Chapter 7 – HUMAN AUTOMATION INTEGRATION

Chapter Lead: S. Galster

Contributors: M. Barnes, K. Cosenzo, S. Galster, E. Hollnagel, C. Miller, R. Parasuraman, J. Reising, R. Taylor, L. van Breda

7.1 INTRODUCTION

Many versions of future concept of operations (CONOPS) rely heavily on UMVs. The pressure to take the human out of immediate control of these vehicles is being driven by several factors. These factors include a reduction in cost for the production and maintenance of the vehicle, operational viability in extreme environments, and the public pressure to keep soldiers further away from immediate harm. In addition to adding more UMVs, there is also a push to have these vehicles perform more complex tasks than they are currently required to perform. These two factors, adding more UMVs and having them perform more complex tasks, will not be realized without augmenting the current structure of control. One way to achieve this augmentation is through the utilization of automation. The automation may be applied on the vehicle itself, through the interface controlling the vehicle, through system design or in any amalgamation of these approaches. Automation, if applied in a responsible and judicious manner, will enable the acquisition of capabilities that will be required to operate under near and far-term CONOPS.

The focus of this chapter is the discussion of the past, present and future automation integration challenges that are faced when adopting a human centric design philosophy. Topics will include the identification and discussion of specific problematic areas that have evolved and will bring to bear the lessons learned as automation was integrated into other domains such as flight operations, air traffic control, and process control for example. The lessons learned may not signify that a particular problem area has been "solved", but may point out that the area deserves consideration when evaluating trade-offs in the system design and engineering process. The anecdotal, operational, theoretical and empirical work completed thus far all provide a sound foundation that should serve as a starting point for human automation integration in the UMV domain. This domain may have specific challenges or specific opportunities available, both now and in the future, to explore and expand the base of human and automation integration knowledge. The remainder of this introduction will focus on some of the more salient areas in the integration problem space. These topics and others will be addressed in various sections throughout the chapter.

7.1.1 Problems with Supervisory Control Tasks

Vehicle control may be considered as a hierarchically structured set of functions. Plan conception and plan selection activities are performed in the navigation function, verification and adjustment of the short-term voyage progress are performed in the guidance function, and typical closed-loop control activities are performed in the control function. *Supervisory control* of vehicles deals with automated vehicle control functions to a large extent. Current technology is not yet prepared to autonomously perform mission guidance under dynamically changing environmental conditions, although navigation, guidance and control functions are widely automated. The operator, who may observe the controlled process, acts as a manager who supervises the system and interacts with the automated system by performing corrective actions. The human operator in current architectures is the primary responsible factor in terms of goal-driven decision-making, and thereby specifying the constraints and demands settings for the automated system.

It is known, however, that supervisory control systems have certain limitations in performance, either on the operator's side due to human capacity limitations or induced by deficiencies of the automation, causing human error intensified by the inability of the automation to perform on the higher level of problem-solving.

7.1.2 Function Allocation

Consider the development of human roles and automation from the traditional "*Left-over principle*", through human engineering optimising *compensatory* principle with human monitoring (Fitts lists), to contemporary *complementary* principle arising from human-computer co-operation/collaboration. Now function allocation can be dynamic according to external system functions, efficiency and system boundary conditions. Levels of autonomy provide bounding of the decision authority and behaviour to promote trust. Key questions:

- Should the human monitor the (technical) system given that humans are poor monitors?
- Should the (technical) system monitor the human?
- If so what roles should the human play and what are their responsibilities?
- Are humans included in systems just to deal with those functions that engineers can not automate?

7.1.3 Levels of Automation

For designing supervisory control, possible structures for the allocation of decision-making tasks between human and computers are complex (up to 10 levels). These have been applied to stage models of human information processing functions (information acquisition, analysis, decision selection, action implementation).

7.1.4 Pilots Associate Levels of Autonomy

Autonomy can be defined simply as the capability to make decisions. Pilots Associate (PA) Levels of Autonomy (LOA) and prime directives recognise the need for the operator/pilot to remain in charge. PA provides selectable LOA (Inactive, Standby, Advisor, Assistant, Associate) defined by operational relationships with bounded structure, but this needs to be readily communicable.

7.1.5 Cognitive Cockpit PACT

User knowledge acquisition produces a practical, communicable set of assisted PACT levels (At Call, Advisory, In Support, Direct Support) for variable and adaptive decision support/automation, supporting situation assessment, decision making and action. Research focus shifts to developing practical procedures for pre-assigning functions and tasks, operator initiated real-time changes, and triggering level changes from context-sensitive adaptive rules. Scripts and play-book tools for delegation of tasks/policy become relevant.

7.1.6 UAV/UCAV Autonomy

PACT LOA have been applied to the management of multiple UAVs to help pilot/ground operator workload. DARPA UAV programmes use four levels of autonomy with intermediate levels of exception and consent (UCAV), and veto and permissive (ICAV). UCAV research focus moves to multiple collaborating, autonomous groups covering complimentary, co-ordinated and co-operative planning and interactions.

7.1.7 Multi-Agent Adjustable Autonomy

Dynamic adaptive and adjustable autonomy is proposed for multi-agent intelligent systems for distributed problem solving structures in complex dynamic environments. Agents have self-direction and goals with capability to form, modify or dissolve the agent organisation. Degree of autonomy becomes linked to individual goal. Focus moves to the decision process for how a goal is pursued free from intervention, oversight or control by another agent. Autonomy with respect to goals is on a variable scale (consensus, master, local, command). Issues become rules for transfer of control, communication protocols, interaction styles, and cognitive strategies for reasoning with adjustable autonomy in operating context.

7.2 AUTOMATION AND HUMAN PERFORMANCE

Through the introduction of automation, the supervision of multiple functions is more and more assigned to a single human operator, the 'process manager' or 'supervisor', who is assisted by a process information system. Consequently, the level of direct involvement of the human operator with the actual process is decreasing. It must be realised that there is ample evidence that lack of attention to the human aspects in early phases of the development process of a system may result in wrong usability. It is therefore essential to consider in the pre-design phase, to what extent the machines should be made automatic and how this affects human factors related issues, with the purpose to minimise the risk of errors and to optimise system performance. The following deals with these human factors considerations, showing the areas that deserve attention.

7.2.1 Humans and Unmanned Military Vehicle

Current UMVs can operate in a more or less autonomous way. This means that the operator-supervisor is no longer directly operating the UMV directly, but parts of the work (semi-autonomous or semi-automatic) or the entire job (fully automatic or autonomous) are operated by the machine itself. In the consideration of semi-automatic operation, there may be various levels of human involvement. In semi-automatic systems, the system may address certain sub-tasks which are normally operated independently (e.g., with one command take-off and go to cruising altitude). As a next step, the machinery could perform complete well-defined tasks (e.g., perform a battle damage assessment task). As a next step, a more complicated set of tasks may be executed, such as fly back to base. Finally, a complete 'mission' may be run automatically. It may be clear that fully automatic systems may be very accurate in performing repetitive tasks. Yet, they typically lack flexibility.

At present, much effort is spent to the development of real-time process control systems as a means to increase the efficiency and the safety of automated systems. It is, however, important to maintain an operator-centred automation philosophy that overcomes limitations, enhances abilities and fosters acceptance when automated systems are introduced. Therefore, focus on human factors issues related to the automation is essential. Below, a generic list of issues is provided together with criteria that should be taken into account when automation is introduced.

In general, the user-centred design process starts with a function analysis, function allocation and task analysis process, in which the demands that UMV moving tasks put on the user, the identification of information to be processed and decisions to be made by the user, are determined. The opportunities provided by new technologies in performing UMV tasks have to be analyzed in terms of where and how they could support the human operator. For this purpose, a generic model of information transfer must be developed, which can be used to support the analysis. On the basis of this generic model, relevant issues are described which focus on

human factors aspects with respect to control and supervision of (semi-) autonomous systems. This must lead to a list of recommendations. Throughout the report, the relation with a practical situation is explained (i.e., UMV operation).

Technological developments more and more allow sub-systems of machinery to be automated. The potential benefits of automation express themselves in different areas: it can make the machines to perform the work faster and more accurate, it can reduce the fuel consumption and increase the economical utilization of the machines. Furthermore, automation can relieve the human operator from dull, repetitive or dangerous work, and reduce the workload. However, there are also potential drawbacks from a human factors point of view when automation is introduced. For example, in the cockpit of commercial airlines it appeared that automation lowered job satisfaction, induced human error, and caused a significant loss of skilled behavior. Parasuraman and Riley [1] give a summary that includes the use, misuse, disuse and abuse of automation. Stokes, Wickens and Kite [2, p. 101] stated that 'a potential danger is to automate tasks that are easy or beneficial to automate, instead of those tasks which are eligible for automation from the operator'. This means that sometimes the overall performance of a system may be enlarged by allocating a certain function to a human, although an automaton would perform better on the specific function. This implies that evaluating the system as a whole with the operator in-the-loop is at least as important as evaluating the performance of individual automated subsystems. "*designing or automating a human-machine system rarely produces an acceptable result without extensive searches through alternative designs plus experimentation to evaluate overall system performance; there are no shortcuts to success*" [3, p. 1459].

In the literature, different definitions of automation are used. The most common definition is used by Wiener [4], Satchell [5], and many others: 'Automation is concerned with replacing human functioning by machine functioning'. This applies for both partial and total replacement [6]. Despite the agreement on the definition of automation, several authors distinguish many sorts and levels of automation. Although these sorts and levels are not necessary to discuss the human factors involved in automation, they may be useful in understanding the conceptual frameworks and the complexity of the underlying knowledge domain. Wickens [7] distinguished automation for replacing the human, and automation for supporting the human; Wiener [4] distinguished automation of control tasks, and automation of monitoring functions (which can be divided into automation of detection, and automation of diagnosis). These distinctions are based on the different levels of human functioning, and resemble the approach of Sheridan [8]. The latter bases his conceptual framework upon the three levels as defined by the model of Rasmussen [9]: Skill based, rule based, knowledge based. Sheridan claims that for successful automation, one has to start at the skill based level; automation of tasks at the knowledge based level is exceptional.

Although the introduction of automation sometimes follows on the availability of new techniques and technological developments, it is also recognized that there are important potential advantages for the operator and the system output when human functions are transferred to a machine. Automatons can simply be better in certain tasks, for example, to perform route planning tasks in order to minimize fuel usage [10,4]. Even simple automatons may be faster, more accurate, more reliable, less stressful, less vulnerable for small errors, and cheaper than human operators. They even may be able to perform tasks, which the operator is able to specify, but not able to execute (including tele-operation). Important considerations may also be aimed at relieving the human from time consuming, dangerous, complex, tiring or dull (repetitive) tasks [2]. However, the main goal concerning the human operator is to reduce workload [3,11]. This may enhance both the well-being of the operator, and the performance of the total system. There are additional effects of automation, which are not directly the goal of the design process, but offer opportunities to further improve system performance (e.g., automation may offer possibilities to introduce on-line simulation of the work process; predictive displays may be used to support the anticipation process; 'fail soft' protection may prevent

operators and equipment to commit certain errors, limiting the influence of individual differences on system performance).

It may be clear that the role of the human operator is considered trivial by some, and critical by others; the fundamental nature and the characteristics of the interaction between operator and automated system is still a matter of debate and design evolution. This introduction underlines the importance of considering the human factor when automatic systems are introduced.

7.2.2 Human Factors and Automation

Automation may be introduced to replace or support operator functions at different levels. In the literature, various sub-divisions are made, ranging from 'no automation' to 'fully autonomous operation'. A useful subdivision was made in AGARD [6] with seven different levels of automation (i.e., no automation, manual augmented, manual augmented limited, co-operative, automatic pre-select, automatic select, and autonomous operation). In this Introduction, these levels of automation are applied to UMVs. An eighth level (unmanned operation) is added. Note that unmanned operation is possible at all levels of automation, however, this is not considered within the scope of this Introduction. Practical applications are suggested for each level of automation, and the implications with respect to human factors considerations are discussed.

1) No Automation

The system is manually operated without any automatic augmentation or support. The operator is performing activities using his human faculties (e.g., visual functions, mental activities, visual inspection, verbal communication). The control movements by the operator directly affect system output.

2) Manual Augmented

Manual control is augmented by an automatic system when an automatic system assists the operator by controlling simple activities (so-called low level activities). Examples are augmented systems that control a single system variable (e.g., speed, frequency, rpm, strength, pressure), or provide low level decision support (e.g., records images when desired).

3) Manual Augmented and Limited

Manual control is augmented and limited when an automatic system assists the operator by controlling low level activities such that over-control and control-errors are avoided. An example is the anti-lock brake system (ABS) in cars; the driver needs less pedal force to execute a brake action, and moreover, when a full stop manoeuvre is made, the braking action is limited in order to avoid break-out of the car. Another example is the speed-limiter in trucks; the truck will stop acceleration when a certain speed is reached, even when the gas pedal is pushed down further. An example in decision making support is the monitoring of input sequences (e.g., a car does not start the engine as long as it is put into gear).

4) Co-operative

Co-operative support of manual control is an automatic function with pre-selection of parameters and their values, in combination with manual control. An example is dynamic positioning of a ship during disturbances (e.g., in wind and water current). Another example is cruise control in cars; the driver selects the desired speed and the car will drive at that speed, irrespective of the road condition (e.g., highway or desert road, hill upward or downward). Note that co-operative support may be superimposed upon the

lower support levels to enable the operator to go beyond the installed limits when certain extreme control conditions are met.

5) Automatic Pre-Select

Support of manual control by automatic pre-select is an automatic function with pre-selection of parameters and their values, without manual control. In fact, automatic pre-select may be conceived as the replay of a set of pre-programmed actions. An example is automatic course control (auto-pilot control) on board ships; the navigator selects a new course, starts the course change manoeuvre by pressing a push-button, and the vessel will automatically alter course with a pre-selected turn rate. Note that pre-select automation can be superimposed upon the lower support levels (e.g., the navigator cannot execute manoeuvres with course changes of over 180 degrees).

6) Automatic Select

Support of manual control by automatic select is an automatic function that performs certain functions automatically. These functions can be selected or deselected (switched on or off) by the operator. An example is an automatic track keeper on a ship: The navigator selects navigation points (way points) to follow, enables automatic select mode by pressing a push-button, and the vessel will determine the optimal track between the two navigation points (considering the effect of water current, wind, and local weather by means of databases), and then start to follow this optimal track. The navigator only has to accept (correct, or reject) the proposed track.

7) Autonomous Manned Operation

During autonomous manned operation, the work is done automatically. The operator is still present on the vehicle, monitoring the system and surveilling the execution of the work procedure.

8) Autonomous Unmanned Operation

The work is done autonomously. There is no operator on the vehicle itself.

The effect of these levels of automation on the human factors of UMVs is roughly evaluated below. For this purpose, the effects of different levels of automation on the main human factors related issues were assessed by the author. Table 7-1 shows a summary of the results. In the Table, a + was rated when a positive effect on the related issue was expected, a ++ when the effect was rated very positively. A – was rated when a negative effect on the related issue was expected, a – when the effect was rated very negatively. The assessment is made subjectively by the author, based on the author' expertise in human factors and on the information derived from the literature. It has to be stressed that some items in the assessment are disputable, but the overall picture gives an indication of the advantages and drawbacks of the different automation levels. All the arguments that led to Table 7-1 are explained hereafter.

Table 7-1: Evaluation of the Application of Automation on Unmanned Military Vehicles (UMVs)

Automation Level	Human Factors Related Issues									System Output	
	Peripheralization: Cognitive Issues		Peripheralization: Control Issues			General Issues					
	Task Involvement 1.1 [1])	Vigilance 1.2	Direct Control 2.1	Maintaining Skills 2.2	Maintaining Awareness 2.3	Few Mental Resources Required 3.1	Relief of Physical Workload 3.2	Comfort 3.3	Little Training Required 3.4	Output Quality	Economical Usage
1) No Automation	+ +	+ +	+ +	+ +	+	+ +	- -	-	- -	+ +	- -
2) Manual Augmented	+ +	+ +	+ +	+ +	+	+ +	-	-	-	+ +	- -
3) Manual Augmented Limited	+ +	+ +	+	++	+	+ +	-	-	-	+ +	-
4) Co-operative	+	+ +	-	+	-	+ +	+	+	+	+	+
5) Automatic Pre-Select	+	+ +	-	+	-	+ +	+	+	+ +	+	+ +
6) Automatic Select	-	+ +	-	-	-	+ +	+	+	+ +	+	+ +
7) Autonomous Manned	-	+	+ +	- -	+ +	-	- -	- -	+ +	-	+ +
8) Autonomous Unmanned	-	-	+ +	- -	+ +	- -	+ +	+	- -	-	+ +

Note: -- = very negative effect
- = negative effect
+ = positive effect
++ = very positive effect

As indicated in Table 7-1, *task involvement (1.1)* – a low state increases the risk of complacency (i.e., a low index of suspicion due to automation) – is significant at lower levels of automation. The operator is (nearly) solely operating the UMV, directly controlling airframe and sensors. The operator is closely 'in-the-loop', that is, he is continuously involved in manual control activities, and, as well, continuously physically influenced by the behavior of the vehicle. The more automatic equipment is installed, the more the operator will rely on this, which increases the risk of complacency. In automation levels 6 – 9 there are situations that complete functions will be taken over by the machine, resulting in a low degree of task involvement (i.e., high risk of complacency).

Vigilance (1.2), defined as the ability to detect infrequent signals over prolonged periods of time, is high when the operator actively participates in the process. At automation levels 1 – 6 there are still a number of manual activities to perform by the operator. This is hardly the case during autonomous operation (levels 7 – 8), when nearly all operator functions are performed by the automated systems; however, the negative effect on vigilance will be limited in these situations when adequate feedback of the process state is provided to the operator.

In the lower levels of automation (levels 4 – 6) there is a direct link between operator control actions and process control activities. In case of high level automation (levels 7 – 8) there is a direct link between automatic system actions and process control activities. In both cases it is well possible to match input and output, which has a positive influence on *direct control (2.1)*. In semi-automated operation (levels 4 – 6), control activities are performed by operator and by machines simultaneously. It is then well possible that the sequencing of the output activities is conflicting; automatic functions will be executed more or less independently from actions taken by the operator (e.g., it is known that pre-select parameters may be changed during execution just to fool the automatic system in order to speed up the sequencing, or to start the sequence earlier). This has a negative influence on direct control.

Automation has a drawback on *maintaining skills (2.2)*, that is, the ability to perform the manual task properly when situations require switching back from automatic to manual control. Literature indicates that, the more automatic systems there are introduced, the more the operator will be placed out-of-the-loop. A lack of operator involvement is the result, with increasing risk in loss of skill. Particularly in the design of autonomous systems this deserves attention.

About *operator awareness (2.3)*, defined as the ability to perceive elements in the environment and to understand their meaning, it is stated that the operator is fully aware of machine state and controlled process state in case there is a high degree of direct control (see 2.1, the levels 1 – 3, and the levels 7 – 8), provided that there is adequate feedback of machine and process state. However, in levels 1 – 3 it may be expected that the operator is not fully aware of what the impact of his actions be on a number of activities (e.g., does his activity still fit in the work plan; what are the actual costs; are there time delays; are there changes in the work plan). Worst operator awareness is rated when a part of his activities is taken over by automated systems (i.e., items 4 – 6). Machine and operator are performing their assigned activities, with the risk that the machine performs activities that are not (or insufficiently) perceived by the operator/supervisor. For example, in case of an automatic visual target tracking system, it is important to have the system dead-reckoning when the target is obscured. Again, adequate feedback is essential in this case.

Part-task automation (levels 1 – 6) has a positive effect on the required *mental resources (3.1)* (i.e., routine tasks require few mental resources, resulting in low mental workload). The operator still directly perceives (e.g., by force feedback, auditory/vibration feedback) the system's activities, and few mental transformations have to be made to understand what is going on, and what (manual) activities have to be performed. It is

assumed that more operator attention is required during autonomous operation, particularly in an unmanned situation. The operator is then tele-operating the system, while collecting information only through feedback systems. In that case it is expected that more mental transformations are needed (e.g., engine rpm has to be verified by means of an indicator on a screen, instead of interpreting engine noise/vibration).

Physical workload (3.2) – the biomechanical conditions of the operator during task execution is fully relieved during autonomous unmanned operation when the operator is located in a remote station. There will be some relief in case certain functions are taken over by automatons (levels 4 – 7); no relief of physical workload is expected when no automation is installed (level 1), or when the operator is manning an autonomous system (level 7). In the latter case, the operator is sitting on board a machine that performs activities autonomously, most of which the operator is not aware of or even may not expect.

Comfort (3.3), in the sense of physical workload, is rated low for low level of automation (levels 1 – 3), and medium for part-task automation (levels 4 – 6). Comfort, in the sense of being informed about the actual status of the total system, is rated very low in case of manned autonomous operation (level 7). The machine will perform actions that are not attended by the operator, who is sitting in the cabin.

Little training (3.4) is required in situations of part-task and full automation (i.e., automation levels 4 – 7). Note that training in autonomous unmanned operation (level 8) may include tasks that are not relevant in the context of the other automation levels (1 – 7).

System output quality is rated positive when few operator functions are automated. The more automatic systems are used, the less flexible the system will be. In case of part-task automation conflicts between operator actions and automatic system activities are likely to happen; in case of autonomous operation the sequencing, the function execution mostly is pre-programmed. In contrast to this, *economical usage* of the system will increase the more automatic systems are used (i.e., use on a continuous basis; fine-tuned in energy consumption for each job; better maintenance scheduling, less influence of environmental disturbances, etc.).

7.2.3 Recommendations

The following recommendations with respect to automation are given:

1) Select the proper level of automation

Each level of automation has its specific implications on the human factors aspects. When considering Table 7-1, a number of conclusions can be drawn. Low level of automation (levels 1 – 3) results in best quality of work and will not lead to operator peripheralization. However, the operator then performs under a high degree of physical load which limits the economical usage of the equipment. Medium level of automation (levels 4 – 6) is rated most positively. It is essential, however, that adequate feedback of the system state is provided to the operator so that optimal awareness is maintained (e.g., by providing visual display information and haptic feedback in the controls). The operator is kept in-the-loop which benefits the quality of task execution. Total (autonomous) operation is only beneficial in an unmanned configuration.

2) Keep the operator in-the-loop, increase operator awareness, and provide adequate feedback

The operator should actively participate in the job. Adequate visual feedback on displays should be presented for monitoring purposes, and haptic feedback by providing active controls for sensory information. Have the

operator activate sequences, or confirm the actual system status, on a regular basis. Avoid simultaneous monitoring and manual control activities. Avoid discrepancies between indicators and actual mode, e.g., use pushbuttons with visual indicators. Introduce dedicated displays, presenting the present status, the oncoming actions, and the current plan. The above mentioned importance of feedback is confirmed in the literature, including lessons learned in cockpit automation, and in generic model descriptions of a human-machine interface confirming the benefits of operator-in-the-loop performance [12]. Automation may result in eliminating or replacing critical cues. For example, specific information in manual control tasks (e.g., haptic information in joystick controls) is needed to detect system state changes [13]. Norman [14], as well as Korteling and Van Gent [15], stated that without appropriate feedback, operators are indeed out-of-the-loop: They may not be aware whether their input has been received, whether actions are performed properly, and what the intentions of the system are. In tracking tasks, error detection is better with manual control, due to the presence of proprioceptive feedback [16]. Kessel and Wickens [17] hypothesized that the fact that subjects operating under manual control had both visual and proprioceptive feedback, contributed to better performance. The need for appropriate feedback will increase with increasing complexity and with increasing autonomy of the system.

3) Introduce flexible automation

The operator should be able to select the automated activities at all time. This puts high demand on the designers of the automation algorithms. A good way to overcome this problem is to have the operator 'learn' the system how to perform the individual activities (e.g., the operator first shows the machine how to equalize a pile of material, or, how to put a load at certain positions when performing fork work activities).

4) Natural operation

The system should perform automatic activities as if this was done by the operator himself during automatic task execution. There will be less unattended actions of the system, which improves the operator's awareness and comfort, increasing total system safety and performance. Natural operation is particularly important when the operator has to override the automatic system by switching back to manual control.

5) Forestall loss of skill

The best way to prevent the loss of operator skill is probably to periodically give the operator dedicated training. Another possibility is to require the operator to perform skill critical tasks manually at certain times, even though the task may have been allocated to the automated system. Furthermore, the use of active controls, and the use of a system in which the operator 'learns' the machine how to perform a task will also help to prevent skill loss. Both these options were mentioned earlier.

6) Smooth mode transitions

Enable a smooth transition between manual and automated control mode, e.g., prevent sudden forces on controls. In fact, this implies that active controls must be installed. The operator continuously feels what the automatic system is doing just by putting his hands on the controls. Taking over manual control could then be just a matter of further activating that control (e.g., putting pressure on the grip, or activating a switch).

This Introduction outlined various human factors considerations in the use of automatic systems. It indicated possible advantages and disadvantages of the various levels of automation.

7.2.4 References

[1] Parasuraman, R. and Riley, V. (1997). Humans and automation: Use, misuse, disuse, abuse. Human Factors.

[2] Stokes, A., Wickens, C.D. and Kite, K. (1990). Display technology, human factors concepts. Warrendale PA: Society of Automotive Engineers.

