COOPERATIVE IMPLICATIONS OF THE ALLOCATION OF FUNCTIONS TO HUMANS AND MACHINES

Jean-Michel HOC

CNRS and LUNAM University – IRCCyN
1, rue de la Noé — B.P. 92101 — 44321 NANTES Cedex 3, France
Contact : Jean-Michel.Hoc@irccyn.ec-nantes.fr

Christine CHAUVIN

University of South-Brittany — CRPCC
Centre Universitaire de Recherche — rue Saint Maudé — 56260 LORIENT, France
Christine.Chauvin@univ-ubs.fr

Document written in 2011

ABSTRACT

Objective: This paper aims at pointing out the cooperative aspects of function allocation to humans and machines in complex systems. Background: It is based on the observation that interference exists mostly between functions devoted to each of these agents. Furthermore, when interfering functions are distributed among several agents, the latter need to perform cooperative activities to manage the former. Method: It uses functional, structural, and social frameworks to distinguish between various levels and forms of cooperation, and different attitudes toward automation. Such frameworks allow us to consider the cooperative implications of the main FA approaches: Fitt’s MABA-MABA method, the LOA (Level Of Automation) taxonomy and the DFA (Dynamic Function Allocation) approach. Results: It emphasizes the adaptive ability of humans who are given supervision of the HMS, and consider FA in terms of function delegation. This paper emphasizes the need to elaborate and maintain a COFOR (COmmon Frame Of Reference) to manage interference. It puts forward the cost of cooperative activity and the need to evaluate its efficiency in terms of the relationship between cost and benefits. Conclusion: Finally, it proposes the MOC (Mode Of Cooperation) framework as an alternative to LOA and presents an application of MOC to car-driving assistance domain. Application: The paper identifies strategic decisions in the design of Human-Machine Systems in order to improve Human-Machine Cooperation and system safety.

KEYWORDS: Function Allocation; Human-Machine Cooperation; Level Of Automation; Adaptive Automation; Mode Of Cooperation.

1 INTRODUCTION

This paper focuses on dynamic situations. Initially, these included process control situations within the process industry domain (Bainbridge, 1978; Hollnagel, Mancini, & Woods, 1988). Later, dynamic situations were extended to include transportation and other situations that share common properties. In the main, these include the following (Hoc, 1993).
Partial control. Human operators do not fully and directly control the dynamics of a situation. Their actions are combined with technical process dynamics. Their actions may also be distributed between fairly autonomous machines and themselves. This implies the diagnosis of the effects of the partners’ actions, as well as one’s own actions and the environment’s dynamics.

Time constants and constraints. It is important to make correct decisions; above all, though, decision implementations must be carried out in time to deal with the temporal aspects of the situation’s dynamics. Cooperation needs communication, and, in this kind of situation, communication has to take place quickly.

Uncertainty and risk. Human operators are not alone in influencing the situation. Thus, they must consider uncertainty. Then, because of time constraints, they must act before certainty is reached. Human operators are better than machines at adapting to unforeseen situations. Thus, machines often return control to their human operators.

As already noted in these situations, more often than not, functions are de facto distributed among various operators and machines. Whatever the timing of such distribution (i.e., before or during activity development), and whatever the rationale behind it (i.e., after mature reflection or in an opportunistic way), it is possible to identify three explicit or implicit Function Allocation (FA) strategies.

Machine-centered strategy or residual strategy. Engineers introduce automation wherever they can, allowing human operators to do the rest. This type of FA is widely implicit. If the functions allocated to machines are well defined, engineers are not necessarily aware of the nature of the functions allocated to human operators.

Human-centered strategy or assistance strategy. This strategy has been mainly defended by ergonomists who consider that machines only serves to assist their human operators. In this case, FA is less implicit, but it is not compatible with innovation and autonomy on the machine design side.

Human-Machine System-centered strategy. Designers consider the whole Human-Machine System (HMS) and its task. FA is explicit in order to optimize HMS efficiency. Some cooperation between humans and machines is considered when they contribute to the main task by performing their own (complementary) functions.

For decades, many authors have provided designers with valuable contributions on FA. Such contributions will be reviewed here. This current paper aims mainly to describe the consequences of implicit or explicit FA on Human-Machine Cooperation (HMC). It will relate to Hoc’s (2001) definition of cooperation, which placed more stress on the notion of cooperative activity (extra activity developed when agents do not work alone) than on the notion of cooperative structure (relationships between the agents). From the former viewpoint, cooperation consists of managing interference between agents by the agents themselves, both beforehand and in real time, in order to facilitate individual or common tasks. Interference is not always negative (e.g., a disturbance); sometimes, it can be positive (e.g., mutual control by cross checking). More often than not, functional decomposition defines the interfering functions, so that the cooperation viewpoint is inescapable when distributing such functions between distinct agents. Following this introduction, Section 2 will delineate three theoretical frameworks on cooperation: Hoc’s functional approach, Schmidt’s (1994) structural-
functional approach, and the attitudinal and social viewpoint. These three frameworks will be the main references for the whole paper.

Sections 3 to 6 will briefly review FA studies from a historical perspective, namely: Fitts’ method, Sheridan’s Levels Of Automation (LOA), FA criteria, and dynamic FA (DFA). At each step, we will underline the cooperative implications of the approach. Section 3 will review the allocation of elementary functions, based mainly on Fitts’ list, which identifies the best performance either by the human or the machine. It will also consider how this has given way to more integrated functions. In this way, we will respond to criticism leveled at Fitt’s list. Section 4 will review the movement from the Sheridan’s concept of Level Of Automation (LOA), which aims to describe increasing automation in HMS and its consequences on performance, to the concept of Mode Of Cooperation (MOC). This movement is contemporary. Nowadays, LOA is the most widespread framework used to describe FA. We will argue that LOA and its re-interpretations, including DFA, are machine-centered and hide implications in terms of cooperation.

Allocation can be static (beforehand) or dynamic (in real time). In any case, the allocation or evaluation criteria are the same and must be considered when making allocation decisions. Section 5 will review the main criteria proposed in the literature. Section 6 will discuss dynamic function allocation specifically (DFA). More often than not, the term “Function Allocation” is used; however, in some cases, “Task Allocation” is also used, and more rarely, “Role Allocation”. For humans, the concepts of task, role, and function are not equivalent. We will argue that, when distributing entities between humans and machines, it is preferable to distribute functions rather than tasks. Task definitions are not the same for humans and machines, nor are roles, because machines cannot have responsibilities. Section 7 will take stock and organize the generic implications of the application of the cooperation viewpoint to FA. Finally, Section 8 will develop MOC (Mode of Cooperation) as an alternative to LOA, based on the car-driving assistance domain. We will conclude on the need to consider HMC in designing HMS.

2 FRAMES OF REFERENCE FOR COOPERATION

In this paper, we will refer to the HMC label in a commercial sense, where a machine is described in terms of being “ergonomic”. On the contrary, while assuming a large asymmetry between the two partners linked within the HMS chimera, we will transfer the models of cooperation between humans in order to consider HMC. With a design perspective in mind, the aim of the transfer is not to impoverish models of human cooperation but to enrich human-computer interaction design. The theoretical frameworks for cooperation are numerous and related to the size of the collective structure, ranging from organizations (sociology) to small teams (psychology). We will favor the latter because HMC is often approached within small teams that include both humans and machines. This was the purpose of the functional framework proposed by one of us, which is already the result of an integration of several relevant frameworks. However, Schmidt’s structural-functional approach, and Millot and Mandiau’s structural approach (1995) provide us with complementary elements that are also relevant to small teams. They are also sometimes applied to human-machine cooperation. Finally, HMC with quite “intelligent” (in terms of adaptation capability) machines can trigger attitudes in humans that are similar to those observed between cooperative humans (Nass, Fogg, & Moon, 1996). Thus, we will briefly review the main concepts currently used within the HMC domain, including trust, complacency, and over-reliance.
2.1 Functional framework

Within the context of HMC, we will adopt Hoc’s (2001) minimal definition of cooperation. This definition has also been used in some human-human cooperation situations.

“Two agents are in a cooperative situation if they meet two minimal conditions.

(1) Each one strives towards goals and can interfere with the other on goals, resources, procedures, etc.

(2) Each one tries to manage the interference to facilitate the individual activities and/or the common task when it exists.” (p. 515).

The term “interference” is taken in the sense of Castelfranchi (1998), which can be positive as well as negative.

“The effects of the action of one agent are relevant for the goals of another: i.e., they either favour the achievement or maintenance of some goals of the other's (positive interference), or threaten some of them (negative interference)” (p. 162).

