



A Workflow Perspective in Aviation

Guy Andre Boy

► To cite this version:

Guy Andre Boy. A Workflow Perspective in Aviation. Cognitive Informatics: Reengineering Clinical Workflow for More Efficient and Safer Care, 2019. hal-02956589

HAL Id: hal-02956589

<https://hal.science/hal-02956589>

Submitted on 3 Oct 2020

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

CHAPTER 7. A Workflow Perspective in Aviation

Guy André Boy, Ph.D.

Abstract

During the last three decades, aircraft cockpits have been highly automated and incrementally digitized to the point that we now talk about “interactive cockpits”, not because pilots interact with the mechanical part of the aircraft directly, but because they use a pointing device to interact with computer screens. More generally, computers invaded the airspace ranging from onboard aircraft to air traffic control (ATC) ground services. In terms of workflow, the elimination of the flight-engineer onboard commercial aircraft cockpits drastically transformed the way pilots fly – this was done in the beginning of the 1980’s. Pilot’s activity shifted from manual control to cognitive management of embedded systems (performing some of the tasks the flight engineer used to do). Aircraft automation considerably improved commercial aviation safety. Today, main issues come from the exponential increase of the number of aircraft in the sky. Air traffic complexity imposes drastic reengineering of air traffic workflow. ATC is shifting towards air traffic management (ATM). Same thing: moving from control to management, expanding cockpit’s single agent problems and solutions to air traffic multi-agent problems and solutions. The main emphasis is therefore social cognition. During the 20th century, we automated physical systems – we now talk about cyber-physical systems (i.e., we shifted from hardware mechanical engineering issues to software cognitive problems). During the beginning of the 21st century, almost all systems are first ideated and designed as pieces of software, and they are transformed into tangible things. Consequently, workflow can be modeled and simulated very early during the design process and, unlike during the 20th century, functions and therefore activities can be tested before anything is physically built. This provides great possibilities that should be operationalized and implemented. This chapter will show that the main issue has become tangibility, instead of automation. At the same time, these new possibilities provided by our emerging digital world enable a brand-new move towards more autonomous systems that need to be coordinated. We will illustrate this shift from automation towards autonomy, by providing salient examples and generic patterns of this evolution of workflow in the aviation domain.

Introduction

Aviation is changing from air traffic control (ATC) to air traffic management (ATM), where both systems and people are immersed into massive amount of software. Workflow is changing because people and systems now interact in a different manner, using different cognitive and socio-cognitive processes and functions. Digitalization of the airspace logically leads to different kinds of function allocation.

Since the 1980s, we never stopped automating aviation systems. Consequently, flying has become more cognitive, moving pilots' cognitive functions from doing to thinking. Today, automation in aviation should be better called digitalization of air-ground socio-technical systems. Digitalization of the airspace is a matter of looking for the right mix of technology, organization and people's activities that should be concurrently designed and tested to discover emergent patterns, which themselves should be incrementally considered.

The massive use of information technology in aviation led to drastic innovations. Main reasons why automation and innovation have been and still are drivers of aviation evolution are: exponential increase of the number of aircraft that causes congested network; higher air traffic complexity; unpredictable delays; and other things that result in severe congestions at key airports, rising fuel costs and pollution.

Aircraft cockpits were greatly transformed during the 1980s involving the development of a large number of embedded systems. Digitalization of aircraft systems increased the need for cognitive engineering developments, especially in the digitalization of commercial aircraft cockpits. More specifically, flying tasks evolved from manual mechanical control to management of embedded systems.

In the same way, air traffic recently started to evolve from control to management. Two major programs are considering this shift, SESAR in Europe (Single European Sky ATM Research) and NextGen (Next Generation Air Transportation System) in the USA. The main goal is for us to better understand new airspace management models that take into account traffic growth, safety constraints and capacity management. Such efforts should result in appropriate human-systems integration of new multi-agent systems of systems, and teams of teams.

Automation in aviation is not new. Autopilots were introduced in commercial aviation in the 1930s (e.g., the Boeing 247 commercial aircraft flew with an autopilot in 1933). However, we learned a lot from aircraft automation during the last three decades of the 20th century, especially shifting from analog to digital automation. Anytime we automate, we rigidify tasks and therefore activities. Automated systems are very context-dependent (i.e., they work fine when they are operated in very well-known contexts, but may dysfunction outside of these contexts).

Digitalization of embedded systems made us evolve toward cyber-physical systems (CPSs) and the Internet-of-things (IoT). We then deal with much larger systems of systems that require new types of investigation. More specifically, such new systems, which include both people and machines, become more autonomous in broader contexts, which also need to be further understood and subsequently defined. This evolution from automation to autonomy requires that we emphasize flexibility issues. More specifically, if automation is associated with rigidity (i.e., procedure based), autonomy should be associated with flexibility (i.e., problem-solving based) where people should be considered at the center.

In this chapter, following up our human-centered design (HCD) approach, we will provide human-systems integration (HSI) solutions more than human factors problems. This will enable us to have a basis for comparing evolutions in aviation and health care. A discussion will be started. We will conclude and provide perspectives.

Using the AUTOS pyramid to support workflow analysis

Workflow evolved with respect to technology (more specifically, automation), organization (in commercial aircraft, we moved from 5 technical crewmembers during the fifties to 4, then 3, and 2 in the beginning of the 1980s), and jobs (different kinds of functions changed drastically because systems were able to execute tasks that were performed by people before).

A typical flight is divided into phases, sub-phases, and so on. For example, after passenger boarding, there is the taxi phase, then the runway rolling phase (before takeoff), takeoff, after takeoff, initial climb, climb, cruise, and so on. These phases are contextual pattern that determine appropriate set of tasks. In order to define workflow patterns, we use the AUTOS pyramid for each of these contextual patterns, which can be normal, abnormal or emergency.

The AUTOS pyramid was first introduced in HCD as the AUTO tetrahedron (Boy, 1998) to help relate four entities: Artifact (i.e. system), User, Task and Organizational environment. We subsequently added contextual patterns, which we called “Situations” that represent the various possible events where the artifact could be used (Boy, 2011). The AUTOS pyramid supports HSI making sure that all important entities (i.e., Artifacts, Users, Tasks, Organizations and Situations) are taken into account, as well as their properties and interconnections (provided on the edges of the pyramid).