[3] Boff, K. and Lincoln, J.E. (1988). Engineering Data Compendium: Human Perception and Performance. Wright Patterson AFB, OH: AAMRL.

[4] Wiener, E.L. (1988). Cockpit automation. In: E.L. Wiener and D.C. Nagels (Eds.) Human factors in aviation (pp. 433-461). San Diego: Academic.

[5] Satchell, P.M. (1993). Cockpit monitoring and alerting system. Brookfield: Ashgate.

[6] AGARD (1986). Improved guidance and control automation at the man-machine interface. Hollister, W.M. (Ed.). AGARD advisory report AR-228. Neuilly-sur-Seine: NATO.

[7] Wickens, C.D. (1992). Engineering Psychology and Human Performance. New York: Harper Collins.

[8] Sheridan, T.B. (1988). The system perspective. In: Wiener, E.L. and Nagel, D.C. (Eds.) Human factors in aviation. San Diego: Academic Press.

[9] Rasmussen, J. (1983). Skills, rules and knowledge; signals, signs, and symbols, and other distinctions in human performance models. IEEE Transactions on Systems, Man, and Cybernetics, 13, 257-266.

[10] Hawkins, F.H. (1987). Human factors in flight. Brookfield: Gower Publishing Company.

[11] AGARD (1984). Human factors considerations in high performance aircraft. AGARD conference proceedings CP-371. Neuilly-sur-Seine: NATO.

[12] Passenier, P.O. and Van Delft, J.H. (1997). Human factors in process control: designing the human interface. Journal A, 37, 10-14.

[13] Young, L.R.A. (1969). On adaptive manual control. Ergonomics, 12, 635-657.

[14] Norman, D.A. (1989). The problem of automation: Inappropriate feedback and interaction not overautomation. ICS Report 8904. La Jolla, CA: University of California, San Diego, Institute for Cognitive Science.

[15] Korteling, J. and Van Gent, R.N.H.W. (1994). Anticipatie en actieve stuurmiddelen bij helicopterbesturing [Anticipation and active controls in helicopter flight]. Report TM 1994 A-2. Soesterberg, The Netherlands: TNO Human Factors.

[16] Parasuraman, R., Mouloua, M. and Molloy, R. (1996). Effects of adaptive task allocation on monitoring of automated systems. Human Factors, 38, 665-679.

[17] Kessel, C.J. and Wickens, C.D. (1982). The transfer of failure detection skills between monitoring and controlling dynamic systems. Human Factors, 24, 49-60.

7.3 HUMAN AUTOMATION INTEGRATION WITH CONTRACTUAL AUTONOMY

7.3.1 Introduction

7.3.1.1 Function Allocation

Automation is continually improving in capability with associated changes in perceptions of appropriate human roles and the suitability of functions for human and/or machine performance. Traditional engineering mostly used the "left over" principle for allocation of function, where the technical system was designed to do as much as is feasible from an efficiency point of view, and the rest was left for the operator. HF engineering introduced the compensatory principle, where human and machine capabilities are compared on salient criteria and the function allocation is made so that the respective capabilities are used optimally. In 1951, Paul Fitts suggested some simple criteria for allocating functions between people and machines to predict roles in future air navigation and air traffic control systems [1]. Fitts distinguished between four kinds of control systems, namely:

1) Fully automatic control;
2) Automatic control with human monitoring;
3) Semi-automatic control supplemented by human performance of critical functions; and
4) Primary control by human operators.

The 1994 NATO RSG workshop on function allocation [2] reiterated the questions posed by Fitts and his colleagues which still lacked general answers:

- Should the human monitor the (technical) system given that humans are poor monitors?
- Should the (technical) system monitor the human?
- If so what roles should the human play and what are their responsibilities?
- Are humans included in systems just to deal with those functions that engineers can not automate?

Options on decision making were noted to range from the principle that the human should make all decisions, because humans are responsible for systems, to the principle that there are some decisions that humans should never be permitted to make.

In the 1980's, with increasingly capable intelligent computing, ideas of human-computer teamwork, cognitive engineering, cognitive automation and joint cognitive systems began to emerge [3]. According to the *complementarity* principle, function allocation serves to support and sustain human ability to perform efficiently. Here, the focus shifts from human-machine interaction to human-computer co-operation, and from the internal functions and structure of the human and machine to the external functions and establishing the system boundaries [4].

The capability for collaborative working with other agents, including humans, is a goal for intelligent automation. Taylor and Reising [5] noted that in order to work collaboratively with humans, intelligent automation probably requires a functional architecture with the following attributes:

- A model of human decision making and control abilities.

- The ability to monitor human performance and workload through behavioural and physiological indices.
- The ability to predict human expectations and intentions with reference to embedded knowledge of mission plans and goals.

7.3.1.2 Automation Reliability, Trust and Use

Issues of "trust" have featured strongly in implementation of ideas of dynamic function allocation, automated decision support and human-computer collaborative adaptive systems. This is in response to concerns about human monitoring of unreliable automation and the need for control strategies to mitigate bias from complacency and over-trust, or alternatively, under-trust and disuse [6].

Early UK MOD research with aircrew investigated the structure and measurement of trust to help design automation safeguards. A study of twin-crew RAF Tornado aircraft operations, elicited tactical decision making scenarios and aircrew rated them for the importance of factors associated with trust [7]. Demand for trust was associated with perceived risk and the probability of negative consequences, whereas the supply of trust was related to the requirement for judgement and awareness, and uncertainty and doubt in making decisions. Relying on others to make risky decisions calls for a large amount of trust. If the decision requires another person exercising a high degree of awareness and judgement, and there is much uncertainty and doubt in the decision provided, then the actual trust engendered by the decision will be low. In a follow-up study on the quality of aircrew teamwork [8], trust was found to be a significant factor in distinguishing between good and poor teamwork performance. Trust was rated at a significantly lower level in single-seat RAF Harrier operations (i.e., human-computer teamwork) than in two-seat RAF Tornado aircraft tactical operations (i.e., both human-computer and human-human teamwork).

Several models of trust have been proposed. Riley [9] developed a model of the relationships between trust, operator skill level, task complexity, workload and risk, self-confidence and automation reliability. Studies in which workload and reliability were varied, led to refinement of the model to include factors of fatigue and learning about system states [10]. Further research has modelled trust as a function of recent performance, and the presence and magnitude of fault, with subjective trust increasing with automation reliability [11]. The relationship between reliability and trust was confirmed by human monitoring performance measures indirectly measuring trust [12]. Recent work has extended modelling to investigate how the dynamics of trust and reliance depend on information sharing [13], and how trust becomes over-trust through unintended use [14].

Experimental evidence has verified that unexpected automation failure leads to a breakdown of trust, and to difficulty in the recovery of trust with a loss of faith in future teamwork performance. As trust declines, manual intervention increases [15,16]. Research has showed how when workload is increased, over-trust or complacency develops with automatic systems, and coupled with vigilance problems, this is likely to lead to failure to detect performance deviations and decrements in automation performance [17]. Operator detection of automation failure is substantially degraded with a static allocation fixed over a period of time, favouring dynamic adaptable allocation [18]. Manual task reallocation has been proposed as a countermeasure to monitoring inefficiency and complacency since short periods of intermittent manual task reallocation, or cycling between manual and automation control, reduces failures of monitoring [19]. By maintaining manual skill levels, and enhancing situational awareness, manual task re-allocation helps when intervention is needed following automation failure. However, without active involvement, it is difficult to maintain an appropriate dynamic internal model of the important changing relationships needed for regaining manual control following automation [20]. Experimental studies have shown that with competing demands for

attention, humans are poor at monitoring automation for occasional malfunctions, exhibiting automation complacency [17]. Humans also have poor awareness of adaptable automation failure [21,22].

Generally, trust is best considered as an *intervening variable* between automation reliability and automation use [6]. For purposes of measuring aiding effectiveness, like "confidence", trust operates as a psychosocial attitude with inherently variable subjective complexity, rather than as a cognitive functional state such as "situation awareness" and "workload". Trust is unlikely to be reliably linked to performance and effectiveness. Furthermore, "trust" is unlikely to provide reliable psychometric or concomitant behavioural measures (qualitative or quantitative), with useful sensitivity, discrimination, diagnosticity or predictive power.

7.3.1.3 Trustworthy Levels of Automation

For the design of adaptable automation and automated decision support, research effort is needed to be directed at constructing and constraining human-automation relationships, interactions and behaviours in a manner that naturally engender trust. Methods for delegating authority to automation are needed that manage and control risks of automation in a sensible, regulated and predictable manner, with appropriate safeguards. Safeguards are needed against breakdown or failure in performance to ensure that operator trust in system functioning is maintained at realistically appropriate levels, without adversely affecting situational awareness. The first Law of Adaptive Aiding states that "computers should take tasks, but not give them" [19]. Automatic reallocation of tasks to manual performance seems close to a violation of this law. In particular, variable assistance and allocation could lead to unacceptable unpredictability. So, awareness of the current task allocation strategy is an important factor for system effectiveness, but this may not be easily achieved by seamlessly adaptable aiding. Careful consideration is needed of the procedures for implementation of dynamic task allocation and re-allocation.

The building of trust between the operator and the computer automation system has been identified as a key issue in enabling the capability of cognitive automation. Trust is built when consistency and correctness is observed in the computer system's decisions and actions. Two important guidelines for building trust have arisen [23]:

- *Define the Prime Directives.* These are overall governing rules which bound the behaviour of the aiding system, and yet provide a logical structure for aiding system to act in a rational and reliable manner, avoiding arbitrary behaviour, so that the human does not experience any surprises, e.g., Asimov's Laws of Robotics.
- *Specify the Levels of Autonomy.* These also bound the behaviour of the aiding system by limiting its decision authority for the performance of specific sub-functions to a set of system configurations specified and set by the operator.

Trust is built on awareness of proven performance. Adaptive strategies for coping with control of complex, dynamic situations, such as automatic unburdening and manual re-allocation, will need careful adaptive logic to ensure their appropriateness. The design of the functional interface needs to ensure that appropriate levels of awareness of the current task allocation are easily maintained. Awareness is needed to avoid task contention, and to ensure that tasks are not overlooked or performed incorrectly.

Ideas of levels of automation have been proposed to represent scales of delegation of tasks to automation, with implications for reliability, use and trust. Sheridan and Verplanck [24] first proposed 10 possible levels of allocation of decision-making tasks between humans and computers. More recently, Parasuraman, Sheridan and Wickens [25] have considered the application of automation to a four-stage model of independent

information processing functions (information acquisition, analysis, and decision selection and action implementation). In doing so, they have sought to apply a revised set of levels of automation.

- The computer decides everything and acts autonomously, ignoring the human.
- The computer informs the human only if it, the computer, decides to.
- The computer informs the human only if asked.
- The computer executes automatically, then necessarily informs the human.
- The computer allows the human a restricted time before automatic execution.
- The computer executes the suggestion if the human approves.
- The computer suggests an alternative.
- The computer narrows the selection down to a few.
- The computer offers a complete set of decision alternatives.
- The computer offers no assistance. The human must make all the decisions and actions.

The term autonomy has been introduced to describe the bounding of functioning and decision authority of advanced automation and intelligent decision systems. Autonomy can be defined simply as the capability to make decisions. Thus, autonomy can be considered in terms of the freedom to make decisions, considering constraints on decision-making (limitations, boundaries, rules, regulations), decision-making abilities (authority, responsibility, competency), and the capability to make different kinds of decisions (classes, functions, levels).

In the 1980's, the DARPA/USAF Pilot's Associate (PA) program provided a practical implementation of intelligent pilot aiding based on prime directives and levels of autonomy (LOA). PA design was guided by a top-level operational philosophy based on the pilot being in charge. The goal of the PA was to provide consistently correct information, and to aid the pilot's decision making by helping to manage workload, reduce confusion, and simplify tasks. This led to the philosophy of the PA as an intelligent subordinate to the pilot, with specific capabilities for decisions and actions. These top level requirements led to specific operational relationships (ORs) for discrete PA sub-functions interactions, with increasing degrees of automation and autonomy. From these ORs, pilot selectable levels LOA were obtained for groups of functions governed by the required pilot operational relationship and interaction [26]. Five discrete LOA modes were proposed, namely: Inactive, Standby, Advisor, Assistant, Associate. Each LOA mode was associated with tailorable functional clusterings for flexible responding to avoid too rigid automation imposed by design. These modes were aimed to provide a bounded, communicable structure for delegated levels of authority, minimising mode confusion, and building trust and confidence. Generally, human factors research indicates that the required control structure should be cognitively simple, and not complex. Pilots tend to view computer autonomy simply as either automatic, with or without status feedback; semi-automatic, telling what will happen and asking permission to proceed; or advisory, providing information only.

7.3.2 Contractual Autonomy

7.3.2.1 Pilot Authorisation and Control of Tasks

In order to address the requirement for a practical implementation of LOA, under the UK MOD Cognitive Cockpit programme a framework was created for pilot interaction and delegation of adjustable levels of autonomy, known as the PACT system (Pilot Authorisation and Control of Tasks). PACT spans the range of

possible levels of allocation of decision making tasks, or levels of autonomy, between humans and computers [24,25]. The PACT framework is intended to provide trustworthiness in the information and behaviour of adaptable automation [27,28]. The PACT framework is summarised in Table 7-2.

Table 7-2: Bonner-Taylor PACT System

Primary Modes	Levels	Operational Relationship	Computer Autonomy	Pilot Authority	Adaptation	Information on performance
Automatic		**Automatic**	Full	Interrupt	Computer monitored by pilot	On/off Failure warnings Performance only if required.
Assisted	**4**	**Direct Support**	Advised action unless revoked	Revoking action	Computer backed up by pilot	Feedback on action. Alerts and warnings on failure of action.
	3	**In Support**	Advice, and if authorised, action	Acceptance of advice and authorising action	Pilot backed up by the computer	Feed-forward advice and feedback on action. Alerts and warnings on failure of authorised action.
	2	**Advisory**	Advice	Acceptance of advice	Pilot assisted by computer	Feed-forward advice
	1	**At Call**	Advice only if requested.	Full	Pilot, assisted by computer only when requested.	Feed-forward advice, only on request
Commanded		**Under Command**	None	Full	Pilot	None performance is transparent.

PACT simplifies the number of automation modes required – fully automatic, assisted, commanded – with a further secondary levels nested within the semi-automatic assisted mode, which can be changed adaptably or by pilot command. The PACT framework employs military terminology for categories of support in British Army land forces (Mike Bonner, personal communication). This provides realistic operational relationships compatible with military user control schemata. It is a logical, practical set of levels of automation, with progressive operator/pilot authority and computer autonomy supporting situation assessment, decision making and action, as illustrated in Figure 7-1.

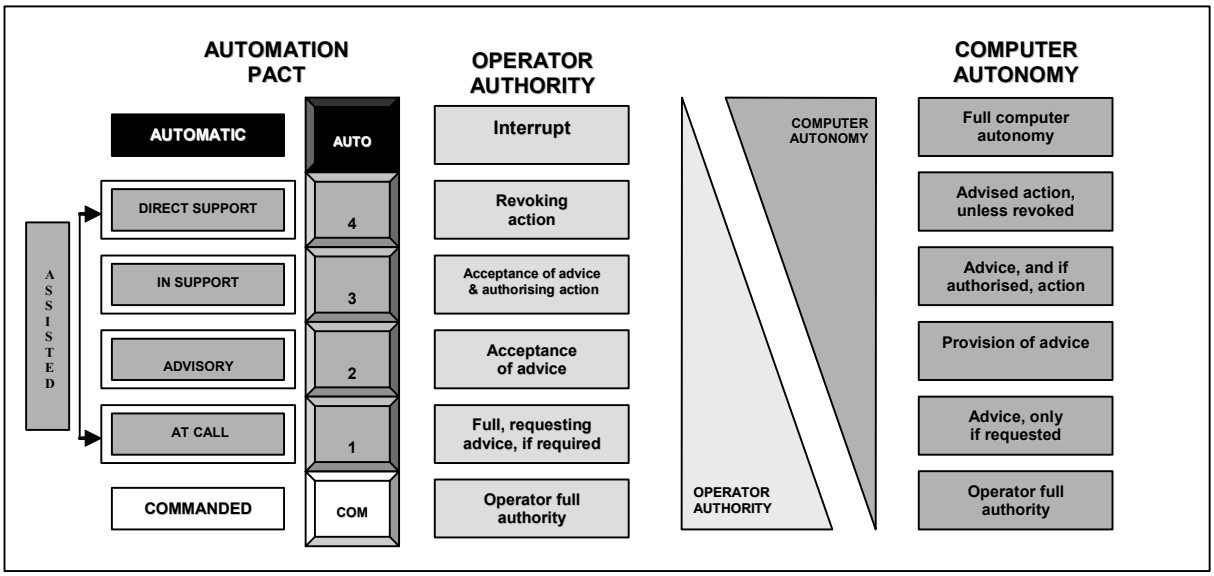

Figure 7-1: Pact Progression of Operator Authority and Computer Autonomy.

The purposes of the PACT framework can be summarised as follows:

- To bound the behaviour of the aiding system;
- To limit its decision authority to the performance of specific sub-functions; and
- To enable a set of system configurations to be specified, set and adjusted by the operator.

PACT is based on the idea of *contractual autonomy*. Using an aircrew term from co-operative air defence, the pilot forms a set of "contracts" with the automation by allocating tasks to PACT modes and levels of automation aiding. The contract defines the specific nature of the operational relationship between the pilot and the computer aiding for co-operative performance of specific sub-functions and tasks. In setting the PACT contract, the operator defines:

- What sub-functions and tasks are aided, when and how;
- What level of assistance is provided as primary or default, and when;
- What levels of assistance are permissible for anticipatable contingencies, and when;
- What are permissible triggers for changing levels of assistance, either contextual or by operator command; and
- What information is provided to the operator, when and how, including status advice, feed-forward/ feedback course of action information and saliency.

Thus, autonomy is limited by the set of contracts made between the pilot and the computer automation system governing and bounding the performance of tasks to a set of sensible and predictable co-operative behaviours according to rules of operation (context, resources). Fighter pilots develop similar inter-personal contracts in planning control of multi-aircraft manoeuvres in co-operative air defence. The PACT autonomy contracts are binding delegation agreements for the computer. Only the pilot can set or modify the PACT contracts, or define a priori the contextual circumstances for real-time adaptation PACT changes. Mission functions and

tasks can be set to PACT levels by allocation individually or grouped in related scripts or plays, at different levels of abstraction, in a number of ways:

- Pre-set operator preferred defaults.
- Operator selection during pre-flight planning.
- Changed by the operator during in-flight re-planning, probably using Direct Voice Input commands.
- Automatically changed according to operator agreed, context-sensitive adaptive rules.

Figure 7-2 illustrates a set of mission functions and tasks with PACT contractual autonomy levels arranged along a timeline in a hypothetical task network. This provides the operator with implicit if not explicit control, so as to engender trust through understanding of automation functioning. Thus, the pilot retains authority and executive control, while delegating responsibility for the performance of tasks in a sensible and predictable manner to the computer.

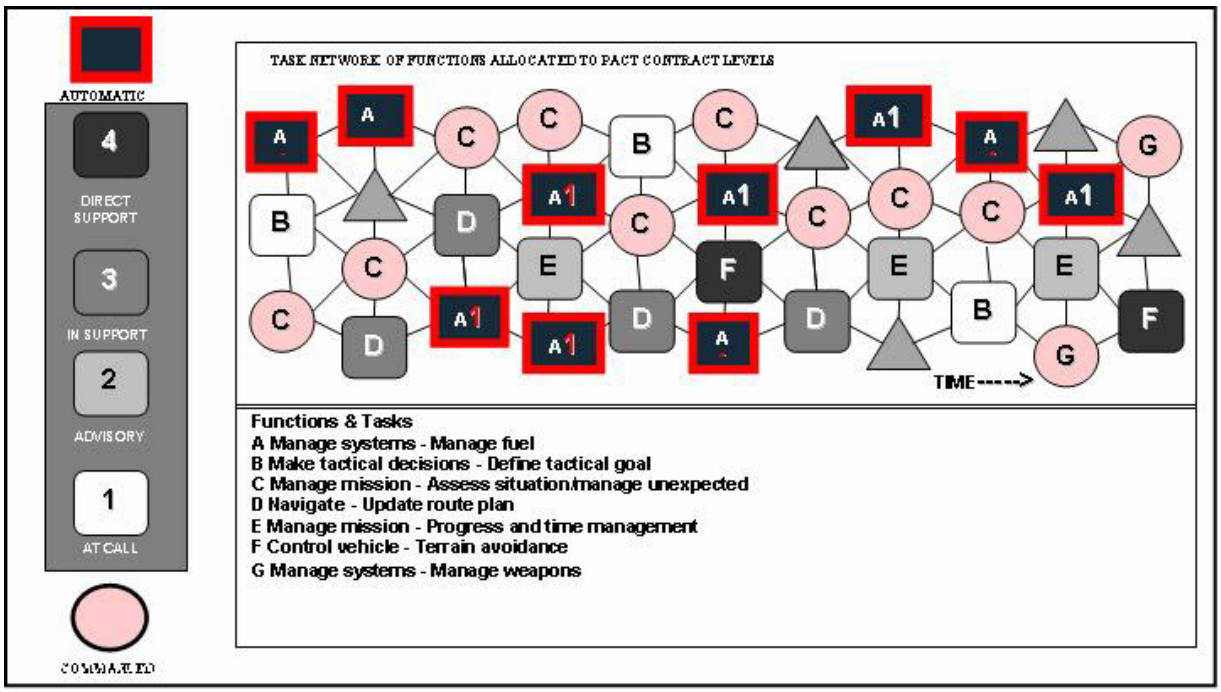

Figure 7-2: Task Network of Functions and Tasks Set to Pact Contract Levels.

In the Cognitive Cockpit implementation, the PACT system operates within an adaptive system architecture that couples on-line monitoring of the pilot's functional state and on-line task knowledge management and decision support for context-sensitive aiding, deriving information to mediate the timing, saliency and autonomy of the aiding. The PACT framework provides the necessary and sufficient levels of autonomy for the management of tasks. Three principle agents with different tasks comprise the Cognitive Cockpit system:

- A *Cognition Monitor* (COGMON) is responsible for monitoring the pilot's physiology and behaviour to provide an estimation of the pilot's functional state.
- A *Situation Assessor Support System* (SASS) is responsible for monitoring the aircraft situation and outside environment, generating advice and recommending courses of action.

- A *Task Interface Manager* (TIM) is responsible for monitoring the mission plan, deciding automation and managing the cockpit interface.

The TIM module provides on-line analysis of higher-order outputs from COGMON and SASS, and other aircraft systems. A central function for this system is maximisation of the goodness of fit between aircraft status, 'pilot-state' and tactical assessments provided by the SASS. These integrative functions enable this system to influence the prioritisation of tasks and, at a logical level, to determine the means by which pilot information is communicated through the TIM and the associated cockpit interfaces. Overall, this system allows pilots to manage their interaction with the cockpit automation, by context-sensitive control over the allocation of tasks to the automated systems.

The TIM functional architecture comprised modules for goal-plan tracking and for interface, timeline, automation and task management utilising a blackboard for goal-plan tracking information. Details of the TIM functional architecture are provided elsewhere [37,38]. The idea of a tasking interface exploits the lessons learnt from the US Army's RPA program [39]. It arose from the need to be able to predict pilot expectations and intentions with reference to embedded knowledge of mission plans and goals. The aim was to provide an adaptive or "tasking" interface that allowed the operators/pilots to pose a task for automation in the same way that they would task another skilled crewmember. It afforded pilots the ability to retain executive control of tasks whilst delegating their execution to the automation. A tasking interface necessitated the development of a cockpit interface that allowed the pilot to change the level of automation in accordance with mission situation, pilot requirements and/or pilot capabilities. It was necessary that both the pilot and the system operated from a shared task model, affording communication of tasking instructions in the form of desired goals, tasks, partial plans or constraints that were in accord with the task structures defined in the shared task model.

Providing flexible or adjustable levels of autonomy for the performance of tasks and functions is a key requirement for implementation of the tasking interface concept. Allowing pilots to choose various levels of interaction for the tasks they are required to conduct can mitigate the problem of unpredictability of automation. TIM utilises the monitoring and analysis of the mission tasks provided by the SASS combined with the pilot state monitoring of the COGMON to afford adaptation of automation, adaptable information presentation and task and timeline management.

In the Cognitive Cockpit implementation, PACT levels are triggered adaptively, in accordance with PACT contracts, in response to contextual input from COGMON, SASS and TIM mission goal-plan tracking (GPT). The intention is to monitor and manage the variability in performance through a barrier system approach (monitor, detect, correct, reflect performance), and through appropriate cognitive streaming interventions (join, break, divert cognition). TIM feedback and feed-forward control messages are used with appropriate multi-modal intervention saliency (background, hinting, influencing, directing, compelling) developed to reduce cognitive bias with decision support systems. All the tasks in the mission scenario are pre-allocated to *possible* PACT level contracts by the pilot. The individual task PACT levels (defaults and contingencies) are set to mitigate the risks to achievement of the individual task goals. The TIM Task Manager distinguishes between pending, active and completed tasks for the current scenario/vignette. Individual tasks progressed from pending, to active, and then to completed, as the scenario progressed.