In order to manage interference, three levels of cooperative activities can be defined in terms of abstraction and anticipation.

Level 1: Action level. At this level, cooperative activities consist in locally creating, detecting, anticipating, and resolving interference. Interference creation can be positive when it consists in mutual control or cross-checking. In this case, one agent can inform another of their disagreement, or can give advice. For example, the non-flying pilot in an aircraft cockpit may detect an error committed by the flying pilot. By voicing that suspicion, the crew can improve its performance.

Level 2: Planning level. At this level, a Common Frame of Reference (COFOR: see Hoc, 2001) is explicitly generated and maintained as a common representation of the situation. The situation includes the environment (currently referred to as Situation Awareness — SA), as well as the agents’ activities. COFOR includes: common goals, common plans, and function allocation. For example, a fighter aircraft crew can re-plan the flight in order to process an opportunity goal that had not been planned beforehand. At the action level, cooperative activities are a sub-product that implicitly feed the COFOR. An appropriate COFOR can facilitate cooperation at the action level, preventing negative interference or enabling anticipation of it, and enabling countermeasures to be taken.

Level 3: Metacooperation level. In the same way that the Planning level activity can facilitate the Action level activity, the Metacooperation level activity can also facilitate the Planning level activity. It includes high-level knowledge that is useful to other levels, such as mental models of the other agents and of oneself. Indeed, when it is difficult to communicate with one’s partner (e.g., a machine), mental models may be useful for generating the necessary COFOR so that one can anticipate the partner’s intention or the action that could interfere with one’s own activity. For example, by observing a surgeon’s behavior, the anesthetist can infer that the operation will take longer than anticipated and thus, prepare a new injection.
This approach is functional in that it defines cooperation more as an activity than a relational structure between agents. Schmidt’s structural approach is related to the functional approach, and is thus of interest to us.

### 2.2 Structural frameworks

Schmidt (1994) adopted a structural-functional conception of cooperation that is close to ours. He mainly defined three forms of cooperation, at the same time implying function allocation and relationships (structure) between the agents. Later, Millot and Mandiau (1995) considered combining these basic forms in order to generate a large variety of cooperation situations.

- **Augmentative Cooperation.** This form is justified when workload increases, thus creating the need to add agents with the same competency. For example, in ATC (see §6), it was the aim of an automatic conflict detection and resolution device to take charge of some aircraft conflicts when the human ATC controller’s Mental Work Load (MWL) was too high.

- **Debative Cooperation.** Within this form, there is also some redundancy between agents. Here, redundancy is not devoted to multiplying work strength, but to carrying out mutual control or cross-checking between the activities, sometimes using different viewpoints. For example, in-car automation (see §8) can be designed in such a way that it does not drive the car but warns the driver when they make an error.

- **Integrative Cooperation.** In this case, the task needs agents with different competencies, an appropriate distribution of functions, and a final integration. This form is frequent in HMC where humans and machines should be considered as complementary rather than similar (see §3).

From a strictly structural point of view, Millot and Mandiau (1995) suggested that human-machine cooperation may take a horizontal form or a vertical form, based on whether or not human and machine components are on the same hierarchical level. For example, when an ATC controller and a machine perform aircraft conflict detection and resolution on distinct conflicts in parallel, it is a horizontal cooperation situation. However, this characterization can be misleading if one considers that the human is responsible for the entire HMS performance and supervises the machine. Horizontal cooperation exists for conflict detection and resolution, but a vertical component is added for supervision.

### 2.3 Attitudinal and social frameworks

A number of concepts borrowed from the social sciences (especially psychology) are currently used to describe human attitudes toward machines. They have been largely reviewed by Parasuraman and Riley (1997), who saw them as factors that influence the use and misuse of automation, and by Lee and See (2004), who placed particular emphasis on trust. In this paper on FA, we will refer to two of these concepts.

- **Trust in automation.** Automation that allows a shift from full manual to full automatic control has been proven to relate to the ratio between trust in automation and self-confidence (Lee & Moray, 1994). When part of a task remains under manual control and another part under automatic control, one must also consider the trust that humans allocate to HMS. For example, Adaptive Cruise Control (ACC) integrates speed and time headway control; however, lateral control remains in the driver’s hands. Thus,
trust in the human-machine team must develop, especially in situations where there is a high level of interference between the driver’s lateral control and ACC’s longitudinal control, such as overtaking. In this case, ACC can slow down the car when the driver needs to accelerate (Rajaonah, Anceaux, & Vienne, 2006). Trust is also characterized by its calibration — the relation between trust and actual automation capability, especially with regard to its validity — and its specificity — the degree to which trust is associated with a particular component or aspect of the trustee (Lee, 2006).

-- Complacency or over-reliance on automation. Following an initially minimal level of trust in the machine, complacency can develop as a “psychological state characterized by a low index of suspicion.” (Wiener, 1981, p. 117). It can result in three behavioral effects: negligence of information gathering needed by the automated function (or too high a decrease in sampling frequency), negligence of supervision of automation (result and procedure assessment), and negligence of manual correction for improving performance (Hoc, Young, & Blosseville, 2009).

3 FROM ELEMENTARY TO INTEGRATED FUNCTIONS

Before getting to the very heart of the matter, we must first stress that the notion of function has different meanings within the fields of engineering sciences and psychology. For the engineering sciences (see, for example, definitions given by AFNOR, 1996), a function is either an external function (or service function) or an internal (or technical) one. A service function is an action that is required or realized to satisfy the customers’ needs, whereas a technical function is an action (means) chosen by the designer to achieve the service function. In psychology and cognitive science, function has the meaning of a high-level cognitive process. Cognitive functions refer to a generic intellectual process by which one becomes aware of, perceives, or comprehends information. This process involves all aspects of perception, thinking, reasoning, and remembering. The literature dedicated to FA refers sometimes to “technical” functions, sometimes to “cognitive” ones, or both.

3.1 Fitts’ list

Most of the time, implicit methods precede those that are explicit or scientific. The first well-known method for allocating functions between humans and machines on a scientific basis was put forward by Paul M. Fitts (1951) in the Air-Traffic Control domain (Table 2-1). He suggested that functions should be allocated to the most skillful partner within that domain. Thus, the double acronym ‘MABA-MABA’ was introduced, standing for ‘MAn is Better At’, ‘MACHINE is Better At’. One should note that, at the time of Fitts’ research, humans were better than machines at visual pattern recognition; this is not quite the case nowadays. The principle expounded by Fitts was simple: (a) identify the elementary (cognitive or technical) functions that are necessary to perform a particular task; and (b) identify the best partner for performing each function. Certainly the definition of functions is relative to both the current state of the art in cognitive science, and machine performance; however, it could be updated. As was the case with Taylorism — the first application of scientific principles to work organizations — Fitts’ rational proposal for distributing functions between humans and machines was widely criticized as being too simplistic.
3.2 Criticisms of Fitts’ list

The most radical criticism of Fitts’ method was also the earliest. It was based on a lack of methodological progress for FA in HMS design since Fitts’ report. Jordan (1963) asserted that Fitts’ approach was not appropriate for solving the FA problem. He noted that humans and machines need to be comparable subsystems in order to determine which is the best agent for performing a particular function. Moreover, the very definition of functions is influenced by this need for comparability at an elementary level. In most HMS, the main reason for maintaining human operators in control rooms, on decks or in cockpits is their incomparable adaptive ability. This ability is a high-level function rather than an elementary one. Thus, Jordan suggested that the notion of comparability be replaced by that of complementarity. His main motivation was to decompose tasks into ‘molecular’ activities instead of elementary functions. “Man and machine should complement each other in getting these activities done in order to accomplish the task.” (p. 163, op. cit.). In this way, Jordan introduced the idea that human and machine must necessarily cooperate in an integrative way.

Several authors pursued this criticism. Bainbridge’s chapter (1987), entitled “ironies of automation” is probably the best example. According to the author, engineers implicitly think that it is best to eliminate the human component from the HMS, as it is known to be unreliable and inefficient. Indeed, this is still the case today. The first irony is that designers themselves created the conditions for human unreliability. At the same time, Reason (1988) stated that: “system designers have unwittingly created a work situation in which many of the normally adaptive characteristics of human cognition are transformed into dangerous liabilities.” (p. 7). In terms of cooperation, the human agent has to perform a task at the very moment when the conditions of correct realization are not fulfilled for this particular agent. The second irony is explicated in the following question: how can humans supervise machines that perform better than them? Designers maintain human operators in HMS in order to perform functions they are unable to automate. According to Jordan, most of the time these functions are activities that are high-level and impossible to decompose. Designers implicitly consider that humans supervise the HMS, possibly because they implicitly consider the human component to be the most adaptive. Sheridan and Verplanck (1978) seriously considered this idea of functional levels a decade before when introducing the notion of Level of Automation (LOA).

