Possible dynamic allocation of functions in support of airborne spacing operations.

P. Le Blaye, T. Coye de Brunélis

Possible dynamic allocation of functions in support of airborne spacing operations

Le Blaye, Patrick

ONERA / BA 701/ 13 661 Salon Air, France
E-mail: Patrick.Le_Blaye@onera.fr

Phone: + 33 (0)4 90 17 01 30

Coye de Brunélis, Thierry

ONERA - University of Provence/ BA 701/ 13 661 Salon Air, France
E-mail: Thierry.coye_de_brunelis@onera.fr

Phone: + 33 (0)4 90 17 01 45

ABSTRACT

Dynamic function allocation (DFA) refers to systems in which both the user and the system can initiate changes in the level of automation in real time (Scerbo, 2005) in order to reach optimal system performances. Even if this concept seems promising for future systems, the sharing of authority between human and machine remains a major issue. Who should be in charge of DFA triggering?

In order to contribute to the elucidation of the problem, an experiment has been conducted at ONERA - the French Aerospace Lab. Two groups of participants were placed in a simplified aircraft cockpit to perform a piloting task featuring an airborne spacing application with conflict avoidance.

Our results emphasize the impact of DFA implementation on pilot performances and the particular importance of communication for the acceptability by the pilots of DFA and of the sharing of authority.

Keywords

Dynamic function allocation, airborne spacing, sharing of authority, trust in system

INTRODUCTION

The research work presented in this paper addresses the allocation of functions between a human user and an automated system, focusing especially on its dynamic aspects. A main issue is to determine which criteria should guide the dynamic function allocation process so that the resulting system can accommodate the various individual preferences or strategies to perform an intended task.

The introduction of a new concept of operation always raises the issue of the allocation of the functions required to perform the new task. This is actually the case with the emergence of new concepts in civil transport aviation, including the delegation of tasks currently performed by the air controller to the flight crew, as well as in military operations, which involve an increasing number of uninhabited autonomous systems.

The paper starts with a presentation of dynamic function allocation and its related issues. It then describes the conditions and the results of a human-in-the-loop experiment involving two groups of participants with different levels of expertise in the context of a new concept of operation envisaged to increase the capacity of the civil transport aviation system. A discussion concludes the paper.

THE ISSUES

Dynamic Function Allocation

Dynamic Function Allocation (DFA) refers to the variable distribution of functions in real time between the system and the operator(s) to achieve optimal system performance (Cook *et al*., 1999). The first idea behind DFA may be to regulate the task load of the operator in order to avoid that he/she may become overloaded for instance in case of an emergency, or on the opposite, that he/she could be subject to complacency if the automation took over all the tasks. A more ambitious goal for DFA is to insure an appropriate level of situation awareness, especially in supervisory control tasks, or to maintain the operator's specific know-how (Kaber *et al*., 2001; Parasuraman *et al*., 1998).

Although the concept of DFA appears to be promising for the design of future systems, some possible drawbacks remain regarding its implementation. One of the main issues is the authority sharing between the technical system and the operator: who should actually initiate the changes of allocation or of level of automation?

Human-machine interaction and levels of automation

Sheridan & Verplanck (1978) described different types of human-machine interaction in terms of levels of automation (LOA). They provided a scale of ten possible LOAs which is presented in a simplified version on Fig. 1 (Sheridan, 1992).

High	10. The computer decides everything, acts autonomously, ignoring the human, or
	9. informs the human only if it, the computer, decides to, or
	8. informs the human only if asked, or
	7. executes automatically, then necessarily informs the human, or
	6. allows the human a restricted time to veto before automatic execution, or,
	5. executes that suggestion if the human approves, or
	4. suggests one alternative, or
	3. narrows the selection down a few, or,
	2. offers a complete set of decision/action alternatives, or
Low	1. The computer offers no assistance: human must take all decisions and actions

Fig. 1: Levels of automation of decision and action selection

The main benefit with this scale is that it provides a clear linear view of the possible solutions that may be envisaged when implementing an automated function, from a minimal level (no assistance) to a maximal level (automation does everything with no feedback to the user). The role of the human is seen as inversely proportional to the role of automation.