7.3.2.2 PACT Evaluation

The operation of the PACT framework was successfully demonstrated to the MOD Cognitive Cockpit customer in 2001. It has subsequently been incorporated into interfaces for UAV control and been demonstrated

successfully [29]. Diethe et al. [36] reported that a successful DARPA Augmented Cognition programme Cognitive Cockpit closed loop trial was completed in November 2003 during which the stability and performance of the Cognitive Cockpit system were examined under different levels of threat/workload in a realistic deep-strike mission.

Analysis has provided additional sources of evaluation information. A risk analysis [40] indicated that generic risks of automation are likely to be mitigated by the Assisted PACT levels, as follows:

- PACT 5 Automatic: Automation bias, poor mode awareness and monitoring, surprise, unexpected action, ROE change, out-of-the-loop performance, unpredictability.
- PACT 0 Commanded: Cognitive bias, complexity, pre-occupation, fixation, time pressure, failure to evaluate options, forgetting rules and procedures, breakdown of skill.

The PACT system is designed to support the pilot's cognitive work. The support ranges from providing advice to providing action. The resultant cognitive work can be represented in terms of a Skills, Rules, Knowledge (SRK) perception-assessment-targeting-execution cognitive decision ladder using state flow transition diagrams. Control task analysis [41,42] has been used to identify the structure of the cognitive work performed by the pilot and by automation at each PACT level [43]. Figure 7-3 illustrates the control task analysis for PACT Level 3 Assisted-In Support, represented in decision ladder flow terms.

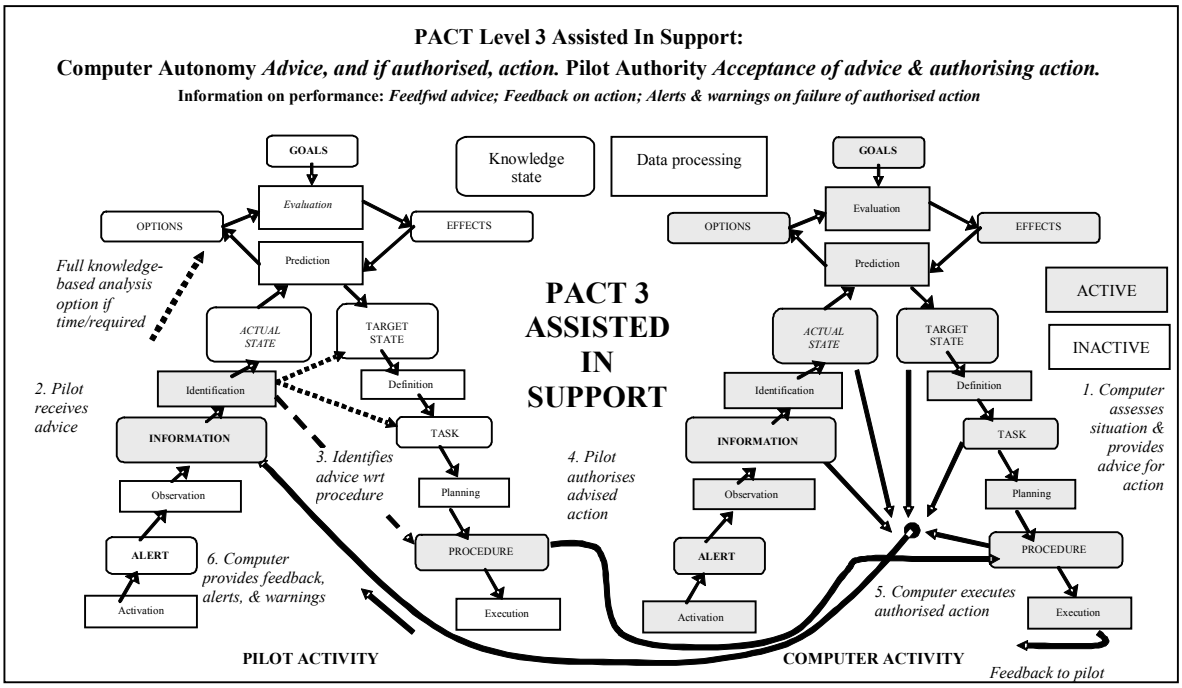

Figure 7-3: Control Task Analysis for PACT Level 3 Assisted In Support.

Figure 7-4 summaries the levels of cognitive work estimated in four decision ladder phases (Perception, Assessment, Decision and Action (PADA). Workload estimates were provided for PACT levels with immediate acceptance, critical acceptance and independent analysis. The analysis indicated that immediate acceptance of advice, associated with high levels of automation trust, was more likely to occur for Perception,

Assessment and Decision phases (i.e., situation assessment, status, goals, options, effects and plans) – but immediate authorisation of action was unlikely to occur, without critical appraisal, indicating a basic lack of trust. This may limit the reduction in cognitive load arising from automation of advised action (Direct Support). Concern about the validity of automated action is understandable during early familiarisation and confidence building. Critical appraisal of recommended courses of action probably will continue until the trustworthiness of the system can be established.

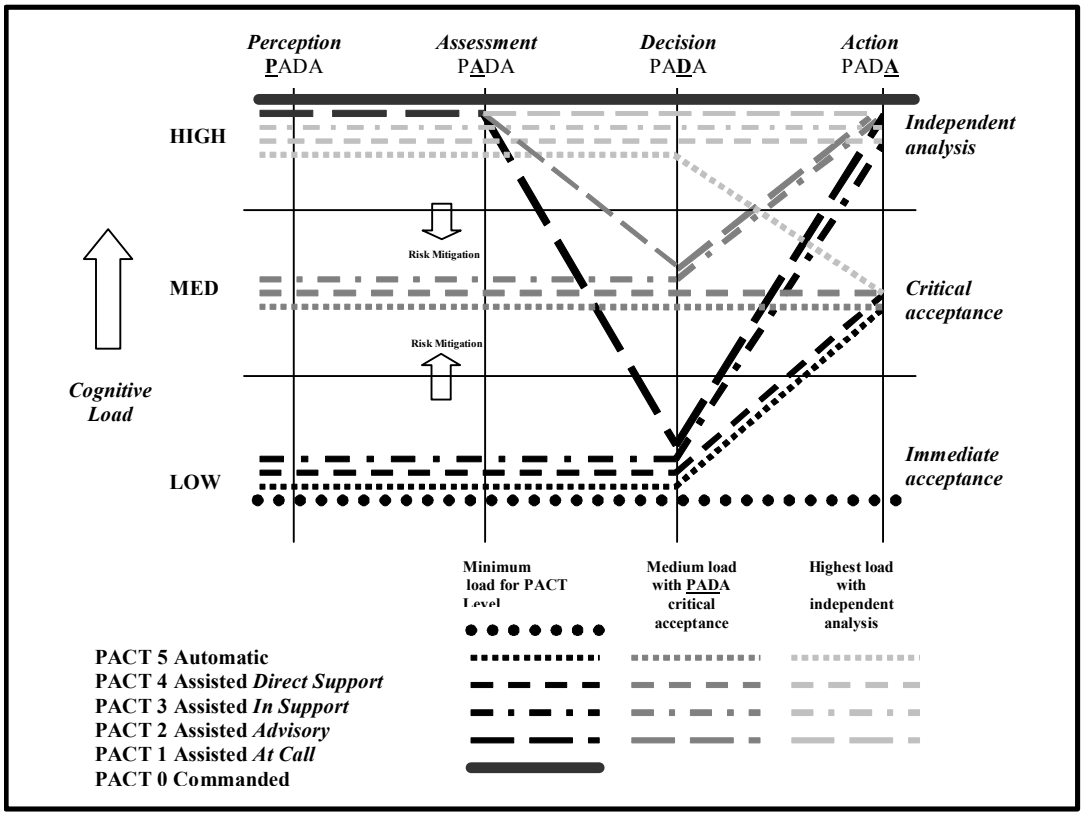

Figure 7-4: Cognitive Load Estimates for PACT Levels.

Some support for this observation on the untrustworthiness of action automation has been reported in an investigation of multiple UAV control. Ruff et al, [44] describe a study to assess the effects of automation reliability and levels of automation (LOA) on supervisory control of multiple UAVs. The LOA used were Management-by-Consent (MBC) and Management-by-Exception (MBE). These LOA equate to PACT Levels 3 and 4 respectively. Under MBC, the operator had to explicitly agree to suggested actions before they occurred. Under MBE, the system automatically implemented suggested actions after a pre-set time unless the operator objected. Results of two experiments showed that participants reported higher workload and difficulty with MBE. Under MBE conditions, time limits were set on manual intervention before automatic performance of the tasks (re-plans, image prosecution). This manipulation of reliability meant that erroneous action could occur. Participants generally chose to complete the tasks manually before MBE automatic action, indicating a lack confidence or trust in the system reliability.

7.3.3 Supervisory Control with Adjustable Autonomy

It seems likely that future manned and uninhabited platforms are both likely to have on-board cognitive automation operating with relatively high levels of autonomy or decision authority. Context sensitive technologies or "intelligent" computer software agents (e.g., Bayesian nets) offer the possibility of being able to control, regulate, direct and adapt system behaviour, within constraint boundaries, even in uncertain, novel, and unpredictable situations. The aspiration is to achieve the requisite cognitive agility, precision, reliability and safety of operations with intelligent systems, with the minimum human supervision and human-computer communication.

The PACT system has been applied in research on the management of multiple UAVs from manned cockpits, to help reduce pilot cognitive workload. It is seen as equally applicable to control of multiple UAVs from ground stations. Furthermore, PACT seems particularly relevant when coupled with intelligent organisation principles, control architectures and tools for structuring and delegating tasking co-ordination and execution workload [29]. DARPA sponsored work on air vehicles (AV) indicates value in similar autonomy solutions. The DARPA UCAV Advanced Cognition Aids Integration project for target engagement and multiple AV identifies four levels of autonomy, namely automate, exception (informs immediate action, OK or revoke), consent (authorisation required), manual [45]. The DARPA ICAV Intelligent Control of Unmanned AV project on mixed initiative distributed intelligence architecture for UAV operations identifies four levels of authorisation, namely autonomous, veto (proposal implicitly accepted after time out), permissive (proposal implicitly rejected after time out), manual [46]. For future envisioned UCAV operations, involving real-time, multiple (group) collaborating autonomous vehicles in joint operations with manned platforms, it seems likely that autonomous control levels will need extending beyond human command and computer support, to cover classes of autonomous complimentary, co-ordinated and co-operative planning and interactions.

Autonomy issues and implementation solutions have been addressed in work on multi-agent intelligent systems for problem solving in complex dynamic environments [34]. Mixed-initiative systems, dynamic adaptive autonomy and adjustable autonomy have been proposed to enable multi-agent systems to perform effectively with adaptability and flexibility. In the context of single-agent to human-user interaction, autonomy has generally been viewed as freedom from human influence – but for multi-agent systems, where the human user may be remote from operations, autonomy becomes a matter of the agent's self-direction and goals, and the capability to dynamically form, modify or dissolve the agent organisation into goal-oriented, problem-solving groups. The degree of autonomy is considered to be implicit or explicitly linked to individual goals, and focuses on the decision making process used to determine how a goal is pursued free from intervention, oversight, or control by another agent (technical or human). Autonomy with respect to goals can be considered to be on a variable scale:

- Consensus or distributed control through consensus (working as a team member, sharing decision-making control equally with all other decision-making agents, all with equal authority);
- Master control (makes decisions alone, may communicate or give orders to other agents with authority);
- Locally autonomous (makes decisions alone, only agent with authority); and
- Command-driven or centralised control (makes no decisions about how to pursue goals, has authority, but must obey orders given by another agent).

Taylor [32,33] adds these agent autonomy levels to the PACT levels with a summary of the responsibilities in cognitive control model terms of advising and performing targeting (or governing), monitoring (or directing),

regulating and controlling (or operating) (Table 7-3). This enables consideration of the flow and transitioning of control in functional context rather than in terms of internal decision-making processes. Further exploitation of the PACT framework can be suggested as follows:

- Assign functions to multi-agent resources in CCII. Use PACT levels to define operational relationships.
- Assign a broad range of inactive reserve functions and operational relationships to PACT Level 1 Assisted At Call, i.e., pre-set at PACT Levels 2, 3, 4.
- Use PACT to define multi-agent support inter-relationships at the Master Control autonomy level.
- Use PACT agents to organise and filter prioritised information in Command and Control Information Infrastructure (CCII) for command intent and SA.

Table 7-3: Adjustable Autonomy Levels for Intelligent Multi-Agent Systems

AUTONOMY	TARGETING (GOVERNING)	MONITORING (DIRECTING)	REGULATING	CONTROLLING (OPERATING)
Consensus Autonomy	Multiple intelligent computer agents	Multiple intelligent computer agents	Multiple intelligent computer agents	Multiple intelligent computer agents
Master Autonomy	Intelligent computer agent	Intelligent computer agent	Intelligent computer agent + Authorised support agents	Intelligent computer agent + Authorised support agents
Local Autonomy	Intelligent computer agent	Intelligent computer agent	Intelligent computer agent	Intelligent computer agent
Automatic/ Commanded Autonomy	Operator	Computer agent performing some interpretation & planning + Operator interrupt	Computer agent performing recognition & scheduling + Operator interrupt	Computer/intelligent agent performing detection & execution agent + Operator interrupt
Assisted Direct Support	Operator	Operator authorising + Computer agent performing some interpretation & planning	Operator authorising + Computer agent performing recognition, & scheduling	Operator authorising + Computer agent performing detection & execution
Assisted In Support	Operator	Operator performing + Optional computer agent advising & performing some interpretation & planning	Operator performing + Optional computer agent advising & performing recognition & scheduling	Operator performing + Optional computer agent advising & performing detection & execution
Assisted Advisory	Operator	Operator performing + Computer agent advising interpretation & planning	Operator performing + Computer agent advising recognition & scheduling	Operator performing + Computer agent advising detection & execution
Assisted At Call	Operator	Operator + Optional computer agent	Operator + Optional computer agent	Operator + Optional computer agent
Command	Operator	Operator	Operator	Operator

Adjustable autonomy gives the agent architecture the ability to adapt their problem-solving to situations particularly in domains with unreliable communications and the possibility of agent failure, high degrees of uncertainty and resource contention needing distribution of tasks and co-ordinated planning to resolve conflicts. Distributed problem solving structures are generally thought to perform faster for complex tasks, when operating under uncertainty and changes in the environment, when few resources are shared, and when communication is unreliable. Centralised structures perform faster for simple tasks, when many resources are shared, when communication is reliable, and when there is no requirement to negotiate. Autonomy level agreements and communication protocols, joint intentions, and employing conventions for explicit commitment to specific interaction styles are considered necessary to establish reliability and trust. A central problem in adjustable autonomy is the determination of whether and when transfers of control to the operator/ user should occur [47]. The transfer of control from agent to human is believed to require a balancing of the costs of interrupting a human user with the benefits for highest quality decision making when the human has

superior decision-making expertise. One technique proposes that transfer should occur when the expected utility of transfer is greater than that of retaining the decision-making. Another forces the agent to relinquish and transfer control if the uncertainty is high. Others transfer if any incorrectness in the agents decision can cause significant harm, if the agent lacks decision-making capability, or on the basis of thresholds of learnt rules. In multi-agent applications, cognitive strategies are needed for reasoning with adjustable autonomy in the operating context (situated autonomy) to provide the correct co-ordination, reordering and scheduling and to balance the costs, benefits, uncertainty and implications within the multi-agent group [48].

There is considerable potential for read-across for control architectures from cognition and joint cognitive systems for the control of distributed multi-agent systems. They use decision resources efficiency and enable the decision agility and adaptiveness needed for the manouevrist approach to military problem-solving. The use of cognitive control models will increase the transparency of control architectures and control authority for human user appreciation of the planning and interaction situation during collaborative problem-solving.

7.3.4 Conclusions

The PACT framework developed for pilot authorisation of control of tasks provides a simplified, practical set of adjustable levels of contractual autonomy capable of engendering trust in automation. An illustration of the idea of PACT as an enabling framework between and automation trust, reliability and use is shown in Figure 7-5. PACT enables the pilot to delegate responsibility for tasks to the computer through a set of contracts that limit autonomy and bound the behaviour of the aiding system, while maintaining the pilot's authority through executive control. Control task analysis, cognitive loading and risk analysis provide useful tools for understanding and modelling the functioning of the PACT system. The PACT framework seems sufficiently robust and useful to be applicable to other systems and environments requiring cognitive control with trustworthy variable levels of autonomy, such as the control of multiple uninhabited vehicles.

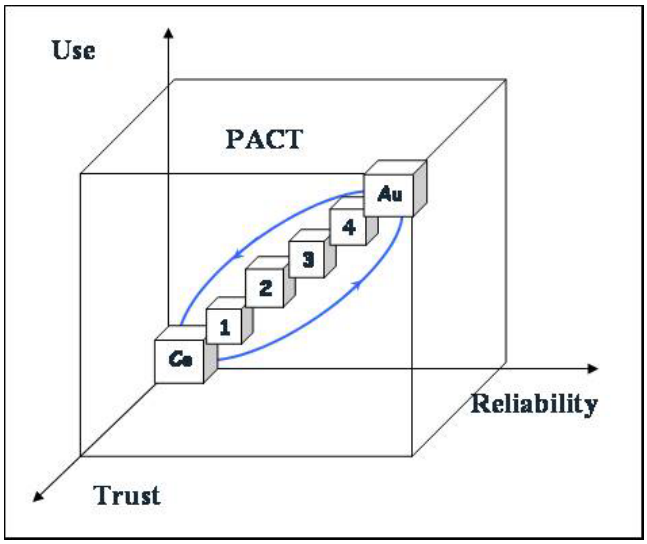

Figure 7-5: PACT Enabling Automation Reliability, Trust and Use.

A number of fundamental questions and key issues can be identified concerning the role of humans in advanced automated and intelligent systems. In particular, there is uncertainty over how to optimise the use of

human and computer decision resources, while preserving a human-centric system. These matters need to be understood in the context of the changing capability requirement responding to new military problem-solving challenges. Important changes are being made in the way in which military force is to be used in the future with the introduction of effects-based approach to the planning and conduct of joint operations. This will be enabled by network CCII, and will provide shared planning and situation appreciation, command intent and Combat Identification. The prime reason for human involvement in military decision-making with automated systems – human control of use of military force for safety assurance – seems established in military law and it is axiomatic for military relevance. Human knowledge, experience and judgement provide unique capability to analyse safety risks and to think ahead in uncertain and novel situations. The challenge is to provide information and decision systems that protect and preserve the human user's key role, and that augment and enhance the user's cognition rather than replaces the user in complex decision making. Recent developments in theory of cognition provide pragmatic approaches that are likely to improve understanding of the human factors issues, problems and solutions of human-computer collaboration. In addition, new approaches to the use of automation propose adjustable levels of computer autonomy with a strong socio-technical and cognition basis. These seem likely to provide sensible architectures for distributed, multi-agent intelligent systems that can be more readily appreciated by human users than traditional automation approaches.

7.3.5 References

[1] Fitts, P.M. (1951). Human Engineering for an Effective Air Navigation and Traffic Control System. Washington, DC. National Research Council.

[2] Beevis, D., Essens, P. and Schuffel, H. (1996). (Eds) Improving Function Allocation for Integrated Systems Design. CSERIAC SOAR 96-01. Crew Systems Ergonomics and Analysis Centre, Wright-Patterson Air Force Base, Ohio.

[3] Hollnagel, E. and Woods, D.D. (1983). Cognitive Systems Engineering: New Wine in Old Bottles. International Journal of Man-Machine Studies, Vol. 18, pp. 583-600.

[4] Hollnagel, E. (1997). Control Versus Dependence: Striking the Balance in Function Allocation. In: M.J. Smith, G. Salvendy, and R.J. Koubek (Eds), Design of Computing Systems. Advances in Human Factors/Ergonomics, 21B, pp. 243-246. Amsterdam: Elselvier.

[5] Taylor, R.M. and Reising, J.M. (1998). The Human-Electronic Crew: Human Computer Collaborative Teamworking. In: Collaborative Crew Performance in Complex Operational Systems. Papers presented at the RTO Human Factors and Medicine Panel (HFM) Symposium held in Edinburgh, United Kingdom, 20-22 April 1998. NATO RTO-MP-4, AC/323(HFM)TP/2, ISBN 92-837-1008-8. Paper 22, pp. 1-17. NATO Research and Technology Organisation, Neuilly-sur-Seine Cedex, December 1998.

[6] Parasuraman, R. and Byrne, E. (2003). Automation and Human Performance in Aviation. In: P.S. Tsang and M.A. Vidulich (Eds). Principles and Practice of Aviation Psychology, Chapter 9, pp. 311-356. Mawhah, NJ: Erlbaum.

[7] Taylor, R.M. (1988). Trust and awareness in human electronic crew teamwork. In: "The Human-Electronic Crew: Can They Work Together? WRDC-TR-89-7008, Wright-Patterson AFB, OH.

[8] Taylor, R.M. and Selcon, S.J. (1993). Operator and automation capability analysis: Picking the right team. AGARD-CP-520, Paper 20, NATO Research and Technology Organisation, Neuilly-sur-Seine Cedex.

[9] Riley, V. (1992). Modelling the dynamics of pilot interaction with an electronic crew. In: The Human Electronic Crew: Is the Team Maturing. WL-TR-92-3078. Wright Patterson AFB, OH. July, pp. 103-107.

[10] Riley, V. (1994). A theory of operator reliance on automation. In: M. Mouloua and R Parasuraman (Eds). Human Performance in Automated Systems: Current Research and Trends. Hillsdale, New Jersey: Lawrence Erlbaum Associates.

[11] Lee, J.D. and Moray, N. (1992). Trust, control strategies and allocation of function in human machine systems. Ergonomics, Vol. 35, pp. 1243-1270.

[12] May, P., Molloy, R. and Parasuraman, R. (1993). Effects of automation reliability and failure rate on monitoring performance in a multi-task environment. Proceedings of the 37th Annual Meeting of the Human Factors and Ergonomics Society, Seattle, WA.

[13] Gao, J. and Lee, J.D. (2004). Information sharing, trust and reliance – A dynamic model of multi-operator multi-automation interaction. In: D.A. Vincente, M. Moustapha and P.A. Hancock (Eds). HPSAA II, Vol. II, Human Performance, Situation Awareness and Automation: Current Research Trends, pp. 34-39, Mawhah, NJ: Erlbaum.

[14] Itoh, M., Inahashi, H. and Tanaka, K. (2004). Overtrust due to unintended use of automation. In: D.A. Vincente, M. Moustapha and P.A. Hancock (Eds). HPSAA II, Vol. II, Human Performance, Situation Awareness and Automation: Current Research Trends, pp. 11-16, Mawhah, NJ: Erlbaum.

[15] Muir, B.M. (1987). Trust between humans and machines and the design of decision aids, International Journal of Man-Machine Studies, Vol. 7, pp. 527-539.

[16] Lerch, F. and Prietula, M. (1989). How do we trust machine advice? In: G. Salvendy and M. Smith (Eds). Designing and Using Human Computer Interfaces and Knowledge-based Systems. Amsterdam, Elsevier.

[17] Parasuraman, R., Molloy, R. and Singh, I. (1993a). Performance consequences of automation induced complacency. International Journal of Aviation Psychology, Vol. 3, (1), 1-23.

[18] Parasuraman, R., Mouloua, M., Molloy, R. and Hillburn, B. (1993b). Adaptive function allocation reduces the cost of static automation, Proceedings of 7th International Symposium on Aviation Psychology, Ohio State University, Vol. 1, pp. 178-181.

[19] Rouse, W.B. (1994). Twenty years of adaptive aiding: Origins of the concept and lessons learnt. In: M. Mouloua and R Parasuraman (Eds.). Human Performance in Automated Systems: Current Research and Trends, pp. 28-33. Hillsdale, New Jersey: Lawrence Erlbaum Associates.

[20] Carmody, M.A. and Gluckman, J.P. (1993). Task specific effects of automation and automation failure on performance, workload and situation awareness. Proceedings of 7th International Symposium on Aviation Psychology, Ohio State University, Vol. 1, pp. 167-171.

[21] Taylor, R.M. and Shadrake, R. and Haugh, J. (1995). Trust and adaptation failure: An experimental study of uncooperation awareness. In: The Human Electronic Crew: Can We Trust the Team? Proceedings of the 3rd International Workshop on Human-Computer Teamwork. WL-TR-96-3039. Wright-Patterson AFB, OH, December 1995. DRA CHS Report DRA/CHS/HS3/TR95001/02, January 1995, pp. 93-98.

[22] Taylor, R.M., Shadrake, R., Haugh, J. and Bunting, A. (1996). Situational awareness, trust and compatibility: Using cognitive mapping techniques to investigate the relationships between important cognitive system variables. In: Situation Awareness: Limitations and Enhancement in the Aviation Environment, Proceedings of AGARD AMP Symposium, Brussels, Belgium, 24-27 April 1995. AGARD-CP-575, pp. 6-1 to 14. NATO, Neuilly-sur-Seine Cedex.

[23] Reising, J.M. (1995). Must the Human-Electronic Crew Pass the Turing Test? In: The Human Electronic Crew: Can We Trust the Team? WL-TR-96-3039, pp.103-108. Wright Patterson AFB, OH, December 1995. Proceedings of the 3rd International Workshop on Human-Computer Teamwork. DRA CHS Report DRA/CHS/HS3/TR95001/02, January 1995.