Artifacts, users and tasks

An artifact is anything that is built by people. In this chapter, artifacts will denote systems. A system is a set of interconnected components (i.e., physical parts) and procedures (i.e., software parts) forming a complex whole that is intended to be useful for doing something. Artifacts may be aircraft, avionics systems, devices and components of these systems for example. Artifacts are often integrated sets of existing technology. Sometime, they are made of brand new technology. Here is a short list of interactive artifacts: force feedback, loudspeakers, screens, signals, buttons, keyboard, joystick, mouse, trackball, microphone, 3D mouse, data suit (or interactive seat), metaphor for interaction, visual rendering, 3D sound rendering, 3D geometrical model and so on. These artifacts are usually integrated with mechanical artifacts such as pipes, containers, engines, pressurizers, turbines, flaps, slats, wheels, brakes and so on.

Users may be novices, experienced personnel or experts, coming from and evolving in various cultures (e.g., pilots, air traffic controllers, dispatchers). They may be tired, stressed, making errors, old or young, as well as in very good shape and mood. Users have been taken into account by human factors and ergonomics (HFE) during the last five decades in the context of engineering-centered engineering, generating the concepts of user interfaces and operational procedures.

Tasks vary from handling quality control, flight management, managing a passenger cabin, repairing, designing, supplying or managing a team or an organization. Each task involves one or several cognitive functions that related users must learn and use. The AUT triangle (Figure 1) enables the explanation of three edges: task and activity analysis (U-T); information requirements and technological limitations (T-A); ergonomics and training (procedures) (A-U).

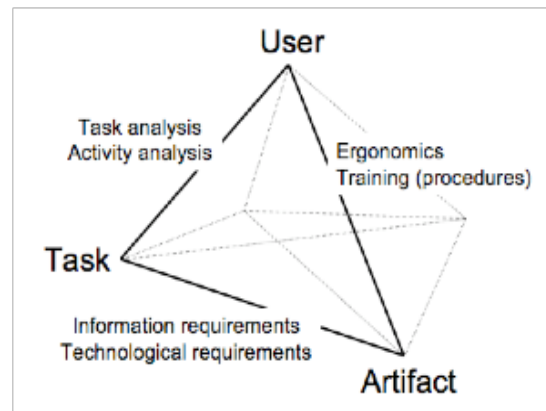


Figure 1. The AUT triangle.

Today, almost any system includes software, which mediates user intentions and provides appropriate feedback. Automation introduces constraints and, as already said, more rigidity. End-users do not have the final action (automation does), they need to plan more than in the past. Work becomes more cognitive and (artificially) social, i.e., there are new social activities that need to be performed in order for the other relevant actors to do their jobs appropriately. This even becomes more obvious when cognition is distributed among many human and machine agents.

Cockpits were incrementally shaped to human anthropometrical requirements in order to ease manipulation of the various instruments. This of course is always strongly related to technology limitations also. Anthropometry developed its own language and methods. It is now actively used in design to define workspaces according to human factors such as accommodation, compatibility, operability, and maintainability by the user population. Workspaces are generally designed for 90% to 95% coverage of the user population. Anthropometric databases are constantly maintained to provide appropriate information to designers and engineers. Nevertheless, designers and engineers need to be guided to use these databases in order to make appropriate choices.

Fatigue is a major concern in aviation, and strongly depends on work organization. Therefore, it is important to know about circadian rhythms and the way people adapt to shift work and long work hours for example. Consequences are intimately associated with health and safety risks. Fatigue studies provide more knowledge and knowhow on how to proceed with work time schedules, appropriate training, systematic checks, and health indicators following. Of course, this needs to be integrated in regulatory procedures. Useful information can be found in the Handbook of Human-Machine Interaction (Gander, Graeber & Belenky, 2011).

In aviation, cognitive factors start with workload assessment. This statement may seem to be restrictive and old fashion, but the reader should think twice about workload before starting any work in human factors. On one side, workload is a concept that is very difficult to define. It is both an output of human performance and a necessary input to optimize performance, i.e., we produce workload to perform better, up to a point where we need to change our work strategy. But on the other side, we need to figure out a model that would quantify a degree of load

produced by a human being while working. Of course, this model should be based on real measurements performed on the human being. Many models of workload have been proposed and used in aviation (Bainbridge, 1978; Hart, 1982; Boy & Tessier, 1985). Workload also deals with the complexity of the task being performed. In particular, people can do several things at the same time, in parallel; this involves the use of several different peripheral resources simultaneously (Wickens, 1992). Sperandio (1980) studied the way air traffic controllers handle several aircraft at the same time, and showed that the time spent on radio increased with the number of aircraft being controlled: 18% of their time spent in radio communication for one controlled aircraft whereas 87% for nine aircraft controlled in parallel. In other words, task complexity tends to increase human operator efficiency.

Human-systems interaction moves into human-systems cooperation when systems become more autonomous. In this case, it is more appropriate to talk about agent-agent cooperation. Hoc and Lemoine studied dynamic task allocation (DTA) of conflict resolution between aircraft in air-traffic control on a large-scale simulator. The more the assistance, the more anticipative the mode of operation in controllers and the easier the human-human cooperation (HHC). These positive effects of the computer support are interpreted in terms of decreased workload and increased shared information space (Hoc & Lemoine, 1998).

Situation awareness (SA) is another concept that is useful to introduce here, especially as a potential indicator for safety in highly automated human-machine systems. During the last decades, lots of efforts have been carried out to assess SA such as the Situation Awareness Global Assessment Technique (SAGAT) (Endsley, 1988, 1996). Several efforts have been developed to assess SA in the aeronautics domain (Mogford, 1997); the main problem is the characterization of the influence of action on situation awareness. Indeed, human operator's actions are always situated, especially in life-critical environments, and SA does not mean the same when actions are intentional as when they are reactive. In human-machine interaction, this is a very important issue since actions are always both intentional (deliberative) and reactive because they are mainly performed in a close loop (Boy, 2015).

Considering organizations in design

The Orchestra model was proposed in aviation to better understand authority sharing (Boy & Grote, 2009; Boy, 2013). Technological design requires multidisciplinary design teams (i.e., a design team must include people who have related background, competence and experience on each of these relevant artifacts incrementally integrated). In addition, design team members need to understand each other (i.e., they need to be able to read the same music theory, even if they do have the same scores). They need to be appropriately coordinated both at the task level (i.e., scores need to be harmonized by a composer) and the activity level (i.e., design team members, as musicians, need to be coordinated by a conductor at performance time).