From these criticisms, it is clear that when a function is allocated to a machine it is much more precisely defined than when it is allocated to a human. Sometimes, there is no explicit allocation to a human. Although we would defend a less extreme position than the “residual” strategy proposed by Chapanis (1970), his viewpoint can be partially defended. He considered that ergonomics should not hinder engineering in its automation endeavor. His message was to allow automation as much as possible and leave humans to do the rest. From his viewpoint, the main role of ergonomics is to check whether the rest is humanly feasible.

3.3 Implications of Fitts’ method in terms of cooperation

We will consider the implications of Fitts’ method from three defined viewpoints: the functional, the structural and the attitudinal.

From the functional viewpoint, it is evident that Fitts’ list or, more precisely, Fitts’ method, makes explicit some allocation of elementary functions; however, it does not explicitly consider the resulting cooperative activities. In addition, the method implies some kind of
residual strategy, even if there is a comparison between human competency and that of machines before allocation. This could produce unanticipated interference in the short term (action level). With both Bainbridge’s first irony of automation and Reason’s criticism, we can identify a recurrent form of interference between humans and machines: an agent — the machine — can create unfavorable conditions for the other agent’s (human) activity. The situation where the human must take charge of the job is borderline or clearly outside the validity domain of this agent’s competency or performance. The functions are not necessarily independent; however, they are separate, so that redundancy (and thus mutual control) is impossible. FA is static, in that it is chosen at the very beginning, without adaptation to the current conditions of performance. The agents do not contribute to FA. This could create difficulties in adapting the plans to situations (planning level). This will be resolved by Dynamic Function Allocation (DFA: see below). In addition, there is no explicit consideration of a common goal or plan; this is defined beforehand, but not by the agents. This considerably reduces the capacity of HMS for cooperation. At the metacognition level, explicit models of the agents are considered, but only beforehand and by the designer, not the agent. This contributes to a reduction in adaptation capacity.

Fitts’ method took some structural aspects into account. Furthermore, Jordan’s concept of complementarity and Bainbridge’s second irony identified the inescapable integrative cooperation aspect of HMC. Nevertheless, FA remains too implicit. We know that high-level and elementary functions need to be integrated, but we do not know the functions concerned. In addition, the human adaptive ability is not considered. There is also an irony that humans are assumed to be able to supervise the machines that are assumed to be more competent than themselves (vertical cooperation). Finally, attitudinal and social aspects are not considered.

4 LEVELS OF AUTOMATION (LOA) AND BEYOND

4.1 Sheridan’s Levels Of Automation (LOA)

Fitts’ list was drawn up on the basis of the identification of elementary cognitive functions, such as pattern recognition or inductive reasoning. Within the domain of telerobotics, Sheridan and Verplanck (1978: cited in Sheridan, 1996) proposed decomposition into more integrated functions (Table 1). Levels 1 and 10 consider allocation of the entire task, either to the human or the machine. From Level 2 to Level 4, decision-making is performed partly or wholly by the machine. From Level 5 to Level 9, in addition, action implementation is performed by the machine. Some cooperative functions are explicitly allocated to the human: human approval (Level 5), human veto right (Level 6), human request for feedback (Level 8). Some cooperative functions are explicitly allocated to the machine: action proposal (Levels 2 to 7), feedback (Levels 8 and 9).

Some human functions are defined, but only those leading to machine input. LOA are mainly described from the machine’s viewpoint and reflect a machine-centered design. Furthermore, some machine functions are not explicitly defined. For example, Level 2 assumes that the machine generates a complete set of action alternatives. Obviously, this implies that the machine performs information gathering and diagnosis before being able to select possible actions. However, nothing is said about the human role. The human operator chooses from among the alternatives and, consequently, participates in decision-making. Thus, the human should also participate in the situation analysis (information gathering and diagnosis).
The LOA hierarchy is based on the progressive expulsion of the human from the direct control of the situation. When reaching Level 4, the machine is assumed to strongly influence action decision. From Level 7, action execution by the machine is imposed. As a result, the human is assumed to forgo direct control. However, the diverse forms of feedback show that the designer believes that the human performs supervisory control.

### Table 1: Levels Of Automation — LOA (after Sheridan, 1996)

<table>
<thead>
<tr>
<th>LOA</th>
<th>Function Allocation</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.</td>
<td>The computer offers no assistance; the human must do it all.</td>
</tr>
<tr>
<td>2.</td>
<td>The computer offers a complete set of action alternatives, and</td>
</tr>
<tr>
<td>3.</td>
<td>Narrows the selection down to a few, or</td>
</tr>
<tr>
<td>4.</td>
<td>Suggests one, and</td>
</tr>
<tr>
<td>5.</td>
<td>Executes that suggestion if the human approves, or</td>
</tr>
<tr>
<td>6.</td>
<td>Allows the human a restricted time to veto before automatic execution, or</td>
</tr>
<tr>
<td>7.</td>
<td>Executes automatically, then necessarily informs the human, or</td>
</tr>
<tr>
<td>8.</td>
<td>Informs him or her after execution only if he or she asks, or</td>
</tr>
<tr>
<td>9.</td>
<td>Informs him or her after execution if it, the computer, decides to.</td>
</tr>
<tr>
<td>10.</td>
<td>The computer decides everything and acts autonomously, ignoring the human</td>
</tr>
</tbody>
</table>

Four main classical functions — information gathering, diagnosis, action decision, and action implementation — are allocated either to human or machine. Of these, Sheridan and Verplank’s LOA (op. cit.) only considered decision-making and action implementation explicitly. Two basic ideas underly this list: (a) to adopt a hierarchical conception of information processing, where, for example, diagnosis and decision-making are the noblest functions; (b) to usually consider (albeit wrongly) that the functions are sequentially implemented.

This first LOA approach is clearly machine-centered. What the machine needs to do is often precisely laid out (e.g., generating a set of action alternatives). This is not the case for its human operator; for example, the choice between action alternatives, and the information gathering and diagnosis activity needed for that. More details on supervisory control functions can be found in Sheridan (1988), although these are not integrated into the LOA.

### 4.2 Further revisions of LOA

Sheridan (2000) revised his initial 1978 proposal, recognizing that the unidimensional character of LOA was a first approximation. He proposed to consider four main stages in activity: information gathering, diagnosis, decision-making, and action implementation. Within each level, he defined eight LOA, resulting in a multidimensional approach to FA. Between the two extreme levels — fully manual and full automation — he only retained six LOA. Levels 2 and 3 concern decision-making, whereby the machine either proposes a set of action alternatives or a specific one. Levels 4 and 5 assume that the machine makes the decision and allow human intervention before action implementation; here, the machine either waits for approval or briefly pauses for veto expression. Finally, Levels 6 and 7 assume that the machine executes an action without the involvement of human decision and gives any feedback to the human; thus, the machine informs the human, either on request or only in certain cases. This is our interpretation of Figure 2 from Sheridan’s study (2000; p.207), in which Levels 6 and 7 are mistakenly defined in the same way.
In parallel, Endsley and Kaber (1999) proposed a totally new LOA typology that has only a few aspects in common with that of Sheridan, except the Levels 1 and 10 (Table 2). They were more explicit about the functions performed by each partner: monitoring (probably including information gathering and diagnosis), solution generation, selection, and implementation. Except at extreme levels, humans and machines always share monitoring. As LOA increases, machines become more prominent in solution generation and selection. They also perform solution implementation, even at the lowest levels.