[24] Sheridan, T.B. and Verplank, W.L. (1978). Human and computer control of undersea teleoperators. Technical Report. MIT Man-machine Systems Laboratory, Cambridge, MA.

[25] 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. Part A: Systems and Humans, Vol. 30, No. 3, pp. 286-297, May 2000.

[26] Krobusek, R.D., Boys, R.M., and Palko, K.D. (1989). Levels of autonomy in a tactical electronic crewmember. WRDC-TR-89-7008, Wright Patterson AFB, OH.

[27] Taylor, R.M. (2001a). Cognitive Cockpit Systems Engineering: Pilot Authorisation and Control of Tasks. In: R. Onken (Ed), CSAPC'01. Proceedings of the 8th Conferences on Cognitive Sciences Approaches to process Control, Neubiberg, Germany, September 2001. University of the German Armed Forces, Neubiberg, Germany.

[28] Taylor, R.M., Abdi, S., Dru-Drury, R. and Bonner, M.C. (2001). 'Cognitive cockpit systems: information requirements analysis for pilot control of cockpit automation'. In: D. Harris (Ed), Engineering psychology and cognitive ergonomics, Vol. 5, Aerospace and transportation systems, Ch. 10, pp. 81-88. Aldershot: Ashgate.

[29] White, A.D. (2002). The human-machine partnership in UCAV operations. In: Proceedings of 17th Bristol UAV Systems Conference, 10-13 April 2002.

[30] Taylor, R.M., Brown, L. and Dickson, B. (2002). From safety net to augmented cognition: Using flexible autonomy levels for on-line cognitive assistance and automation. RTO-MP-086 AC/323(HFM-085)TP/42. ISBN 92-837-0028-7. Paper No 27, NATO RTO Human Factors and Medicine Panel, Symposium on Spatial Disorientation in Military Vehicles: Causes, Consequences and Cures, La Coruna, Spain, 15-17 April 2002.

[31] Waters, M. and Taylor, R.M. (2004). A Bayesian Agent Approach to Autonomous Decision Making for an Unattended Cognitive Underwater Vehicle (UCUV). In: Proceedings of the Workshop on Uninhabited Military Vehicles (UMVs) – Human Factors of Augmenting the Force, RWS-010-P4, held in Leiden, The Netherlands, 10-13 June 2003. NATO RTO Human Factors and Medical Panel 078/Task Group 017, NATO RTO Neuilly-sur-Seine.

[32] Taylor, R.M. (2002). Capability, Cognition and Autonomy. In: Proceedings of the NATO RTO Human Factors and Medicine Panel Symposium, HFM-084/SY-009, "The Role of Humans in Intelligent and Automated Systems", Warsaw, Poland, 7-9 October 2002. NATO, RTO, Neuilly-sur-Seine Cedex.

[33] Taylor, R.M. (2003). Cognition and Autonomy in Distributed Intelligent Systems. In: D. Harris, V. Duffy, M. Smith and C. Stephanidis (Eds). Human Centred Computing: Cognitive, Social and Ergonomic Aspects. Vol. 3, pp. 330-334, Lawrence Erlbaum Associates, Mahwah, New Jersey.

[34] Barber, K., Goel, A. and Martin, C. (2000). Dynamic Adaptive Autonomy for Multi-agent Systems. Journal of Experimental and Theoretical Artificial Intelligence. Vol. 12, Part 2, pp. 129-148.

[35] Hollnagel, E. (2002). Cognition as Control: A Pragmatic Approach to the Modelling of Joint Cognitive Systems. Special Issue of IEEE Transactions on Systems, Man and Cybernetics A: Systems and Humans – "Model-Based Cognitive Engineering in Complex Systems" (In Press) http://www.ida.liu.se/~eriho/

[36] Diethe, T.R., Dickson, B.T., Schmorrow, D. and Raley, C. (2004). Toward an augmented cockpit. In: D.A. Vincente, M. Moustapha and P.A. Hancock (Eds). HPSAA II, Vol. II, Human Performance, Situation Awareness and Automation: Current Research Trends, pp. 65-69, Mawhah, NJ: Erlbaum.

[37] Bonner, M., Taylor, R.M. and Miller, C. (2000). Tasking interface manager: Affording pilot control of adaptive automation and aiding. In: P.T. McCabe, M.A. Hanson and S.A. Robertson (Eds.), Contemporary Ergonomics 2000, pp. 70-74, London: Taylor and Francis.

[38] Taylor, R.M. (2001a). Cognitive Cockpit Systems Engineering: Pilot Authorisation and Control of Tasks. In: R. Onken (Ed), CSAPC'01. Proceedings of the 8th Conferences on Cognitive Sciences Approaches to process Control, Neubiberg, Germany, September 2001. University of the German Armed Forces, Neubiberg, Germany.

[39] Miller, C.A., Guerlain, S. and Hannen, M. (1999). The Rotorcraft Pilot's Associate Cockpit Information Manager: Acceptable behaviour from a new crew member. Proceedings of the American Helicopter Society, 55th Annual Forum, Montreal, Quebec, May 25-27, 1999.

[40] Taylor, R.M. (2001b). Cognitive Cockpit Control Task Analysis. DERA Memo, DERA/CHS3/6.3/14/7, 7 March 2001.

[41] Sanderson, P., Naikar, N., Lintern, G. and Goss, S. (1999). Use of cognitive work analysis across the system life cycle: From requirements to decommissioning. Proceedings of the Human Factors Society 43rd Annual Meeting, Houston, TX, pp. 318-322, Santa Monica, HFES.

[42] Vincente, K.J. (1999). Cognitive Work Analysis: Towards Safe, Productive and Healthy Computer-based Work. New Jersey: Lawrence Erlbaum.

[43] Taylor, R.M. (2001c). Cognitive Cockpit Risk Analysis. DERA Memo, DERA/CHS3/6.3/14/7, 26 February 2001.

[44] Ruff, H.A., Calhoun, G.L., Draper, M.H., Fontejon, J.V. and Guilfoos, B.J. (2004). Exploring automation issues in supervisory control of multiple UAVs. In: D.A. Vincente, M. Moustapha and P.A. Hancock (Eds). HPSAA II, Vol. II, Human Performance, Situation Awareness and Automation: Current Research Trends, pp. 218-222, Mawhah, NJ: Erlbaum.

[45] Leahy, M. (2001). DARPA UCAV Advanced Cognition Aids Integration. Paper presented to Unmanned Systems 2001. The 28th Annual Technical Symposium of the Association of Unmanned Vehicles International, Baltimore, July 2001.

[46] Elmore, W. (2001). DARPA ICAV Intelligent Control of Unmanned Air Vehicles, Paper presented to Unmanned Systems 2001. The 28th Annual Technical Symposium of the Association of Unmanned Vehicles International, Baltimore, July 2001.

[47] Scerri, P., Pynadath, D.V. and Tambe, M. (2001). Adjustable Autonomy in Real World Multi-Agent Environments. In: Agents '01, Proceedings of the 5th International Conference on Autonomous Agents.

[48] Hexmoor, H. (2000). A Cognitive Model of Situated Autonomy. In: Proceedings of PRICAI-2000. Workshop on Teams with Adjustable Autonomy, pp. 11-20.

7.4 ADAPTIVE AUTOMATION FOR ROBOTIC MILITARY SYSTEMS

7.4.1 Introduction

Future Combat Systems (FCS) is a US Army program that will transform the battlefield Future force structure; doctrine and tactics will change as new systems are introduced; possibly in ways that can not be anticipated. Units of Action (UA) are being designed to be flexible, reconfigurable components of FCS tailored to specific combat missions. One aspect of increased flexibility will be the introduction of numerous robotic systems. The term robot is used in a generic sense to describe systems that are unmanned with some degree of autonomy that include aerial, ground, subterranean, naval surface and sub-surface vehicles. These systems will be an essential part of the future force because they extend manned capabilities, are force multipliers and most important, they can save lives.

Any major change in current doctrine implies problems as well as solutions. Robotic systems with diverse roles, tasks and operating requirements are being designed to exploit future battle spaces. The role of the human operator is not well understood; however most of the contemplated systems will require either active human control or supervision with the possibility of intervention. In the most extreme case, soldiers will operate multiple systems while on the move and while under enemy fire. In all cases, the workload and stress will be variable and unpredictable – changing rapidly as a function of the military environment. The purpose of this chapter is to investigate technologies that unload the warfighter interacting with unmanned systems during multi-tasking missions. First, we will investigate automation technologies, specifically their positive and negative effects on human performance and situation awareness. Next, we will discuss adaptive and adaptable processes as methods that potentially overcome the disadvantages of preset automation. The last section will survey diverse physiological measures that can be used to trigger adaptive processes emphasizing the rapid development of these methods and their current limitations.

Future robotic systems are being designed to be used in all facets of the modern battlespace and be, to the degree possible, autonomous. This requires both rapid response capabilities and intelligence built into the system. However, ultimate responsibility for system outcomes always resides with the human and in practice; even highly automated systems usually have some degree of human supervisory control [1]. Particularly in combat, some oversight and the capability to override and control lethal systems will always be a human responsibility for the following reasons [2]:

- System safety;
- Change in the commander's goals;
- Implied meta-goals; and
- Fratricide.

However, automation is not an all or nothing phenomenon. Automation can vary in the degree to which a particular function that was previously carried out by a human operator is allocated to a machine agent. This is the concept of "level of automation" (LOA) as discussed by Sheridan [3]. However, automation can vary in other dimensions as well, for example in the stage of human information processing that the automation is applied, whether to stages such as information acquisition and analysis, or to stages such as decision making and response execution. Parasuraman, Sheridan and Wickens [4] developed a taxonomy of human automation control that is two dimensional. Figure 7-6 shows rows as degree of automation and columns as type of processing function addressed. The four processing functions of information acquisition, information analysis, decision making/action selection, and action implementation are similar to the Observe-Orient-Decide-Act or OODA loop in the parlance of military command and control. The taxonomy also captures the multiplicity of control options from fully automated to fully manual for each of these functions. The decision space is not only complex, but it implies that there is not a single solution to partitioning control. Specifically, as the type of task the operator performs changes the control logic may need to change as well. This is understating the problem because the taxonomy does not consider either other tasks the operator is performing or the overall workload and stress imposed by the current environment. A review of the human performance literature reinforces the notion that there is not a single solution to partitioning; human performance varies greatly depending on the operator task and the current environment [5,4].

Processing task-type of auto	Info acquisition	Info analysis	Action selection	Action implementation
Full auto				
Auto-human informed sometimes				
Auto-human Informed				
Auto-human veto time limit				
Auto executes only if human approves				
Auto suggests				
Auto narrows				
Auto shows all options				
Manual				

Figure 7-6: Human-Automation Taxonomy with Rows Representing Degree of Automation and Columns Processing Functions. (Adapted from [4]).

7.4.2 Human Performance Issues for Automated Systems

Numerous problems related to human performance in automated systems have been identified in the literature. One problem with automated systems is the operator's trust and level of use of the automation. Parasuraman and Riley [5] compiled both research and real world examples of automation misuse, disuse and abuse. They showed that the human operator ignored important indicators, failed to use reliable systems, misused unreliable systems, or misunderstood the true state of the system. The paper catalogued various human deficiencies related to supervisory control and, most important; the review motivated a good deal of subsequent research.

In another review article, Mosier and Skitka [6] examined what they termed cases of automation bias. The operator tended to over rely on automated systems even in cases where appropriate operator intervention would have averted performance problems. They did not identify an automation bias per se, but rather identified a number of performance problems related to lowered vigilance, high workload, time stress and loss of situation awareness (SA). SA problems resulted from a combination of automation complexity and poor display design. The best example was the Three Mile Island accident wherein operators were misled by too many malfunction indicators that were unrelated to the underlying problem. Again, there did not seem to be a coherent theory explaining automation bias rather the authors identified multiple causes of over reliance. Other researchers found that while there were circumstances where humans over relied on automation, there were other equally important instances where they should have relied on automation and did not.

For example, an Army supported study investigated operator trust issues related to the Battlefield Combat Identification System (BCIS) [7]. In an experiment with college students, the simulated aid's target identification rate was varied from 60 – 90% accuracy and the participants' task was to affirm or override the target decisions. Overall the subjects were twice as likely to be wrong when they agreed with an erroneous aid decision (p (error/aid error) = .27) then when they made override errors for cases when the aid was correct (p (error/aid correct) = .13) supporting the automation bias hypothesis.

However, other research from the same authors indicated exactly the opposite bias- disuse of appropriate automation. In a study similar to the BCIS study, college students were given 200 trials in which they had to decide whether a target was present on the display or not [8]. After each trial, target advisories were given to the students purported to be from either an aid (automatic target recognition (ATR) device) or a "peer". Participants were told the relative accuracy rates for the aids ("peers") and their own decisions and then had to decide whether to base future decisions on their own performance or that of the automated advisory (or peer). Surprisingly, even when the aid made ½ as many errors and the subjects reward depended on accuracy, 80% of the subjects chose to make their own decisions. Also, subjects trusted "peer" advisories more than the aid advisory with the same accuracy level. They rationalized their decisions in terms of self reliance. However, the most salient difference between this study and the BCIS study was that subjects were told the ATR- "peer" decision **after** they had made their own decision. Thus there was no workload advantage to using the aid advisory because the operator's decision was made before the aids results were known.

The results are important because a simple manipulation (the order in which decisions were required) caused an automation bias to shift to a self–reliance bias. The same results were repeated in subsequent experiments, but the self-reliance bias was mitigated if the test subject was informed the reason for the ATR errors and were given appropriate feedback during the initial trials [9]. Anecdotal data also indicated mistrust of aids in cases where there was a high false alarm rates: the "cry wolf" phenomenon [10]. In summary, humans are neither universally over reliant or under reliant on automated systems. The crucial factors seem to be workload, time stress, false alarm rate and decision order.

Automation reliability has the same contradictory effects on performance depending on the task, workload and type of errors the automated device makes. A number of experimenters found no effect of aid reliability on performance [7,11,12]. The most likely reason is the lack of calibration of the subjects. For example, in the Dzindolet et al. study each subject received only one reliability level (60%, 75% and 90%) and they were not given the feedback that may have allowed them to respond effectively. The literature suggests that humans have problems understanding probabilities and may need some form of intervention in order to perform efficiently [13,14]. Reliability level effects depend both on operator strategies and the type of error the aid manifests. Meyer [15] has shown that when automation reliability is such that malfunctions are almost always correctly indicated – that is, the automation makes few misses, then the operator has high *reliance* on the

automation. This is an effective strategy, but can result in a problem when the automation *does* miss, because of the complacency effect. On the other hand, if automation reliability is such that few false alarms are made, then the operator has high *compliance*: if an automated alarm sounds, then the operator tends to immediately comply with the alarm and tend to the situation. Reliance on automated aids permits the operator to attend to tasks other than the automated task until the alert is triggered thus improving multitask performance and not just the performance on the automated task.

Research using realistic unmanned aerial vehicle (UAV) operator tasks indicted that reliant behaviors are affected principally by the misses. In contrast, compliance errors were affected by the false alarm rate, but not affected by miss rate of the automated device. However, increasing the operator's workload and decreasing the aid's reliability level had adverse effects on both compliance and reliance errors [16].

The issue is complicated because performance depends on the type of processing task the operator performs and paradoxically high reliability can result in costs as well as benefits. Rovira, McGarry and Parasuraman [17] investigated automation of artillery targeting decisions. For their particular task, they found that reliable automation improved the surrogate commander's decision latency without sacrificing accuracy. However, particularly for decision tasks related to choosing a course of action, higher reliability hurt the operator's performance when the aids gave incorrect information. The surrogate commanders trusted 80% accurate aids more than the 60% ones in cases where they should have been more skeptical (i.e., when the aids gave them incorrect information). Apparently, the advisories from 'trusted" aids were not scrutinized as thoroughly as those from less reliable aids.

One interpretation of the previous results is to assume that reliable aids lulled the operator into a false sense of complacency. However, the complacency literature suggests that the results depend on other factors besides trust. A number of researchers had failed to show complacency effects, motivating Parasuraman, Molloy and Singh [11] to investigate possible reason for this in a multitask aviation environment. There most important finding was that complacency did not occur for low workload (single task) conditions. For the high workload task, the operator became complacent (over relied) on the aid when the aid had a constant reliability level; however, when the reliability level varied over a block of trials the performance decrement was ameliorated. This suggests that complacency was not so much a matter of trust, but a strategy to deal with high workload. If an aid acted in a predictable manner (constant reliability) then the operators would commit their resources to other tasks in a high workload environment causing a performance decrement in the unmonitored automation task.

The forcing function in most of these studies was high workload. The operator basically traded off situation awareness for workload reduction by depending on aids even in cases when it was not beneficial to do so. This was not universally true, in some cases automation improved overall performance even when the automated task required intervention because the operator's residual cognitive capacity was allocated effectively among the set of tasks [18,19]. However, too often, the loss of situation awareness related to inefficient automation monitoring leads not only to performance decrements, but also to an increasingly impoverished understanding of the work environment which can over time result in catastrophic errors [20,6,5]. Because of the uncertainty and risk associated with military environments, automating any but the most trivial tasks must be done with extreme caution. The soldier and his command chain need to maintain situation awareness; keeping the soldiers out of the loop will not only have consequences for the immediate task, but also predispose them to miss important cues signaling change [20]. Conversely, requiring soldiers to engage in multiple tasks could very well have the same consequences.

7.4.2.1 Adaptive Principles

A possible solution is to create enough flexibility in the system to ensure more automation during peak workload and greater operator engagement during workload lulls. However, workload itself is a theoretical construct and it is not always obvious how it affects performance. For example, the traditional view of vigilance was that underload was responsible for the observed human performance decrement as a function of time of watch. Recent research indicates that it is overload and not underload that is responsible for the deleterious effects of vigilance [21]. A possible work around would be to leave it up to the operator to decide when to automate and which tasks to automate as mission requirements change. However, this is not always practical because it would burden operators with additional tasks precisely when they are already heavily loaded. For this reason, a number of researcher have suggested using some form of behavioral indictor to change levels of automation dynamically as a function of the changing work environment [22-25].

Adaptive automation uses mitigation criteria that drive an invocation mechanism to maintain an effective mixture of operator engagement and automation for a dynamic multitask environment (Figure 7-7). The invocation mechanism is triggered by whatever measurement process is used to represent the current task state. If properly instrumented the results of the measurement process should be displayed to operators in order to keep them informed of the state of the invocation process.

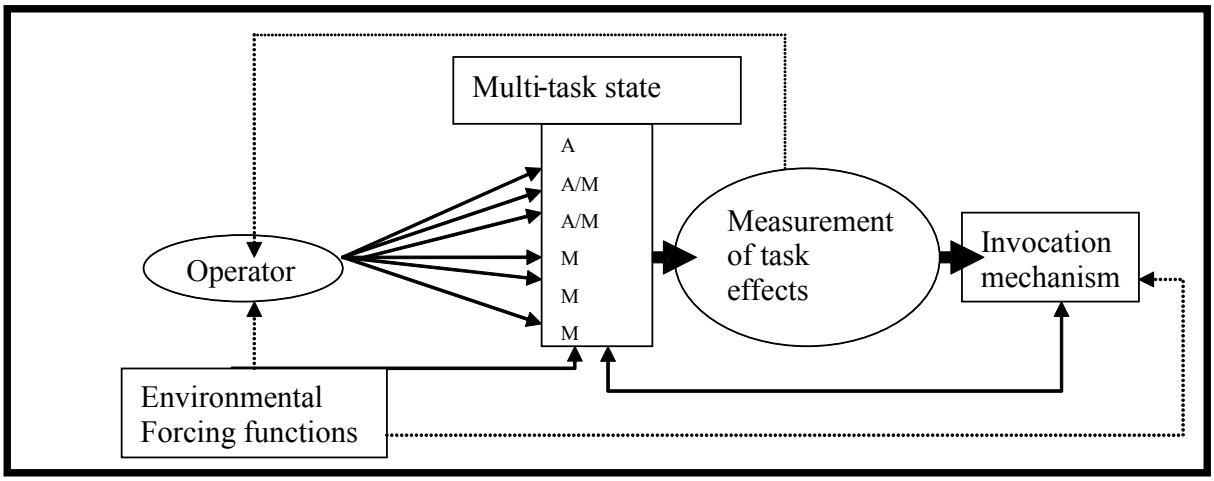

Figure 7-7: Example of a Closed Loop Adaptation for A – Automated, A/M – Automated/Manual, and M – Manual Task Sets.

This construct is more complex than simply unloading (or engaging) the operator because to be effective the invocation process must be sensitive to the operator's combined tasking environment which depends on interactions among tasks as well as overall workload, stress and safety considerations [26]. For example, the algorithm might automate auditory tasks when the communication traffic reaches a predefined level, but not change other task states until the overall workload measure (or physiological index) reaches criterion [27]. Furthermore, whenever certain critical events occur, the invocation mechanism must be sensitive to indices that imply that the operator requires emergency automation (e.g., G-loss of consciousness as evidenced by physiological measurements) [28].

7.4.3 Some Characteristics of Adaptive Automation Systems

7.4.3.1 Invocation Methods

In adaptive systems, the "division of labor" between human and machine agents is not fixed but dynamic, in contrast, to systems where provision of computer aiding is pre-determined at the design stage, and task allocation is fixed during system operations. Although the adaptive automation concept is not new, having been proposed about 25 years ago [29], technologies needed for its effective implementation were not readily available until recently.). A key issue in adaptive automation is the method of invocation. Parasuraman et al. [23] reviewed the major techniques and found that they fell into five main categories:

- Critical events;
- Operator performance measurement;
- Operator physiological assessment;
- Operator modelling; and
- Hybrid methods.

The critical-events method is exemplified by the work of Barnes and Grossman [28]. In this approach, automation is invoked when certain tactical environmental events occur, but not otherwise. For example, in an aircraft air defence system, the beginning of a "pop-up" weapon delivery sequence leads to the automation of all defensive measures of the aircraft. If the critical events do not occur, the automation is not invoked. Hence this method is inherently flexible and adaptive, because it can be tied to current tactics and doctrine during mission planning.

However, a disadvantage of the method is its possible insensitivity to actual system and human operator performance. For example, this method will invoke automation irrespective of whether or not the pilot requires it when the critical event occurs. Operator performance and physiological measurement attempts to overcome this limitation. In these methods, various operator mental states (e.g., mental workload, or more ambitiously, operator intentions) may be inferred on the basis of performance or other measures and then input to adaptive logic. For example, performance and physiological measurements may allow the inference that a human operator is dangerously fatigued or experiencing extremely high workload. An adaptive system could use these measurements to provide computer support or advice to the operator that would mitigate the potential danger. Alternatively, human operator states and performance may be modeled theoretically, with the adaptive algorithm being driven by the model parameters. Intelligent systems that incorporate human intent inferencing models have been proposed [30]. Finally, hybrid methods could be used that combine one or more of these different invocation techniques, so that their relative merits can be maximized.

7.4.3.2 Adaptive and Adaptable Systems

In adaptive systems, the decision to invoke automation or to return an automated task to the human operator is made by the system, using any of the previously described invocation methods. This immediately raises the issue of user acceptance of such a system. Human operators may be unwilling to accede to the "authority" of a computer system that mandates when and what type of automation is or is not to be used. Apart from user acceptance, however, is the issue of system unpredictability and its consequences for operator performance. Billings and Woods [31], for example, raised the caution that truly adaptive systems may be problematic because their behavior may not be predictable to the user. To the extent that automation can hinder the operator's situation awareness by taking him or her out of the loop, unpredictably invoked automation by an adaptive system may further impair the user's SA. However, if the automation were explicitly invoked by the user,

then presumably the unpredictability will be lessened – but involving the human operator in making decisions about when and what to automate can increase workload. Thus, there is a tradeoff between increased unpredictability versus increased workload in systems in which automation is invoked by the system or by the user, respectively. Opperman [32] characterized these alternatives as 'adaptive' and 'adaptable' approaches to system design (see also [33]). In either case, the human + machine system adapt to various contexts, but in adaptive systems automation determines and executes the necessary adaptations, whereas in adaptable systems, the operator is in charge of the desired adaptations. The distinction is primarily one of authority. In an adaptable system, the human always maintains authority to invoke or change the automation, whereas this authority is shared in an adaptive system. Inagaki's [34] design concept of "situation-adaptive autonomy" is related to this view of an adaptive system, but in his approach, control of a process is traded off between human and computer in real time based on time criticality and the expected costs of human and machine performance.

While in this review we primarily consider how adaptive automation affects system performance, it is important to keep in mind that adaptable automation may provide an alternative approach with its own benefits. The LOA concept introduced by Sheridan [3] does not specify which level should be used or who decides that there should be a change in level.

When the decision is made by a designer prior to system operation, it is a part of system design and corresponds to picking an appropriate LOA for that system design. The decision can also be made by automation itself (or some expert system) during system operators, as a part of a truly adaptive automation system. In both of these cases, the human operator is not involved in the decision. In adaptable systems, however, the human operator is more akin to a supervisor of a human team who delegates tasks to team members, or in this case, to automation. The challenge for developing such an adaptable automation system is that the operator should be able to make decisions regarding the use of automation in a way that does that create such high workload that any potential benefits of delegation are lost.