An organizational environment for design does not only include all design team players (i.e., human agents), but also technological means (i.e., system agents). At this point, human-systems integration is not only for the sake of the product, but also for the sake of the design team itself. For this reason, design cards constitute useful support (Boy, 2013). Considering organizations in

design introduces three additional edges (Figure 2): social issues (U-O); role and job analyses (T-O); emergence and evolution (A-O).

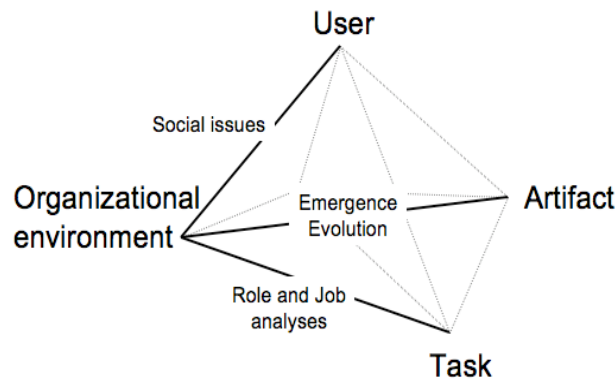


Figure 2. The AUTO tetrahedron.

There are two fields of research that grew independently for the last three decades: crew resource management (CRM) in aviation, and computer-supported cooperative (CSCW) work in HCI. The former was motivated by social micro-world of aircraft cockpits where pilots need to cooperate and coordinate to fly safely and efficiently. CRM started during a workshop on *resource management on the flight deck* sponsored by NASA in 1979 (Cooper, White, & Lauber, 1980). At that time, the motivation was the correlation between air crashes and human errors as failures of interpersonal communications, decision-making, and leadership (Helmreich et al., 1999). CRM training developed within airlines in order to change attitudes and behavior of flight crews. CRM deals with personalities of the various human agents involved in work situations, and is mainly focused on teaching, i.e., each agent learns to better understand his or her personality in order to improve the overall cooperation and coordination of the working group. The same kind of issues should be taken into account in design teams and solutions incrementally implemented and evaluated.

Interaction is also influenced by the organizational environment that is itself organized around human(s) and system(s). More explicitly, HSI could focus on someone facing his/her laptop writing a paper; it could also be someone driving a car with passengers; it could be focused on an air traffic management system that includes pilots, controllers and various kinds of aviation systems. People are now able to interact with computerized systems or with other people via computerized systems. We recently put to the front authority as a major HCD concept. When a system or other parties do the job, or part of the job, for someone, there is delegation. What is delegated? Is it the task? Is it the authority in the execution of this task? By authority, we mean accountability (responsibility) and control. Such questions should find answers within the design team both analytically and experimentally through human-in-the-loop simulations (HITLS).

Organization complexity is linked to social cognition, agent-network complexity, and more generally multi-agent management issues. There are four principles for multi-agent management: agent activity (i.e., what the other agent is doing now and for how long); agent activity history (i.e., what the other agent has done); agent activity rationale (i.e., why the other agent is doing what it does); and agent activity intention (i.e., what the other agent is going to do).

next and when). Multi-agent management needs to be understood through a role (and job) analysis. To summarize, O-factors mainly deal with the required level of *coupling* between the various purposeful agents to handle the new artifact.

Testing in a large variety of situations

The AUTOS framework (Figure 3) is an extension of the AUTO tetrahedron that introduces a new dimension, the “Situation”, which was implicitly included in the “Organizational environment”. The three new edges are: usability/usefulness (A-S); situation awareness (U-S); situated actions (T-S); cooperation/coordination (O-S).

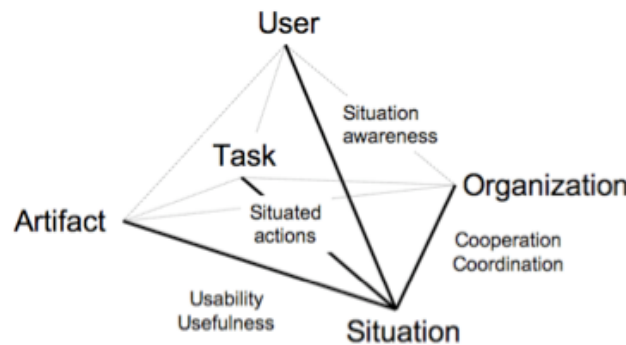


Figure 3. The AUTOS pyramid.

Interaction depends on the situation where it takes place. Situations could be normal or abnormal. They could even be emergencies. This is why we will emphasize the scenario-based approach to design and engineering. Resulting methods are based on descriptions of people using technology in order to better understand how this technology is, or could be, used to redefine their activities. Scenarios can be created very early during the design process and incrementally modified to support product construction and refinement.

Scenarios are good to identify functions at design time and operations time. They tend to rationalize the way the various agents interact among each other. They enable the definition of organizational configurations and time-wise chronologies.

Situation complexity is often caused by interruptions and more generally disturbances. It involves safety and high workload situations. It is commonly analyzed by decomposing contexts into sub-contexts. Within each sub-context, the situation is characterized by uncertainty, unpredictability and various kinds of abnormalities. To summarize, situational factors deal with the *predictability* and *appropriate completeness* (scenario representativeness) of the various situations in which the new artifact will be used.

Using the AUTOS pyramid in practice

Software is very easy to modify. Consequently, design teams develop prototypes that they modify all the time! Interaction is not only a matter of end product; it is also a matter of agile development process. End-users are not the only ones to interact with a delivered product; designers and engineers also interact with the product in order to fix it up toward maturity... even after its delivery. This is why agile approaches based on design cards (Boy, 2016) are extremely

useful and effective (Schwaber, 1997; Sutherland, 2014). In addition, scenario-based design is an HCD approach that fosters understandability (situation awareness), complexity, reliability, maturity and induced organizational constraints (rigidity versus flexibility).

