrections and updates to ACAS-X. Examples include the formal validation efforts of Jeannin and Platzer [41, 42]. Efforts at reducing the memory requirements of ACAS-X have also led to validation efforts for ACAS-X derivatives encoded as deep neural networks [43]. As surprising as may be, we claim that such formal and extensive validation efforts are *not* always necessary for such systems to properly embody *Ariadne*, as will be discussed below.

B. Assurance of *Ariadne*-inspired systems

Plan B assurance is considered to be one of the key tenets of safe autonomy, and constitutes the core object of various programs articulated around the "safe autonomy engineering", funded through multiple current research programs. As such, there are multiple efforts going on and described in a multitude of papers, see [44, 45]

Probably one of the most interesting developments, independent from designing the Plan B itself, is the safety of the various switching mechanisms that allow the vehicle to "switch" from one mode to another mode, say from "nominal mode" to "backup mode".

In that context, Copilot is a runtime verification framework for real-time embedded systems. Copilot monitors are written in a compositional, stream-based language with support for a variety of logics including Temporal Logics, which results in robust high-level specifications that are easier to understand than their traditional counterparts. The framework translates monitor specifications into C code with static memory requirements, which can be compiled to run on embedded hardware [46, 47].

Copilot monitors are software and the researchers, notably at NASA are investigating how to ensure the code generated could be certified say under NASA's level C or B classification. Copilot was intended to minimize the

instrumentation of the system so it samples specified system state either C global variables or data provided by middleware software bus.

Galois, through a contract with DARPA's Assured Autonomy program, is leading a small group that is using Copilot to verify the composition of small controllers. They have now integrated SMT solvers into the framework so you can check simple properties of say three monitor specs to make sure they don't miss coverage or interfere with each other.

III. Verification in *Ariadne*

The issue of verification is *central* to *Ariadne*. Indeed, *Ariadne* aims at formalizing and extending a framework essentially built around a human infrastructure, whose safety has been proven by many years of practice and the intervention of safety specialists, often called Designated Engineering Representatives (DERs), mandated by the FAA to assure the safety of all products built by industry. It is therefore essential that *Ariadne*'s safety meet the same safety standards as it aims at including automation either in the air, on the ground, of both.

A. Verification vs. Validation vs. Certification

In systems engineering, verification is the step where some element of a system performs according to its specifications. Verification is not sufficient to guarantee that the whole system meets its functional requirements, but it is considered to be a necessary step towards achieving that goal, named validation. Verification is often brought together with validation as part of the overall process of certification. Several differences exist between these steps. Unlike verification, which focuses on individual elements of a system only, validation is about establishing reasonable evidence that the assembled system meets its functional requirements. Verification and Validation are usually highly technical tasks that rely on precise engineering analysis techniques to be established. Certification is the final and official stamp of approval of the Air Navigation Services Provider (ANSP) or similar authority before operational use, based on the foregoing verification and validation steps. Certification is not a technical activity.

Verification is usually done at the component level and may be performed repeatedly at various levels of implementation of that element, from concept to full implementation. In the case of Plan B strategies the concept is typically described by means of block diagrams, whereas implementation might be a mix of software and communication systems. It may also simply be a procedure executed by a human, where the human, together with the procedure form the component to be verified. Several verification procedures exist that are adapted to each kind of component, whether they are structural components, propulsive components, electronic components, or software component. The latter, perhaps with a computer architecture, is that of concern to *Ariadne*. Component verification is guided by several reference documents published by organizations, such as RTCA (Radio Technical Commission for Aeronautics) and SAE (Society of Automotive Engineers). In the case of airborne software/hardware components, a reference document is SAE's ARP-4761A [48] which pertains to system and subsystems safety assessment [49]. The ARP 4761A then refers to a number of other reference documents [40, 48, 50–52] that consider lower levels of safety verification and validation shown in Fig. 5. For ground equipment and procedures, guidance documents are fewer. However they do exist, see [53, 54] However, it is most often possible to apply guidance documents for airborne equipment to ground-based operations. Indeed, airborne systems usually must meet higher safety requirements than ground-based systems. Other variations include the differing roles of humans in the cockpit rather than those on the ground.

1. Formal verification in *Ariadne* : *Guaranteeing safety with unproven components*

Mathematical formalism paves the way of many engineering projects to success, and aerospace systems are no exception to this fact. Rigorous mathematical formalism is presented in civil aviation and offers a powerful and expressive language to specify not only avionics systems, but also structural and aerodynamic characteristics. In software-intensive components, computer-assisted mathematical verification is named "Formal Methods" and is present as a supplement, RTCA DO-333 [55] to RTCA DO-178C [40]. Formal, mathematical analysis of software/hardware integrated systems at the functional level is less present, other than as they relate to control systems. As a source of inspiration, the document developed by ASTM-F3269-17 [56] offers an approximate pathway to the embodiment of *Ariadne* in the context of introducing Artificial intelligence and, in general, "unproven algorithms" in air operations. Perhaps one of the most surprising aspects of that embodiment is the significant and genuine possibility of reducing the verification requirements that come with it, in fact far below those expressed in ASTM F3269-17. In essence, two philosophies can be applied to the Plan B mindset.

On the one hand, Plan B may be nominally inactive and be only called upon when it is necessary to do so. In that

Figure 13. ARP 4754A Document Interrelationship

Figure 5 A map of guidance documents relative to embedded aerospace software. Source: NASA [49]

case, the recommendations present in ASTM F3269-17 do apply, and the backup control strategies discussed there must meet stringent safety requirements and form prime candidates for the exhaustive and formal verification methods commonly seen in "classical" control systems and that are slowly making their way into AI-driven controlled systems.