Static allocation of functions: H-M distribution of control

Some methods have been proposed to guide the static allocation of functions when designing the system required in support of a new task (e.g. Dearden et al., 2000; Grote et al., 2000). They typically propose the following steps:

- Identify the functions required to perform the new task (functional analysis)
- Decide which functions should be kept in charge of the operator (based on the role of the operator) or –even partially- automated
- Determine the appropriate level of automation of each partially automated function.

The last step requires some further considerations regarding what is actually an appropriate level of automation.

In particular, the level 6 of the Sheridan & Verplanck scale appears as a critical threshold regarding the involvement of the human operator, as it is the higher level where the operator keeps an active role. Above that level (LOA>6), the operator becomes a passive agent in terms of decision and action implementation. The long phases of relative inactivity induced by the highest level of automation (LOA>6) may contribute to an out-of-the-loop syndrome or may negatively affect the situation awareness or the know-how of the operator.

A recommendation is to leave the critical decisions to the operator (LOA $\leq$ 6) and to delegate the less critical decisions to the system (LOA>6) (Parasuraman et al., 2000). A consensus is also emerging to say that moderate levels of automation should be applied to the information analysis and decision making stages of information processing, while higher levels may be envisaged for the sensory-motor stages: information acquisition and action implementation (Di Nocera et al., 2005; Kaber et al., 2005).

DFA and human-machine cooperation

DFA obviously impacts human-machine cooperation which is also a broad area of research. Millot (1988) for instance defines 2 cooperative structures of HM cooperation, vertical and horizontal. The vertical cooperation refers to the fact that the operator is responsible for all tasks to be performed and decides explicitly with or without a system assistance who does what. The horizontal cooperation refers to a dynamic function allocation between human and machine where task or function can be dynamically allocated to human or machine according to triggering criteria. The triggering can be operated by the human or by the system.

Authority and criteria for the dynamic allocation of functions

By nature, a dynamic allocation of functions in fact moves the decision regarding the appropriate level of automation from the design stage to the actual operation of the system. The dynamic allocation of function raises the central issue of the *authority* regarding the allocation decision (Inagaki, 2003): who between the human and the system should be given the authority to initiate the change of allocation, i.e. the shift from one level of automation to another?

If the initiative is left to the operator, the DFA is termed *explicit*. There the operator is kept in control of the system, but the initiation of the changes may induce an extra workload or may produce its expected effects with an unnecessary time delay.

The initiative may rather be given to the system; the DFA is then called *implicit*. The system there needs to be able to identify a situation requiring a change of allocation (allocation criteria) and to decide which change has to be made (allocation change).

The mechanism for dynamic allocation control is generally implemented as *production rules* (if criteria C is fulfilled then change allocation from A1 to A2). Four types of criteria have been proposed on which the changes of allocation may be based:

- Critical events based, where the allocation is determined by an objective and observable number of events occurring at any time.
- Operator performance based, where the allocation is made dependent upon the operator’s observable performance of tasks.
- Physiological measurement based, where the allocation is determined by physiological indicators of the operator’s overload or situation awareness.
- Model based, requiring a predictive model of operator performance.

DFA IN THE CONTEXT OF NEW CIVIL AVIATION OPERATIONS

Objective

This experiment has been conducted in order to identify the effects of different solutions for a dynamic implementation of functions in support of the pilot of a civil aircraft performing a particular task. This simulated task corresponds to a new concept of operations involving a delegation of tasks currently performed by the air traffic controller to the flight crew (maintain a given spacing from a leading aircraft and detection of potential conflicts with the surrounding traffic).

A central issue of DFA is whether it is possible to design an automated system which can accommodate to different users' strategies or preferences. In other words, what are the preferences of users with different levels of expertise having several levels of automation at their disposal, and so, what should be the control mechanisms for a dynamic allocation of functions in order to better match these preferences?

To address this issue, four experimental conditions have been tested, involving different types of function allocation ranging from static allocation up to implicit DFA and applied to the various modes available for the pilot to perform a conflict avoidance maneuver. A detailed description of the experiment is provided below.

Experimental set-up

Simulation environment

The experiment makes use of a simplified aircraft cockpit, featuring a projected view of the external world with a head-up display symbology for short term aircraft control, and a head-down navigation and traffic display. A specific interface based on a touch screen is also installed on the right side of the cockpit.