One such architecture for adaptable automation that can provide for flexible tasking of automation is the "Playbook" [35-37]. The objective of the Playbook interface is to provide a human supervisor the ability to delegate tasks to automation with much of the flexibility available in human-human task delegation, and to do so dynamically at the time of system operation rather than at system design. The Playbook interface facilitates the "teaching" of automation (an idea first proposed by Sheridan [3] when he created the supervisory control concept) by creating a shared knowledge structure of tasks and their relationships within which task performance can be discussed by human and automation. For the Playbook concept to work, the automation must have substantial knowledge about how to perform tasks and achieve goals. This knowledge is also used to improve the efficiency and/or safety of plans developed by allowing automation to review and critique human plans. Finally, Playbooks streamline the process of delegation by the human operator by providing a compiled set of plans, or 'plays', with short, easily-commanded labels that can be further modified as needed. This is the critical aspect of the concept that allows this form of adaptable automation not to increase the workload associated with delegation, much as a sports team has an approved set of plays that facilitate task delegation by the team leader. For example, the quarterback in football selects plays that are executed by the team members (the other players).

Support for the efficacy of the Playbook approach to adaptable automation has come from two sources. First, an example Playbook prototype for mission planning tool for commanding Unmanned Combat Air Vehicles (UCAVs) has been developed as a proof-of-concept [37,38]. Second, initial experimental studies of the effects of Playbook interfaces on human performance have been carried out [39,40]. These studies examined the use of a simple Playbook interface on system performance during simulated human-robot teaming using the RoboFlag simulation environment. The RoboFlag Playbook provides the operator the

ability to command simulated robots, individually or in groups, at two levels of granularity: via providing designated endpoints for robot travel or via commanding higher level behaviors (or modes or plays) such as "Patrol Border." The RoboFlag simulation was modified to emulate a typical unmanned vehicle (UV) mission involving a single operator managing a team of robots. The simulated mission goal was to send the robots from a home area into enemy territory, access and obtain a specified target, and return home as quickly as possible with minimum loss of assets. In the Parasuraman et al. [39] study, the Playbook interface was evaluated as a function of two sources of task demand adversary "posture" in which the enemy engagement style was changed unpredictably between three types, offensive, defensive, or mixed; and environmental uncertainty, as manipulated by variation in the effective visual range of each robotic vehicle under the control of the operator. The results showed that the multi-level tasking of the simplified Playbook interface allowed effective user supervision of robots, as evidenced by the number of missions successfully completed (percent games won) and the time for mission execution. As expected, significantly fewer games were won when the opponent posture was mixed rather than entirely offensive. Nevertheless, users still won a moderately high proportion of games (about 62%) and in a relatively short time (about 51 seconds) in the mixed posture condition. These findings suggest, but do not prove, that the Playbook interface, as a simple example of a delegation interface, allowed users to respond effectively to unexpected changes in opponent posture by tasking robots appropriately. In a subsequent study [40], the Playbook interface was pitted against less flexible interfaces and to manual control, and was found to have significant benefits over both. Further confirmation of the efficacy of the Playbook approach to adaptable automation requires studies in which more complex versions of the Playbook interface are evaluated.

7.4.3.3 Human Interaction with Adaptive Systems

Since the theoretical frameworks for adaptive automation proposed by Rouse [41] and Parasuraman et al. [23], there has a steady stream of empirical work aimed at examining the effects of adaptive automation on human and system performance in different application domains. The initial studies were designed to investigate whether the performance costs of certain forms of static automation (described previously), such as reduced situation awareness, complacency, skill degradation, etc., can be mitigated by adaptive automation. Most of these studies used either a critical event or model-based approach to adaptive automation. A task was allocated dynamically to either human or machine control at some point in time during a simulated mission, either when some critical event occurred, or as dictated by a simple model of operator and system performance. For example, Hilburn et al. [42] examined the effects of adaptive automation on the performance of military air traffic controllers who were provided with a decision aid for determining optimal descent trajectories of aircraft – a Descent Advisor (DA). The DA was either present at all times (static automation) or came on only when the traffic density exceeded a threshold. Hilburn et al. found significant benefits for controller workload (as assessed using pupillometric and heart rate variability measures) when the DA was provided adaptively during high traffic loads, compared to when it was available throughout (static automation) or only at low traffic loads. In addition to physiological measures of workload, other measures can also be used to assess the workload-leveling effect of adaptive automation. Kaber and Riley [43], for example, used a secondary-task measurement technique to assess operator workload in a target acquisition task. They found that adaptive computer aiding based on the secondary-task measure enhanced performance on the primary task.

The results of these and other studies (see [44] for a review) indicate that adaptive automation can serve to reduce the problem of unbalanced workload, with attendant high peaks and troughs, that static automation often induces. As discussed previously, under high workload operators tend to adopt an attention allocation strategy that results in diminished monitoring of an automated task [11,45]. As a result, operators can miss malfunctions in the task, or fail to correct suboptimal performance by the automation because they are busy

attending to other tasks. Adaptive automation in the form of a temporary return of the automated task to human control can mitigate this so-called complacency effect. In a study with the Multi Attribute Test (MAT) flight simulation battery, Parasuraman et al. [46], showed that temporary return of an automated engine-systems task to human control benefited subsequent operator monitoring of the task when it was returned to automated control. It is important to emphasize that the reallocation to human control was *brief*. If the benefit could only be obtained by prolonged human intervention in the task, that would defeat the purpose of automating the task in the first place. Parasuraman et al. found that the benefit of adaptive reallocation was found for either of two methods of invocation, a model-based approach in which the temporary return to human control was initiated at a particular time specified by the model; and a performance-measurement approach in which the adaptive change was triggered only when the operator's performance on the engine-systems task fell below a specified level. A subsequent study showed that the operator (and system) performance benefit could also be sustained for long periods of time, in principle indefinitely, by repetitive or multiple adaptive task allocation at periodic intervals [47]. Such brief, periodic, adaptive reallocation of an automated task to human control can enhance overall system performance by either maintaining the operator's awareness of the automated task parameters or by refreshing the operator's memory (his or her "mental model") of the automated task behavior. In support of the latter explanation, Farrell and Lewandowsky [48] showed that they could successfully computationally model the complacency effect and the benefit of adaptive reallocation in a three-layer connectionist network with a memory decay function for nodes representing automation performance.

These results show that adaptive automation can balance operator workload and reduce automation complacency. However, Parasuraman et al. [46] also showed that performance benefits can be eliminated if adaptive automation is implemented in a clumsy manner, supporting the concerns of Billings and Woods [31]. Moreover, these studies, while clearly pointing to the potential benefit of adaptive automation, had some limitations. The model-based invocation method used in many of the studies has the advantage that the model can be implemented off-line and easily incorporated into rule-based expert systems. However, this method requires a valid model, and many models may be required to deal with all aspects of human operator performance in complex task environments.

7.4.4 Physiological Measurement Techniques

There are many ways in which the system changes can be initiated: subjective workload assessments, operator performance and physiological measures. The final section will discuss physiological measurement processes and their relationship to adaptation. Operator physiological assessment offers another potential input for adaptive systems [49,23]. Physiological measures can provide additional information that can be tapped for control of adaptive systems. Technology is available to measure a number of physiological signals from the operator, from autonomic measures such as heart rate variability to central nervous system measures such as the EEG and event-related potentials or ERPs, as well as measures such as eye scanning and fixations. Measurement technology is developing rapidly showing improvements in the areas of non-intrusiveness, precision and prediction. For these reasons, the authors felt it was important to survey these methods in some detail. The goal of using physiological methods for adaptive automation is to enhance operator performance by restructuring the environment and or adjusting the demands of the situations [50]. The emphasis is on the operator's capabilities not the system's [49].

One way in which systems may be able to enhance operator performance is via online measures of workload. According to Scerbo, Freeman, and colleagues [51] mental workload is the critical factor in determining when and what type of system changes need to be made. There are individual differences in how a person handles the demands of a situation and these individual differences may impact on performance. According to Gopher

and Donchin [52] people often increase their mental and physical effort as task load increases. The concept of workload is used to account for the aspect of the interaction between the person and the task that cause task demands to exceed the person's capacity to successfully complete the task. The concept of workload implies that there are limitations in information processing capacity. Two theories form the foundation of much of the research conducted on adaptive automation, Kahneman's Capacity and Resource Theory and Wicken's Multiple Resource Theory [22]. According to Kahneman, information processing resources are limited, but are a function of arousal. As task load increases, arousal increases beyond an optimal level causing cognitive capacity to decrease. In single resource theory, capacity can be allocated to different tasks but the result is a depletion of overall capacity. According to Wicken's multiple resource theory, information processing resources are limited but this limitation is a function of the type of mental resource being used (i.e., visual, auditory). Capacity decreases more when two task are sharing the same resources than when they are using different resources. The models are not mutually exclusive; Wickens, Dixon and Chang [27] recently found that a combination of single resource and multiple resource models best described operator performance for controlling two unmanned vehicles in a multi-task environment.

There is now a substantial literature indicating that different psychophysiological measures can be used for real-time assessment of mental workload [53-56]. Researchers have examined the use of Heart Rate (HR), Heart Rate Variability (HRV), Electroencephalography (EEG), Event Related Potentials (ERP), Transcranial Doppler Sonography (TCD), Functional Magnetic Resonance Imaging (fMRI) and more recently Functional Near Infrared Imaging (fNIR). Prinzel, Freeman, Scerbo, Mikulka, and Pope [57] have also specifically demonstrated the feasibility of an adaptive system based on EEG measures. Each of these measures has advantages and disadvantages and we discuss their potential utility in an adaptive automation system. An overview of the most likely indices, limitations and advantages is presented in Table 7-4. A more detailed review can be found in Scerbo, Freeman, Mikulka, Parasuraman, DiNocera and Prinzel [51].

Table 7-4: Matrix of Physiological Measures – Advantages and Disadvantages

Measures		What does it measure?	Advantages	Disadvantages	How Can The Measure Be Used?
Electroencephalography	EEG	-Electrical activity of neuronal assemblies - Recorded potential reflects cortical activity in area under electrode	-Good temporal resolution -Localization of a behavior to a cortical region -Extensive research base -New systems more field practical	-Affected by artifacts such as muscle activity and heart beats -Poor spatial resolution	-Assess cortical involvement in different contexts and situations (e.g. effect of practice on cerebral activity)
Event Related Potentials (Derived from EEG)	ERP	-Component of EEG related specifically to a stimulus -Discrete time locked responses to a stimulus	-Good temporal resolution -Can examine time-based changes in cortical activity to a stimulus	-Affected by artifacts such as eye blinks and heart activity -Poor spatial resolution	-Assess time locked changes in cortical involvement in different contexts and situations
Transcranial Doppler Sonography	TCD	-Blood flow velocity in the main stem intra-cranial arteries	-Good spatial & temporal resolution -Non-invasive -Less restrictive than other brain imaging techniques (e.g. EEG) -Low-cost & field practical	-Can not identify the specific brain area utilizing the metabolic resources (i.e. blood flow) -Relatively new measurement technique	-Potential metric for cognitive readiness -Assess how information-processing resources are utilized and distributed in different situations
Functional Magnetic Resonance Imaging	fMRI	-BOLD response (blood oxygenation level response), i.e., cerebral blood flow	-Excellent spatial resolution -Extensive medical research base, growing cognitive science base	-Less good temporal resolution -Expensive -Restrictive environment	-As with EEG, ERP, and others, good for cognitive modeling validation
Functional Near Infrared Imaging Devices	fNIR	-Light is transmitted and the reflected light from the cortical level is encoded and reconstructed into a map of brain activity	-Non-invasive -Less restrictive than other brain imaging techniques (e.g. EEG) -Field practical	-New technology -Still in development -Unable to measure signals from deep brain tissue	-Real time assessment of warfighter's cognitive state that can be utilized in operational environments
Heart Rate Variability (Derived from an electrocardiogram)	HRV	-Cardiac activity	-Assess the variability in R-R intervals -Discriminate between parasympathetic and sympathetic influences on the heart	-Influenced by respiratory changes	-HRV related to changes in cortical brain areas -Changes in HRV may be related to cognitive events and individual differences

7.4.4.1 Heart Rate and Heart Rate Variability

Cardiac activity is measured by electrocardiogram (ECG/EKG). An ECG records the electric activity generated by the action potentials of the cardiac muscle cells. Two measures of cardiac activity that can derived from an ECG are heart rate and heart rate variability. Heart rate is measured in beats per minute. Research has shown that heart rate increases with increases in workload demands [51]. There is variability in the heart cycle. This variability is called Heart Rate Variability (HRV) which can be measured in the time or frequency domain. Research has suggested that variation in HRV may be used to differentiate between levels of task difficulty, the type of task and mental workload. For example, the more effort a task requires, the greater the suppression of HRV. Nickel and Nachreiner [58] examined if the .1 Hz component of HRV can discriminate between levels of workload. Results showed that the .1 Hz component discriminated between periods of work and rest. However, HRV was not different between types of tasks or level of task difficulty suggesting that tasks demands must be very high for the HRV components to be able to discriminate between workload conditions. In general, cardiac activity is probably the most commonly used measure in workload assessment [59]. HR and HRV may reflect energetic arousal, emotional processes and cognitive processes that may impact on task performance [58]. Cardiac activity can be easily measured and with the development of small telemetric systems it is a field practical measure.

7.4.4.2 Electrocortical Activity

7.4.4.2.1 Electroencephalography (EEG)

EEG is a non-invasive recording of the fluctuations in electrical activity of large ensembles of neurons in the brain which is taken from the scalp. Activity can be measured from numerous locations and across various bandwidths (e.g., alpha 8-12 Hz). There is an extensive research base on EEG activity during various cognitive tasks. For example, Gevins, Smith, Leong, McEvoy, Du and Rush [53] showed that neural networks could be used to discriminate differences in memory load based on EEG. In the adaptive automation literature, a general assumption is made that changes in EEG reflect arousal and workload [51]. Pertinent to our research interests, Scerbo and colleagues (e.g., [57]) have conducted an elegant series of experiments which use EEG to drive adaptive automation. It is a closed-loop system that moderates workload by decreasing the task demands when workload increases. Increases in workload are assessed via EEG. EEG is measured and an EEG engagement index is derived from components of the frequency domain. The system allocates the tasks based on the engagement index. A high engagement index is related to a high state of alertness and an increased ability to attend to stimuli [60]. The MAT task is used as the test-bed. There is a monitoring task, resource management task, communications task and a tracking task. The tracking task shifts between manual and automated depending on the engagement ratio of the operator. Tracking performance improved using an adaptive policy wherein high engagement EEG ratios invoked automation and low ratios invoked manual tracking compared to the opposite invocation policy (non-adaptive – switch to auto-low and manual-high). Prinzel et al. [57] showed that when the engagement index increased and the system automated the task, performance on the tracking task was better than when the task was always in the manual condition. Further, Prinzel, Pope and Freeman [61] provided biofeedback about the participants' performance and engagement level, which improved performance on the tracking task during automation and when the task was returned to manual. The engagement index modifies the system to meet the real-time needs of the operator and as a result improves performance. Using biofeedback, the participant was aware of his/her state which allowed the participant to be an active participant in the task environment.

However, these studies failed to show that other invocation policies or simply automating the tracking task would not have been as effective. For example, a recent study by Mikulka, Scerbo and Freeman [50] used a similar paradigm to investigate vigilance performance under low, medium and high event rates. They also

introduced a control to ensure that the invocation policies based on the EEG's were the cause of the performance differences. Each EEG participant had a yoked partner who switched to automation (or manual) mode in the same time hack and task level (unrelated to his or her EEG state) as the EEG partner. The group that had an adaptive EEG policy was again superior to the non-adaptive EEG engagement policy group. However, the yoked groups followed the same pattern implying that the results were caused by task difficulty and switching policy and not the EEG indices per se. This suggest that modeling the task environment might be as effective as using the EEG engagement policy. In summary, EEG based invocation policies show promise, but more research needs to be conducted to confirm their superiority in complex task environments in comparison to automated systems and to other invocation policies.

7.4.4.2.2 Event Related Potentials (ERP)

An EEG based system such as the one described above is triggered by gross changes in the level of engagement and may not be sensitive to different types of task and levels of task difficulty [63]. An ERP is a component of EEG related specifically to the presentation of a stimulus. It is an average of the EEG at the electrode sites of interest; a time locked response to a stimulus. ERPs are characterized by their polarity (positive or negative) and time of occurrence (latency from the onset of the stimulus). An ERP may reflect various information processing activities such as attention, intention and expectation. For example, the P300 reflects the availability and distribution of information processing resources and it may be sensitive to changes in workload. Prinzel et al. [62] assessed the sensitivity of the P300 to changes in workload and performance. Participants completed the MAT task (with or without adaptive automation) and an auditory oddball task in which they counted the number of high tones. Performance was better on the tracking and auditory task in the adaptive automation condition relative to the control and yoked control. Further, the amplitude of the P300 was greater in the adaptive automation condition than the control and yoked control condition. The amplitude of the P300 may be proportional to the attentional resources "invested" in the task. Kramer, Trejo and Humphrey [54] suggest that ERPs may be used to detect variations in workload. Kramer et al. had participants complete a radar monitoring task during which time task-irrelevant auditory probes were presented. The ERP amplitude (i.e., N100, N200), elicited by the deviant auditory probe, decreased as task load increased. Results showed that ERPs are sensitive to task type and level of task difficulty. Further, Prinzel and colleagues [62] suggest that the P300 may be an indicator of the efficacy of the adaptive automation and may also be used as a trigger for it.

In summary, electrocortical activity has been used with limited success in adaptive automation. It is important to note that as with all measurement techniques, EEG and ERPs are not without limitations. These measurements are not sensitive enough to assess activity in narrow cortical regions. EEG and ERPs have poor spatial resolution. Further, the EEG signal can affected by artifacts such as muscle activity and heartbeats. Filtering techniques can be used to eliminate some of the artifacts from the EEG. Researchers must be cautious when filtering their data that they do not filter out the EEG signal as well as the artifact. Current EEG systems are not field practical, but advances in technology are being made and systems are being designed to be more robust and field-ready.

7.4.4.3 Blood Flow

7.4.4.3.1 Positron Emission Tomography (PET) and Functional Magnetic Resonance Imaging (fMRI)

Brain imaging studies using PET and fMRI measures of cerebral blood flow have revolutionized cognitive psychology. There is hardly a domain of cognition where such measures have not been used to localize elements of cognitive function within the brain. It is therefore not surprising that researchers have also

attempted to examine whether these measures might be used for investigating more applied aspects of cognition such as mental workload.

The notion of mental workload as reflecting how hard one's mind is working at any given moment is intuitively appealing. Given that the mind is a function of the brain, it follows that mental work should be associated with *brain* work [63]. Brain work can be linked to both global and local changes in cerebral blood flow associated with mental activity. Sir Charles Sherrington, the great 19th century physiologist, first suggested that brain work was related to the regulation of the blood supply of the brain. Sherrington demonstrated that there is a close coupling between the electrical activity of neuronal cells, the energy demands of the associated cellular processes, and regional blood flow in the brain. His pioneering work suggested (but did not prove, since he lacked the technology) that if mental activity results in increased neuronal response in localized regions of the brain, then mental work could be assessed by measuring regional cerebral metabolism and blood flow. Autoradiographic studies in animals later confirmed Sherrington's principle for the regulation of brain blood flow and its coupling to neuronal activity and energy usage – but it would take several years before sensitive techniques were developed for measuring regional brain blood flow in humans. The development of PET paved the way for less invasive measurement of regional cerebral metabolism and blood flow in humans. PET is an adaptation of autoradiographic techniques originally developed for measuring blood flow in animals. Regional cerebral glucose metabolism can be non-invasively determined using PET and radioactively labeled glucose (18-fluoro-deoxyglucose), while regional cerebral blood flow may be assessed with PET and radioactively-labeled oxygen (O-15) in water.

PET is more accurate than the older methods in localizing the specific cortical regions activated by cognitive task demands. Nevertheless, the spatial resolution of PET, particularly in individual subjects, could be improved. Furthermore, the need for ionizing radiation, although safe when used within exposure limits, is an impediment against frequent use in studies with normal human subjects. The recent development of fMRI has overcome both these limitations. fMRI provides non-invasive, high-resolution assessment of regional cerebral blood flow.

How do PET and fMRI compare with EEG or ERP measures of cognition and mental workload? Because brain electrical activity is recorded from the scalp, the EEG or the ERP, while having excellent temporal resolution in identifying the neural correlates of mental processing, do not provide strong evidence for the localization of neural activity associated with mental workload. The poor spatial resolution of EEG and ERPs can be overcome to a degree through the use of such techniques as dipole modeling and spatial deconvolution. For example Gevins et al. [64] reported that midline frontal EEG theta activity is a correlate of mental workload. This EEG signal is thought to be generated in the anteromedial frontal cortex, possibly the anterior cingulate cortex which has also been proposed to be a high-level central executive control center on the basis of PET and lesion studies. In general, however, brain imaging techniques offer superior spatial resolution to EEG/ERPs and have provided recent evidence on the cortical localization of mental workload.

7.4.4.3.2 Blood Flow Velocity

Cerebral blood flow velocity (BFV) is assessed by Transcranial Doppler Sonography (TCD). TCD is used to assess the hemodynamic changes in the major cerebral arteries [65]. More specifically, TCD measures moment-to-moment changes in blood flow velocity (BFV). BFV values are obtained by recording frequency shifts in the ultrasound that is reflected from the blood flowing the cerebral artery. The frequency shifts are recorded by spectral analysis.

TCD could possibly be used to 'quantify attentional effort' [66]. Changes in BFV may reflect the changes in attention. According to the resource utilization model, demanding tasks consume more processing resources

than less demanding tasks [67]. When there is a decrease in performance this may reflect a depletion of the information processing resources. The amount of resources available may be related to BFV. For example, Maybelen [68] showed that as performance on vigilance tasks declined, there was a decrease in blood flow velocity over time. The amount of blood flow was also dependent on the type of task utilized; an absolute judgment task was associated with greater amount of blood flow and a greater decrease in blood flow over time than a comparative judgment task.

Hitchcock et al. [66] compared the blood flow during a vigilance task. Participants were presented low or high salience cues that were presented at various intervals during the experiment (e.g., high salience cues were presented 40% of the time). Results showed that for low salience cues there was a greater amount of blood flow and a greater decrease in blood flow over time in the right cerebral artery relative to that for high salience cues which were detected more often. The authors suggested that blood flow may be an index of the expenditure of information processing resources, which is greater for low than high salience cues.

TCD is a low cost non-invasive diagnostic tool. Unlike other measures of brain activity, such as fMRI or EEG, TCD has good temporal resolution while placing participants under less restrictive conditions. TCD may be able to be used in a laboratory or field setting. The use of TCD would allow researchers the flexibility to examine military issues in an appropriate context without the restrictions imposed by other measurement techniques. The disadvantage of TCD is that it has poor spatial resolution; researchers are only able to generalize the BFV to a cerebral artery. The cortical region that is utilizing the metabolic resources can not be identified. Furthermore, this is a relatively new technique in cognitive psychology and has not been used in adaptive automation applications.

7.4.4.4 Hybrid Measures

It is possible that one physiological measure will not be able to capture the complexity inherent in human performance. Wilson, Lambert and Russell [69] used a multiple measure approach to design a physiologically based adaptive automotive system. Wilson et al. had participants complete the MAT task with two levels of difficulty (varied the number of events that occurred in five minutes). EEG, ECG, EOG and respiration were measured during the task. Wilson et al. trained an artificial neural network (ANN) to recognize the physiological patterns that differentiate states of rest, low task difficulty and high task difficulty. The ANN was then used to determine which condition a participant was performing and when the high difficulty task was detected the monitoring and auditory task were automated. Results showed that the ANN correctly identified the task conditions and when adaptive automation was implemented tracking error decreased and performance on the resource management task increased compared to the manual condition. No comparison was made between fully and adaptively automated performances. (Wilson et al.).

7.4.4.5 Measurement Conclusions

Most of the measures were based on the premise that the measures are physiological correlates of workload and that as the human operator's workload increases beyond a certain point performance will start to deteriorate. Correlations with perceived workload and increased task difficulty have been shown for a number of the above indices. Some are currently not usable in a combat environment and others are not precise enough to be useful in their current configuration. However, these limitations are being overcome rapidly and the developing technology will soon be practical and precise enough for future combat systems. One of the most promising technologies is one of the oldest: EEG. However, the initial results are pertinent to artificial task environments and the usefulness of an adaptive systems based on EEGs has not been established. More research needs to done in the following areas:

1) Establish the usefulness of these measures for adaptive and adaptable processes compared to non-adaptive automation and manual control in complex environments.
2) Compare various adaptive invocation policies in these environments.
3) Investigate some of the newer techniques (including the neural net modeling) as their precision and practicality increases.