Software complexity can be split into internal complexity (or system complexity) and interface complexity. Internal complexity is related to the degree of explanation required to the user to understand what is going on when necessary. Concepts related to system complexity are: flexibility (both system flexibility and flexibility of use); system maturity (before getting mature, a system is an accumulation of functions —the “another function syndrome” — and it becomes mature through a series of articulations and integrations); automation (linked to the level of operational assistance, authority delegation and automation culture); and operational documentation. Technical documentation complexity is very interesting to be tested because it is directly linked to the explanation of artifact complexity. The harder a system is to use; the more related technical documentation or performance support are required in order to provide appropriate assistance at the right time in the right format.

What should we understand when we use a product? How does it work? How should it be used? At what level of depth should we go inside the product to use it appropriately? In the early ages of the car industry, most car drivers were also mechanics because when they had a problem they needed to fix it by themselves; the technology was too new to have specialized people. These drivers were highly skilled engineers both generalists and specialists on cars. Today, things have drastically changed; drivers are no longer knowledgeable and skilled to fix cars; there are specialists that do this job because software is far too complex to understand without appropriate help. Recent evolution transformed the job of mechanics into system engineers who know how to use specialized software that enables to diagnose failures and fix them. They do not have to fully understand what is going on inside the engine, a software program does it for them and explain problems to them; when the overall system is well-designed of course. This would be the ideal case; in practice, most problems come from organizational and situational factors induced by the use of such technology (e.g., appropriate people may not be available at the right time to fix problems when they arise).

Interface complexity is characterized by content management, information density and ergonomics rules. Content management is, in particular, linked to information relevance, alarm management, and display content management. Information density is linked to decluttering (Doyon-Poulin, Ouellette & Robert, 2014), information modality, diversity, and information-limited attractors, i.e., objects on the instrument or display that are poorly informative for the execution of the task but nevertheless attract user’s attention. The “PC screen do-it all syndrome” is a good indicator of information density (elicited improvement-factors were screen size and zooming). Redundancy is always a good rule whether it repeats information for crosschecking, confirmation or comfort, or by explaining the “how”, “where”, and “when” an action can be performed. Ergonomics rules formalize user friendliness, i.e., consistency, customization, human reliability, affordances, feedback, visibility and appropriateness of involved cognitive functions.

Task complexity involves procedure adequacy, appropriate multi-agent cooperation (e.g., air-ground coupling in the aerospace domain) and rapid prototyping (i.e., task complexity cannot be

properly understood if the resulting activity of agents involved in it is not observable). Task complexity is linked to the number of sub-tasks, task difficulty, induced risk, consistency (lexical, syntactic, semantic and pragmatic) and the temporal dimension (perception-action frequency and time pressure in particular). Task complexity is due to operations maturity, delegation and mode management. Mode management is related to role analysis. To summarize, T-factors mainly deal with *task difficulty* according to a spectrum from best practice to well-identified categories of tasks.

Do not forget that CFA requires HITLS to observe activity. Activity analysis could be defined as the identification and description of activities in an organization, and evaluation of their impact on its operations. Activity analysis determines: (1) what activities are executed; (2) how many people perform the activities; (3) how much time they spend on them; (4) how much and which resources are consumed; (5) what operational data best reflects the performance of activities; and (6) how much value these activities provide to the organization. This analysis is accomplished through direct observation, interviews, questionnaires, and review of the work records addressed to users of prototypes at different stages of design and development.

Cockpit evolution: from control to management

Twentieth century engineering was dominated by mechanical engineering. Engineers built trains, cars, airplanes and power plants by assembling mechanical things. During the last decades, computer science and information technology massively penetrated mechanical machines to incrementally create systems, which included physical hardware and cognitive software. Everything started with the automation around the center of gravity using yoke or side stick and thrust levers (Figure 4). The first embedded system was a single agent regulating parameters, such as speed and heading, one parameter at a time. Time constant of the feedback was around 500 milliseconds. Pilots had to adapt to this embedded system by changing from control of flight parameters to supervision of embedded system behavior with respect to a set point.

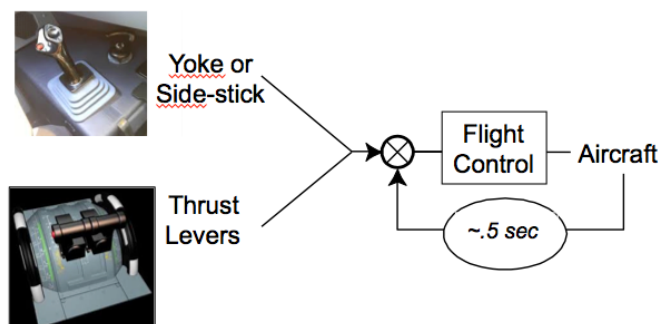


Figure 4. Trajectory control embedded system: Flying around the center of gravity¹.

¹ The 4-loop approach to aeronautical automation evolution was inspired to me by Etienne Tarnovski during a keynote that he gave at HCI-Aero'06, Seattle, USA.

The guidance system was developed circa the early eighties (Figure 5). This second feedback loop took into account several parameters. Its time constant was around 15 seconds. Note that this feedback loop was implemented on top of the trajectory control system. High-level modes of automation appeared and were managed on the flight control unit panel. At the same time, integrated and digital autopilot and auto-throttle were installed.

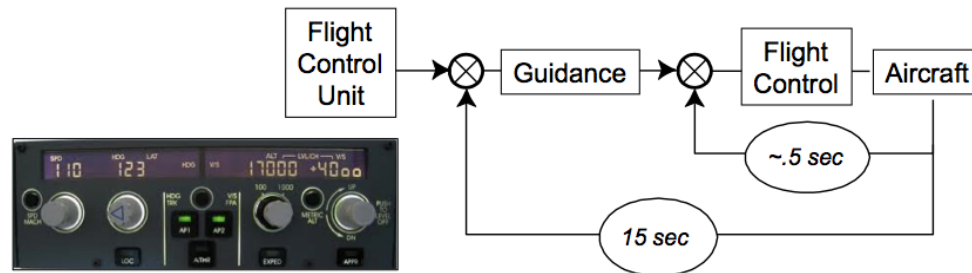


Figure 5. Guidance embedded system: Guiding on basic trajectory.

The third embedded system concerned navigation automation with a time constant of about one minute (Figure 6). Guidance and flight management became integrated. This was the first real revolution in the evolution of aeronautical embedded systems. We were shifting from control of flight parameters to management of embedded systems. Software became dominant and the number of artificial agents on aircraft grew exponentially. For that matter, pilots have now to deal with a variety of embedded systems that are not only humans but also software-based agents. Problems may emerge when these software-based agents communicate among each other. This issue will be analyzed later in the chapter.