The aircraft model used here is a generic civil transport aircraft, with a simplified flight control system (autopilot and auto-throttle) providing various modes and guidance laws, plus some advanced functions which have been developed specifically to support the proposed operational concept and this experiment. These modes and functions are described later in this paper.

Navigation and traffic display

The primary task of the pilot is to follow a pre-defined flight plan, using this display. It integrates the navigation data (flight plan) together with the traffic information (symbols, speed, and altitude) (Fig. 2). The traffic information is assumed to be perfectly reliable (i.e. no traffic may be ignored by the display).

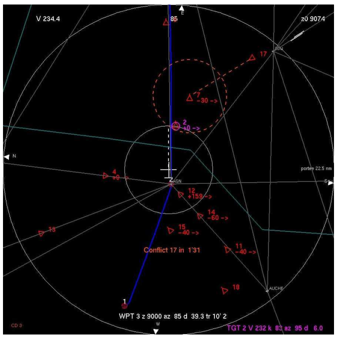

Fig. 2: The navigation and traffic display showing a potential conflict

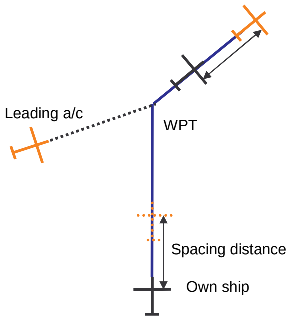

Fig. 3: The spacing task ("heading then merge")

Airborne spacing

The pilot has also to capture and maintain a spacing distance from a leading aircraft. This spacing task is delegated by the controller. At the beginning of the experiment, the aircraft is located at a distance from the leader aircraft larger than the instructed spacing distance (6 NM). The pilot's task is then to merge at the next waypoint at the spacing distance form the leader and then to maintain this distance along the whole flight plan, with an acceptable margin of +/- 0.5 NM (Fig. 3).

This task could be performed manually, based on the traffic information provided by the navigation and traffic display. However, the task is costly in terms of workload and it is not considered as a central task in the pilot's role. So, the spacing function appears as a candidate for at least partial automation and a specific speed control law has been developed to support this task, as described below.

Conflict detection and resolution
The navigation and traffic display provides an improved awareness of the surrounding traffic, which is not only limited to the assigned leading aircraft for the spacing task. As a consequence, the monitoring of the traffic, including the detection of potential conflicts (at less than 5 NM horizontal distance and +/- 1000 feet vertical distance, in less than 150 seconds), appears as a new task to be performed by the crew, and a conflict detection function has been developed in order to support this task.

When a potential conflict is detected, this function displays a message including the time to conflict and shows the predicted trajectories of the two potentially conflicting aircraft (Fig. 2). Today, such a function cannot be assumed to be perfectly reliable, as some surrounding aircraft may not be properly equipped. For this reason, and also with the aim to keep the participant actively involved in the conflict detection task, the automated function was disabled for one potential conflict out of 4.

Available modes
Three different types of modes (control, navigation and speed) are available to the pilot in this experiment, with two possible modes in each type, one manual and one automated. These modes are further detailed below:

Control modes :
- MAN Manual mode, in fact a piloting mode through control laws which provide direct control of the flight path shown on the head-up display.
- AP Auto pilot selected mode, the auto pilot automatically guides the aircraft to the waypoint currently selected by the pilot.

Navigation modes :
- WPT Basic waypoint mode, the navigation data and a guidance symbol to the waypoint selected by the pilot are shown on the head-up display.
- NAV Same as WPT, plus an automatic change of the selected waypoint to the next waypoint in the flight plan, when passing over the current waypoint.

Speed/Thrust modes :
- RSPD A recommended speed value is calculated to insure the required spacing and provided to the pilot on the head-up display.
- AUTO The same speed control law is used to actually command the thrust level, so that the spacing distance is automatically captured and maintained.

The current mode combination is displayed and controlled through a special touch-screen interface, inspired from the displays existing on modern civil aircraft (Fig. 4).

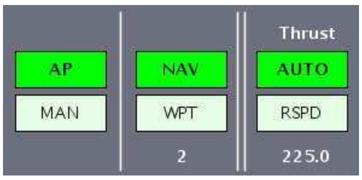

Fig. 4: The touch screen interface for flight modes display and control

Scenarios
The scenarios consist of a pre-defined flight plan with a moderate level of surrounding traffic. The trajectories of the traffic other than own ship are pre generated, based on flight plans especially adapted to provoke the desired interactions.