Table 2: LOA proposed by Endsley and Kaber (1999)

<table>
<thead>
<tr>
<th>LOA</th>
<th>Monitoring</th>
<th>Generating</th>
<th>Selecting</th>
<th>Implementing</th>
</tr>
</thead>
<tbody>
<tr>
<td>Manual Control</td>
<td>Human</td>
<td>Human</td>
<td>Human</td>
<td>Human</td>
</tr>
<tr>
<td>Action Support</td>
<td>Human/Computer</td>
<td>Human</td>
<td>Human</td>
<td>Human/Computer</td>
</tr>
<tr>
<td>Batch Processing</td>
<td>Human/Computer</td>
<td>Human</td>
<td>Human</td>
<td>Computer</td>
</tr>
<tr>
<td>Shared Control</td>
<td>Human/Computer</td>
<td>Human/Computer</td>
<td>Human</td>
<td>Human/Computer</td>
</tr>
<tr>
<td>Decision Support</td>
<td>Human/Computer</td>
<td>Human/Computer</td>
<td>Human</td>
<td>Computer</td>
</tr>
<tr>
<td>Blended Decision Making</td>
<td>Human/Computer</td>
<td>Human/Computer</td>
<td>Human/Computer</td>
<td>Computer</td>
</tr>
<tr>
<td>Rigid System</td>
<td>Human/Computer</td>
<td>Computer</td>
<td>Human</td>
<td>Computer</td>
</tr>
<tr>
<td>Supervisory Control</td>
<td>Human/Computer</td>
<td>Computer</td>
<td>Computer</td>
<td>Computer</td>
</tr>
<tr>
<td>Full Automation</td>
<td>Computer</td>
<td>Computer</td>
<td>Computer</td>
<td>Computer</td>
</tr>
</tbody>
</table>

4.3 Evaluation of LOA

A group of authors with close research links have carried out several experiments based on dynamic micro-worlds to formulate rules for choosing between LOA (Endsley, 1996; Endsley & Kaber, 1999; Kaber, Onal, & Endsley, 2000; Kaber, Riley, Tan, & Endsley, 2001). Their main results are as follows.

- Preference for partial automation over full automation

The “human-out-of-the-loop” syndrome has been known for some time (e.g., Endsley & Kiris, 1995; Ephrath & Young, 1981). Full automation reduces situation awareness (SA), so that returning to manual control can be difficult. However, HMS performance can be better in complex situations. Intermediary LOA (partial automation) can be a good way to reach an acceptable balance between SA, mental workload, and performance.
In normal operation periods, an optimal choice would be to only automate action implementation. However, when there are automation failures, performance deteriorates if the human operator does not perform action implementation. This was shown, for example, by Endsley and Kaber (1999), who used a dynamic micro-world in which moving targets had to be suppressed. It is possible that participation in action implementation could reduce the Out-Of-the-Loop (OOL) syndrome when automation fails. Option generation that is shared between the human and the machine results in a lower performance than if allocation was carried out only by the human operator or the machine. Possibly, mutual control would be more efficient serially, allowing the human to reflect on their own options before being confronted by those offered by the machine. The allocation of option generation to the human is efficient when the machine supports the human in action implementation. When the decision-making function (option selection) is fully automated, the mental workload (MWL) is reduced and SA is improved, but performance is not as good. More generally, Kaber et al. (2009) noticed that selecting the best LOA depends on the chosen evaluation criterion. Complementary to these results, Squire and Parasuraman (2010) showed within a multiple-tasks paradigm that higher LOA is more costly when task switching. A recurrent difficulty with the literature is the lack of a firm theoretical framework that could help to interpret the behavioral results. Such a framework could make it easier to transfer of results from these studies carried out in dynamic micro-worlds to a set of wider situations. One could consider that, before further theoretical elaboration is possible, it is necessary to carry out experiments that vary the type of task, even in dynamic environments.

4.4 Implications of LOA in terms of cooperation

As compared to MABA-MABA, LOA’s approach defines more macroscopic functions and accounts for increasing function delegation from the human to the machine. Thus, it takes structural aspects of cooperation into account. However, the consequences of function allocation on human activity, and particularly on the action level, are often implicit within the two frameworks. Within LOA, mutual control is considered several times. Action proposal, human approval, and human veto right are various forms of mutual control. However, the time given to the human to elaborate a sizable representation of the situation (SA) is only implicitly considered. LOA is a valuable solution to the OOL syndrome. It emphasizes the interests of human action implementation in certain situations where this could have a positive effect on SA. However, FA is defined beforehand, without consideration of the current conditions. This is a limitation in the adaptive capacity of HMS. At the planning level, COFOR maintenance is envisioned when feedback is considered. The machine can give some feedback on an activity to the human. However, metacooperation is still not taken into account. If structural aspects of cooperation are considered, since supervision involves a vertical cooperation, attitudinal and social aspects are neglected.

5 CRITERIA FOR ALLOCATION DECISIONS

Several authors have produced more or less exhaustive lists of criteria that are relevant to FA design and evaluation. In common with most critical studies of Fitts’ approach (1951), especially that by Jordan (1963), research by Kantowitz and Sorkin (1987) opened the way for methodological reflection. They certainly abandoned the “residual” strategy proposed by Fitts (1951) and Chapanis (1970) on the basis that there are diverse kinds of risk: under-load or
over-load, lack of meaning of the functions given to the human, or inconsistency of the remaining functional structure. They stressed that an important criterion for FA is to recognize human responsibility in the HMS. They also noted that a function could not be allocated to the machine when human responsibility is too important. An extreme example is the decision to launch a nuclear bomb. With regard to the unanticipated effects of automation (such as lack of trust, and a difficulty in returning to manual control), they defended the idea of dynamic allocation, as discussed below. However, they were overly optimistic with regard to the cost of allocation when a human operator is put in charge.

The work analysis method suggested by Rasmussen, Pejtersen, and Goodstein (1994), and systematically presented five years later by Vicente (1999), defined FA as a “division of work”. FA was introduced within the third step of their method, following an analysis of the work domain and the “analysis of work organization and system users”. They mainly considered the distribution of work among humans, although their method can be applied both to humans and machines. They listed six criteria: norms and practice, load-sharing, safety and reliability, competency, information access, and functional decoupling. Older, Waterson, and Clegg (1997) enriched these criteria (Kantovitz & Sorkin, op. cit.; Rasmussen et al., 1994) by adding new ones: adaptability and user acceptability. These criteria refer to the functional, structural, and attitudinal approaches to cooperation, which have been previously identified. Sometimes, they are related to several approaches and, thus, are listed under the category that matches it most closely.

5.1 Criteria related to the functional approach to cooperation

5.1.1 Action level

- Functional decoupling. This criterion is an attempt to optimize necessary cooperative activities between agents in relation to performance. It was developed further by Dearden, Harrison, and Wright (2000) when they considered function in relation to the role of the human. As a result, these authors proposed a continuum of functions that can be allocated to the machine and those that should remain allocated to the human: separable functions (minimal interference between the agents), easy to coordinate functions, critical functions (critically affect the human’s performance related to the human role), crucial to the human’s role. The more interference between distributed functions, the more cooperation activities are needed. Wright, Dearden, and Fields (2000) illustrated the point of “work articulation” with an aircraft scenario in which an engine fire was provoked by bird ingestion at a low altitude. In this situation, there are two parallel technical functions to perform: shutting down the engine and maintaining thrust to gain altitude. On one hand, surface analysis of the problem could lead to a decision to automate the shutting down function, because it is easy to do and imposes high workload. On the other hand, shutting down the engine at the wrong time could jeopardize the aircraft’s ability to achieve sufficient altitude. This case is extreme in that the authors recommended allocating both functions to the pilot because cooperation is not feasible in such a highly dynamic situation.

- Need for supervision. In any case, allocating a function to a machine goes hand in hand with the allocation of the supervision of that function to the human operator (Ephrath & Young, 1981). Even with last-minute on-board collision avoidance TCAS, the pilots and air-traffic controllers remain responsible for traffic safety. In 2002, there was a tragic accident between two aircraft near Überlingen; one descended on the action recommendation of the TCAS, the other descended on the ATC controller’s order, which conflicted with the TCAS recommendation (BFU, 2004). In this
particular case, there was a function delegation to automation, but without adequate information for the human controller to supervise that automation. The pilot was the main decision maker in this kind of decisional conflict and gave priority to the ATC order. After the accident, a clear instruction was given that stated the mandatory status of the TCAS recommendation (allocation of decision authority: see below). However, this does not mean that the entire HMS should not be supervised and that the ATC controller is the best human operator for doing that. The relative cost of this supervision as compared to the human execution cost is a criterion to consider, as well as safety in urgent situations. In other words, even though the functions allocated to the human and to the machine do not interfere very much, a large interference is created by the emergence of a supervision function.

5.1.2 Planning level

− Information access. In dynamic situations where temporal constraints can be high, the agent in charge of a particular function must be allowed easy access to the necessary data to perform the function. When the partner is in possession of the required information, this creates a precondition interference that must be managed with appropriate COFOR and models of partners. At first, SA is distributed, with each partner aware of who has the necessary information; thus, communication enables the elaboration of a COFOR, resulting in appropriate individual SA (Salmon, Stanton, Walker, Jenkins, & Rafferty, 2010). This was particularly illustrated by studies in the military command and control domains. When agents are redundant for the execution of a particular function, the agent who has access to the relevant information is the one to perform the function. Loiselet and Hoc (1999) illustrated this with their example of a two-seater fighter aircraft: the first agent to be able to illuminate the target with the laser beam will fire the missile.