On the other hand, the proper *Ariadne* embodiment is one where Plan B is *always active* and there is *always a safe alternative* that is ready to step in for each likely failure, including combinations thereof. Moreover, the scenarios leading to triggering the run-time assurance mechanism *include cases when whatever algorithm in charge of producing Plan B fails to do so*. As a result, it becomes entirely possible to *verify Ariadne* embodiments, although these embodiments only rely on unverified, and perhaps unverifiable algorithms, such as the numerous black-box algorithms that usually are the preferred product of subcontractors who are keen on keeping control over their industrial property, or organizations that are keen on maintaining human supervision of traffic operations. TO build a concrete idea of how *Ariadne* is embodied into a verified path planning unit with unverified components, we consider the situation shown in Fig. 6. An aircraft is attempting to pass through a hole in a squall line. The aircraft is shown at times t_1 , t_2 , and t_3 . The aircraft is color-coded with corresponding colors, red for t_1 , light blue for t_2 and green for t_3 . The trajectory followed by the aircraft is shown in purple. The dashed part of the trajectory represents "a future trajectory" and may be unknown. It is assumed that the aircraft has a path planner (either in the mind of a human or installed electronically). This path planner is a "black box", that is, its function is known, but it may or may not return an answer within a prescribed amount of time. It is only assumed that the black box works "most of the time" and requires computations that last only a fraction of the durations $t_2 - t_1$, $t_3 - t_2$, and $t_4 - t_3$. We initially assume the aircraft is at its initial position at time t_1 and that a feasible trajectory is available from t_1 and t_2 , *together with a backup trajectory*, shown in red. That red trajectory is a "safe" trajectory because it ends in a racetrack pattern, which can be followed as long as the aircraft does not run out of fuel. At time t_1 , the *nominal* trajectory of the vehicle is the segment from t_1 to t_2 concatenated with the red backup trajectory. The time segment $[t_1 t_2]$ is then used by the pilot (human, but increasingly autonomous) in the case when Visual Flight Rules (VFR) apply or by ground-based Air Traffic Control, in case of Instrument Flight Rules (IFR) apply, to determine whether (i) the segment $[t_2 t_3]$ is flyable and (ii) whether a new trajectory resulting in a safety trajectory ending up with racetrack beginning at t_3 can be found, my means of computer or human intellectual effort. If not, then then the aircraft engages into the red backup trajectory. If yes, then the aircraft has obtained its "ticket" to proceed with the point t_3 . The same process repeats from then on.

This story, with its imperfect realism, shows what the core mechanism allowing an aircraft to fly safely is. That mechanism is proven safe by means of the way it is constructed: While part of the planned trajectory is "useful" from the perspective of getting to destination, much of the planned trajectory is a safe "recovery trajectory". It is often the

Figure 6 An aircraft passing through a hole in a squall line. In this simplistic scenario, weather perturbations are assumed to be non-moving. Many other simplifications have been made deliberately for clarity purposes.

case that pilots manually handle the aircraft, in which case a computerized system may have no information available to compute a backup, safe trajectory from a short-term path prediction. One solution to recover some level of trajectory predictability is to narrow the length of the intervals $[t_1 \ t_2]$, $[t_2 \ t_3]$ and $[t_3 \ t_4]$. The trade-off that needs achieving is then the necessity to increase the "tempo" at which "Plan B"s need computing, which may or may not be possible. However, commercial products, such as Xavion, a tablet app designed to back up the avionics of general aviation aircraft and provide emergency routings to the nearest airport in case of engine failure are a strong indication that the more general concept described in this paper can also be implemented [57]. Many westbound, Europe-America transatlantic flights perfectly embody *Ariadne*'s philosophy, as shown in Fig. 7. Indeed, these flights initially *do not* file a flight plan to their final destination, but their final destination is Gander a popular alternate airport located in Newfoundland. Once close to their destination, they then amend the flight plan to continue to their destination. The reason for this procedure is fuel savings: Although most flights are fueled well-enough to reach their final destination, aeronautical regulations do not consider direct flights to their destination to be safe enough with the allotted fuel. That allotted fuel is, however, "legal" to support a flight to Gander and, once close enough, whatever fuel remains usually become "legal" to proceed to destination. This example illustrates particularly well the inverted way of planning, whereby the nominal trajectory is the safety trajectory, and the actual trajectory needs to be "earned" prior to proceeding with it.

From the perspective of the authors' past work, the procedures and guaranteed safety outlined in the foregoing can be traced as far back as 1997 [58], where a procedure to control nonlinear systems was designed by introducing a sequence of Lyapunov functions and corresponding invariant sets, and control from one trim point to another was ensured by aiming at one equilibrium, only to switch to the next equilibrium as soon as getting within its region of attraction, as shown in Fig. 8.

This initial concept, which will undoubtedly remind the reader of Extended Twin-Engine Operations (ETOPS) procedures, was later refined by Frazzoli *et al* [59] to use the approach for arbitrary path planning tasks using randomized, and therefore eminently unverifiable algorithms, including a safe slew attitude planning for space telescopes [60], see Fig 9. Last, such "Plan B" engineering approaches were implemented in an actual flight test implementing an early version of the "loyal wingman", whereby a manned F-15 and a large sized UAS were operating together. The large UAS was guided by a real-time implementation of a mixed-integer program, and always maintained a "Plan B" backup loitering pattern in case of computational failures, which happened more than once, as shown in Fig. 10.

Figure 7 Typical transatlantic flight from CDG to JFK. Initial flight plan lands the aircraft in YQX (Gander). Flight plan is amended close to YQX to continue onto JFK, its final destination

Figure 8 Hopping from equilibrium to equilibrium: Plan A is a succession of Plan Bs [58].

Figure 9 Hopping from equilibrium to equilibrium via randomized path planning: Plan A is a succession of Plan Bs [59].

Figure 10 Planning for loitering. Blue: F-15, Yellow: J-UCAS [21]. J-UCAS proxii (an instrumented T-33) flew autonomously.

2. Starting and ending "Plan B" for aircraft

The way embodiments of *Ariadne* can be designed to effectively support the air traffic control system is predicated on common sense, existing air traffic control rules, a few axioms, and accurate mathematical models of aircraft and aircraft control operations. One of the axioms needed for completeness is the existence of a state, or a set of states, where passenger and crew safety can be held indefinitely. We choose the set of states to be the apron, where the aircraft can be safely unloaded/loaded, boarded/de-boarded, serviced, and fueled up. The apron may be close to a boarding gate or not, as is frequently the case in non-US airports. It also provides a good initial condition to insert the aircraft into airspace operations, since it is inherently equipped with a backup trajectory, that is, remaining stopped at the apron. The reader can imagine equivalent "absolutely safe states" for unmanned or remotely operated vehicles. The *Ariadne* issue as seen from the aircraft cockpit is then to keep earning appropriate "Plan B"s throughout the execution of the flight, from departure apron to arrival apron as shown in Fig. 11. If some parts of the flight do not have a "Plan B" or do not have enough "Plan B"s, that is conceivable as well as long as operators are conscious about the absence of such plans as well and the risk of system failure may be properly quantified.