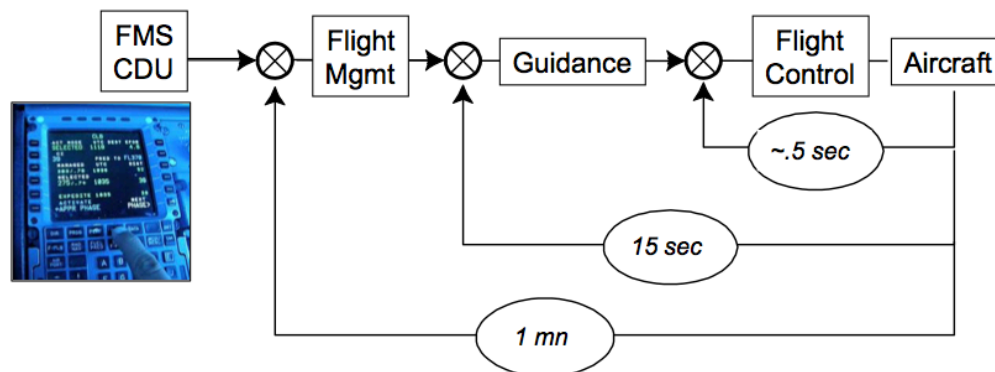


Figure 6. Navigation automation loop: Guiding the flight plan.

Whenever people extend their capabilities with appropriate embedded systems that can be called “cognitive prostheses” (Hamilton, 2001), they increase their performance. However, if these cognitive prostheses use automation to a point that is not clearly understood from an operational standpoint, performance may decrease and, in some cases, cause serious problems. This is what Earl Wiener called “clumsy automation” (Wiener, 1989). When I was working on the

Orbital Refueling System of the Space Shuttle in the mid-eighties (Boy, 1987), I found out that there is an optimum P_0 in terms of autonomy level and performance of the overall human-machine system (Figure 7).

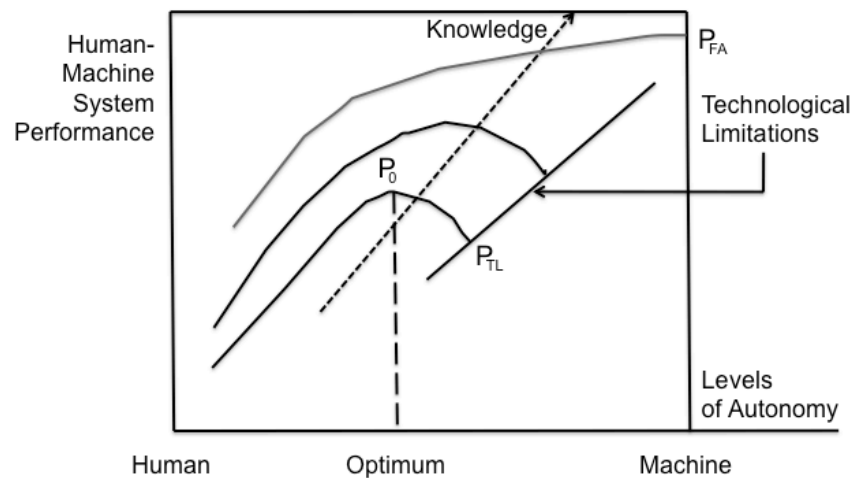


Figure 7. Human-machine system performance versus levels of autonomy (adaptation of Boy's NASA technical document, 1988).

Technology-centered engineering typically automates at the point P_{TL} (i.e., on the technological limitations limit). It takes experimental tests and efforts to go back to P_0 . HCD takes into account the existence of P_0 , and incrementally tries to find out this optimum through creative design and formative evaluation using HITLS. Interestingly, the more we know about autonomy of the various human and machine agents, the more the optimum P_0 move to the right and goes up on Figure 7. Of course, if we knew everything about the environment and the various agents involved in the various interactions, the optimum P_{FA} would be on the full autonomy of the machine (e.g., in the case of an aircraft, P_{FA} would correspond to a drone).

This shift of optimum P_0 to the right is strongly related to maturity. In HCD, we distinguish between technology maturity and maturity of practice. The former deals with reliability, availability and robustness of the technology being designed and developed. The latter deals with people's adaptation to and resilience of the technology; often new practices emerge from technology use. We then need to observe usages as early as possible to anticipate surprises before it is too late. HCD proposes methods and tools that enable HCD teams to detect these emergent properties at design time (e.g., using HITLS). In any case, there is a maturity period required to assess if a product can be delivered or not.

Organizational automation and management

We have seen that pilot's job moved from control of flight parameters to management of embedded systems during the 1980s. This job revolution in the cockpit is now shifting to air traffic (i.e., air traffic control is moving toward air-traffic management). In other words, single agent's

shift from control to management is currently evolving to a multi-agent shift. Why? What are underlying organizational issues?

The main cause is the number increase effect. According to Boeing's Current Market Outlook 2014-2033, average airline traffic yearly growth is estimated at a rate of 5%. Knowing that most big airports, such as Hartsfield–Jackson Atlanta International Airport, are already over-saturated, the air traffic capacity issue requires complexity science approaches, and more specifically a multi-agent approach, where agents are aircraft. Connectivity among these agents has to become explicit. If aircraft separation has to be reduced during approach and landing for example, current human-centered air traffic control techniques need to be re-visited and in many cases drastically changed. This is a matter of technology, organizations and people. On the technology side of the problem to be solved, each aircraft should know about the location and identity of the other aircraft around it. Aircraft should then be equipped with appropriate sensors and receptors, such as Automatic Dependence Surveillance – Broadcast (ADS-B) system. Satellite data have to be used to identify a clear dynamic model of the sky, in terms of both traffic and weather. We can also use weather radar data from all aircraft and fuse them with satellite data to increase 3D validity of weather models. Same kind of fusion can be done for air traffic using more conventional radar system data. Resulting data can be used to develop an embedded system providing each aircraft with a protection safety net (i.e., each aircraft knows the traffic around it and is able to decide tactical maneuvers to increase global safety). The protection safety net of each aircraft is an embedded system related to other aircraft equivalent and a coordination ground system that orchestrates the overall traffic.