A number of 4 scenarios is used. Each scenario lasts for 12 minutes and includes the encounter of two potential conflicts, plus false potential conflicts (traffic at a flight level far from own ship, or traffic with changing trajectories).

Participants
Two groups of ten participants were involved in the experiment. The first group was composed of volunteer, non pilots, aeronautical engineers. The second group was

composed of student pilots having a limited flight experience.

Training and experimental trial

For each participant, a session of 2 hours is dedicated to training. The participants are asked to read a training manual and then they practice the simulation with the scenarios especially prepared to put the subject in the various possible situations. The experimental trial starts once the subject feels comfortable with their tasks.

Each participant proceeds to 4 flights, each with a different function allocation condition. The flight parameters, the main events and the pilot's inputs are recorded for post analysis of the simulation. A detailed debriefing is then conducted with the subject.

Tasks

The participants were asked to perform the following tasks, ranked by order of priority:

1) To follow their flight plan at the assigned flight level (FL 90);

2) To capture and to maintain the spacing distance with their assigned leading aircraft;

3) To report any potential conflict with the surrounding traffic, even before it was detected by the assistance function, and, if asked to by the controller, to perform a vertical avoidance manoeuvre to an assigned flight level (FL 70 or 110), and once cleared of conflict, to report and to come back to the original flight level.

4) When prompted to by an audio signal and a visual message, to give an evaluation of their workload, using a 5 levels Instantaneous Self-Assessment (ISA) keypad.

The production rules for implicit DFA

The implicit DFA is implemented as production rules based on critical events, as shown by the following pseudo-algorithms:

If (conflict_detected) then
previous_mode_combination = mode_combination
mode_combination = (MAN-WPT-RSPD)

If (no_conflict) and (flight_level==flight_plan_value+/-60ft) then
mode_combination = previous_mode_combination

When activated, these rules control the change of the current mode combination, reverting to a manual mode combination so that the pilot has control when a conflict is to be avoided, and coming back to the previous (preferred) mode combination once the conflict has been avoided and the aircraft is back on its assigned flight level.

Experimental conditions

The experimental conditions relate to the type of function allocation. Four different conditions are tested:

C1) Static: Only the mode combination at the lowest level of automation is used during the total duration of the scenario. This condition corresponds to level 1 of the LOA scale.

C2) Explicit: Any mode combination (among the eight possible combinations) can be used during the scenario, as preferred by the participant. Note that the avoidance manoeuvres require the use of the manual control mode (MAN). This condition corresponds to level 2 of the LOA scale (Millot's vertical cooperation).

C3) Semi-implicit: The modes are dynamically controlled by production rules as described above, but the participant is prompted by a message a few seconds before the mode combination changes; he/she can validate or invalidate (veto) the proposed change of modes before the time delay expires. This condition corresponds to level 6 of the LOA scale (Millot's horizontal cooperation).

C4) Implicit: The modes are dynamically controlled by the production rules as in the previous condition, but the change of modes is made automatically without any prompt. This condition corresponds to level 10 of the LOA scale (Millot's horizontal cooperation).

RESULTS

A one-way ANOVA with one within-subjects factor was carried out on the number of altitude overshoots made at the current flight level (FL) 90. For both groups, the experimental conditions appear to have a significant effect ($F(3,27) = 17,837$, $p < .0001$ and $F(3,27) = 11,199$, $p < .0001$). A Newman-Keuls post-hoc test shows that participants perform less overshoots in C2, C3 and C4 than in C1 (Fig. 5).