7.4.5 General Conclusions

Future combat environments will be radically different with robotic and automated systems becoming a ubiquitous component of future aerial and ground inventories. The authors discussed human performance advantages and disadvantages of battlefield automation and concluded:

1) A review of the human performance literature suggests that soldiers tend to both over rely and under rely on automated systems depending on the following factors:
    - Decision order;
    - Operator overload;
    - False alarm rate; and
    - Reliability of the system.
2) Poorly designed automation resulted in loss of situation awareness and operator complacency.
3) Adaptive and adaptable automated systems were evaluated as flexible alternatives to preset automation.
4) Various physiological measures were contrasted as potential components of adaptive systems in terms of their relative maturity and intrusiveness.
5) Although some of these measures were extremely promising as non-intrusive indexes of operator loading; more research is needed before practical adaptive or adaptable systems can be fielded.

7.4.6 References

[1] Woods, D. (1996). Decomposing automation: Apparent simplicity, real complexity. In: R. Parasuraman and M. Mouloua (Eds). Automation and human performance: theory and application. Human factors in transportation (pp. 3-17) Mahwah NJ: Lawrence Erlbaum Associates.

[2] Barnes, M.J., Wickens, C.D. and Smith, M. (2000). Visualizing uncertainty in an automated National Missile Defense simulation environment. Proceedings of the 4th Annual FedLab Symposium: Advanced Displays and Interactive Displays (pp. 117-122). Adelphi, MD: U.S. Army Research Laboratory.

[3] Sheridan, T.B. (1992). Telerobotics, automation, and supervisory control. Cambridge, MA: MIT Press.

[4] 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 – Part A: Systems and Humans, 30, 286-297.

[5] Parasuraman, R. and Riley, V. (1997). Humans and automation: Use, misuse, disuse, abuse. Human Factors, 39, 230-253.

[6] Mosier, K.L. and Skitka, L.J. (1996). Human decision makers and automated decision aids: Made fro each other? In: R. Parasuraman and M. Mouloua (Eds). Automation and human performance: theory and application. Human factors in transportation (pp. 163-176) Mahwah NJ: Lawrence Erlbaum Associates.

[7] Dzindolet, M.T., Pierce, L.G., Beck, H.P., Dawe, L.A. and Anderson, B.W. (2001). Predicting misuse and disuse of combat identification systems. Military Psychology, 13(3), 147-164.

[8] Dzindolet, M.T., Pierce, L.G., Beck, H.P. and Dawe, L.A. (2002). The perceived utility of human and automated aids in a visual detection task. Human Factors, 44(1), 79-94.

[9] Dzindolet, M.T., Beck, H.P., Pierce, L.G. and Dawe, L.A (2001). A framework for automation use (ARL-TR-2412) Aberdeen Proving Ground, MD: U.S. Army Research Laboratory.

[10] Parasuraman, R. and Byrne, E.A. (2003). Automation and human performance in aviation. In: P. Tsang and M. Vidulich (Eds.) Principles of Aviation Psychology. (pp. 311-356). Mahwah, NJ: Erlbaum.

[11] Parasuraman, R., Molloy, R. and Singh, I.L. (1993). Performance consequences of automation-induced "complacency." The International Journal of Aviation Psychology, 3, 1-23.

[12] Singh, I.L., Molloy, R. and Parasuraman, R. (1997). Automation-related monitoring inefficiency: The role of display location. International Journal of Human-Computer Studies, 46, 17-30.

[13] Barnes, M. (2003). Human dimension of battlefield visualization: Research and design issues (ARL-TR- 2885). Aberdeen Proving Ground, MD: U.S. Army Research Laboratory.

[14] Wickens, C. and Hollands, J. (2000). Engineering psychology and human performance. Upper Saddle River, NJ: Prentice Hall.

[15] Meyer, J. (2001). Effects of warning validity and proximity on responses to warnings. Human Factors, 43, 563-572.

[16] Dixon, S. and Wickens, C.D. (2003). Imperfect automation in unmanned aerial vehicle flight control. (AHFD)-03-1/MAAD-03-1) Savoy, IL: University of Illinois Research Lab.

[17] Rovira, E., McGarry, K. and Parasuraman, R. (2004). Effects of imperfect automation on decision making in a simulated command and control task. Under review, Human Factors.

[18] Galster, S.M. and Parasuraman, R. (2003). The application of a qualitative model of human-interaction with automation in a complex flight task. In: Proceedings of the 12th International Symposium of Aviation Psychology, 411-416, Dayton, Ohio: Ohio State University.

[19] Lorenz, B., DiNocera, F., Rottger, S. and Parasuraman, R. (2002). Automated fault management in a simulated space flight micro-world. Aviation, Space and Environmental Medicine, 73, 886-897.

[20] Endsley, M. (1996). Automation and situation awareness. Automation and human performance: theory and application. Human factors in transportation (pp. 163-179) Mahwah NJ: Lawrence Erlbaum Associates.

[21] Warm, J., Dember, W.N. and Hancock, P. (1996). Vigilance and workload in automated systems. In: R. Parasuraman and M. Mouloua (Eds.) Automation and human performance: Theory and applications. (pp. 183-200). Mahwah, NJ: Erlbaum.

[22] Byrne, E.A. and Parasuraman, R. (1996). Psychophysiology and adaptive automation. Biological Psychology, 42, 249-268.

[23] Parasuraman, R., Bahri, T., Deaton, J.E., Morrison, J.G. and Barnes, M. (1992). Theory and design of adaptive automation in aviation systems. (Technical Report No. NAWCADWAR-92033-60). Warminster, PA: Naval Air Warfare Center, Aircraft Division.

[24] Rouse, W.B. (1977). Human-computer interaction in multi-task situations. IEEE Transactions on Systems, Man and Cybernetics, SMC-7, 384-392.

[25] Scerbo, M.W. (1996). Theoretical perspectives on adaptive automation. In: R. Parasuraman and M. Mouloua (Eds.), Automation and human performance: Theory and applications (pp. 37-63). Mahwah, NJ: Lawrence Erlbaum Associates.

[26] Wickens, C. and Hollands, J. (2000). Engineering psychology and human performance. Upper Saddle River, NJ: Prentice Hall.

[27] Dixon, S., Wickens, C.D. and Chang, D. (2004). Unmanned aerial vehicle flight control: False alarms versus misses. In: Proceedings of the 48th Annual Human Factors and Ergonomics Society Meeting. New Orleans, LA.

[28] Barnes, M. and Grossman, J. (1985). The intelligent assistant for electronic warfare systems (NWC TP 5885) China Lake CA: US Naval Weapons Center.

[29] Rouse, W.B. (1976). Adaptive allocation of decision making responsibility between supervisor and computer. In: T.B. Sheridan and G. Johannsen (Eds.), Monitoring behavior and supervisory control (pp. 295-306). New York: Plenum Press.

[30] Geddes, N. (1985). Intent inferencing using scripts and plans. In: Proceedings of the First Annual Aerospace Applications of Artificial Intelligence Conference. (pp. 123-127). Washington DC: AAAI.

[31] Billings, C.E. and Woods, D. (1994). Concerns about adaptive automation in aviation systems: In: R. Parasuraman and M. Mouloua (Eds). Automation and human performance: Current research trends (pp. 264-269) Hillsdale NJ: Lawrence Erlbaum Associates.

[32] Opperman, R. (1994). Adaptive User Support. Hillsdale, NJ; Erlbaum.

[33] Scerbo, M. (2001). Adaptive automation. In: W. Karwowski (Ed.) International encyclopedia of human factors and ergonomics. London: Taylor and Francis, Inc.

[34] Inagaki, T. (1999). Situation-adaptive autonomy: Dynamic trading of authority between human and automation.

[35] Miller, C., Pelican, M. and Goldman, R. (2000). 'Tasking' interfaces for flexible interaction with automation: Keeping the operator in control. In: Proceedings of the Conference on Human Interaction with Complex Systems. Urbana-Champaign, IL.

[36] Miller, C. and Parasuraman, R. (2003). Beyond levels of automation: An architecture for more flexible human automation collaboration. In: Proceedings of The Human Factors and Ergonomics Society, Denver, CO, October 2003.

[37] Miller, C. and Parasuraman, R. (2002). Designing for flexible human-automation interaction: Playbooks for supervisory control, Technical Report, SIFT, MN.

[38] Miller, C., Pelican, M. and Goldman, R. (2000). 'Tasking' interfaces for flexible interaction with automation: Keeping the operator in control. In: Proceedings of the Conference on Human Interaction with Complex Systems. Urbana-Champaign, IL.

[39] Parasuraman, R., Galster, S. and Miller, C. (2003). Human control of multiple robots in the RoboFlag simulation Environment. In: Proceedings of the 2003 IEEE International Conference on Man, Systems and Cybernetics. Washington, DC.

[40] Squire, P.N., Galster, M. and Parasuraman, R. (2004). The effects of levels of automation in the human control of multiple robots in the RoboFlag simulation environment. In: D.A. Vincenzi, M. Mouloua and P.A. Hancock, Human Performance, Situation Awareness, and Automation Volume II (pp. 48-53). Mahwah, NJ: Erlbaum.

[41] Rouse, W.B. (1988). Adaptive aiding for human-computer control. Human Factors. 30, 431-441.

[42] Hilburn, B., Jorna, P.G., Byrne, E.A. and Parasuraman, R. (1997). The effect of adaptive air traffic control (ATC) decision aiding on controller mental workload. In: M. Mouloua and J. Koonce (Eds.), Human-automation interaction: Research and practice (pp. 84-91). Mahwah, NJ: Erlbaum Associates.

[43] Kaber, D.B. and Riley, J.M. (1999). Adaptive automation of a dynamic control task based on secondary task workload measurement. International Journal of Cognitive Ergonomics, 3, 169-187.

[44] Parasuraman, R. (2000). Designing automation for human use: Empirical studies and quantitative models. Ergonomics, 43, 931-951.

[45] Moray, N., Inagaki, T. and Itoh, M. (2000). Adaptive automation, trust, and self-confidence in fault management of time-critical tasks. Journal of Experimental Psychology – Applied, 6, 44-58.

[46] Parasuraman, R., Mouloua, M. and Hilburn, B. (1999). Adaptive aiding and adaptive task allocation enhance human-machine interaction. In: M.W. Scerbo and M. Mouloua (Eds.) Automation technology and human performance: Current research and trends. (pp. 119-123). Mahwah, NJ: Erlbaum.

[47] Mouloua, M., Molloy, R. and Parasuraman, R. (1993). Monitoring automation failures: Effects of single and multiadaptive function allocation. In: Proceedings of the Annual Meeting of the Human Factors Society, Human Factors Society, Santa Monica, CA, pp. 1-5.

[48] Farrell, S. and Lewandwosky, S. (2000). A connectionist model of complacency and adaptive recovery under automation. Journal of Experimental Psychology: Learning, Memory, and Cognition 26, 395-410.

[49] Byrne, E.A. and Parasuraman, R. (1996). Psychophysiology and adaptive automation. Biological Psychology, 42, 249-268.

[50] Mikulka, P.J., Scerbo, M.W. and Freeman, F.G. (2002). Effects of a biocybernetic system on vigilance performance. Human Factors, 44(4), 654-664.

[51] Scerbo, M.W., Freeman, F.G., Mikulka, P.J., Parasuraman, R., DiNocera, F. and Prinzel, L.J. (2001). The efficacy of psychophysiological measures for implementing adaptive technology. (Technical Report No. NASA/TP-2001-211018). Hampton VA: NASA, Langley Research Center.

[52] Gopher, D. and Donchin, E. (1986). Workload: An examination of the concept. In: K. Boff, L. Kauffman, and J.P. Thomas (Eds.), Handbook of Perception and Human Performance, (pp. 41.1-41.49). New York: Wiley & Sons.

[53] Gevins, A.S., Smith, M.E., Leong, H., McEvoy, L., Du, R. and Rush, G. (1998). Monitoring memory load during computer-based tasks with EEG pattern recognition methods. Human Factors, 40, 79-91.

[54] Kramer, A.F., Trejo, L.J. and Humphrey, D.G. (1996). Psychophysiological measures of workload: Potential applications to adaptively automated systems. In: R. Parasuraman and M. Mouloua (Eds.), Automation and human performance: Theory and applications (pp. 137-162). Mahwah, NJ: Lawrence Erlbaum Associates.

[55] Parasuraman, R. (1990). Event-related brain potentials and human factors research. In: J.W. Rorbaugh, R. Parasuraman and R. Johnson (Eds.), Event-related potentials: Basic and applied issues (pp. 279-300). New York: Oxford University Press.

[56] Scerbo, M.S., Freeman, F.G., Mikulka, P.J., Parasuraman, R., Di Nocera, F. and Prinzel, L.J. (2001). The efficacy of psychophysiological measures for implementing adaptive technology. Technical Paper NASA/TP-2001-211018. Hampton, VA: NASA Langley Research Center.

[57] Prinzel, L.J., Freeman, F.G., Scerbo, M.W., Mikulka, P.J. and Pope, A.T. (2000). A closed-loop system for examining psychophysiological measures for adaptive automation. International Journal of Aviation Psychology, 10, 393-410.

[58] Nickel, P. and Nachreiner, F. (2003). Sensitivity and diagnosticity of the 0.1-Hz component of heart rate variability as an indicator of mental workload. Human Factors, 45(4), 575-590.

[59] Lenneman, J.K. and Backs, R.W. (2000). The validity of factor analytically derived cardiac autonomic components for mental workload assessment. In: Backs, R.W. and Boucsein, W. (Eds.), Engineering Psychophysiology: Issues and Applications. Mahwah, NJ: Lawrence Erlbaum.

[60] Freeman, F.G., Mikulka, P.J., Prinzel, L.J. and Scerbo, M.W. (1999). Evaluation of an adaptive automation system using three EEG indices with a visual tracking task. Biological Psychology, 50(1), pp. 61-76.

[61] Prinzel, L.J., Pope, A.T. and Freeman, F.G. (2001). Physiological self-regulation and adaptive automation. The International Journal of Aviation Psychology, 12, 179-196.

[62] Prinzel, L.J., Freeman, F.G., Scerbo, M.W., Mikulka, P.J. and Pope, A.T. (2000). A closed-loop system for examining psychophysiological measures for adaptive automation. International Journal of Aviation Psychology, 10, 393-410.

[63] Parasuraman, R. and Caggiano, D. (2002). Mental workload. In: V.S. Ramachandran (Ed.) Encyclopedia of the human brain. San Diego: Academic Press.

[64] Gevins, A. and Smith, M.E. (2003). Neurophysiological measures of cognitive workload during human-computer interaction. Theoretical Issues in Ergonomic Science, 4, 113-131.

[65] Stroobant, N. and Vingerhoets, G. (2000). Transcranial doppler ultrasonography monitoring of cerebral hemodynamics during performance of cognitive tasks: A review. Neuropsychology Review, 10, 213-231.

[66] Hitchcock, E.M., Warm, J.S., Dember, W.N., Matthews, G.R., Shear, P.K., Rosa, R.R., Tripp, L., Mayleben, D.W. and Parasuraman, R. (2000). Proceedings of the Human Factors and Ergonomic Society 44th Annual Meeting (3-382 – 3-385).

[67] Parasuraman, R., Warm, J.S. and Dember, W.N. (1987). Vigilance: Taxonomy and utility. In: L.S. Mark, J.S. Warm and R.L. Huston (Eds.) Ergonomics and human factors: Recent research. (pp. 11-32). New York: Springer-Verlag.

[68] Maybelen, D. (1998). Cerebral blood flow during sustained attention. Unpublished doctoral dissertation. University of Cincinnati, Cincinnati, OH.

[69] Wilson, G.F., Lambert, J.D. and Russell, C.A. (2000). Performance enhancement with real-time psychophysiologically controlled adaptive aiding. Proceedings of the Army Science Conference, ASC00-0350.

7.5 DESIGNING FOR FLEXIBLE HUMAN-AUTOMATION INTERACTION: PLAYBOOKS FOR SUPERVISORY CONTROL

As systems become more complex, there is increasing temptation to control them via "automation" – that is, a device, machine, or system that accomplishes, fully or partially, a function which was or could be performed by a human [1]. Examples of this trend are rife in domains ranging from aviation and air traffic management to health care and bioinformatics [2-4].

Automation can provide clear benefits. Billings [2] documents payoffs of increased automation in commercial aviation in four key areas: safety, reliability, economy, and comfort – but automation has also been shown to pose novel problems for human operators in some circumstances: to increase workload and training requirements, to result in decreased situation awareness and, in specific instances, to cause accidents (e.g., [1,5-7]).

This 'double-edged sword' of automation use [8] has motivated repeated questions about which tasks should be automated to which level or degree for optimal control, performance, and safety. Technologists tend to push to automate tasks as fully as possible – what has been called the 'technological imperative' [9]. Human factors engineers and others concerned with safety and the human role in advanced systems have tended to highlight the risks of increased automation (e.g., [10-12]) and to argue against the use of higher levels of automation, especially if the human role in the resulting system is decided by default.

An approach to human-automation relationships that retains the benefits of automation while minimizing its costs and hazards is needed. For reasons discussed below, we believe that such an approach requires that neither human nor automation be exclusively in charge of most tasks, but rather demands flexibility in the role and level of automation while placing control of that flexibility in the human operator's hands. This implies

that, for most tasks, automation levels should be at neither end of a possible spectrum, but rather at some intermediate, and *adjustable*, point. Human operators need to be able to delegate tasks to automation, and receive feedback on their performance, in much the same way that delegation can be performed in successful human-human teams and organizations: at various levels of detail and granularity, and with various constraints, stipulations, contingencies, and alternatives.

In this paper, we begin by reviewing the literature supporting the advantages of a delegation-based, intermediate approach to human-automation relationships. Then we examine traditional approaches to characterizing automation levels and discuss why a delegation approach demands they be extended to explicitly represent task decomposition. Finally, we present one specific implementation of a delegation approach, based on a sports team 'playbook' metaphor.

7.5.1 Intermediate Automation Levels-Costs of Automation Extremes

The promised, and frequently realized, benefits of automation have generally been sufficient to argue against relegating many tasks to the lowest levels of automation—purely manual performance. The economic benefits of automation are a strong, but not the only, motivator. Automation does offer substantial benefits in human workload reduction, increased performance and safety, when it is properly designed and used [1,2,5,7,13,14]. The case against applying automation wherever, whenever and at the highest levels technologically feasible has been harder to make. Much research has been devoted to showing the disadvantages of reduced human engagement in system control and problem solving characteristic of high-level automation [1,2,7,8,15-22]. These problems will be characterized next:

1) *Reduced Situation and System Awareness:* High levels of automation, particularly of decision-making functions, may reduce the operator's awareness of system and environmental dynamics [23,24]. Humans tend to be less aware of changes when those changes are under the control of another agent (whether automation or human) than when they make the changes themselves [25]. Mode errors also illustrate the impact of automation on the user's awareness of system characteristics [26,27]. Mode errors arise when the operator executes a function that is appropriate for a mode other than the one the system is currently in [28]. When a system is capable of automatically changing its mode, as is the case with aircraft Flight Management Systems (FMS), mode errors become more common precisely because the operator is less likely to be aware of the current system mode [26]. Several aviation incidents and accidents have involved this type of error [2,29].

2) *Trust, Complacency, and Over-Reliance:* Trust is an indicator of how accurately the operator understands the system [30,31]. Operators may not use well-designed, reliable automation if they believe it to be untrustworthy, or they may continue to rely on automation even when it malfunctions if they are overconfident in it. Highly automated systems are prey to both sorts of errors [7].

   The problem of excessive trust or complacency has been documented in several studies showing that students in laboratory experiments [32], trained pilots in simulation [33], and experienced air traffic controllers using decision aiding automation [34] are poor at monitoring automation for occasional malfunctions if their attention is occupied with other tasks. In these studies, users grant more autonomy to a system than it was designed to support by almost always accepting recommendations even though they are sometimes incorrect. Over-reliance on automation can also manifest as a decision bias stemming from a heuristic to reduce the cognitive effort involved in solving a problem. This heuristic may result in "automation bias" [35] – a tendency to uncritically accept automation recommendations. Although reliance on automation may be an effective strategy in many cases, over-reliance can lead to errors when automation is less than perfect.

3) *Skill Degradation:* If higher levels of automation can result in complacency and loss of situation awareness, it is perhaps not surprising that they can also result in skill degradation if allowed to persist. The pilots of increasingly automated aircraft feared this effect with regards to psychomotor skills such as aircraft attitude control [2], but it has also been demonstrated for decision making skills [24]. In both cases, the use of an intermediate, lower level of automation alleviated skill degradation if the skills had been learned in the first place.

4) *Unbalanced Mental Workload:* Automation can sometimes produce extremes of workload, either too low or too high. That high levels of automation could leave an operator bored and inattentive is, perhaps, to be expected. That automation can increase workload, on the other hand, is one of the "ironies of automation" [8] because many automated systems are introduced as workload-saving moves, but this does not always occur. First, if automation is implemented in a "clumsy" manner, e.g., if executing an automated function requires extensive data entry or "re-programming" by human operators at times when they are very busy, workload reduction may not occur where it is most needed [36]. Second, if engagement of automation requires considerable "cognitive overhead" [37], i.e., extensive cognitive evaluation of the benefit of automation versus the cost of performing the task manually, then users may experience greater workload in using the automation.

5) *Degraded Overall Performance:* The most significant effect of using too high an automation level may be degraded overall performance of the human + machine system. In an experiment involving aircraft navigation and route planning, Layton et al. [38] provided operators with one of three levels of support ranging from 'sketching only', where the human sketched a desired route and the system provided feedback about its feasibility, to 'full automation' where the system automatically provided a recommended 'best' route according to its optimization criteria. An intermediate level allowed the user to ask for a route with specific characteristics that the system then provided if possible. They found that humans in the intermediate and high automation conditions frequently explored more routes because, in the highly manual sketching condition, the process of arriving at a route was too difficult to allow trying many alternatives consistently and fully. By contrast, as in [35], in the full automation condition, users tended to accept the first route suggested without exploring it or its alternatives deeply. Even when they did explore, the system's recommendation tended to narrow and bias their search. Users tended to check the route provided for obvious mistakes rather than generating a preferred route on their own. Particularly in trials when the automation performed suboptimally (e.g., because it failed to adequately consider uncertainty in weather predictions), the intermediate level of automation produced better overall solutions.

6) *Lower User Acceptance:* Finally, an additional problem with high levels of automation is lack of user acceptance. This may be particularly true of highly trained and skilled operators of complex, high-criticality systems such as aircraft, military systems, process control, etc. For example, Miller [39], interviewed several pilots and designers to develop a consensus list of prioritized goals for a "good" cockpit configuration manager for the Rotorcraft Pilot's Associate [40]. Even though this system provided advanced automation capable of managing information displays and cockpit functions to conform to pilot intent, two of the top three items on the pilots' consensus list were "Pilot remains in charge of task allocation" and "Pilot remains in charge of information presented."

   Similarly, Vicente [41] and Bisantz et al. [42] cite examples of human interactions with even such low criticality automation as food planning aids in fast food restaurants, showing that operators can become frustrated when forced to interact with automation that removes their authority to do their jobs in the best way they see fit. Vicente [41] summarizes findings from [43] showing that jobs in

which human operators have high psychological demands coupled with low decision latitude (the ability to improvise and exploit one's skills for job performance) lead to higher incidences of heart disease, depression, pill consumption, and exhaustion.

7.5.2 Tradeoff Space for Effects of Automation Level

The findings summarized above show that a mixture of human and automation involvement is frequently desirable rather than the extremes of full or no automation. In these cases, human + machine systems must be designed for an appropriate relationship between operators and automation allowing both parties to share responsibility, authority and autonomy over many work behaviors in a safe, efficient and reliable fashion. The effects of different human-automation relationships over a task or function can be viewed as a 'tradeoff space'. Figure 7-8 presents a conceptual view of this space over three relevant parameters:

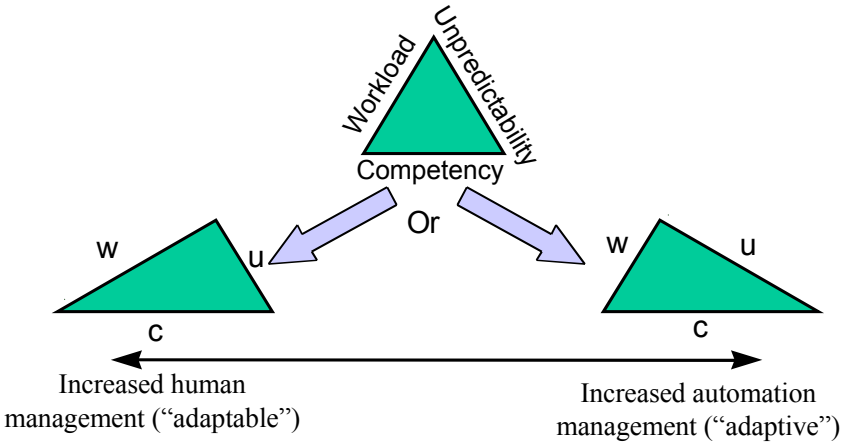

Figure 7-8: The Tradeoff Relationship between System Competency, Human Workload and Unpredictability.

The Competency of the Human-Machine System. Competency refers to correct behavior in context. Therefore, a system becomes more 'competent' whenever it provides correct behaviors more frequently or encompasses a greater number of contexts.

The Mental Workload Required for the Human to Interact with the System. Workload refers to the combined amount of attentional and cognitive "energy" the human must exert to use the system [44-46]. Mental workload for the human is modeled because it is the major constraint on the other two dimensions.