Figure 11 Typical aircraft flight cycle (modified from [61])

3. Validation needs for "Plan B engineering"

The fundamental difficulty of "Plan B engineering", that is, the need for model-based safety validation, has been trivialized by the foregoing. However, there remain significant validation efforts to be done in order to make the "Engineered Plan Bs" appropriate for insertion in the air transportation landscape. For example, one of the most important validation elements is managing the number of false alarms. False alarms are a highly subjective measure of performance that rates the "Plan B" potential to trigger for inappropriate reasons, and, although false alarms do not constitute immediate threats to safety, they can, especially when human operators are present, become the source of multiple frustrations and lower attention that can lead to behaviors that compromise safety. Fortunately, the evaluation of false alarms may not require sophisticated mathematical tools and may be performed via simulations only, possibly with a human in the loop whenever relevant [62].

Another important factor is the development of "Plan B"s that do not contradict existing procedures, especially those applied by humans on existing traffic. One example of conflict between procedures that necessitated reconciliation between several human parties after generating great controversy is "Land and hold short" (LAHSO) operations at busy airports, whereby aircraft are offered to land, under the condition that they hold short of another runway operated for full-length landing or take-off operations, or a busy taxiway with aircraft crossing the active runway, as shown in Fig. 12. During the late 1990s and early 2000s, the FAA entered into contentious discussions with airline pilots about LAHSO in the context of the urgent need to increase airport capacity. This was especially the case for Chicago O'Hare International Airport. Back then, the airport consisted of several crossing runways, and LAHSO [64] was seen as an essential mechanism to improve capacity without "pouring more concrete". American Airlines and its APA pilot union entered into heated discussions with the FAA over the well-foundedness of LAHSO, in particular in the context of

FIG 4-3-8
Land and Hold Short of an Intersecting Taxiway

FIG 4-3-7
Land and Hold Short of an Intersecting Runway

Figure 12 Land and hold short examples. Left: busy taxiway crossing active runway. Right: Two intersecting runways [63]

how certain LAHSO could *de facto* remove some of the available "Plan B", such as those necessary to handle rejected landings. Later incidents supported American Airlines' position, see for example [65] where an aborted landing resulted in a near miss in Canada.

It must be noticed that the practice of "sacrificing" available "Plan B"s, as perceived by the APA for the purpose of increasing system capacity is not unique to aviation and it is, in fact, far more frequent in ground transportation: For example, it is difficult for drivers not to remember situations where other vehicle operators engaged in maneuvers and behaviors that clearly demonstrate the absence of Plan B on their part. It is often a lot harder for drivers to realize they, themselves, engaged into such activities without a Plan B in mind. In addition, it is commonplace for highway operators in the US to run road traffic in the emergency lanes during rush hours although these lanes are considered an important resource for drivers to find a safe haven in case of vehicle breakdown, thereby showing that economics sometimes prevails over safety concerns.

B. Clever "Plan B"s architectures that cover many failures, including their own: The Westinghouse breaking system

One of the key attributes of *Ariadne* is the necessity to make the time of its triggering "Plan B" as small as possible when it becomes necessary and making the probability of its execution as close to one as possible. Thus, and perhaps paradoxically, the system under operation should always be "very close" to triggering "Plan B", close enough that the state of "Plan B" triggering be accessible with as few logical operations as possible, on the one hand, and that be true over the largest number of faulty system states. In other terms, "Plan B" triggering must be "robust" against as many external factors as possible.

The history of emergency braking in railroad systems offers a beautiful example in case, discussed thereafter. For a detailed story, see [66] The preferred braking mechanism for trains is *air braking*, whereby the locomotive generates pressured air that travels through an elaborate piping system all over the train that directly accesses the individual brakes of every car.

A naive implementation of the air brake system would consist of keeping atmospheric air pressure inside the piping system during cruise operations, resulting into brakes not being applied. During braking and emergency braking, pressure in the pipe system would then go up and brakes would be applied via a pistons connected to the now pressured pipe system. In 1869, George Westinghouse patented a fail-safe train braking system, whereby the *normal* state of the pneumatic system is to be under high pressure, and the *abnormal* state is for the same piping system to be at atmospheric pressure. A notional view of the classic vs. Westinghouse brake system is shown in Fig 13. The left arrangement shows how the brake pad is pressed against the wheel when pressure is applied. The right arrangement shows that

Figure 13 Classical (left) vs. Westinghouse airbrake (right)

in the absence of pressure applied to the piston, the brake pad is pressed against the wheel by the spring, which is under tension, and breaking action occurs. Releasing the brake requires pressure to be applied for the brake pad to be released. The actual Westinghouse airbrake is somewhat more complex because the extended spring is replaced by a complementary pneumatic system. The effect, however, is the same. The advantages of the Westinghouse arrangement are not obvious at first. However, Westinghouse brake system offers remarkable safety characteristics, such as the automating triggering of the brake for any event affecting the train integrity. For example, separation between any two train cars will result in damage to the pneumatic line, thus automatically reducing pressure applied to all train brakes and triggering strong braking action. Cornelius Vanderbilt, a train owner and operator opposed the introduction of Westinghouse's invention in the train stock he owned. It is one of history's ironic twists that the university named after him now leads and promotes one of the largest research efforts in autonomous system safety, see [67].

Robust, "built in" safety mechanisms do carry over in aviation applications. For example, the fuselage door system in many Boeing aircraft is arranged so it is passively stable, a feature that is not shared with many Airbus aircraft, whose doors are secured against the fuselage via a more complex system of latches. Other examples include almost all aircraft control surfaces: The "natural equilibrium" of control surfaces mounted on trailing edges of wings, stabilizer and tail fin such as rudder, ailerons, and elevators is about zero deflection angle, which ensures some level of safety that would not be matched by control surfaces mounted at the leading edge of the the same aircraft elements.

C. Adaptability and flexibility in *Ariadne*

Air transportation specialists, be they air traffic controllers or pilots, often maintain not one, but several backup plans in mind. Sometimes the reason for doing so is for safety: One backup plan alone might not be enough to guarantee safety. For example, pre-flight planning may include the definition of several alternate airports throughout the trip so as to make sure one of them remains accessible despite local convective weather. Another reason might be because of the temporary nature of the safety offered by "safe conditions". For example, consider the safety afforded by introducing a racetrack pattern, such as shown in Fig 6. Racetrack uses often induces pilot concerns about fuel reserves and increased use of radio bandwidth to discuss these concerns. A conceivable approach to handling these concerns is to always include a genuine "forever safety state" in the RTA, such as an alternate landing site, not for the purpose to ever using this site, but in order to address the safety-oriented part of the concern and convince the pilot of a genuinely realistic "last resort solution", as illustrated in Fig. 14

IV. *Ariadne* and automation

The range of applicability of *Ariadne*, because it ALREADY pervades the current air transportation system, is very wide. It can be thought of in the context of current operations. It can also be thought of as a healthy basis upon which the future evolution of the air traffic control system may be considered.