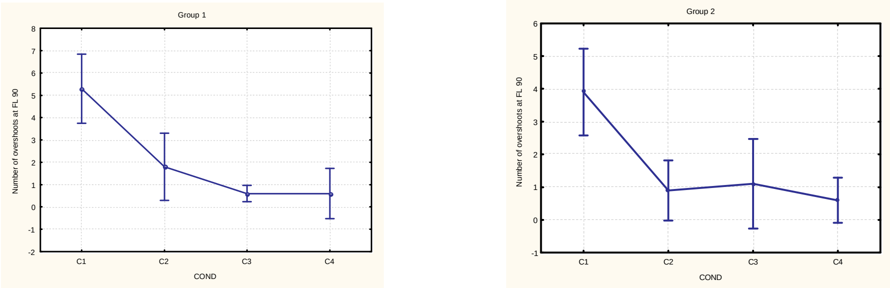

Fig. 5 (a and b): Number of overshoots at FL 90
for group 1 (left graph) and group 2 (right graph)

A similar ANOVA was performed for the overshoots made at FL 70 and 110 used during conflict avoidance. For the first group, it only appears a trend of effect of the experimental conditions on the number of overshoots ($F(3,27) = 2,551$, $p <.0765$). For the second group, the ANOVA shows a significant effect ($F(3,27) = 5,226$, $p <.0005$). The Newman-Keuls post-hoc analysis indicates that the number of overshoot made in C4 is higher than those made in C1 to C3. (Fig. 6)

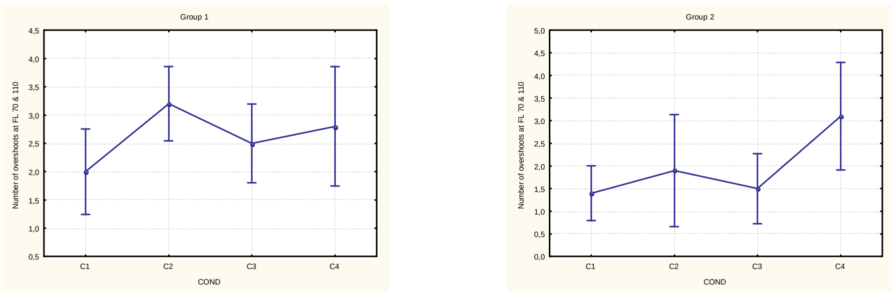

Fig. 6 (a and b): Number of overshoots at FL 70 & 110
for group 1 (left graph) and group 2 (right graph)

Otherwise, a one-way ANOVA was carried out regarding the workload declared by the participants (ISA value). A main effect of the type of allocation appears for both groups ($F(3,27) = 16,451$, $p <.0001$ and $F(3,27) = 17,180$ $p <.0001$). Newman-keuls post-hoc analyses of the 1st group data show that the ISA value is higher in C1 than in C2 to C4. For the 2nd group, post-hoc analysis indicates that ISA value is higher in C1 than in in C3 and C4 which themselves are higher than C2. (Fig. 7)

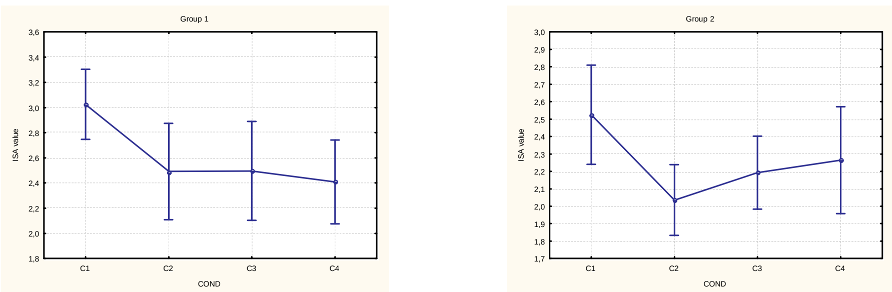

Fig. 7 (a and b): ISA value for group 1 (left graph) and group 2 (right graph)

From the 1st group, 50% of the participants ranked the semi-implicit condition (C3) as their preferred condition; at the same time, the static and implicit conditions (C1 and C4) are rated as the less preferred conditions by respectively 60 and 50% of the participants. Regarding the group 2, 50 % of the participants ranked the semi-implicit condition (C3) and the explicit condition (C2) as their preferred one. As for the first group, the static and implicit conditions (C1 and C4) are ranked as the less preferred conditions, by respectively 60 and 40 % of the participants (Table 1).

	Rank							
	1		2		3		4	
Group	G1	G2	G1	G2	G1	G2	G1	G2
Static	0	0	10	10	30	30	**60**	**60**
Explicit	**30**	**50**	60	50	10	0	0	0
Semi-implicit	**50**	**50**	10	40	30	10	10	0
Implicit	20	0	10	10	20	10	**50**	**40**

Table 1: Preferences expressed by the participants of the 2 groups for the different types of DFA (% of participants)

The main results are summarized in the table below.