− Need for a shared context. Rognin, Semailbier, and Zouinar (2000) stressed the role of cooperation in system reliability, especially in the maintenance of a shared context. The ease or difficulty of maintaining the shared context can be a function allocation criterion. This allocation criterion aims to reduce the risk of high interference density, but also considers the adaptive loop by trying to avoid breaking it through inappropriate FA.

5.1.3 Metacooperation

− Competency implies an integrative form of cooperation. Obviously, a function cannot be allocated to an incompetent agent. Nobody will allocate a non-automatable function to a machine. However, this introduces the need for an appropriate COFOR between the agents and for minimal models of the other agents to be made available. In an experiment on shifting between manual and automatic control within an industrial process control micro-world environment (pasteurizer), Lee and Moray (1994) showed that participants allocated the function to the agent they considered would perform most competently.

5.2 Criteria related to the structural approach to cooperation

− Norms and practice induce a vertical form of cooperation. They stress the constraints that are related to the social and hierarchical status of the agents. The non-satisfaction of these types of constraints can lead to significant social damage. FA is not only a
matter of abstract cognition. We have cited above the extreme example of launching a nuclear bomb, which cannot be delegated to a machine because of inescapable norms. One of the sources of difficulty in automating functions is the understandable resistance of human operators to the loss of their social status.

- **Allocation of decision authority.** Grote, Weik, Wäfler, and Zöch (1995), and Grote, Ryser, Wäfler, Windischer, & Weik (2000) proposed the KOMPASS method for function allocation. This method is more macroscopic, integrating such organizational aspects as polyvalence of work system members or work groups. We will see below (§6.1) that when a function implies a decision between reasonable options, there is a question of authority allocation, which is related to a vertical cooperation structure. If the machine performs the function, it can pass on authority to the human, as is the case for intermediate LOA. Authority is a slightly different concept to that of responsibility. When the designer gives authority to the machine, there is no transfer of responsibility, which remains with the operator or the designer. Simply, in a particular situation where the operator is not able to perform the task, the machine can take the control without human intervention. The Überlingen accident cited above is an illustration of this. This is a way of preventing negative interference; something that is all the more necessary because the temporal constraints are so high. However, this solution is debatable and does not lead to the economy of supervision.

- **Safety and reliability** can be ensured by a debative form of cooperation, the agents being able to exert a mutual control over each other. From the human viewpoint, the control of the machine is performed by supervision, and the reverse is the validation (or invalidation) of the human decision. Paradoxically, in risky systems, FA can lead to the same function being allocated to several agents to prevent redundancy, and thus insure safety and reliability. Nevertheless, in this case, the final allocation choice is made in real time and this activity increases the cooperation cost. This point will be discussed below.

- **Adaptability** is required by the function. Humans are more adaptable than machines and, thus, are preferable when unexpected events occur. It also implies integrative cooperation and stresses the need for feedback from the machine operation (COFOR and model of the machine). The delegation of authority to a machine cannot be definitive and is temporary because, more often than not, the machine is not able to fully analyze the situation for adopting a long-term solution to the problem. Delegation to a TCAS is an illustrative example.

### 5.3 Criteria related to the attitudinal approach to cooperation

- **User acceptability.** This criterion, already defined in §2.3, depends not only on the global quality of the machine, but also on the class of situations in which the machine is reliable. It is related to trust (Lee & See, 2004).

- **Complacency or over-reliance.** Some forms of function distribution can lead to the operator’s disinvestment. Maintaining the operator inside the loop is a way to prevent this drawback; for example, the operator and the machine can share control of the same function, as proposed for car lateral control assistance (Deroo, Hoc, & Mars, 2011; Mulder, Abbink, & Boer, 2008).

Such allocation criteria can be used beforehand or in real time (adaptive or dynamic allocation), as evaluation criteria for design choice (beforehand) or for dynamic allocation.
6 DYNAMIC FUNCTION ALLOCATION (DFA)

Many studies into Dynamic Function Allocation (DFA) have referred to “Adaptive Automation” (AA), whether or not the allocator is a human or a machine. Scerbo (2001) suggested that a distinction should be made between two types of DFA: one that restricts “Adaptive Automation” to allocation changes at the human’s initiative, and one that uses “Adaptable Automation” to refer to changes at the machine’s initiative. In order to avoid confusion, adaptive automation in Scerbo’s terms will be termed explicit allocation, and adaptable automation will be termed implicit allocation, following the current distinction in supervisory control (e.g., Rieger & Greenstein, 1982). At a more general level, we will use the term “Dynamic Function Allocation” to cover the two types of allocation.

6.1 Function, task, or role

Until now, we have exclusively made use of the wording “function allocation”, although “task allocation” is also used within this context. This choice was not made by accident, but deliberately, on the basis of long experience of this question within the context of Air-Traffic Control (ATC). Task allocation and function allocation are very different, provided that one makes a difference between task and function, particularly with regard to the following human properties:

- A task is often considered as a more macroscopic structure than a function. Task complexity is often an impediment to the allocation of an entire task to a machine.
- A task is concrete and action-oriented, whereas a function is abstract, in that it can contribute to several tasks (Fuld, 2000).
- From a psychological viewpoint, as highlighted in the present paper, a task is associated with an explicit goal, which corresponds to an intention to protect it from concurrent ones. On the other hand, a function is closer to a routine, as a means of servicing several tasks.

For example, after several years of research into task allocation within the ATC domain, we have found it more appropriate to distribute functions instead of tasks between humans and machines (Hoc & Debernard, 2002). The problem arose from a high rate of increase in air traffic. Several solutions were envisioned in order to manage the human controllers’ workload, independently of the reduction in control sector volume. Indeed, in relation to aircraft speed and response lag, it is necessary to state a minimum distance to the conflicting point between aircraft to have time to send a manageable deviation instruction to a particular aircraft. From among the various solutions, the possible use of an Automatic Conflict Resolution Device (ACRD) has been explored. The first paradigm to be considered was task allocation (e.g., Hoc & Lemoine, 1998; Vanderhaegen, 1999). Within this context, tasks were aircraft conflicts. In a particular dynamic allocation paradigm (see below), the controller had a veto right when ACRD decided to allocate a conflict to itself. More often than not, the reason was a major discrepancy between task definitions by ACRD or the controller. ACRD was designed to define tasks as binary conflicts between aircraft, whereas the controllers considered multiple-aircraft structures in order to find the most relevant solution. On this basis, we chose a function allocation paradigm where the controller was in charge of identifying the conflicts (the tasks). The controller was able to delegate functions (within these tasks) to the machine whilst keeping control of the task themselves. For example, a function could be to make aircraft 1 go behind aircraft 2.
The question of role delegation is of a quite different nature. Role cannot be delegated to the machine if it is defined as a task that is associated with the corresponding responsibility. Authority is another concept that is close to, but distinct from, responsibility. In many domains, the question of allocation of authority is crucial. For example, in car driving with automation, when the machine diagnoses that the Time To Contact (TTC) with a threatening obstacle is less than the driver’s response time, is it reasonable to delegate the human authority to the machine for emergency braking? Possible questions could be the probability of false alarms, and the driver’s intention to skirt the obstacle. We have already given the example of the midair collision above Überlingen (BFU, 2004), where there was a conflict between the decision given by the TCAS (onboard collision avoidance device) and that given by the human ground controller. After the accident, the inquiry committee recommended that authority be delegated to the TCAS. In this particular case, such a decision was correct, but this might not always be so. Indeed, any delegation of authority raises the question of time and condition of return to manual control.

If the term “function” is more appropriate in this context than task or role, a complex work system must be decomposed into a hierarchy of functions. It is not appropriate to remain at an elementary level, even if only elementary functions are the components to be distributed. Moreover, within the framework of human-machine cooperation, the concept of delegation could be more appropriate than that of allocation. Delegation means that, although the main task remains under human responsibility, complementary functions should also be considered, such as human supervision of the automated function and human cooperation.

### 6.2 From static to dynamic allocation

The idea of a dynamic allocation of functions is not new (Rieger & Greenstein, 1982), particularly in dynamic situations. As already stated, several authors have criticized FA from the viewpoint of its rigidity in relation to circumstances. Dynamic Function Allocation (DFA) has been defined as, the “dynamic allocation of system functions to a human operator and/or automated controller over time based on operator states and task/contextual information for the purposes of optimizing system performance” (Kaber, Wright, Sheik-Nainar, 2006, p. 528: These authors used the term “Adaptive Automation” — AA). The goals of DFA include moderating operator workload and supporting system/situation awareness by facilitating a better match between task demands and operator cognitive resources (Kaber, Perry, Segall, McClernon, & Prinzel, 2006). Human under-load has its own drawbacks, as does overload. Under-load can lead to vigilance decrement, and overload to well-known attentional difficulties. Dynamic function allocation between humans and automation has been considered as a means to control MWL. When MWL is anticipated to increase, automation can perform some functions, whereas when MWL is decreasing it could be a good choice to return functions to the human. We have already cited an example in ATC where an ACRD can resolve binary conflicts or allow the human to take charge of them.