A. Initial possibilities

Considering today's air traffic control, *Ariadne* can be embodied as a mechanism to assist air traffic controllers and pilots with current traffic control and management and decision-making. In its simplest form, *Ariadne* may simply be embodied as a decision aid, whereby air traffic controllers focus on mission execution, while one embodiment of *Ariadne* maintains "backup option situational awareness". For example, air traffic control positions could be equipped with supporting software that computes complete safety trajectories in real-time. The same kind of services may be made available to pilots, perhaps as an additional display, and only after a thorough Human Factors evaluation. When the aircraft is on autopilot, this functionality could act as an alarm system indicating a change in "Plan B". One implicit or

Figure 14 An example of cascaded RTA mechanisms allowing complete model-based safety

explicit objective to seek would be to develop "Plan B"s that minimize the conservatism of air transportation operations. Indeed, it is known that the required care for air transportation safety, and the necessity to develop "Plan B"s, leads to a much larger use of airspace, both in space and time, than a naive look at air traffic control screens suggest.

B. A solid basis for moving the system forward

In future operations, *Ariadne* can guide the air transportation system evolve towards increased system productivity and reduced inefficiencies.

Beyond this, there is also the possibility of fully automating some air traffic control functions by implementing both "Plan A" and "Plan B" functions according to the foregoing principles. With an optimistic perspective, it is possible to imagine an entirely automated sector control activity. However, obtaining the proper certifications will include a very close evaluation of the resulting embedded software, on the one hand, and the development of appropriate "letters of agreement", many of which are already written with *Ariadne* in mind [68]. The complete system may then, barring unknown other obstacles, be accepted for operations.

It also becomes possible to consider remotely piloted flights while interacting with human-operated traffic. Quoting the Aviation Week Network issue of December 3, 2020: "The vision: unmanned cargo aircraft plying the skies to meet the ever-growing express logistics needs of the e-commerce giants. The reality: a pair of startups that are converting the Cessna Caravan, see Fig. 15, into a remotely piloted regional cargo aircraft as a first step." New issues and possible failures arise when introducing RPAS (Remotely Piloted Aircraft Systems). One such new failure mode is the loss of link between the RPAS ground operator and the vehicle, thereby also creating a loss of link between the ground operator and air traffic control. Several "Plan B"s may be considered that require close discussion between manufacturers/operators and air traffic controllers. Consider for example the two following Plan B:

- 1) Upon loss-of-link, make a U-turn and perform a racetrack pattern at a location where link was active.
- 2) Upon loss-of-link, continue flight plan execution in autonomous mode.

Upon discussion with the Reims, France, Air Route Traffic Control Center (ARTCC), the second solution was deemed far more preferable than the first solution, although both looked "reasonable" at first. One of the key reasons was the need for high RPAS behavior predictability, which, during this interview, surpassed the desire for recovering communication with a human pilot immediately. Recent news, such as the approval of Garmin's autoland system [70] provide a promising way to, indeed, implement continued flight plan conformance by an RPAS. It is worth noting that the implementation of any "plan B" will automatically account for the gradual improvement of communication capabilities offered, for example, by 6G communication systems and, consequently, the reduction of occurrences of

Figure 15 Xwing's Cessna Caravan RPAS under test [69]

loss-of-link. Below a certain reliability threshold, the "Plan B" obligation may then be eliminated.

Ariadne makes it also possible to consider formation flight by a group of commercial aircraft from a rational and implementable perspective. Formation Flight involves aircraft flying in close formation, with aircraft flying precisely into each other's wake for reduced fuel consumption and increased range, which might be critical to the early introduction of all-electric or hydrogen aircraft over medium and long-haul flights. Such operations can be implemented only if appropriate "Plan Bs" can be developed to handle likely failures, such as loss of engine. Such "Plan B"s may be derived from similar procedures currently in use during military formation flight operations.

V. Computational issues

One of the benefits of *Ariadne* is the attractive computational options that it offers, together with relatively easy software validation requirements. However, the foregoing arguments indicate that, since the default mode of system operation is in "safety mode", the challenge remains with computing the next "ticket" towards achieving a later "guaranteed safe" trajectory. And if available computational time to find the next solution is too short, the vehicle will *have to* engage in its currently available safe trajectory, keeping in line with the policy that "Plan B" computation failures are treated like any failures and, perhaps surprisingly at first, can be taken care of by the Plan B strategy that *already exists*. Alternatively, simpler "Plan B" decision spaces may be chosen instead that enable shorter computation times. A good example in point is "last minute" collision avoidance, that falls within the current range of airborne collision avoidance systems. The need for a high refresh rate of "Plan B" trajectories in order to limit the number of safe but premature collision avoidance maneuvers may initially imply reliance on a small maneuver portfolio. As of today, this portfolio consists of a small number of vertical maneuvers that have been optimized, not only for safety, but also for minimizing their impact on the intended trajectory of all aircraft involved. As embedded computer power increases and aircraft fleets modernize following recommended safety management practices such as those discussed in SAE ARP 4761, ARP 4754, and RTCA DO-178C among others, it can be foreseen that collision avoidance will be eventually handled with a much broader array of available maneuvers, possibly using artificial intelligence techniques, but also commercially available models and algorithms from Operations Research, such as mixed-integer linear programming, already demonstrated in part by Feron *et al* [71] and detailed in Schouwenaars' PhD thesis [72].

VI. Operational challenges and Recommended approach for progressive implementation in current ATM environments

One of the foreseen challenges is the eventual necessity of closer coordination between cockpit, dispatch and air traffic control/management [73].

Indeed, *Ariadne* is fundamentally based on maintaining a number of safe options for ground and cockpit operators alike. Keeping in mind the importance of ground control in ensuring the smooth flow of traffic and absence of collisions, on the one hand, and pilot ultimate responsibility for aircraft integrity, on the other hand, it could be imagined that these shared responsibilities be integrated somehow in the future, which would then require intense information sharing.