This approach definitely defines a new kind of organizational automation, which is multi-agent. This multi-agent approach involves interaction among people and systems, among systems, and among people (often through information technology). Since there will be a layer of information technology gluing the various air and ground systems, new factors are emerging such as cyber-security. Air traffic management of the future will be almost entirely based on highly interconnected cyber-physical systems (CPSs). New kinds of risks will emerge from the activity of this giant airspace CPS-based infrastructure. In particular, malicious actors will be able to attack these systems from anywhere in the world. Consequently, we need to further develop methods and tools that enable studying security and resilience of such CPS-based infrastructure, and find appropriate solutions. Protection will have to be found from technology (safety and security nets), organizations (collaboration among agents), and people (increasing training and expertise).

Air-ground integration is an example of human-systems integration point of view that involves function allocation. Since Fitt's law that provided the HABA-MABA² recommendations for single-agent function allocation, very little has been done on multi-agent function allocation. Function allocation rationale is to determine which functions should be carried out by humans and which by machines (Fitts, 1951). Cognitive function analysis (CFA) is an effective approach to this kind of problem in a multi-agent environment (Boy, 1998, 2011). CFA includes physical functions also (remember that a cognitive function is defined by a set of resources that may be cognitive

² Fitt's HABA-MABA (humans-are-better-at/machines-are-better-at) approach provided generic strengths and weaknesses of humans and machines.

functions and/or physical functions). In addition, the Orchestra model (Boy, 2013) provides a very useful framework to model and simulate resulting cognitive functions of the human-systems system being developed. Consequently, agent needs to be further modeled in terms of cognitive and physical functions, as well as the way they are inter-connected – this is typically doing a CFA. CFA results can be presented as a cognitive/physical function network (or interactive map), which is a good support for studying interaction complexity, distributed situation awareness and multi-agent decision-making.

Starting a discussion

Tangibility

In the beginning of the 21st century, embedded systems are reframing the structure-function duality. Instead of functionalizing structure to create automated machines (the 20th century approach), we are now structuralizing software functions. Therefore, automation problems, which were created during the last decades of the 20th century, are no longer the main issue because software can be tested from the early days of the design process using advanced virtual engineering, making HCD possible (i.e., taking into account human factors at design time). Cognitive function analysis can be validated because, in addition to task analysis, activity can be observed in HITLS using virtual prototypes and real users. Consequently, technological and organizational requirements can be nurtured by early solid function analyses.

The main issue has then become structure, and therefore tangibility. Since almost everything can be modeled and simulated on computers, and we now have 3D printing, structure can be very easily obtained. Traditionally in aeronautics, this was taken into account by flight tests. This is needed for the validation of any life-critical system. The right balance between cognitive and physical functions should be tested, as well as between abstractions and physical structures. Tangibility needs to be tested in terms of physical tangibility and figurative tangibility. The former is based on criteria such as simple reachability, complex accessibility, fatigue, noise management, resource availability and so on. The latter is based criteria such as monitoring, situation awareness, decision-making, risk taking and so on. Of course, these criteria can be detailed with respect to the level of granularity required at the current stage of the design process. Digital HITLS at design time contributes to the development of more appropriate HSI requirements. It does not remove the need for actual flight tests once the real system is developed, but contributes to improve their effectivity and cost.

Maturity

Lots of efforts and money have been spent to increase system reliability that contributed to aviation safety. As already said, safety of the future ATM involves increased attention on interactions among its various agents. The qualification of these “new” agents is an issue that concerns this investigation on authority sharing. We would like to go beyond the safety-reliability discussion, and propose to focus on maturity.

Since the general trend is to manage short-term benefits instead of long-term sustainability, it is not surprising that we currently observe a discrepancy in provision and qualification of human operators. Since human reliability remains very critical, if human operators are even less available and qualified, the situation will become worse. We are beginning to understand that human factors issues are no longer so much the result of engineering decisions but of economically-induced decisions.

Therefore, if we want to keep or improve the current level of safety, with an increase of airspace capacity, some drastic changes will need to be made in the way we understand and manage technology and organizations. In particular, safety-critical systems should be mature for safe use when they are delivered. The concept of maturity could be misleading because we already have in industry the quality-based Capacity Maturity Model (Paulk et al., 1993) that supports maturity of manufacturing processes. We are interested in product and practice maturity. Product maturity requires a strong focus on human-centered high-level requirements, as well as participatory design and development all along the life cycle of the product (Boy, 2005). Product maturity engineering involves a careful elicitation of the attributes that shape the related maturity of practice. We typically say that maturity of practice is reached when a reasonable number of surprises or emerging factors have been identified and related causes fixed. Obviously, product maturity engineering addresses the long term and is not appropriate for short-term goals and practice of current economy-driven organizations. It should be!

During the early stages of a development process such as the ATM of the future, the participation of the various representative actors is mandatory. This participatory approach requires that not only pilots and controllers, i.e., end-users and musicians, but also designers and regulators, i.e., the music instrument makers and composers, share a common frame of reference. In addition, the job definitions process cannot be improvised and must be planned since we know that initial definitions will have to be revised all along the life cycle of the overall ATM development process. Consequently, authority distribution for the design of the various instruments (this is where HCI specialists enter into play) is a matter of incremental development and testing. In PAUSA, we took a scenario-based approach (Carroll, 1995) to carry out such an authority distribution supported by the development of human factors principles and criteria, socio-technical models and HITLS.

The growing number of interdependencies among ATM agents led us to propose a measure of socio-cognitive stability (SCS) derived from various contributions, including Latour's account on socio-technical stability (Callon, 1991; Latour, 1987), emerging cognitive functions (Boy, 1998), distributed cognition (Hutchins, 1995), and socio-cognitive research and engineering (Hemingway, 1999; Sharples et al., 2002). We make a distinction between local and global SCS. Local SCS is related to agent's workload, situation awareness, ability to make appropriate decisions and, finally, correct action execution. It can be supported by appropriate redundancies and various kinds of cognitive support such as trends, relevant situational information and possible actions. Global SCS is concerned with the appropriateness of functions allocated to agents, pace of information flows and related coordination. It is very similar to the level of synchronization of rhythms in a symphony. Globally, socio-cognitive support could be found in a

safety net that would take into account the evolution of interacting agents and propose a constraining safety envelope in real time.