Within a horizontal cooperation structure between a human operator and a machine, a function allocator is needed to decide which agent will be working on the function in real time; for example, on the basis of MWL. When the machine makes allocation decisions, FA is considered as “implicit”. A mixture between vertical and horizontal structures can be reached with a multi-level organization when the human is not only able to perform the function at the tactical level, but is also able to play the role of function allocator. In this case, the authors termed this allocation “explicit”. In order to pass the allocation decisions to the human without overloading this agent, a second human can be implied as a function allocator. The allocation is explicit because the human operators can negotiate the allocation decisions. This
situation can be considered as “assisted explicit allocation”. Within the context of ATC, where these principles were evaluated, there is a Radar Controller (RC) who has particular charge of collision avoidance maneuvers in the controlled sector and a Planning Controller (PC), who is in charge of preparing the RC’s work. The allocation role was given to the PC. Allocation was doubly “assisted”, by the PC in relation to the RC, and by a predicting tool at the PC’s disposal, which anticipates future conflicts and is thus future RC’s MWL (Hoc & Lemoine, 1998).

With a dual route approach to function allocation, van Dongen and van Maanen (2005) proposed that either the machine alone makes the allocation decisions when temporal constraints are high or the human-machine cooperation. In the latter case, the human-machine team determines whether actual or anticipated performance degradation is problematic and a re-allocation is needed. They recommended basing FA not only on MWL, but also on the ratio between trust and self-confidence (Lee & Moray, 1994).

Specifically within the context of LOA, Endsley (1996) presented a short review of studies aimed at designing and evaluating DFA. The common aim to these studies was to fight against the OOL syndrome and the loss of SA provoked by high LOA (cf. §4.3). She envisioned several alternatives, the first one being alternating allocation to the human and allocation to the machine. Other solutions could be to choose a particular LOA in relation to the circumstances. Miller and Parasuraman (2007) pursued this line, stressing the need for appropriate delegation interfaces.

6.3 Evaluation of dynamic allocation

As already stated, dynamic task allocation was evaluated within the context of ATC by a French laboratory (LAMIH in Valenciennes) over the course of ten years. The two first studies dealt with aircraft conflict allocation, either to an RC or to an ACRD (Vanderhaegen, Crevits, Debernard, & Millot, 1994; and then Hoc & Lemoine, 1998). No study considered full automation because the ACRD was not able to resolve every possible conflict; rather, it was limited to a restricted class of conflicts — those that take place between two aircraft at the same level. In each case, dynamic task allocation paradigms were compared with a control situation without assistance (manual control). The results of these studies led the laboratory to define a function allocation paradigm with a principle of function delegation to the machine within the framework of a task defined by the human controller (Debernard et al., 2009; Hoc & Debernard, 2002).

The first study was conducted with the RC controlling alone, but with the possible assistance of an ACRD. Dynamic allocation improved performance, and implicit allocation more than explicit allocation. Despite a non-balanced experimental design, clinical data analysis enabled the authors to conclude that the explicit allocation’s lower performance was due to the reluctance to delegate a binary conflict to the machine when it was more appropriate to integrate it into a more complex structure. Thus, the machine resolved more conflicts in the implicit condition than in the explicit one; thus, the MWL was lower and the performance higher.

The second study did not make use of the implicit allocation condition because the controllers rejected it on the basis of their feeling that control of the situation had been lost in the previous experiment. Two paradigms were compared with manual control: explicit and assisted explicit allocation, with two controllers (RC and PC in charge of the allocation).
Dynamic allocation led to a better performance in terms of anticipation and quality of cooperation between controllers. Performance was also better with assisted explicit allocation, with a drawback in terms of complacency (see below). However, a number of conflicts were not delegated to the machine on the basis of their complexity and of the inappropriateness of their decomposition into binary conflicts.

Within the LOA approach, several experimental studies have been conducted on the effect of DFA. For example, in uninhabited vehicle supervision, de Visser, Horvath, and Parasuraman (2008) showed that DFA led to better results than full manual or automatic control on the basis of performance decrement. However, the conclusions of this type of study must be balanced because the results are related to the type of situation (Endsley, 1996). DFA is more appropriate in dynamic situations than static situations. It has differential effects on psychomotor, perceptual, and cognitive tasks. According to Kaber, Wright, Prinzel & Clamann (2005) and Kaber, Perry, Segall, McClenon, & Prinzel (2006), humans appear to adapt better to DFA applied to information acquisition and action implementation (sensory and psychomotor information-processing functions) than to DFA applied to higher cognitive functions (analysis and decision making). The choice of the best LOA in relation to the situation type explains the benefit of DFA. In a micro-world experiment into robot supervision and control, DFA was managed by humans in order to adapt to the situation (Parasuraman, Galster, Squire, Furukawa, & Miller, 2005). The alternation of manual and automatic control is preferable to purely manual or automatic control (Kaber & Endsley, 2004; Parasuraman, Mouloua, & Molloy, 1996). Nevertheless, full automation could be the best LOA in time critical situations (Inagaki, 2003; Moray, Inagaki, & Itoh, 2000). It refers, once again, to the question of authority.

Several studies have stressed the cost of explicit dynamic allocation that implies the human operator. It was clear in the LAMIH experiments and was confirmed by Kaber and Riley (1999), who showed that in explicit dynamic allocation, the individuals remained more in manual control mode and achieved a lower performance than in implicit allocation. A possible interpretation is the cost of the allocation decision, which must be reduced by support to human-machine communication (Kaber et al., 2001). In the last LAMIH experiment (Debernard et al., 2009), explicit function allocation was supported by an efficient interface. However, its introduction modified the human-human cooperation within the team. At first designed to assist the RC, the delegation interface appeared to be more efficiently used by the PC. The PC is usually in charge of preparing the RC’s job, particularly in terms of anticipating conflicts. Thus, the PC utilized the interface for defining new conflicts and went as far as proposing resolution decisions. This alleviated the RC’s MWL and improved performance.

6.4 Implications of dynamic allocation in terms of cooperation

The implications of dynamic allocation on cooperation may, once again, be considered in terms of functional, structural and attitudinal aspects.

Considering these implications from a functional point of view allows us to point out several possible consequences. As shown, the distribution of tasks between humans and machines within the ATC domain can create negative interference (action level) when the human’s task representation is different from that of the machine. Major negative interference can intervene when DFA is related to high-level functions because of a loss of control of the situation by the human operator. A feeling that situation mastery has been lost implies interference that can be
a serious impediment to task execution. DFA obviously raises the same questions about cooperation between distributed functions as it does about static allocation. However, within our human-machine cooperation framework (Hoc, 2001), function allocation is a cooperative activity in itself, located at the planning level at which a COFOR is elaborated and maintained. In order to manage interference between individual activities, the cooperating agents must share a minimal COFOR. If function allocation is possible, it is assumed that the agents are partly redundant. They must perform this activity; indeed, we have already cited several empirical studies that stress the importance of designing appropriate interfaces to support the allocation process (see Kaber et al., 2001, for a review). However, for MWL reason, these interfaces could be handled by another operator. Thus, DFA can allow the work team to be restructured. As a particular cooperative activity, DFA enables cooperation to contribute to HMS adaptation to unforeseen situations, when the HMS structure is not able to insure resilience. It opens the way to restructuring or re-configuring the HMS in order to deal with a wider class of situations beyond the validity domain of the current HMS structure. Several authors have stressed the contribution of cooperation to safety (e.g., Rognin et al., 2000). Cooperation not only introduces mutual control (or cross-checking), but also redundancy and the re-allocation of functions. Although the question remains open to debate, automatic allocation to the machine can be the best solution for a limited duration, in an emergency situation. This is a delegation of authority, rather than responsibility. However, the designer should ensure that, in an emergency, the human operator is unable to produce any acceptable response. The debate stems from the validation of this condition. The availability in each agent of a model of the other agent metacooperation level) can greatly support the FA decisions.

With regard to structural aspects, distribution is possible only because the agents are partly redundant. This opens the way to both augmentative and debative cooperation.