One possibility is to complement the existing ADSB-out system, which is a mechanism for aircraft to communicate with ground control many data, including position data as a replacement for traditional Secondary radar technology. In the United States, there are two methods for achieving ADS-B Out. One is using the next generation of transponders operating on the 1090 MHz band. The other is using a new technology called Universal Access Transceiver (UAT). Another possible information sharing mechanism might be a "safety-critical" version of SWIM, the System-Wide Information Management operation initiated in Europe and adopted by the FAA and ICAO. It is believed that the most natural way to evolve the communication system to meet the needs of *Ariadne* Conops will be via concurrent evolution of the entire CNS infrastructure, especially as enabled by NextGen in the US and SESAR in Europe.

VII. Operational challenges and Recommended approach for new ATM environments

New ATM environments can be characterized by two different evolutions: First, there are changes in Concepts of Operations. Examples include the transition to Free Flight [74], which is now implemented in Poland, see [75] and the Maastricht air traffic management center, see [76] The existence of few, but operational implementations makes the two foregoing examples prime locations where *Ariadne* may be tested experimentally.

Second, there are changes in the type of vehicles involved. In particular, there are very significant efforts currently aimed at urban air mobility, on the one hand, and plain unmanned aviation, on the other hand. Urban air mobility involves a wide-ranging number of vehicles, ranging from minuscule machines, such as DJI's Mavic Mini shown in Fig. 16, on the one hand, to Super Jumbo jets, on the other hand. Nowhere else than in Hong-Kong's old airport is this

Figure 16 The DJI Mavic mini is an example of micro-drone capable of high quality imagery

situation better illustrated, where large aircraft used to land and take-off in urban environments as shown in Fig. 17. While Kai Tak airport is now closed, slightly less dramatic but similar situations remain at urban airports, such as London City airport in the UK and Reagan National airport in the US.

New environments where large unmanned vehicles interact with classical commercial traffic face several issues, some having to do with vehicle integrity, and some having to do with its interaction with the surrounding environment. In the latter case, unmanned traffic faces unique safety requirements that must be translated into specific *Ariadne* provisions. Consider for example issues of engine failures, typical of single-engine autonomous aircraft. The requirement for the vehicle not to crash in unequipped areas could be handled by adapting software, such as Xavion.

Consider then issues of loss of communication link: Indeed, most unmanned air vehicles today are in fact remotely piloted vehicles (RPVs), and examples of such systems include General Atomics' Predator and its successors, and autonomous Cessna Caravans developed by Xwing. Communication link integrity currently constitutes one obstacle to certification. *Ariadne* may, however, offer a valuable option to handle that problem.

VIII. Keeping a systems-oriented approach

One of the key elements of *Ariadne* is the necessity to follow a comprehensive, system-oriented approach to its multiple embodiments within the air transportation system. Any modification of the concept of operations may result in ripple effects affecting pilots in the cockpit, air traffic management specialists at air traffic control centers, airports and flow control facilities, dispatchers at airline operations centers, and other distinct entities even. We believe that the *Ariadne* paradigm is capable of not affecting the current arrangement of functions of the air transportation infrastructure. However, we also believe that the implementation of some of the *Ariadne* functionalities, such as full safety trajectories

Figure 17 Hong Kong urban aerial mobility environment included a Boeing 747 when Kai Tak International Airport was operational [77]

as illustrated in Fig. 14, may require modifications to the current paradigm. Indeed, a full safety requirement considers not only the aircraft along with airborne "Plan B trajectories", such as S-turns and loiter patterns, but they must be followed by an airport approach and a landing for completeness. On the one hand, this completeness requirement is, we believe, a necessity for such a tool to be implemented. On the other hand, it requires increased cross-communication and joint action (thus multivariable processing) to be implemented in today's air traffic environment. For example, today, loitering patterns are decided at the discretion of the air traffic controller. However, the completion of the safe trajectory up until an alternate landing site would have to be encoded as an alternate flight plan that may need updating in real-time, possibly forcing a tempo and information flow that may be hard for a human operator on the ground to digest. Alternatively, the responsibility for alternate flight plan awareness may fall upon the pilot and the cockpit. But then the cockpit would need to know about the loiter pattern decided upon by the air traffic management. It is worthy to note that using loiter patterns constitute a point of friction between ground control and cockpit in many cases, because pilots DO worry about their fuel autonomy. Thus, rather than raising a new issue unknown to air traffic control today, our framework makes an existing "pain point", and why it is difficult to address, more explicit. *Ariadne* fully supports other pain points between ATM and the cockpit raised by previous accidents, such as the foregoing Überlingen collision, and the follow-up modifications brought to TCAS, see Fig. 3.

IX. Deterministic vs. probabilistic frameworks and how to handle a stochastic system

In the mathematical world underlying the present discussion, deterministic reasoning is implicitly used for the sole purpose of simplifying the discussion. In actual operations, and even in a very professional and regulated environment, the presence of many uncertainties make it absolutely necessary to consider a probabilistic framework instead. Such a probabilistic framework, coupled with the necessity to collect extensive statistical data, will undoubtedly create considerable challenges to the reliance of *Ariadne* to support the design of future air transportation operations. These challenges are of the same nature as those encountered by any new development today. However, *Ariadne* brings forward the quasi certainty that these efforts will lead to a successful conclusion unlike most prior efforts aimed at automating the air transportation system and accepting unmanned aviation, for example.

X. A program to implement *Ariadne* in everyday operations

A. Acts of God

Ariadne is predicated on common sense and a comprehensive knowledge of all past mishaps and corrective actions. However, *Ariadne* is *not* predicated on unexpected mishaps, for the authors reject the common belief that intelligent systems may be so smart as to anticipate unknown unknowns, regardless of technological progress in Artificial Intelligence and other disciplines. As a consequence, *Ariadne* does not claim anything could have been done to TCAS in order to prevent the Überlingen accident. On the other hand, *Ariadne* would have led to the prevention of the well-known conditions at the air traffic control center (facility under repair, air traffic control position under-staffing and other incriminating factors) the same way as any other process would have, thereby saving the lives of countless passengers and one air traffic controller.

B. Incorporating Lessons learnt: a continuously learning system

Consistent with the foregoing, the best embodiment of *Ariadne* should include all new "acts of God" as soon as they are identified for the purpose of making them very difficult to occur again. Ideally, the process of incorporating these acts of God into the concepts of operation inspired by *Ariadne* and determining which operations these acts of God are most relevant to account for should be automated if possible. Certain strong difficulties remain, however. Among them is the translational learning that occurs among many humans, whereby experiences gathered by someone in a given operational environment (eg ground vehicle driving) translates into valuable and proactive experience into another operational environment (eg air transportation operations). Such knowledge transfer, which happens "automatically" in people, still constitutes a research challenge whose solution would enhance the quality of some of the embodiments of *Ariadne*, but whose absence would not initial development activities.