	Group 1	**Group 2**
# overshoots at FL90	(C2,C3,C4) < C1	
# overshoots at FL70/110	Trend	(C1,C2,C3) < C4
Workload (ISA)	(C2,C3,C4) < C1	C2 < (C3,C4) < C1
Preferred conditions	C3 (50%) – C2 (30%)	C3 (50%) – C2 (50%)
Less preferred conditions	C1 (60%) – C4 (50%)	C1 (60%) – C4 (40%)

Table 2: Summary of the main results
("Cn<Cm" should be read as "less effect in Cn than in Cm)

The participants declare having two different kinds of strategies when reporting a potential conflict: either they «report and forget» or they prefer to «wait then report». An ANOVA indeed confirmed a main effect of the strategy over the number of potential conflicts detected ($F(1,24) = 19{,}785$, $p = .002$) for the non-pilot group, while the ANOVA for the other group indicates no main effect ($F(1,24) = 0{,}18$, $p = .682$).

DISCUSSION

A first result of our experiment is that the piloting performance (overshoots at FL 90) is better in the conditions allowing the use of automation than in the static-manual condition. This result confirms that the automated functions works as excepted and can actually be used to improve the precision in the task performance.

The second result gives an insight on the effect of automation when reverting to manual piloting. The ANOVA indicates that the participants –especially the pilots- make more altitude overshoots when avoiding a conflict (at FL 70/110) when the allocation is implicit. It shows a negative effect on the piloting performance of the implicit automated reversion from autopilot to manual piloting when performing the avoidance maneuvers. It may also reveal that the recovery of the manual control after a long period with no direct control action results in more deviations than if all the tasks were performed in a manual mode.

The only difference between the semi-implicit (C3) and the implicit condition (C4) is the feedback provided to the participant before the change of allocation occurs. The results (overshoot and preferences) show that this feedback is essential for the acceptability of DFA. Although the system eventually provides the same allocation change, the implicit condition induces a lower performance and is rejected by a majority of participants. The absence of HM communication and possibly its timing substantially disturbs the interaction.

The results regarding the declared workload show an effect of the type of allocation.

While for both groups the full manual condition implies a relatively high workload as expected, the lowest workload is declared by the pilot's group in the explicit allocation. This suggests that a workload may be induced by the monitoring of the (semi-)implicit behavior of the system. The participants actually prefer to keep control of the system even if their resulting control task load is higher. Moreover, some mentioned that the absence of communication in the implicit condition actually induce an extra workload as they have to explore the interface to confirm the state of the system and the current allocation of its control modes. In this case, the DFA results in an increase of workload while it is expected to decrease it.

In the first group, half of the participants prefer the semi-implicit allocation. However, it has to be noticed that 20% of them consider that the implicit DFA is the most appropriate. These participants declare not being affected by the automatic reversion of the piloting modes. Their lack of experience associated to an *a priori* trust in automation could explain those counter-intuitive results. In the 2nd group, the implicit DFA is never ranked first while the explicit triggering is preferred by 50 %. Those two variations could be due to the level of expertise of the participants. The more experimented keep the system manually in control. Most of them explain this choice because "they are the pilot and so are responsible for the flight's safety". Implicitly, they do not have trust in the system *a priori*. Non experimented pilots more easily give trust to automation (relying on it to regulate their own workload) contrary to more experimented pilots.

The acceptability of the semi-implicit allocation, according to the pilot debriefing, relies on the way the system provides feedback on what it is doing. With the feedback, the participants feel they still have control and they perceive an appropriate aiding of the system. The sharing of authority in this case seems to be well accepted. So the acceptability of the sharing of authority by the pilots may not only lie in the level of automation by itself but rather in the associated situation awareness.

Last, different strategies appeared among the participants regarding their reporting of potential conflicts. This result for the first group may be linked to the strategies developed by the individuals to regulate their workload. The participants using a "declare and forget" strategy declare a potential conflict early in order to evacuate the monitoring task and to better concentrate on the control task. On the other hand, the participants using a "wait then declare" strategy favor the reliability of their detection rather than the time margin before the potential conflict. However, this effect was not significant for the second group: the pilots declare adapting their strategy regarding the particularity of each potential conflict (back threat, front threat, lateral threat).