Three kinds of metrics have been deduced during the PAUSA project:

- **Complexity** is expressed as the number of relevant aircraft to be managed per appropriate volumetric zone (AVZ) at each time. An AVZ is calculated with respect to the type of flow pattern, e.g., aircraft crossing, spacing and merging. The definition of such an appropriate volumetric zone requires the assistance of operational ATC controllers. From a socio-cognitive perspective in ATM, complexity should be considered together with capacity. This is what the COCA (COMplexity & CAPacity) project investigated (Athènes et al., 2002; Cummings & Tsonis, 2006; Hilburn, 2004; Laudeman et al., 1998; Leveson et al., 2009).
- **Time pressure criticality** is the amount of workload that an agent (or a group of agents) requires to stabilize an ATM system after a disturbance. Such workload measure could be assessed as the ratio between the sum of required times for each action on the total available time (Boy, 1983).
- **Flexibility** is defined as the ease of modification of an air-ground contract in real-time. Flexibility assessments should guide ATM human-centered automation and organizational setting. Overall, increasing capacity also increases complexity and uncertainty, which need to be managed by finding the right balance between reducing uncertainties through centralized planning and coping with uncertainties through decentralized action. Loose coupling is required for actors to use their autonomy in accordance with system goals (Grote, 2004).

This chapter was written to provide salient workflow aspects, as well as evolutions of human-systems integration in aviation. How can we compare these aspects and evolutions with those in healthcare? Can increase of the number of aircraft be compared to increase of the number of patients? For sure, complexity analysis is required in both domains. Of course, we cannot substitute aircraft by patients! However, the need for dispatch is similar. In aviation, interconnectivity among various kinds of aircraft have to be always considered along with attributes such as long-haul flights, failures, aircraft size and performance. Same could said in healthcare, interconnectivity among various kinds of patients have to be always considered along with attributes such as seriousness of their health.

Time pressure criticality is also a factor shared by aviation and healthcare, as all in life-critical systems. We tend to plan trajectories in advance (i.e., 4D trajectories), and this led to trajectory-based operations (TBO). However, this works fine in normal situations, but may fail in abnormal and emergency situations where procedure-based planning rigidity becomes an obstacle to problem-solving flexibility requirements. This involves human skills and knowledge not only on basic flying capabilities but also on embedded systems and their capabilities. This is the reason why organizational automation, especially workflow automation, should be developed considering appropriate function allocation. Cognitive function analysis is strongly advised.

Conclusion and perspectives

Workflow design and management is a matter of functions and structures that need to be articulated correctly. The first difficulty comes from system complexity (i.e., systems of systems, teams of teams, several critical attributes to be considered, emergent phenomena and properties to be incrementally elicited and re-injected into the overall system). It is then important to identify non-linear processes and bottlenecks (bifurcations in the complexity science sense). For that matter, the AUTOS pyramid and cognitive function analysis greatly supported analysis, design and evaluation of highly automated systems in aviation. We started to automate aircraft and we are now automating air traffic. The actual shift is from rigid (low-level) automation to flexible (high-level) autonomy, where authority sharing has to be considered seriously (i.e., who is in charge and accountable to whom).

The socio-technical evolution of aviation systems led to very successful results in terms of accident deaths, decreasing exponentially since the 1980s toward zero³. This evolution includes automation (now digitalization), regulations and a unique safety culture. However, digitalization has become the most important issue in aviation human-systems integration. Software is very easy to modify, but involves us into a virtual world where new phenomena emerge such as cybersecurity. If we can carry out HCD development very early during the life cycle of a system (i.e., activity and function analyses can be performed very early using human-in-the-loop simulations), tangibility testing remains a highest priority (i.e., physical tangibility regarding structures and figurative tangibility regarding functions).

References

- Athènes, S., Averty, P., Puechmorel, S., Delahaye, D., & Collet, C. (2002). ATC complexity and Controller Workload: Trying to bridge the gap. *Proceedings of HCI-Aero'02*, J. Hansman, S. Chatty & G. Boy (Eds.), Boston (USA).
- Bainbridge, L. (1978) Forgotten alternatives in skill and workload. *Ergonomics*, 21, 169-85.
- Boy, G.A. (1983). The MESSAGE system: a first step toward computer-supported analysis of human-machine interactions (in French). *Le Travail Humain Journal*, 46 (2).
- Boy, G.A. (1987). Operator Assistant Systems. *International Journal of Man Machine Studies*. Vol. 27, pp. 541-554.
- Boy, G.A. (1998). *Cognitive Function Analysis*. Praeger/Ablex; ISBN 9781567503777.

³ "Airlines recorded zero accident deaths in commercial passenger jets last year, according to a Dutch consulting firm and an aviation safety group that tracks crashes, making 2017 the safest year on record for commercial air travel" (<https://www.reuters.com/article/us-aviation-safety/2017-safest-year-on-record-for-commercial-passenger-air-travel-groups-idUSKBN1EQ17L>).

Boy, G.A. (2019). A Workflow Study in Aviation. In Cognitive Informatics: Reengineering Clinical Workflow for More Efficient and Safer Care, by Kai Zheng, Johanna Westbrook, Thomas G. Kannampallil, Vimla L. Patel. Springer, USA.

Boy, G.A. (2005). Maturity, Automation and User Experience (Maturité, Automatisation et Expérience des Utilisateurs). *Proceeding of the French Conference on Human-Computer Interaction*. ACM-Press, New York.

Boy, G.A. (2011). (Ed.) *Handbook of Human-Machine Interaction: A Human-Centered Design Approach*. Ashgate, UK. ISBN: 978-0-7546-7580-8.

Boy, G.A. (2013). *Orchestrating Human-Centered Design*. Springer, U.K. ISBN 978-1-4471-4338-3.

Boy, G.A. (2015). On the Complexity of Situation Awareness. Proceedings 19th Triennial Congress of the IEA, Melbourne, Australia, 9-14.

Boy, G.A. (2016). *Tangible Interactive Systems: Grasping the Real World with Computers*. Springer, U.K. ISBN 978-3-319-30270-6.

Boy, G.A. & Grote, G. (2009). Authority in Increasingly Complex Human and Machine Collaborative Systems: Application to the Future Air Traffic Management Construction. *In the Proceedings of the 2009 International Ergonomics Association World Congress*, Beijing, China.