C. The system in the ecosystem

The implementations of *Ariadne* should also steadily keep into account the evolution of its surrounding environment. To begin with, the entire existing air traffic management system, including both cockpit- and ground-based systems, should be reviewed in detail to validate *Ariadne*'s as the thread that drives its operations. Then, *Ariadne* should evolve according to technical changes in Communication, Navigation, and Surveillance mechanisms, as well as generational changes among pilots, air traffic controllers, and dispatchers. Indeed, the embodiments of *Ariadne* will always operate in closed-loop with its ecosystem, and any evolutions of the embodiments of *Ariadne* or its ecosystem may destabilize the resulting closed-loop system.

XI. Conclusion

While *Ariadne* should underlie any research effort in air transportation operations, it comes only as one element to be incorporated in design considerations, first and foremost human factors, existing procedures, and task partitions across different operational units. *Ariadne* is essential in that many of the perceived verification needs for Plan B procedure acceptance are trivialized, and the validation requirements do not require as many formal proofs as needed. In essence such needs do not exceed those associated with evaluating the system for false alarms. *Ariadne* constitutes a flexible, ubiquitous, and adaptable thread that should support any innovation for existing air traffic control operations to handle unmanned traffic and relaxed routing rules, and any design for new air traffic control operations.

References

- [1] Flowers, J., and Shalgar, P., *The incredible killer tree that broadcasts an SOS to its neighbours.*, 2015 (accessed December 7, 2020). URL <https://uk.blastingnews.com/world/2015/10/the-incredible-killer-tree-that-broadcasts-an-sos-to-its-neighbours-00610971.html>.
- [2] Pisano, R. G., and Storer, T. I., "Burrows and Feeding of the Norway Rat," *Journal of Mammalogy*, Vol. 29, No. 4, 1948, pp. 374–383. URL <http://www.jstor.org/stable/1375126>.
- [3] Sapolsky, R. M., "Stress in the wild," *Scientific American*, Vol. 262, No. 1, 1990, pp. 116–123.
- [4] Greek Mythology, *Ariadne*, 2020 (accessed December 7, 2020). URL <https://www.greekmythology.com/Myths/Mortals/Ariadne/ariadne.html>.

- [5] Coley, M. D., Maynard, I., Seyed Alavi, K., and Chudoba, B., “Ariadne: A Space Program Forecasting and Planning Decision Support Tool for Strategic Planners,” *2018 AIAA SPACE and Astronautics Forum and Exposition*, 2018, p. 5298.
- [6] Jet Propulsion Laboratory, *43 Ariadne (A857 GA)*, 2003 (accessed December 7, 2020). URL <https://ssd.jpl.nasa.gov/sbdb.cgi?sstr=43>.
- [7] Neudecker, D., “ARIADNE—a program estimating covariances in detail for neutron experiments,” *EPJ Nuclear Sciences and Technologies*, Vol. 4, No. LA-UR-17-30469, 2018.
- [8] Hargitai, H., Wang, J., Stooke, P., Karachevtseva, I., Kereszturi, A., and Gede, M., *Map Projections in Planetary Cartography*, 2017. https://doi.org/10.1007/978-3-319-51835-0_7.
- [9] Cristello, S., *Ariadne’s Thread: The Needle, The Haystack, The Thread // Arts Club of Chicago*, 2018 (accessed December 7, 2020). URL <https://thesejournal.org/ariadnes-thread/>.
- [10] Jensen, N. M., *Philosophy in labyrinths*, (accessed December 7, 2020). URL http://www.lavigne.dk/labyrinth/ela_phil.htm.
- [11] Stieger, A., *Myth and Creativity: Ariadne’s Thread and a Path Through the Labyrinth*, 2014 (accessed December 7, 2020). URL https://www.creativitypost.com/article/myth_and_creativity_ariadnes_thread_and_a_path_through_the_labyrinth.
- [12] Langford, J. S., “Triumph of Daedalus,” *National geographic*, Vol. 174, No. 2, 1988, pp. 191–199.
- [13] Chandler, D., *Record-breaking Daedalus project marks 20th anniversary*, 2008 (accessed December 7, 2020). URL <https://news.mit.edu/2008/daedalus-0422>.
- [14] Conner, M., *Daedalus Human-Powered Aircraft*, 2017 (accessed December 7, 2020). URL <https://www.nasa.gov/centers/dryden/multimedia/imagegallery/Daedalus/EC87-0014-8.html>.
- [15] Drela, M., “Low-Reynolds-number airfoil design for the MIT Daedalus prototype-A case study,” *Journal of Aircraft*, Vol. 25, No. 8, 1988, pp. 724–732.
- [16] *Contes de Perrault / Le Petit Poucet*, (accessed November 9, 2020). URL [https://fr.wikisource.org/wiki/Contes_de_Perrault_\(%C3%A9d._1902\)/Le_Petit_Poucet](https://fr.wikisource.org/wiki/Contes_de_Perrault_(%C3%A9d._1902)/Le_Petit_Poucet).
- [17] Opie, I., and Peter, “The Classic Fairy Tales,” *Oxford University Press*, 1974, p. 21.
- [18] Rivera, J., Danylyszyn, A., Weinstock, C., Sha, L., and Gagliardi, M., “An Architectural Description of the Simplex Architecture,” Tech. Rep. CMU/SEI-96-TR-006, Software Engineering Institute, Carnegie Mellon University, Pittsburgh, PA, 1996. URL <http://resources.sei.cmu.edu/library/asset-view.cfm?AssetID=12521>.
- [19] Clark, L., “MIT team guides airplane remotely using spoken English,” *News Office journal, Massachusetts Institute of technology*, 2004.
- [20] Schouwenaars, T., Valenti, M., Feron, E., How, J., and Roche, E., “Linear programming and language processing for human-unmanned aerial-vehicle team missions,” *Journal of guidance, control, and dynamics*, Vol. 29, No. 2, 2006, pp. 303–313.
- [21] Schouwenaars, T., “Safe trajectory planning of autonomous vehicles,” Ph.D. thesis, Massachusetts Institute of Technology, 2006.
- [22] Gurriet, T., Singletary, A., Reher, J., Ciarletta, L., Feron, E., and Ames, A., “Towards a framework for realizable safety critical control through active set invariance,” *2018 ACM/IEEE 9th International Conference on Cyber-Physical Systems (ICCPS)*, IEEE, 2018, pp. 98–106.
- [23] Gurriet, T., Mote, M., Singletary, A., Nilsson, P., Feron, E., and Ames, A. D., “A Scalable Safety Critical Control Framework for Nonlinear Systems,” *IEEE Access*, Vol. 8, 2020, pp. 187249–187275.
- [24] Ames, A. D., Coogan, S., Egerstedt, M., Notomista, G., Sreenath, K., and Tabuada, P., “Control barrier functions: Theory and applications,” *2019 18th European Control Conference (ECC)*, IEEE, 2019, pp. 3420–3431.
- [25] Squires, E., Pierpaoli, P., and Egerstedt, M., “Constructive barrier certificates with applications to fixed-wing aircraft collision avoidance,” *2018 IEEE Conference on Control Technology and Applications (CCTA)*, IEEE, 2018, pp. 1656–1661.
- [26] Ames, A. D., Xu, X., Grizzle, J. W., and Tabuada, P., “Control barrier function based quadratic programs for safety critical systems,” *IEEE Transactions on Automatic Control*, Vol. 62, No. 8, 2016, pp. 3861–3876.