The Unpredictability of the System to the Human Operator. Unpredictability refers to the inability of the human to know exactly what the automation will do when. Unpredictability is a consequence of the human's not personally taking all actions in the system—of not being 'in control' directly and immediately. It is inversely correlated with situation awareness (at least awareness pertinent to automation functions) and, generally, workload. Good system and interface design and good hiring and training practices can serve to reduce unpredictability, but any form of task delegation—whether to automation or other humans—must result in a degree of unpredictability if it offloads tasks.

The triangle in Figure 7-8 illustrates the relationship, or 'tradeoff space', among these three dimensions. The performance of a set of functions to a given level of competency can only be achieved through some mix

of workload and unpredictability. A user's workload can be reduced by allocating some functions to automation, but only at the expense of increased unpredictability; conversely, reducing unpredictability by having the user perform functions increases workload. Alternate designs for a level of competency represent different mixes of workload and unpredictability – corresponding to varying the lengths of the triangle's sides while holding the base constant. It is sometimes possible to reduce both workload and unpredictability for a given level of competency through better design – corresponding to shortening the height of the triangle.

In this tradeoff model, any increase in human-machine system competency must affect the human in that either:

1) The added functionality must be fully controlled by the human(s), resulting in workload increases; or
2) It must be managed by automation, resulting in unpredictability increases.

Opperman [47] identified these alternatives as 'adaptive' and 'adaptable' approaches to system design (see also [48]). In either case, the human + machine system can adapt to various contexts, but in adaptive systems automation determines and executes the necessary adaptations, while in adaptable systems, the operator is in charge of the desired adaptations. The persistent debate (e.g., [49,50]) in the Human-Computer Interface community over intelligent agents vs. direct manipulation interfaces is a similar manifestation of these alternative approaches.

Another implication of the tradeoff space is that these approaches are the endpoints of a spectrum with many alternatives in between, each representing a different tradeoff between workload, unpredictability and competency (as in Figure 7-9). The range of alternatives available may be constrained by automation and/or human capabilities, but within the range of feasible systems, an alternative must be selected to assign roles and responsibilities between human(s) and automation. This decision is the process of selecting an automation 'level' or relationship and it corresponds to picking a point in the tradeoff space [1]. When the division of labor is done by a supervisor for a human team, the process may be called 'delegation'; when done by a designer prior to system operation, it is part of system design. Our objective is to provide a human supervisor the ability to flexibly delegate tasks to automation at the time of use rather than relying on a static design point.

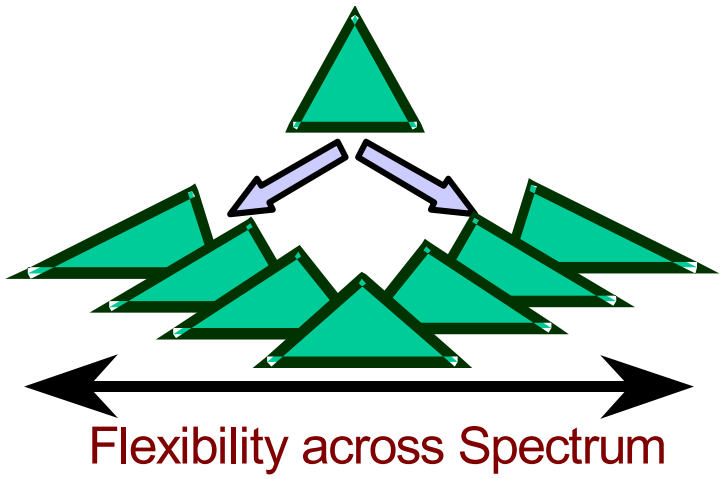

Figure 7-9: A Tasking Interface Provides Flexibility Within the Tradeoff Space.

7.5.2.1 Flexible Automation Levels

Traditionally, the chief difference between task delegation performed by a supervisor and design of an automated system has been that the supervisor had much more flexibility in what, when and how to delegate and better awareness of the task performance conditions, while the designer had to fix a relationship at design time and incorporate it statically into the to-be-built system for all contexts of use. More recently, "adaptive automation" [48,51-56], has been developed which chooses a level of automation for tasks based on contextual criteria such as location, situation, workload, experience, physiological state, etc. In adaptive systems, the division of labor between human and machine is not fixed. An adaptive aiding system may provide assistance through context-dependent forms of automation including the adaptive presentation of information.

The adaptive automation concept was proposed over 25 years ago [57], however the technologies for its effective implementation have only recently matured and empirical evidence of its effectiveness been provided. Studies have shown that adaptive systems can regulate operator workload and enhance performance, while preserving the benefits of automation [51,58,59]. The performance costs of certain forms of automation described previously – reduced situation awareness, complacency, skill degradation, etc – may also be mitigated [54,55,60-63]. Adaptive aiding systems have recently been successfully flight tested in the Rotorcraft Pilot's Associate [64].

Such systems are truly "adaptive" in Opperman's [47] sense since they choose the level of automation to apply. By contrast, human task delegation within a team is more nearly an "adaptable" system, since the human supervisor can choose which tasks to give to a subordinate, how much to dictate about how (or how not) to perform subtasks, how much attention to devote to monitoring, approving, reviewing and correcting task performance, etc.

The ability to delegate tasks to subordinates – and to do so flexibly, adapting both the tasks allocated and the degree of supervision, instruction and monitoring to the context and the capabilities of team members – is clearly a powerful form of organizing and performing work. It is also deeply familiar to anyone who has worked in a team or a supervisory setting. As described below, we have sought to extend such delegation capabilities to control over flexible automation. Before presenting that work, however, we must first describe what is meant by an automation level and then argue for an extension to traditional definitions to support delegation-based approaches to supervisory control.

7.5.2.2 Characterizing Automation Levels

To discuss alternative forms of automation, it is helpful to have a scheme for characterizing automation types, roles and responsibilities. In general, such characterizations have been made in terms of 'levels' of automation: defining a spectrum of possible relationships ranging from full human control authority to full automation.

7.5.2.3 Prior Automation Level Spectra

Bright [65] was perhaps the first to propose such a spectrum. Bright viewed automation as evolving through 17 levels of competency and noted that intermediate stages might well demand greater human workload, skill and training requirements than lower levels. Sheridan [66], is generally credited with defining "supervisory control" – a specific relationship where control automation allows the human to behave as if interacting with an intelligent, human subordinate. He described a ten point spectrum of automation levels whose endpoints are full control autonomy for the human (essentially no role for automation) and vice versa [66].

The intermediate levels in this spectrum, then, represent alternative forms of supervisory control interactions. A version of this list is shown in Table 7-5.

Table 7-5: Levels of Automation (after [66])

1	Human does it all
2	Computer offers alternatives
3	Computer narrows alternatives down to a few
4	Computer suggests a recommended alternative
5	Computer executes alternative if human approves
6	Computer executes alternative; human can veto
7	Computer executes alternative and informs human
8	Computer executes selected alternative and informs human only if asked
9	Computer executes selected alternative and informs human only if it decides to
10	Computer acts entirely autonomously

A problem with these simple, uni-dimensional models of human-automation relationships is that they are ambiguous about what the relationship is defined over. Parasuraman, Sheridan, and Wickens [1] recently noted that Sheridan's levels referred mainly to automation which makes decisions, offers suggestions and/or executes actions. There are, however, other jobs automation can do: for example, sensing and processing information to detect situations of interest. Parasuraman et al. [1] applied a four-stage model of human information processing to arrive at four functions that must be accomplished to perform most tasks:

- Information acquisition;
- Information analysis;
- Decision and action selection; and
- Action implementation.

Since these functions can be performed by either human or automation in various mixes, Parasuraman et al. [1] added a second dimension to Sheridan's spectrum. Most human + automation systems can be characterized by a mix of levels of automation across these four stages, as in Figure 7-10. One system (A) might be highly autonomous in information acquisition, but comparatively low on the other three functions, while a second system (B) might offer a high level of automation across all four sub-functions.

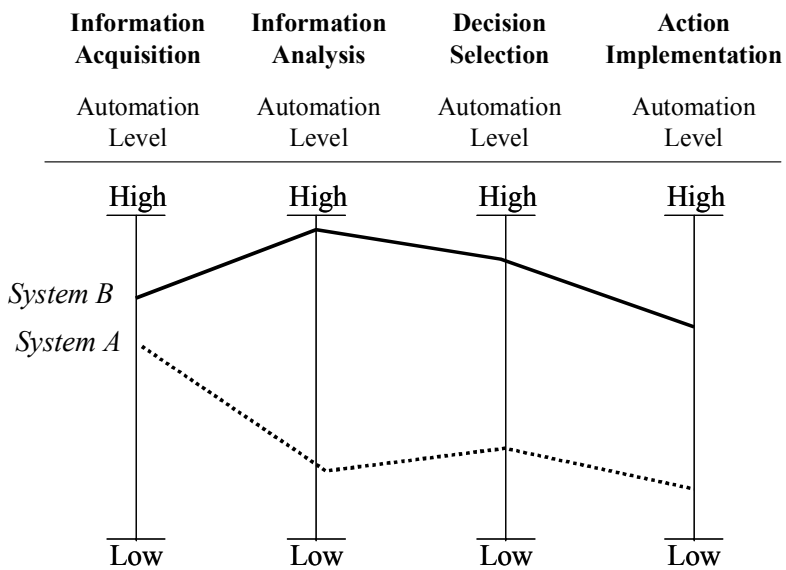

Figure 7-10: Levels of Automation by Information Processing Phase for Two Systems (from [1]).

7.5.2.4 Extending Automation Levels

An important implication of this two-dimensional, levels and stages model of automation [1] is that a parent task can be decomposed – at least into four stages – and that a single automation level need not be applied homogenously. However, in this approach a parent task is decomposed into abstract subtasks based on information processing stages, whereas other task decomposition methods arguably provide more insight into how a task may be performed. Chief among these are functional decompositions (e.g., Operator Functional Modeling [67], Goals-Operations-Methods-Scripts (GOMS) models [68] and Plan Goal Graphs [69]) that stress the sub-functions required to achieve a parent, and sequential process models (e.g., PERT or CPM charts (e.g., [70], Petri nets [71]) that stress the temporal ordering and duration of a function's steps. Such decompositions are inherently hierarchical and may proceed through any number of levels to some primitive, "stopping" level [72] that may be imposed by biology, physics or, more commonly, the functional purpose of the decomposition.

Thus, while the two-dimensional model of automation roles offered by Parasuraman et al. [1] represents a major advance over earlier uni-dimensional models, it arguably does not go far enough. The subdivision of a parent task into four information processing phases represents only a single level of decomposition into an abstract set of task categories. In practice, tasks are accomplished by hierarchical sequences of specific activities – the parent task's subtasks. Automation may be applied differently to each and every subtask that comprises the parent task. Thus, the profile of automation levels sketched in Figure 7-10 should stretch not merely over the four information processing phases at one level of decomposition, but over as many subtasks and levels as we want or need to divide a parent task into.[1]

[1] In fact, the relationship between automation level and task decomposition is more complex than this. As is well understood [1,2,10,26], automation does not merely shift responsibility for tasks but can change their nature as well. In a task decomposition, this means that some sub-tasks may be eliminated while others may be added. This implies that there will generally be multiple alternate decompositions of a parent task depending on, among other things, what level of automation is used. Each alternative constitutes a different combination of human and automation subtasks. The particular set represented should be a function of the model's purpose: if used for design, the full set of possible and reasonable alternatives should be explored; if used to represent possible actions in an existing system, then only the available methods need be represented.

When one identifies a level of automation for a system using Bright's or Sheridan's spectra, one is identifying something like an average or modal level over the subtasks the system accomplishes. Similarly, when one uses a model of levels and stages of automation [1], one is clustering the subtasks by information processing stage and again averaging. Assigning levels by stages offers more sensitivity than assigning them only to the parent task, but it is still an abstraction. In practice, one could identify the *specific* subtasks to be performed and represent an automation level for each of them.

Why would one want to? More sub-tasks are not necessarily better and, in fact, Parasuraman et al. [1] explicitly chose a four-stage model over more elaborate alternatives to simplify design considerations. Precision in representation may be inherently desirable for some purposes (such as training and detailed design), but our interest is in supporting flexible task delegation with automation. As we saw above, for any intermediate level of automation for a task, there are roles for both humans and automation on its sub-tasks. Yet, *someone must coordinate those roles*. Insofar as the human is required to manage, or at least be aware of, that division of labor, s/he must understand the decomposition of the task in question. Supervisory control is a process of task delegation and delegation requires task decomposition. Task decomposition seems to reflect the way humans delegate responsibilities to subordinates, and reason about task performance when receiving feedback [74]. Delegation is, in short, a process of assigning specific roles and responsibilities for the subtasks of a parent task for which the delegating agent retains authority (and responsibility). Furthermore, communication about intent to the agent's subordinates is frequently in terms of specific goals, methods and/or constraints on how, when and with what resources the subtasks should be accomplished [73-75]. For these reasons, it is essential that those subtasks be modeled explicitly.

7.5.2.5 Using Extended Automation Levels in Design

Above, we argued that an automation level is a defined combination of roles and responsibilities between human and automation for a task to be performed, and that a task can generally be decomposed into subtasks – each of which may have its own automation level. This realization opens the doors for more explicit communication, either during design or operations, about the relationship between human and automation concerning the tasks to be accomplished.

We are developing a method for pushing this communication into the run-time environment – more closely emulating delegation in human-human work relationships and allowing the operator to smoothly adjust the 'amount' and level of automation used depending on such variables as time available, workload, decisions criticality, trust, etc.

While this does not eliminate the tradeoff presented in Figure 7-8, it mitigates it by allowing operators to choose various points on the spectrum for interaction with automation (Figure 7-9). The fundamental tradeoff remains, but the operator is in charge of choosing a tradeoff point in that space. This strategy follows both Rasmussen's [76] and Vicente's [41] approach of allowing the operator to 'finish the design' at the time and in the context of use. This allows more control and authority over how and when the user interacts with automation.

Such a system must avoid two problems. First, it must make achievable the task of commanding automation to behave as desired without excessive workload. Second, it must ensure safe and effective overall behavior. We have created a design metaphor and system architecture that addresses these two concerns. We call the general class of systems that can take delegation instruction at various levels tasking interfaces, because they allow posing a task to automation in the different ways one might 'task' a knowledgeable subordinate. Examples we would label tasking interfaces can be found in [75,77]. Our particular approach to enabling,

facilitating and ensuring correctness from a tasking interface, we call a Playbook – because it is based on the metaphor of a sports team's book of approved plays. We have been exploring alternate methods to achieve the flexibility and ease of human-human delegation within human-machine systems. Two examples of our work on delegation systems – a "Playbook®" for task- and constraint-based delegation and "Policy" interactions for setting high level priorities for automation – are discussed in the next section.

7.5.3 References

[1] Parasuraman, R., Sheridan, T. and Wickens, C. (2000). "A Model for Types and Levels of Human Interaction with Automation," IEEE Trans. Syst., Man, Cybern. – Part A: Syst. and Hum., Vol. 30, pp. 286-297.

[2] Billings, C. (1997). Aviation Automation: The search for a human-centered approach. Mahwah, NJ: Lawrence Erlbaum.

[3] Parasuraman, R. and Mouloua, M. (Eds.) (1996). Automation and Human Performance: Theory and Applications. Mahwah, NJ: Erlbaum.

[4] Sheridan, T. (2002). Humans and automation. Santa Monica and NY: Human Factors and Ergonomics Society & Wiley.

[5] Reason, J. (1997). Managing the risks of organizational accidents. Aldershot, UK: Ashgate Publishing Ltd.

[6] Casey, S. (1993). Set Phasers on Stun, Santa Barbara, CA: Aegean Publishing Company.

[7] Parasuraman, R. and Riley, V. (1997). "Humans and automation: Use, misuse, disuse, abuse," Human Factors, Vol. 39, pp. 230-253.

[8] Bainbridge, L. (1983). "Ironies of automation," Automatica, Vol. 19, pp. 775-779.

[9] Sheridan, T. (1987). "Supervisory Control". In: Handbook of Human Factors. G. Salvendy, Ed. New York: John Wiley & Sons, pp. 1244-1268.

[10] Woods, D. (1996). "Decomposing automation: Apparent simplicity, real complexity". In: Automation and Human Performance: Theory and Applications, R. Parasuraman and M. Mouloua, Eds., Mahwah, NJ: Erlbaum, pp. 1-16.

[11] Perrow, C. (1984). Normal Accidents: Living with high-risk technologies. USA: Basic Books.

[12] Reason, J. (1997). Managing the risks of organizational accidents. Ashgate Publishing Ltd.; Aldershot, UK.

[13] Bright, J. (1955). "Does automation raise skill requirements?", Harvard Bus. Rev., Vol. 36, pp. 85-98.

[14] Lorenz, B., Nocera, F., Rottger, S. and Parasuraman, R., "Human Interaction with automated fault management during simulated space flight operations: The role of information sampling activities," To appear in Aviation, Space and Environmental Medicine.

[15] Amalberti, R. (1999). "Automation in aviation: A human factors perspective". In: Handbook of aviation human factors, D. Garland, J. Wise and V. Hopkin, Eds., Mahwah, NJ: Erlbaum, pp. 173-192.

[16] Lewis, M. (1998). "Designing for human-agent interaction," Artificial Intelligence Magazine, (Summer), pp. 67-78.

[17] Rasmussen, J. (1986). Information processing and human-machine interaction. Amsterdam: North-Holland.

[18] Sarter, N., Woods, D. and Billings, C. (1997). "Automation surprises". In: Handbook of human factors and ergonomics, 2nd ed., G. Salvendy, Ed. New York: Wiley, pp. 1926-1943.

[19] Satchell, P. (1998). Innovation and automation, Aldershot, UK: Ashgate.

[20] Sheridan, T. (1992). Telerobotics, automation, and supervisory control. Cambridge, MA: MIT Press.

[21] Wickens, C., Mavor, A., Parasuraman, R. and McGee, J. (1998). The future of air traffic control: Human operators and automation. Washington DC: National Academy Press.

[22] Wiener, E. and Curry, R. (1980). "Flight-deck automation: Promises and problems," Ergonomics, Vol. 23, pp. 995-1011.

[23] Endsley, M. and Kiris, E. (1995). "The out-of-the-loop performance problem and level of control in automation," Human Factors, Vol. 37, pp. 381-394.

[24] Kaber, D., Omal, E. and Endsley, M. (1999). "Level of automation effects on telerobot performance and human operator situation awareness and subjective workload". In: Automation Technology and Human Performance: Current research and Trends, M. Mouloua and R. Parasuraman, Eds., Mahwah, NJ; Erlbaum, pp. 165-170.

[25] Wickens, C. (1994). "Designing for situation awareness and trust in automation". In: Proc. IFAC Conf. on Int. Syst. Eng.. Baden-Baden, Germany, pp. 174-179.

[26] Sarter, N. and Woods, D. (1995). "How in the world did we ever get into that mode? Mode error and awareness in supervisory control," Human Factors, Vol. 37(1), pp. 5-19.

[27] Sarter, N. and Woods, D. (1994). "Pilot interaction with cockpit automation II: An experimental study of pilots' model and awareness of the flight management system," Int. Jour. of Av. Psych., Vol. 4, pp. 1-28.

[28] Sarter, N., Woods, D. and Billings, C. (1997). "Automation Surprises". In: Hndbk. of Hum. Fact. and Erg., 2nd ed., G. Salvendy, Ed., New York: Wiley, pp. 1926-1943.

[29] Parasuraman, R. and Byrne, E. (2002). "Automation and human performance in aviation". In: Principles and practice of aviation psychology, P. Tsang and M. Vidulich, Eds., Mahwah, NJ: Erlbaum.

[30] Lee, J. and Moray, N. (1992). "Trust, control strategies, and allocation of function in human-machine systems," Ergonomics, Vol. 35, pp. 1243-1270.

[31] Lee, J. and Moray, N. (1994). "Trust, self-confidence, and operators' adaptation to automation," Int. Jour. Hum.-Comp. Studies, Vol. 40, pp. 153-184.

[32] Parasuraman, R., Molloy, R. and Singh, I. (1993). "Performance consequences of automation-induced 'complacency,'" Int. Jour. of Av. Psych, Vol. 3, pp. 1-23.

[33] Riley, V. (1994). Human Use of Automation, unpublished dissertation, University of Minnesota.

[34] Metzger, U. and Parasuraman, R. (2001). "The role of the air traffic controller in future air traffic management: An empirical study of active control versus passive monitoring," Human Factors, Vol. 43, pp. 519-528.

[35] Mosier, K. and Skitka, L. (1996). "Human decision makers and automated decision aids: Made for each other?" In: R. Parasuraman and M. Mouloua (Eds.), Automation and Human Performance: Theory and Applications. Mahwah, NJ: Erlbaum, pp. 201-220.

[36] Weiner, E. (1988). "Cockpit Automation". In: E. Weiner and D. Nagel, Eds., Human Factors in Aviation, San Diego: Academic, pp. 433-461.

[37] Kirlik, A. (1993). "Modeling strategic behavior in human-automation interaction: Why an 'aid' can (and should) go unused," Human Factors, Vol. 35, pp. 221-242.

[38] Layton, C., Smith, P. and McCoy, E. (1994). "Design of a cooperative problem solving system for en-route flight planning: An empirical evaluation," Human Factors, Vol. 36(1), pp. 94-119.

[39] Miller, C. (1999). "Bridging the Information Transfer Gap: Measuring Goodness of Information Fit," Jour. Visual Lang. and Comp., Vol. 10, pp. 523-558.

[40] Colucci, F. (1995). "Rotorcraft Pilot's Associate Update: The Army's Largest Science and Technology Program," Vertiflite, March/April, pp. 16-20.

[41] Vicente, K. (1999). Cognitive work analysis: Towards safe, productive, and healthy computer-based work. Mahwah, NJ; Erlbaum.

[42] Bisantz, A., Cohen, S., Gravelle, M. and Wilson, K. (1996). "To cook or not to cook: A case study of decision aiding in quick-service restaurant environments," Georgia Institute of Technology, College of Computing, Atlanta, GA, Report No. GIT-CS-96/03.

[43] Karasek, R. and Theorell, T. (1990). Healthy work: Stress, productivity, and the reconstruction of working life. New York; Basic Books.

[44] Moray, N. (1979). Mental workload. New York: Plenum.

[45] Parasuraman, R. and Hancock, P. (2001). Adaptive control of workload. In: P.A. Hancock and P.E. Desmond (Eds.) Stress, workload, and fatigue. Mahwah, NJ: Erlbaum, pp. 305-320.

[46] Wickens, C.D. and Holland, J.G. (1999). Engineering psychology and human performance. New York: Longman.

[47] Opperman, R. (1994). Adaptive User Support. Hillsdale, NJ; Erlbaum.

[48] Scerbo, M. (1996). "Theoretical perspectives on adaptive automation". In: Automation & human performance: Theory & applications, R. Parasuraman and M. Mouloua, Eds. Mahwah, NJ: Erlbaum, pp. 37-63.

[49] Maes, P. (1994). "Agents that Reduce Work and Information Overload," Comm. of ACM, Vol. 37(7), pp. 31-40.

[50] Shneiderman, B. (1997). "Direct manipulation for comprehensible, predictable, and controllable user interfaces," Proc. of the ACM Int'l Wkshp on Intelligent User Interfaces, New York, NY, pp. 33-39.

[51] Rouse, W. (1988). "Adaptive automation for human/computer control," Human Factors, Vol. 30, pp. 431-488.

[52] Banks, S. and Lizza, C. (1991). "Pilot's Associate; A Cooperative Knowledge-Based System Application," IEEE Expert, Vol. 6(3), pp. 18-29.

[53] Morrison, J. (1993). "The adaptive function allocation for intelligent cockpits (AFAIC) program: Interim research and guidelines for the application of adaptive automation," Naval Air Warfare Center Tech. Rep. #NAWCADWAR-93031-60.

[54] Parasuraman, R., Mouloua, M. and Molloy, R. (1996). "Effects of adaptive task allocation on monitoring of automated systems," Human Factors, Vol. 38, pp. 665-679.

[55] Scallen, S., Hancock, P. and Duley, J. (1995). "Pilot performance and preference for short cycles of automation in adaptive function allocation," Ap. Ergon., Vol. 26, pp. 397-403.

[56] Miller, C. and Hannen, M. (1999). "The Rotorcraft Pilot's Associate: Design and evaluation of an intelligent user interface for cockpit information management," Knowledge Based Systems, Vol. 12, pp. 443-456.

[57] Rouse, W. (1976). "Adaptive allocation of decision making responsibility between supervisor and computer". In: Monitoring behavior and supervisory control, T.B. Sheridan and G. Johannsen, Eds. New York: Plenum Press, pp. 221-230.

[58] Hancock, P., Chignell, M. and Lowenthal, A. (1985). "An adaptive human-machine system," Proc. IEEE Conf. on Syst., Man and Cyber., Vol. 15, pp. 627-629.

[59] Parasuraman, R., Bahri, T., Deaton, J., Morrison, J. and Barnes, M. (1992). Theory and design of adaptive automation in aviation systems. (Progress Report No. NAWCADWAR-92033-60). Warminster, PA: Naval Air Warfare Center.