The choice of DFA allows attitudinal aspects to be taken into account. Allocating a function to a machine should not involve a loss of feeling of situation mastery on the part of the human operator. In addition, the ratio between trust and self-confidence should be considered, as well as trust in the overall HMS. These attitudes are related to circumstances. Blind trust or self-confidence should be avoided because neither the human nor the machine can be reliable in any kind of situation. This situational knowledge is required in order to make the right allocation decision.

Alternating the allocation of a function to the human and the machine is recommended to reduce any human-out-of-the-loop problems. This could improve SA and reduce complacency drawbacks.

7 COOPERATION WITHIN FUNCTION ALLOCATION

Following this review, we will now summarize the main implications of FA on cooperation. The distribution of functions between agents introduces cooperative cost, although it can be beneficial; thus the question of efficiency is the first that needs to be addressed. In addition, FA implies the execution of various cooperative activities, which need to be identified before attempts can be made to improve them and reduce their costs. Distributing functions between a human and a machine must deal with the structural asymmetry that exists between them. Finally, when allocating a function to a machine, one must examine the attitude of the human operator toward automation, particularly with regard to the issues of trust and complacency.
7.1 Efficiency

HMC must be evaluated from the point of view of efficiency. Efficiency describes the extent to which time and effort is used for an intended purpose. In the case of HMC evaluation, the notion of effort refers to the cognitive costs implied by function allocation. They are the costs of knowing, reasoning and deciding, which are generally recognized as MWL (Dowell & Long, 1998). One must also take into account the costs of cooperation activity that result from function distribution, as well as the extra cost of allocation decision in the case of DFA. This kind of cost can be estimated by looking at the number and duration of behaviors that are required. Thus, van Dongen & van Maanen (2005) focused on the costs of coordination, which they defined as the effort and time required to engage and disengage automation, as well as the effort and time required to decide how to reallocate tasks.

7.2 Cooperation processes

7.2.1 Cooperation at the action level

Before distributing functions, one must decompose a particular activity into functions and this decomposition is not unique. It is impossible to decompose a complex system into strictly independent functions. However, some decomposition alternatives can generate far more interference opportunities than others. Among the FA criteria, the implications for cooperation are the most important to consider, especially in dynamic situations where temporal constraints can prevent the human operator from managing interference in real time. At the very least, some support for such management should be envisioned at the time of HMS design. However the designer must always manage a balance between the benefits of automating a function and the drawbacks of the interference that the automated function creates; especially in terms of negative preconditions for the human action.

Particular attention should be devoted to the consequences of allocating a function to the machine when this function needs higher adaptability than that provided by the machine. In this case, the human operator has to frequently return to manual control in order to recover from incidents.

Interference is not always negative. We have seen that it can be generated in a positive way, using the machine as a partner that is capable of mutual control; there is no intention to replace the human operator, but rather to just try to improve the performance. Another positive way to generate interference is the use of the machine as an assistant: in other words, it creates favorable preconditions for the human action.

7.2.2 Cooperation at the planning level

In order to improve interference management at the action level, a COFOR must be elaborated and maintained. Part of the COFOR can be fed by cooperation at the action level, but most of it must be fed by more explicit communication. Restriction in a machine’s communication capability renders COFOR elaboration difficult, but COFOR maintenance can be greatly improved if a machine can return feedback on their activities to the human operator (Kaber, Wright, & Sheik-Nainar, 2006; Skjerve & Skraaning, 2004). Although humans are able to communicate with machines, the cost of such communication is reduced each time the machine is capable of inferring their human partner’s activity from their behaviors, without the need for explicit communication. In dynamic situations, representations of the partners’
activities can play a major part in COFOR (Hoc & Carlier, 2002). With regard to DFA, we have already seen that dynamic allocation controlled by the human (explicit allocation) is preferable but costly; it must also be assisted. It constitutes a typical activity of COFOR elaboration.

Building a “common work space” (CWS) consists in implementing a COFOR on a machine. Pacaux-Lemoine and Loiselet (2002) explained that the definition of its content depends on the domain and the tasks to be achieved, as well as on how agents interact with each other. The CWS should allow agents to share goals or plans but also relevant contextual information and allocation of function. In case of implicit allocation, a crucial point is to inform the human agent about the change of mode. It should rely on a shared task model (Miller & Parasuraman, 2007). Kaber et al. (2001) stated the principles of its design interface: (a) it should foster effective communication of activities, task status and mission goals, as well as the development of useful and realistic concepts, (b) enhance operator awareness of his or her own responsibilities, capabilities, and limitations, as well as those of other team members, (c) support dynamic FA that is quick, easy and unambiguous.

However, a CWS does not resolve all the COFOR problems. First, it implies a maintenance activity, which can be costly for the human operator when the machine is not able to infer aspects of the human activity that are related to the CWS. Second, ergonomics has long stressed the action-oriented property of external representations. Even between humans, a CWS should be easily read by each agent from its action viewpoint (as shown in a two-seater fighter aircraft, for example, by Loiselet and Hoc, 1999). In other words, a CWS must be translated in each agent’s terms.

### 7.2.3 Metacooperation

COFOR elaboration and maintenance can be greatly facilitated thanks to knowledge that is developed in the long term; for example, using models of the partners. These models enable the human to anticipate machine behavior, especially in terms of any possible failure or interference. However, the reverse is also true. The designer should introduce a minimal model of the human operator in the machine in order to improve cooperation, especially by creating positive preconditions to the human activity.

A model of oneself (metaknowledge) is also necessary for appropriate cooperation. Allocation of functions originally performed by humans to the machines creates a need for adaptation. In such a case, human operators must re-structure their activities, combining the models they have of themselves and the models they have of the machines. In particular, norms and practices have an effect on human models of machines. When these models are invalidated, this can generate misunderstanding and negative interference. In any case, the machine should be understandable.

### 7.3 Structural aspects of HMC

In line with the debate on comparability and complementarity between humans and machines, we must underline the large degree of asymmetry between humans and machines with regard to FA. When trying to use a machine within an augmentative cooperation form, we frequently discovered that HMC is not closed into this form because of the necessary human supervision. Thus, sometimes an integrative cooperation form takes place. Automating a function creates the need for supervision, but does not always result in a decrease in MWL. Another reason for integration is that human competency is always different from machine competency. The
same argument is true as far as the opposition between horizontal and vertical cooperation are concerned. There is no pure horizontal cooperation within HMC, but there is always a vertical component.

Cooperation between humans can develop in a variety of situations, from collaboration where the distribution of functions is minimal — with the partners doing almost everything together — to distributed cooperation where functions are largely distributed. In the latter case, it happens that a particular agent supervises cooperative activities, or even performs them. This is the case for HMC, because the machine has limited cooperative skills or know-how (Millot & Mandiau, 1995; Skjerve & Skraaning, 2004). If the machine is not able to perform all the necessary cooperative activities, the human must perform them. Thus, automating a function does not only create a need for supervision, but also a need for cooperative activities to be performed, particularly “work articulation” (Wright et al., 2000).

Another kind of asymmetry between humans and machines occurs when functions are designed to assist the human. In this case, a machine has to watch over the human performance, provide information in time and identify the errors made by its partner. It has to be adaptive and must, therefore, have a good task model, as well as a good model of human behavior.

7.4 Human attitude toward automation

Dzindolet, Beck & Pierce (2006) hypothesized that people may fail to use an automated aid when the perceived utility of the aid is overestimated and, likewise, when it is underestimated; the perceived utility of the system results from a comparison between the perceived ability of the automated system and one’s own perceived ability.

7.4.1 Trust and automation disuse (underutilization)

In dynamic situations in particular, the main human operator’s objective is to master the situation; that is to say, keep the situation under control (Hoc & Amalberti, 2007). Allocating a human function to a machine immediately raises the question of trust in the machine. The level of trust must be at least as high as self-confidence (Dzindolet et al., 2006; Lee & Moray, 1994). Obviously, trust (in a machine or in the HMS) and self-confidence must be calibrated to a particular class of situations. Automation failures and random false alarms are likely to dramatically reduce trust and generate automation disuse.