Callon, M., (1991), Techno-economic networks and irreversibility, in Law, J., (Eds.), *A sociology of monsters: essays on power, technology and domination*, Routledge, London, pp. 132-161.

Carroll, J.M. (1995). *Scenario-based design: Envisioning work and technology in system development*. Wiley, New York.

Cooper, G.E., White, M.D. & Lauber, J.I. (1980). Resource Management on the Flight Deck. Proceedings of a NASA/Industry Workshop Held at San Francisco, California June 26-28, 1979. NASA Conference Publication 2120, NASA Ames Research Center, CA.

Cummings, M. L. & Tsonis, C.G. (2006). Partitioning Complexity in Air Traffic Management Tasks. *International Journal of Aviation Psychology*, Volume 16, Issue 3 July 2006, pp. 277–295.

Doyon-Poulin, P., Ouellette, B. & Robert, J.M. (2014). Effects of Visual Clutter on Pilot Workload, Flight Performance and Gaze Pattern. *Proceedings of HCI-Aero 2014*, Santa Clara, CA. Also in the ACM Digital Library.

Endsley, M.R. (1988). *Situation awareness global assessment technique (SAGAT)*. Paper presented at the National Aerospace and Electronic Conference (NAECON), Dayton, OH.

Endsley, M.R. (1996). Automation and situation awareness. In R. Parasuraman & M. Mouloua (Eds.), *Automation and human performance: Theory and applications*, Mahwah, NJ: Laurence Erlbaum, pp. 163-181.

Fitts, P.M. (Ed.) (1951). Human engineering for an effective air navigation and traffic control system. National Research Council, Washington, DC.

Boy, G.A. (2019). A Workflow Study in Aviation. In *Cognitive Informatics: Reengineering Clinical Workflow for More Efficient and Safer Care*, by Kai Zheng, Johanna Westbrook, Thomas G. Kannampallil, Vimla L. Patel. Springer, USA.

Gander, P., Graeber, C. & Belenky, G. (2011). Operator Fatigue: Implications for Human-Machine Interaction. *Handbook of Human-Machine Interaction: A Human-Centered Design Approach*. Ashgate, U.K.

Grote, G. (2004). Uncertainty management at the core of system design. *Annual Reviews in Control*, 28, pp. 267-274.

Hamilton, S. (2001). Thinking Outside the Box at the IHMC. *IEEE Computer*, January, 0018-9162/01, pp. 61-71.

Hart, S.G. (1982). Theoretical Basis for Workload Assessment. TM ADP001150, NASA-Ames Research Center, Moffett Field, CA, USA.

Helmreich, R.L., Merritt, A.C., & Wilhelm, J.A. (1999). The evolution of Crew Resource Management training in commercial aviation. *International Journal of Aviation Psychology*, 9(1), 19-32.

Hemingway, C.J. (1999). "Toward a Socio-cognitive Theory of Information Systems: An Analysis of Key Philosophical and Conceptual Issues", *IFIP WG 8.2 and 8.6 Joint Working Conference on Information Systems: Current Issues and Future Changes*. Helsinki, Finland: IFIP. 275-286.

Hilburn, B. (2004). Cognitive complexity in air traffic control : A literature review. Project COCA - CComplexity and CAPacity. EEC Note No. 04/04.

Hoc, J.M. & Lemoine, M.P. (1998). Cognitive evaluation of human-human and human-machine cooperation modes in Air Traffic control. *The International Journal of Aviation Psychology*, 8 (1), pp. 1-32.

Hutchins, E. (1995). How a Cockpit Remembers its Speeds. *Cognitive Science*, 19, 265-288.

Latour, B., (1987). *Science in action: how to follow scientists and engineers through society*. Harvard University Press, Cambridge, MA.

Laudeman, I. V., Shelden, S. G., Branstrom, R. & Brasil, C.L. (1998). Dynamic Density. An Air Traffic Management Metric. California: National Aeronautics and Space Administration, Ames Research Center, NASA/TM-1998-112226.

Leveson, N., Dulac, N., Marais, K. & Carroll, J. (2009). Moving Beyond Normal Accidents and High Reliability Organizations: A Systems Approach to Safety in Complex Systems. *Organizational Studies* (Sage Publishers). Vol 30, Issue 2-3.

Mogford, R.H. (1997). Mental Models and Situation Awareness in Air Traffic Control. *The International Journal of Aviation Psychology*. Volume 7, Issue 4, pp. 331-341.

Paulk, M., Curtis, B., Chrissis, M. & Weber, C. (1993). *Capability Maturity Model for Software (Version 1.1)*. Technical Report CMU/SEI-93-TR-024.

Boy, G.A. (2019). A Workflow Study in Aviation. In *Cognitive Informatics: Reengineering Clinical Workflow for More Efficient and Safer Care*, by Kai Zheng, Johanna Westbrook, Thomas G. Kannampallil, Vimla L. Patel. Springer, USA.

Sharples, M., Jeffery, N., du Boulay, J.B.H., Teather, D., Teather, B. & du Boulay, G.H. (2002). Socio-cognitive engineering : A methodology for the design of human-centered technology. *European Journal of Operational Research*, Volume 136, Issue 2, January, pp. 310-323.

Sperandio, J.C. (1980). *La psychologie en ergonomie (Psychology in ergonomics)*. PUF, Paris.

Schwaber, K. (1997). Scrum development process. In *OOPSLA Business Objects Design and Implementation Workshop Proceedings*, J. Sutherland, D. Patel, C. Casanave, J. Miller & G. Hollowell, Eds. Springer, London, U.K.

Sutherland, J. (2014). *Scrum: The Art of Doing Twice the Work in Half the Time*. Crown Business. ISBN-13: 978-0385346450.

Wiener, E.L. (1989). *Human factors of advanced technology ("glass cockpit") transport aircraft*. (NASA Contractor Report No. 177528). Moffett Field, CA: NASA-Ames Research Center.

Wickens, C. D. (1992). *Engineering Psychology and Human Performance* (2nd ed.). Harper Collins Publishers Inc., ISBN: 0673461610, New York.

<https://files.acrobat.com/a/preview/5078a021-37e2-4f34-9f55-8acc7cf913bb>