- [27] Kochenderfer, M. J., Holland, J. E., and Chryssanthacopoulos, J. P., “Next-generation airborne collision avoidance system,” Tech. rep., Massachusetts Institute of Technology-Lincoln Laboratory Lexington United States, 2012.
- [28] Holland, J. E., Kochenderfer, M. J., and Olson, W. A., “Optimizing the next generation collision avoidance system for safe, suitable, and acceptable operational performance,” *Air Traffic Control Quarterly*, Vol. 21, No. 3, 2013, pp. 275–297.
- [29] Ahmadi, M., Singletary, A., Burdick, J. W., and Ames, A. D., “Safe policy synthesis in multi-agent POMDP via discrete-time barrier functions,” *2019 IEEE 58th Conference on Decision and Control (CDC)*, IEEE, 2019, pp. 4797–4803.
- [30] Johnson, W., *Why a Dangerous Braking Problem Makes the 2016 Volvo XC90 a Model To Avoid*, 2020 (accessed December 7, 2020). URL <https://www.motorbiscuit.com/why-a-dangerous-braking-problem-makes-the-2016-volvo-xc90-a-model-to-avoid/>.
- [31] Schierman, J. D., DeVore, M. D., Richards, N. D., and Clark, M. A., “Runtime Assurance for Autonomous Aerospace Systems,” *Journal of Guidance, Control, and Dynamics*, 2020, pp. 1–13.
- [32] Singletary, A., Gurriet, T., Nilsson, P., and Ames, A. D., “Safety-critical rapid aerial exploration of unknown environments,” *2020 IEEE International Conference on Robotics and Automation (ICRA)*, IEEE, 2020, pp. 10270–10276.
- [33] Clark, M., Koutsoukos, X., Porter, J., Kumar, R., Pappas, G., Sokolsky, O., Lee, I., and Pike, L., “A study on run time assurance for complex cyber physical systems,” Tech. rep., Air Force Research Lab Wright-Patterson AFB, Ohio, 2013.
- [34] German Federal Bureau of Aircraft Accident Investigation, *Investigation Report AX001-1-2 (English)*, 2004 (accessed December 6, 2020). URL http://www.bfu-web.de/EN/Publications/Investigation%20Report/2002/Report_02_AX001-1-2_Ueberlingen_Report.pdf?__blob=publicationFile.
- [35] Arinio, T., “Decision criteria for regulatory measures on TCAS II version 7.1,” *Safety Issue Rectification Extension Plus Project*, 2008.
- [36] AviationWeek, *Auto-GCAS Saves Unconscious F-16 Pilot—Declassified USAF Footage*, 2016 (accessed December 10, 2020). URL https://www.youtube.com/watch?v=WkZGL7RQBVw&ab_channel=AviationWeek.
- [37] Manfredi, G., and Jestin, Y., “An introduction to ACAS Xu and the challenges ahead,” *2016 IEEE/AIAA 35th Digital Avionics Systems Conference (DASC)*, 2016, pp. 1–9. <https://doi.org/10.1109/DASC.2016.7778055>.
- [38] Nelson, R. C., et al., *Flight stability and automatic control*, Vol. 2, WCB/McGraw Hill New York, 1998.
- [39] Stevens, B. L., Lewis, F. L., and Johnson, E. N., *Aircraft control and simulation: dynamics, controls design, and autonomous systems*, John Wiley & Sons, 2015.
- [40] RCTA, “DO-178C: Software considerations in airborne systems and equipment certification,” 2011.
- [41] Jeannin, J.-B., Ghorbal, K., Kouskoulas, Y., Gardner, R., Schmidt, A., Zawadzki, E., and Platzer, A., “Formal verification of ACAS X, an industrial airborne collision avoidance system,” *2015 International Conference on Embedded Software (EMSOFT)*, IEEE, 2015, pp. 127–136.
- [42] Jeannin, J.-B., Ghorbal, K., Kouskoulas, Y., Gardner, R., Schmidt, A., Zawadzki, E., and Platzer, A., “A formally verified hybrid system for the next-generation airborne collision avoidance system,” *International Conference on Tools and Algorithms for the Construction and Analysis of Systems*, Springer, 2015, pp. 21–36.
- [43] Katz, G., Barrett, C., Dill, D. L., Julian, K., and Kochenderfer, M. J., “Reluplex: An efficient SMT solver for verifying deep neural networks,” *International Conference on Computer Aided Verification*, Springer, 2017, pp. 97–117.
- [44] Desai, A., Ghosh, S., Seshia, S. A., Shankar, N., and Tiwari, A., “SOTER: A Runtime Assurance Framework for Programming Safe Robotics Systems,” *2019 49th Annual IEEE/IFIP International Conference on Dependable Systems and Networks (DSN)*, IEEE, 2019, pp. 138–150.
- [45] Phan, D. T., Grosu, R., Jansen, N., Paoletti, N., Smolka, S. A., and Stoller, S. D., “Neural Simplex Architecture,” , 2020.
- [46] Pike, L., Goodloe, A., Morisset, R., and Niller, S., “Copilot: a hard real-time runtime monitor,” *International Conference on Runtime Verification*, Springer, 2010, pp. 345–359.
- [47] Pike, L., Wegmann, N., Niller, S., and Goodloe, A., “Copilot: monitoring embedded systems,” *Innovations in Systems and Software Engineering*, Vol. 9, No. 4, 2013, pp. 235–255.