7.4.2 Complacency and automation misuse (over-reliance)

As soon as the human operator reaches a minimum level of trust, a general and serious effect of automation, which is related to the OOL syndrome, can take place. Intermediate LOA are preferable if one wants to reduce complacency. Parasuraman, Sheridan, and Wickens (2000) showed that complacency is higher when decision-making is automated. According to them, humans are likely to be less aware of environmental changes caused by another agent. Furthermore, Scallen and Hancock (2001) were among the authors who recommended the regular shifting of allocation.
8 MODES OF COOPERATION

HMC has been a prominent concern within the car-driving assistance domain (Hoc et al., 2009). Although it is feasible to design an entirely automated car, the adaptive power of this kind of car remains still very limited with regard to the complexity of road traffic conditions. The airspace situation is much simpler; for this reason, commercial aviation can be highly automated. In addition, drivers may be reluctant to buy an automated car that is likely to take too much control of the trajectory. The philosophy is to maintain the driver as the main entity in charge of driving. Thus, the car engineer’s target must be to find the lowest LOA that is capable of keeping the driver-car system within a safe envelope. Conscious of the cooperative implications of LOA, Hoc at al. (op. cit.) sought to identify cooperation modes behind LOA (Figure 1).

![Figure 1: Cooperation Modes in the Car-Driving Assistance Domain (after Hoc, Young, & Blosseville, 2009, revisited)](image)

<table>
<thead>
<tr>
<th>Function Delegation Mode</th>
</tr>
</thead>
<tbody>
<tr>
<td>Mediatized Mode</td>
</tr>
<tr>
<td>Control Mode</td>
</tr>
<tr>
<td>Prescription Mode</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Full Automation Mode</th>
</tr>
</thead>
<tbody>
<tr>
<td>Emergency braking</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Shared Control Mode</th>
</tr>
</thead>
<tbody>
<tr>
<td>Corrective Mode</td>
</tr>
<tr>
<td>Continuous Shared Control Mode</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Mutual Control Mode</th>
</tr>
</thead>
<tbody>
<tr>
<td>Warning Mode</td>
</tr>
<tr>
<td>Action Suggestion Mode</td>
</tr>
<tr>
<td>Limit Mode</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Perceptive Mode</th>
</tr>
</thead>
<tbody>
<tr>
<td>Augmented Vision (HUD)</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Auditory, Haptic Warning</th>
</tr>
</thead>
<tbody>
<tr>
<td>Motor Priming</td>
</tr>
</tbody>
</table>

**Perceptive Mode**

Automation provides the driver with enhanced perceptive capability (e.g., infrared vision or the enhancement of a particular leading point in a bend: Mars, 2008). It only intervenes in the information gathering activity module, which is performed partly by the driver, and partly by automation, with the driver being in charge of the rest. Although light, this kind of automation intervention can be very intrusive, directing attention to a particular piece of information to
the detriment of others. It can influence situation diagnosis, without providing any diagnosis at all. In more generic terms, the perceptive mode makes use of the perceptive competency of the machine to create a positive precondition for the other partner’s activity. However, the machine designer must have a correct model of the human operator in order to satisfy this property.

**Mutual Control Mode**

Within the car-driving assistance domain, diagnosis is more often than not related to the driver-car-road system. From the driver’s viewpoint, most of the diagnoses emitted by car automation can be interpreted as criticisms of the driver’s behavior (e.g., auditory warning informing the driver of speed excess). In terms of cooperation, the activity of one agent that “controls” partner activity “over the shoulder”, without performing any action, is called “mutual control” or “cross-checking”. Mutual control is not restricted to warning signals, but integrates action suggestion (e.g., directional steering wheel oscillation: Navarro, Mars, & Hoc, 2007) and limitation (e.g., gas pedal resistance when reaching a speed limit: Hjälmdahl & Várhelyi, 2004). The only restriction is that automation does not act on the vehicle, but on the driver. This cooperative mode assumes that both agents perform information gathering, diagnosis, and decision making, with the driver alone staying in charge of action implementation.

**Shared Control Mode**

At this level, automation begins to act on the vehicle in order to correct the trajectory. This kind of control may be discontinuous and qualified as corrective. For example, when the driver does not produce a sufficient steering angle in a bend, automation can add some torque to avoid lane departure. The control may also be continuous. Using the same example, automation can always give a certain percentage of the torque, which is expected by the driver, thus rendering the task easier (Saleh, Chevrel, Mars, Lafay, & Claveau, 2011). In this case, the driver and automation share information gathering, diagnosis, action decision and implementation.

**Function Delegation Mode**

The driver consciously delegates one of the driving functions to automation. The best-known example is Cruise Control (CC), which enables the driver to temporarily delegate longitudinal control of the vehicle to automation. Automation performs the entire activity that is related to the delegated function, with the driver being in charge of the other functions. However, such delegation introduces a new function, that of supervising automation. Critical problems with this mode are the creation of acceptable preconditions for returning to manual control and the control of negative consequences in terms of complacency.

**Full Automation Mode**

In this case, the driver is out of the loop, except if supervision is possible. However, within the car-driving domain at this point in time, this mode is limited to rare situations, such as emergency braking with an additional triggering of air bags. This mode definitively prevents the driver from controlling the car and, thus, delegates full authority to the machine. It poses the problem of justifying this full authority should it fail and raises questions about the conditions and timing of a return to manual control.
This proposal for stressing MOC instead of LOA has been proven to be useful within the car-driving automation domain. It needs to be applied to other domains in order to show its generic value.

9 CONCLUSION

Three Function Allocation (FA) strategies exist; these are centered on a machine, on a human operator, or on a Human-Machine System (HMS). Strategies centered on a machine ignore the notion of Human-Machine Cooperation (HMC), whereas human-centered strategies see the machine as an assistance to human activity. Within the framework of HMS design, FA may be addressed from two different points of view. The first approach – and the most widespread – considers humans and machines as two separate entities. Functions are allocated either of them in a static way (LOA) or in a dynamic way (DFA), in order to reach an optimal balance between SA, mental workload, and HMS performance. The second approach may be seen as complementary to the first one. It takes a particular interest in the human-machine relationship. It considers that it is impossible to decompose a system into independent functions, and that interfering functions may be allocated to a human agent and to a machine agent. It stresses that FA implies cooperative activities (Hoc, 2000), and therefore deals with HMC. This position points out that interference management between functions, rather than the allocation of functions itself, is the main design problem. As Grote et al. (2000) stated: “the interaction of human and machine is to be designed, taking into account reciprocal dependencies between task components and the ways they are carried out by human and technical system.” (p. 269). In line with their Joint Cognitive Systems framework, Hollnagel and Bye (2000) proposed a method for modeling FA in order to be able to predict the consequences of a particular allocation. They developed a similar conception, shifting from allocation on the basis of the agents’ own skills to designing a situation where several agents are capable of collaboration and mutual support to reach a satisficing performance (in accordance with Simon, 1969, or Hoc and Amalberti, 2007). Dekker and Woods (2002) went a little further, suggesting that FA introduces a qualitative change in human activity. It is not just a case of humans and machines needing to cooperate (human-automation coordination in the authors’ terms); rather, humans must also restructure their activity, by developing adaptation processes. For example, within the context of car-driving assistance, many studies are devoted to behavioral adaptation; that is to say, the generation of a new way to control activity (Hoc et al., 2009).

This approach also focuses also on the adaptive ability of human and explains that supervision of the HMS is always given to the human. Thus, one can consider FA in terms of “function delegation”. Furthermore, it emphasizes the need to elaborate and maintain a COFOR to manage interference and the need to design a “common work space”. Its main message is as follows: When tasks or functions are distributed among humans and machines, cooperative activities are created, which should not be neglected in the measurement of costs and benefits related to efficiency. Recognizing the importance of HMC may finally lead to the design of MOC instead of LOA, allowing studies to be carried out which extend beyond the car-driving domain. In this way, it may be possible to generalize the framework.

ACKNOWLEDGEMENTS

This work has been funded by the CNRS Research Network GDR 3169 Psycho Ergo.
REFERENCES


**Jean-Michel Hoc** is Research Director at CNRS (French national research center) in IRCCyN (Research institute in communications and cybernetics of Nantes, France) within the cognitive ergonomics domain. For about forty years he has been working on human-computer and human-automation interaction, especially from the cooperation viewpoint. His work has dealt with various application areas, such as psychology of programming, planning and scheduling, industrial process control, air-traffic control and aircraft piloting, or car-driving assistance. He has published about fifteen books and eighty journal articles. He was recently the head of a national CNRS research network in cognitive ergonomics. He has got his PhD in 1979 and his Habilitation to supervise research in 1987.

**Christine Chauvin** is Professor of Psychology and Ergonomics at the University of South Brittany, France. Her main research interests are cognitive modeling for human integration in complex systems, decision making in dynamic situations, and design and evaluation of human-machine systems. In recent years, she has worked in the field of transportation systems, specifically in the field of ship navigation. She earned her Ph.D. in cognitive ergonomics from the University of Paris V in 1996 and her Habilitation to supervise research in 2008.