CONCLUSION

The results of our experimentation stress 3 issues related to the DFA implementation: the acceptability of the sharing of authority, the importance of the communication in the DFA acceptability, and the effect of different levels of expertise.

The acceptability of DFA depends on the user's experience. In our experiment, although the DFA is related to a low level task (reversion of piloting modes), its acceptability is lower for the participating pilots than for the non-pilots. The feeling of loss of control in authority sharing between the pilot and the system is often expressed as a central issue for the DFA acceptability by the pilots. This is certainly to be considered for the implementation of DFA in a real system.

Secondly, in case of DFA, the way the system provides some feedback to the pilot about tasks it realized appears as a central issue to maintain the pilot's situation awareness. Furthermore, the way DFA is designed could create new added tasks for the pilot (*e.g.* supervising what the system is doing, looking for information to know who is doing what). The acceptability of a given implementation may not depend on the nature of the allocated functions but rather more on the particular appropriation of the functions by each individual user and on the way the communication is implemented.

Finally, the limitations of this experiment require to moderate the conclusions and should be taken into account for further experiments: participant profiles and background that are not directly linked to the end users', representativeness of the surrounding traffic, complexity of the task handed by the DFA.

ACKNOWLEDGEMENTS

We would like to thank the participants in the experiment, particularly the student pilots from the French air force academy. We especially thank Mr Gal. Maire, Mr. Lt-Co Versailles, and Mr. Col. Partiot for their support to the experiment.

REFERENCES

Cook C., Corbridge C., Morgan C. and Turpin E., 1999, Investigating methods of dynamic function allocation for naval command and control. International conference on human interfaces, 21-23 June 1999. Conference publication N°463, IEE.

Dearden, A., Harrison, M. & Wright, P., 2000, Allocation of function: scenarios, context and the economics of effort. Int. J. Human-Computer Studies, 52, 289-318.

Di Nocera F., Lorenz B. and Parasuraman R., 2005, Consequences of shifting from one level of automation to another: main effects and their stability. In D. de Waard, K.A. Brookhuis, R. van Egmond, and Th. (Eds.), Design, Safety, and Management (pp. 363 - 376). Maastricht, the Netherlands: Shaker Publishing.

Grote G., Ryser C., Wafler T., Windischer A. and Weik S., 2000, KOMPASS: a method for complementary function allocation in automated work systems, International Journal of Human-Computer Studies, 52, pp 267-287.

Inagaki T., 2003, Adaptive automation : sharing and trading of control, In E. Hollnagel (Ed.), Handbook of cognitive task design, pp. 147-169. Mahwah, NJ: Lawrence Erlbaum associates.

Kaber D.B., Riley J.M., Tan K.-W. and Endsley M.R, 2001, On the design of adaptive automation for complex systems, International Journal of Human-Computer Studies, 5(1), pp. 37-57.

Kaber D.B., Wright M.C., Prinzel L.J., Clamann M.P., 2005, Adaptive automation of human-machine system information processing functions. Human Factors. Volume 47, Number 4, pp. 730-741(12).

Millot, P., 1988, Supervision des procédés automatisés. Editions HERMES. Paris.

Parasuraman R., Mouloua M. and Hilburn B. 1998, Adaptive aiding and adaptive task allocation enhance human-machine interaction, In M.W. Scerbo and M. Mouloua (Eds.), Proceedings of the third conference on automation technology and human performance, 119-123. Norfolk, VA: USA.

Parasuraman R., Sheridan T.B., and Wickens C.D., 2000, A model for types and levels of human interaction with automation, IEEE Transactions on systems, man and cybernetics, Vol.30, No.3, 286-297.

Scerbo, M.W. 1996, Theoretical perspectives on adaptive automation. In R. Parasuraman and M. Mouloua (Eds.), *Human performance in automated systems: Theory and* applications, 37-64. Mahwah, NJ: Erlbaum.

Sheridan T.B., 1992, Telerobotics, automation, and human supervisory control. Cambridge, MA: MIT Press.

Sheridan T.B. and Verplanck W.L., 1978, Human and computer control of undersea teleoperations (Man-Machine Systems Laboratory, Tech. Rep.x). Cambridge, MA: MIT.