- [48] SAE International, “ARP4761: Guidelines and Methods for Conducting the Safety Assessment Process on Civil Airborne Systems and Equipment,” 1996.
- [49] Goossen, E. R., and Buster, D. A., “Regulatory Compliance in Multi-Tier Supplier Networks,” 2014.
- [50] SAE International, “ARP4754: Guidelines for Development of Civil Aircraft and Systems,” 2010.
- [51] RCTA, “DO-254: Design Assurance Guidance for Airborne Electronic Hardware,” 2000.
- [52] RCTA, “DO-297: Integrated Modular Avionics (IMA) Design Guidance and Certification Considerations,” 2006.
- [53] RCTA, “DO-278: Guidelines for Communication, Navigation, Surveillance and Air Traffic Management (CNS/ATM) Systems Software Integrity Assurance,” 2011.
- [54] Geisinger, K., “Guide to methods & tools for safety analysis in air traffic management,” *Series Guide to Methods and Tools for Safety Analysis in Air Traffic Management. Global Aviation Information Network*, 2003.
- [55] RCTA, “DO-333: Formal Methods Supplement to DO-178C and DO-278A,” 2011.
- [56] ASTM International, “ASTM-F3269: Standard Practice for Methods to Safely Bound Flight Behavior of Unmanned Aircraft Systems Containing Complex Functions,” 2017.
- [57] Aviation Pros, *Laminar Research Announces Wireless ADS-B Receiver for Xavion Mobile Aviation Application*, 2013 (accessed December 7, 2020). URL <https://www.aviationpros.com/engines-components/aircraft-airframe-accessories/avionics/press-release/10953251/laminar-research-announces-wireless-adsb-receiver-for-xavion-mobile-aviation-application>.
- [58] McConley, M. W., Appleby, B. D., Dahleh, M. A., and Feron, E., “A control Lyapunov function approach to robust stabilization of nonlinear systems,” *Proceedings of the 1997 American Control Conference (Cat. No. 97CH36041)*, Vol. 1, IEEE, 1997, pp. 329–333.
- [59] Frazzoli, E., Dahleh, M. A., and Feron, E., “Real-time motion planning for agile autonomous vehicles,” *Journal of guidance, control, and dynamics*, Vol. 25, No. 1, 2002, pp. 116–129.
- [60] Frazzoli, E., Dahleh, M., Feron, E., and Kornfeld, R., *A randomized attitude slew planning algorithm for autonomous spacecraft*, Vol. 4155, AIAA Reston, VA, 2001.
- [61] Lee, J., and Mitici, M., “An integrated assessment of safety and efficiency of aircraft maintenance strategies using agent-based modelling and stochastic Petri nets,” *Reliability Engineering System Safety*, 2020, p. 107052. <https://doi.org/10.1016/j.res.2020.107052>.
- [62] Winder, L. F., and Kuchar, J. K., “Generalized philosophy of alerting with applications for parallel approach collision prevention,” Tech. rep., 2000.
- [63] *Pilot Responsibilities When Conducting Land and Hold Short Operations (LAHSO) / AIM*, 2017 (accessed December 7, 2020). URL <https://m.blog.naver.com/daal2004/221155761188>.
- [64] Hilkevitch, J., *FAA Order on Landings, Takeoffs Scares Pilots*, 2000 (accessed December 7, 2020). URL <https://www.chicagotribune.com/news/ct-xpm-2000-04-29-0004290147-story.html>.
- [65] Transportation Safety Board of Canada, *Aviation Investigation Report A02O0272*, 2002 (accessed December 7, 2020). URL <https://tsb.gc.ca/eng/rapports-reports/aviation/2002/a02o0272/a02o0272.html>.
- [66] Humphrey, A., “Forty-five Years of Air-brake Evolution,” *Scientific American*, Vol. 110, No. 25, 1914, pp. 498–511.
- [67] *Institute for Software Integrated Systems*, (accessed December 7, 2020). URL <https://www.isis.vanderbilt.edu/>.
- [68] IVAO, France & Switzerland Divisions, “Letter of Agreement between the Marseille FIR (LFMM) and the Geneva FIR (LSAG),” 2020 (accessed December 20, 2020). URL https://storage.iviao.fr/ao_public/LOA/Inter/Etranger/LOA-LFMM-LSAG_EN_v3.pdf.
- [69] Bogaisky, J., *Xwing Aims To Usher In The Era Of Autonomous Flight Sooner By Robotizing Small, Old Cargo Planes*, 2020 (accessed December 20, 2020). URL <https://www.forbes.com/sites/jeremybogaisky/2020/08/20/xwing-aims-to-usher-in-the-era-of-autonomous-flight-sooner-by-robotizing-small-old-cargo-planes/?sh=792d07966235>.

- [70] Sarsfield, K., *Regulators approve Garmin's Autoland safety system on Vision Jet*, 2020 (accessed December 20, 2020). URL <https://www.flightglobal.com/business-aviation/regulators-approve-garmins-autoland-safety-system-on-vision-jet/139981.article#:~:text=Cirrus%20Aircraft's%20SF50%20Vision%20Jet,engined%20type%20on%2031%20August>.
- [71] Feron, E., Bicchi, A., and Pallottino, L., "Mixed integer programming for aircraft conflict resolution," *AIAA guidance, navigation, and control conference and exhibit*, 2013, p. 4295.
- [72] Schouwenaars, T., De Moor, B., Feron, E., and How, J., "Mixed integer programming for multi-vehicle path planning," *2001 European control conference (ECC)*, IEEE, 2001, pp. 2603–2608.
- [73] Jackson, P., "SWIM in the Sky" Aircraft Access to SWIM (AAtS)," *Air Transportation Information Exchange Conference*, 2012.
- [74] Jensen, D., *The Father of Free Flight*, 2000 (accessed December 7, 2020). URL <https://www.aviationtoday.com/2000/11/01/editors-note-the-father-of-free-flight/>.
- [75] Polish Air Navigation Services Agency, *POLFRA - Polish Free Route Airspace*, 2019 (accessed December 7, 2020). URL <https://www.pansa.pl/en/polfra/>.
- [76] Euro Control, *Free route airspace at Maastricht UAC*, 2019 (accessed December 7, 2020). URL <https://www.eurocontrol.int/service/free-route-airspace-maastricht-uac>.
- [77] PlaneSpotting, *Cathay Pacific 747 on final, on the notorious "Checkerboard" approach into Hong Kong Kai Tak Airport*, 2019 (accessed December 7, 2020). URL https://www.reddit.com/r/aviation/comments/bcmkhg/cathay_pacific_747_on_final_on_the_notorious/.