[60] Duley, J. and Parasuraman, R. (1999). "Adaptive information management in future air traffic control". In: Automation technology and human performance: Current research and trends, M. Scerbo and M. Mouloua, Eds., Mahwah, NJ: Erlbaum, pp. 86-90.

[61] Hilburn, B., Jorna, P., Byrne, E. and Parasuraman, R. (1997). "The effect of adaptive air traffic control (ATC) decision aiding on controller mental workload". In: Human-automation interaction, M. Mouloua and J. Koonce, Eds., Mahwah, NJ: Erlbaum, pp. 84-91.

[62] Kaber, D. and Riley, J. (1999). "Adaptive automation of a dynamic control task based on workload assessment through a secondary monitoring task". In: Automation technology and human performance: Current research and trends, M. Scerbo and M. Mouloua, Eds., Mahwah, NJ: Erlbaum, pp. 129-133.

[63] Parasuraman, R. (1993). "Effects of adaptive function allocation on human performance". In: Human factors and advanced aviation technologies, D.J. Garland and J.A. Wise, Eds., Daytona Beach: Embry-Riddle Aeronautical University Press, pp. 147-157.

[64] Dornheim, M. (1999). "Apache tests power of new cockpit tool," Av.Week & Space Tech., October 18, pp. 46-49.

[65] Bright, J. (1958). "Does automation raise skill requirements?". Harvard Business Review, Vol. 36, July-August, pp. 85-98.

[66] Sheridan, T. and Verplank, W. (1978). "Human and Computer control of undersea teleoperators," MIT Man-Machine Systems Laboratory, Cambridge, MA. Technical Report.

[67] Mitchell, C. (1987). "GT-MSOCC: A domain for research on HCI and decision aiding in supervisory control systems," IEEE Trans. on Sys., Man and Cybern., Vol. 17(4), pp. 553-572.

[68] Card, S., Moran, T. and Newell, A. (1983). The Psychology of Human Computer Interaction, Mahwah, NJ; Lawrence Erlbaum Associates.

[69] Geddes, N. (1989). Understanding human operators' intentions in complex systems. Ph.D. Dissertation, Georgia Institute of Technology.

[70] Martinich, J. (1996). Production and Operations Management: An Applied Modern Approach, New York; John Wiley.

[71] Murata, T. (1989). "Petri Nets: Properties, Analysis and Applications," Proceedings of the IEEE, Vol. 77(4), pp. 541-580.

[72] Diaper, D. (1989). Task Analysis for Human-Computer Interaction, Chichester, U, Ellis Horwood.

[73] Shattuck, L. (1995). Communication of Intent in Distributed Supervisory Control Systems. Unpublished dissertation. The Ohio State University, Columbus, OH.

[74] Klein, G. (1998). Sources of Power; How People Make Decisions. Cambridge, MA; MIT Press.

[75] Jones, P. and Jasek, C. (1997). "Intelligent Support for Activity Management (ISAM): An architecture to support distributed supervisory control," IEEE Trans. on Sys. Man & Cyber. – Pt. A: Sys & Hum, Vol. 27, pp. 274-288.

[76] Rasmussen, J. and Goodstein, P. (1987). "Decision support in supervisory control of high-risk industrial systems," Automatica, Vol. 23, pp. 663-671.

[77] Pande, A. and Hayes, C. (2002). "Plan-Edit: An Interactive Critic for User Guided Generation of High Quality Manufacturing Setup Sequences", Submitted for inclusion in Proc. of ASME Int'l Mftg. Cong. & Exposition, Montreal, Canada.

7.6 DELEGATION ARCHITECTURES: PLAYBOOKS AND POLICY FOR KEEPING OPERATORS IN CHARGE

We argue that as Unmanned Military Vehicles become more intelligent and capable, and as we attempt to control more of them with fewer humans in the loop, we need to move toward a model of delegation of control rather than the direct control that characterizes current practice. We identify and describe five delegation methods which can serve as building blocks from which to compose complex and sensitive delegation systems: delegation through (1) providing *goals*, (2) providing full or partial *plans*, (3) providing *negative constraints*, (4) providing *positive constraints* or *stipulations*, and (5) providing priorities or value statements in the form of a *policy*. We then describe two implemented delegation architectures that illustrate the use of some of these delegation methods: a "playbook" interface for UAV mission planning and a "policy" interface for optimizing the use of battlefield communications resources.

7.6.1 UMV Control as Human-Automation Delegation

While Unmanned Military Vehicles (UMVs) hold the promise of radical change and improvement for a wide range of military applications they also pose a host of challenging problems. Chief among these is how to enable a human operator, who may well be heavily engaged in tasks of his or her own, to retain sufficient control over the UMV(s) to ensure safe, efficient and productive outcomes. This problem is, of course, magnified when the UMVs may be responsible for the lives of many soldiers or civilians, may be capable of unleashing lethal force on its own, and when a single human may be striving to control groups or even swarms of potentially autonomous and independent actors and may be concurrently engaged in other, high tempo and criticality tasks of his or her own.

Yet this problem is not completely novel. Humans have been striving to retain control and produce efficient outcomes via the behavior of other autonomous agents for millennia. It just so happens that those "agents" have been other humans. Not surprisingly, we have developed many useful methods for accomplishing these goals, each customized to a different domain or context of use. When we have some degree of managerial authority over another human actor and yet will not be directly commanding performance of every aspect of a task, we call the relationship (and the method of commanding task performance) *delegation*. Delegation allows the supervisor to set the agenda either broadly or specifically, but leaves some authority to the subordinate to decide exactly how to achieve the commands supplied by the supervisor. Thus, a delegation relationship between supervisor and subordinate has many requirements:

1) The supervisor retains overall responsibility for the outcome of work undertaken by the supervisor/ subordinate team and retains the authority commensurate with that responsibility.
2) The supervisor has the capability to interact very flexibly and at multiple levels with the subordinate. When and if the supervisor wishes to provide detailed instructions, s/he can; when s/he wishes to provide only loose guidelines and leave detailed decision making up to the subordinate, s/he can do that as well – *within the constraints of the capabilities of the subordinate*.
3) To provide useful assistance within the work domain, the subordinate must have substantial knowledge about and capabilities within the domain. The greater these are, the greater the potential for the supervisor to offload tasks (including higher level decision making tasks) on the subordinate.
4) The supervisor must be aware of the subordinate's capabilities and limitations and must either not task the subordinate beyond his/her abilities or must provide more explicit instructions and oversight when there is doubt about those abilities.

5) There must be a "language" or representation available for the supervisor to task and instruct the subordinate. This language must (a) be easy to use, (b) be adaptable to a variety of time and situational contexts, (c) afford discussing tasks, goals and constraints (as well as world and equipment states) directly (as first order objects), and (d) most importantly, be shared by both the supervisor and the subordinate(s).
6) The act of delegation will itself define a window or space of control authority within which the subordinate may act. This authority need not be complete (e.g., checking in with the supervisor before proceeding with specific actions or using some resources may be required), but the greater the authority, the greater the workload reduction on the supervisor.

Items 4 and 6 together imply that the space of control authority delegated to automation is flexible – that the supervisor can choose to delegate more or less "space," and more or less authority within that space (that is, range of control options), to automation. Item 5 implies that the language available for delegation must make the task of delegating feasible and robust – enabling, for example, the provision of detailed instructions on how the supervisor wants a task to be performed or a simple statement of the desired goal outcome.

7.6.2 Types of Delegation

We have developed a variety of architectures within which to support human delegation interactions with automation. Of particular interest as a core enabling technology is the "language" or representation for delegation described in item #5 above. As Klein points out [1], without successfully sharing an understanding of the tasks, goals and objectives in a work domain, there can be no successful communication of intent between actors. We believe there are five kinds of delegation actions or delegation methods that should be supported within such a representation, as described in the Figure 7-11 below. Note that each method forms a building block and they can be combined into more effective and flexible composite delegation interactions. Note also that the subordinate has a specific responsibility in response to each method, as articulated in the table below:

Table 7-6: Supported Delegation Actions and Methods

Supervisor's Delegation Action	Subordinate's Responsibility
1. Stipulation of a goal to be achieved – where a goal is a desired (partial) state of the world.	Achieve the goal(s) if possible (via any means available), or report if incapable.
2. Stipulation of a plan to be performed – where a plan is a series of actions, perhaps with sequential or world state dependencies.	Follow the plan if possible (regardless of out-come) or report if incapable.
3. Provide constraints in the form of actions or states to be avoided.	Avoid those states or actions if possible, report if not.
4. Provide "stipulations" in the form of actions or states (i.e., sub-goals) to be achieved.	Achieve those states or perform those actions if possible, report if not.
5. Provide an "optimization function" or "policy" that enables the subordinate to make informed decisions about the desirability of various states and actions.	Work to optimize value within the "optimization function" or "policy".

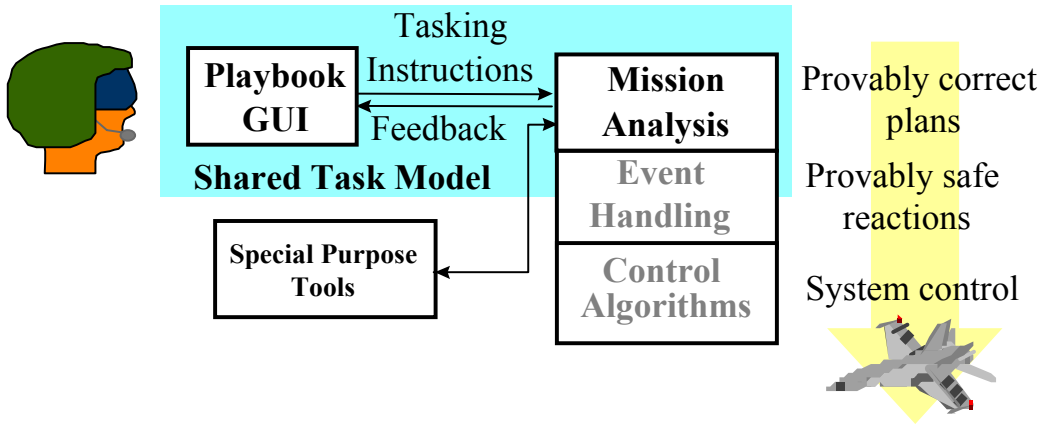

Figure 7-11: General Playbook Architecture.

In the remainder of this section, I will describe two delegation architectures we are developing. While neither system enables all of the types of delegation described above, and neither is fully implemented yet, collectively they illustrate the five types of delegation and provide a rich and highly flexible set of interactions for human-automation delegation.

7.6.2.1 Playbook – Delegation of Goals, Plans and Constraints

The first architecture is based on the metaphor of a sports team's playbook. A playbook works because it provides for rapid communication about goals and plans between a supervisor (e.g., a coach) and a group of intelligent actors (the players) who are given the authority to determine how to act within the constraints inherent in the coach's play. Our Playbook architecture supports delegation action types 1 – 4 in principle and has been implemented in prior prototypes to include action types 2 and 4.

The basic Playbook system architecture is presented in Figure 7-8. The Playbook 'proper' consists of a User Interface (UI) and a constraint propagation planner known as the Mission Analysis Component (MAC) that communicate with each other and with the operator via a Shared Task Model. The operator communicates instructions in the form of desired goals, tasks, partial plans or constraints, via the UI, using the task structures of the shared task model. The MAC is an automated planning system that understands these instructions and (a) evaluates them for feasibility and/or (b) expands them to produce fully executable plans. The MAC may draw on special purpose planning tools (e.g., an optimizing path planner) to perform these functions, wrapping them in its task-sensitive environment. Outside of the tasking interface, but essential to its use, are two additional components. An Event Handling component, itself a reactive planning system capable of making momentary adjustments during execution, takes plans from the Playbook. These instructions are sent to control algorithms that actually effect behaviors.

Operator interaction with the Playbook can be via a variety of user interfaces customized to the needs of the domain and work environment, but operator commands are ultimately interpreted in terms of the Shared Task Model. To date, we have developed prototype playbooks for UCAV teams [2], and Tactical Mobile Robots [3], and are currently developing prototypes for the RoboFlag game [4] and for real-time interaction with teams of heterogeneous UMVs. Below, we provide a description of user interaction with one Playbook interface we developed with Honeywell Laboratories to illustrate the general concept.

We developed the playbook illustrated in Figure 7-12 to enable a human leader to create a full or partial mission plan for UCAVs. This initial work was intended as a ground-based tasking interface to be used for a priori mission planning, but current playbook work is exploring interface modifications to enable real-time and in-flight tasking and task performance monitoring as well.

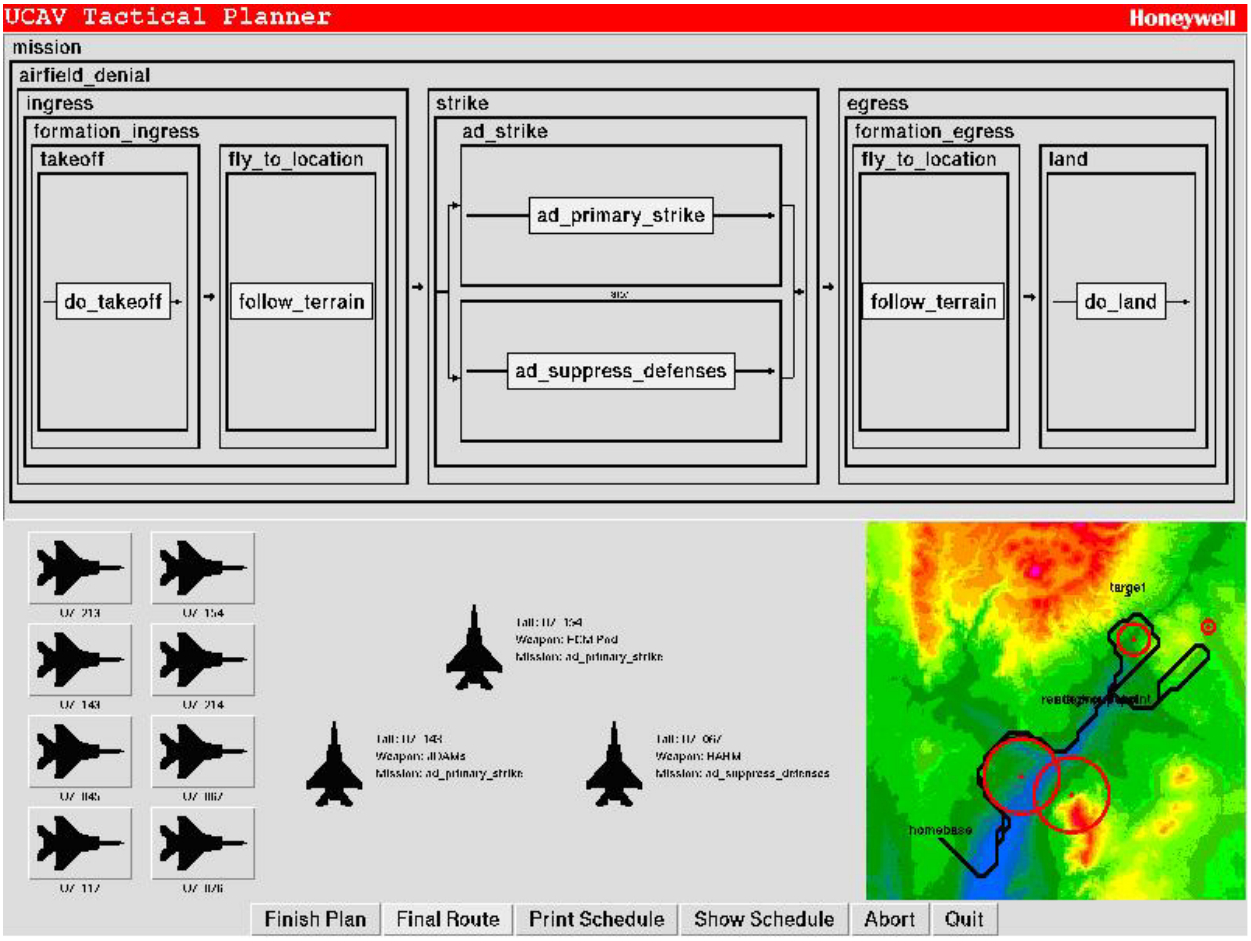

Figure 7-12: Prototype Playbook Interface for UCAV Mission Planning.

Figure 7-12 shows five primary regions of this Playbook UI. The upper half of the screen is a Mission Composition Space that shows the plan composed thus far. In this area, the operator can directly manipulate the tasks and constraints in the plan. The lower left corner of the interface is an Available Resource Space, currently presenting the set of aircraft available for use. The lower right corner contains an interactive Terrain Map of the area of interest, used to facilitate interactions with significant geographic information content. The space between these two lower windows (empty at startup) is a Resource in Use Space – once resources (e.g., UCAVs, munitions, etc.) are selected for use, they will be moved here where they can be interacted with in more detail. Finally, the lower set of control buttons is always present for interaction. This includes options such as "Finish Plan" for handing the partial plan off to the MAC for completion and/or review and "Show Schedule" for obtaining a Gantt chart timeline of the activities planned for each actor, etc.

At startup, the Mission Composition Space presents the three top-level plays (or 'mission types') the system currently knows about: Interdiction, Airfield Denial, and Suppress Enemy Air Defences (SEAD). The mission leader would interact with the playbook to, first; declare that the overall mission "play" for the day was, say, "Airfield Denial." In principle, the user could define a new top-level play either by reference to existing play structures or completely from scratch, but this capability has not been implemented yet.

This action is an example of type 2 delegation – providing a specific task for subordinates to perform – but because this is a very high level task in a hierarchical task network, the supervisor has left a great deal of freedom to the subordinates (in this case, the MAC and the UAVs themselves) to determine exactly how an "Airfield Denial" mission is to be performed. If this were the only delegation information the supervisor provided, the subordinates would be obligated to do their best to perform that action (an Airfield Denial mission), but would have a great deal of authority as to how best to accomplish it.

At this point, having been told only that the task for the day is "Airfield Denial," a team of trained pilots would have a very good general picture of the mission they would fly. Similarly, the tasking interface (via the Shared Task Model) knows that a typical airfield denial plan consists of ingress, attack and egress phases and that it may also contain a suppress air defence task before or in parallel with the attack task – but just as a leader instructing a human flight team could not leave the delegation instructions at a simple 'Let's do an Airfield Denial mission today,' so the operator of the tasking interface is required to provide more information. Here, the human must provide four additional items: a target, a homebase, a staging and a rendezvous point. Each of these is a *stipulation,* or positive constraint, telling the subordinates that whatever specific plan they come up with to accomplish the higher level mission must include these attributes – and thus, they are examples of type 4 delegation interactions. Most of these activities are geographical in nature and users typically find it easier to perform them with reference to a terrain map. Hence, by selecting any of them from the pop up menu, the user enables direct interaction with the Terrain Map to designate an appropriate point. Since the Playbook knows what task and parameter the point is meant to indicate, appropriate semantics are preserved between user and system. As for all plans, the specific aircraft to be used may be selected by the user or left to the MAC. If the user wishes to make the selection, s/he views available aircraft in the Available Resource Space and chooses them by clicking and moving them to the Resources in Use Area.

The mission leader working with a team of human pilots could, if time, mission complexity or degree of trust made it desirable, hand the mission planning task off to the team members at this point. The playbook operator can do this as well, handing the task to the MAC via the "Finish Plan" button. The leader might wish, however, to provide substantially more detailed delegation instructions. S/he can do this by progressively interacting with the playbook UI to provide deeper layers of task selection, or to impose more stipulations on the resources to be used, waypoints to be flown, etc. For example, clicking on "Airfield Denial" produces a pop-up menu with options for the user to tell the MAC to "Plan this Task" (that is, develop a plan to accomplish it) or indicate that s/he will `Choose airfield denial' as a task that s/he will flesh out further. The pop-up menu also contains a context-sensitive list of optional subtasks that the operator can choose to include under this task. This list is generated by the MAC with reference to the existing play structures in the play library, filtered for current feasibility.

After the user chooses 'Airfield Denial' the system knows, via the Shared Task Model, that this task must include an Ingress subtask (as illustrated in Figure 7-12). The supervisor does not have to tell intelligent subordinates this; it is a part of their shared knowledge of what an 'Airfield Denial' task means – and how it must be performed. To provide detailed instructions about how to perform the Ingress task, however, the user can choose it, producing a "generic" Ingress task template or "play". This is not a default method of doing "Ingress", but a generic, uninstantiated template – corresponding to what a human expert knows about what

constitutes an Ingress task and how it can or should be performed. A trained pilot knows that Ingress can be done either in formation or in dispersed mode and, in either case, must involve a "Take Off" subtask followed by one or more "Fly to Location" subtasks. Similarly, the user can select from available options (e.g., formation vs. dispersed Ingress, altitude constraints on takeoff, etc.) on context-sensitive, MAC-generated menus appropriate to each level of decomposition of the task model. One of our current challenges in creating Playbooks for real-time interactions is to enable them to be sensitive to the current state of affairs and of task performance so as to make intelligent assumptions about task performance possible – for example, if the supervisor wishes to command a currently airborne UAV, perhaps in a holding pattern, to perform an 'Airfield Denial' mission, both supervisor and subordinate should know that the Takeoff portion of an Ingress task is no longer necessary and should either be eliminated or be shown as already accomplished.

The user can continue to specify and instantiate tasks down to the "primitive" level where the sub-tasks are behaviors the control algorithms (see Figure 7-11) on board the aircraft can be relied upon to execute in flight. Alternatively, at any point after the initial selection of the top level mission task and its required parameters, the supervisor can hand the partly developed plan over to the MAC for completion and/or review. In extreme cases, a viable "Airfield Denial" plan for multiple aircraft could be created in our prototype with as few as five selections and more sophisticated planning capabilities could readily reduce this number further – but potentially more important, the operator (like a human supervisor dealing with intelligent subordinates) can also provide more detailed instructions whenever s/he deems them necessary or useful to performing the mission successfully and in the way s/he sees fit.

This Playbook illustrates delegation interactions 2 and 4 (plans and stipulations). The subordinates' role in these types of interaction are described in the table above – to perform the plan through any set of sub-methods that adhere to the stipulations provided by the supervisor, or to report that this is infeasible. One of the MAC's roles in the above example is to report when it is incapable of developing a viable plan within the constraints imposed, (e.g., if the user has stipulated distant targets that exceed aircraft fuel supplies). In a real-time delegation system, the MAC will be responsible for continual monitoring of performance to report when world states mean that plan performance is no longer capable of (or likely to) accomplish the user's parent plan (e.g., because of equipment failures, adverse head winds, enemy countermeasures, etc.)

The Playbook architecture is, we believe, also capable of supporting delegation interaction types 1 and 3 (goals and negative constraints) as well. Supporting goal-based delegation interactions would require a slight modification to the shared task representation. Currently, we have used a representation that explicitly includes only hierarchically organized and sequenced tasks (i.e., actions to be performed). Tasks implicitly encode the goals they accomplish, but there are representations (such as Geddes Plan-Goal Graphs [5]) that explicitly interleave both plans and goals and a linked hierarchy. Use of such a representation, along with related modifications to the UI and MAC, would enable the supervisor to say, effectively, "Today we're going to achieve a State" (e.g., the destruction of a given airfield) rather than or in addition to, the plan-based representation used above which allows only the issuing of task-based delegation commands (e.g., "Today we're going to fly an airfield-denial mission"). The incorporation of negative constraints into the interaction (delegation interaction method #3), would require a less substantial modification to the Playbook architecture – potentially requiring only a UI addition to enable the supervisor to incorporate negative commands about task types and state parameters (e.g., "do NOT fly through this valley or use this type of munition") and then requiring the MAC to create plans which avoid those negative constraints.

7.6.2.2 Policy – Delegation via Abstract Value Statements

The final type of delegation interaction offers the ability to provide priorities between alternate goals and states and to do so more abstractly than the above methods. Sometimes supervisors don't have a single,

concrete world state goal in mind, much less a specific plan for accomplishing it. Sometimes supervisors must issue commands well in advance to cover a wide range of largely unanticipatable circumstances. In these cases, the delegation instructions will be less a specific statement of actions to take or world states to be sought or avoided, but rather a general statement of outcomes that would be more or less good or valuable (or, conversely, bad or to be avoided) than others. We refer to the set of such abstract value statements that a supervisor might provide as his or her "policy" for performance in the domain.

We have developed policy-based architectures for two applications: providing commanders' guidance to a resource controller for battlefield network communications [6], and providing visualization and feedback to dispatchers in upset contexts in commercial aviation [7]. We will describe the first of these below.

A *policy statement* is an abstract, general, a priori statement of the relative importance or value of a goal state in the domain. In its simplest form, policy provides a method for human operators to mathematically define what constitutes "goodness" in terms of the outcomes of the delegation. Once defined, a policy statement can be treated as a rule and evaluated against a current or hypothetical context – if the rule is true in the context, then the context incurs the "goodness" (or badness) value stipulated by the rule. Alternate contexts (which could be tied to the expected outcomes of alternate decisions) can then be evaluated against each other by examining the set of policy rules that are satisfied or violated and the resulting set of goodness/badness values accrued. A set of individual policy statements can be bundled together, and these *policy bundles* can be used to flexibly define the priorities that apply in a given situation (priorities can change given different circumstances).

A policy-based delegation system requires at least three components:

1) A representation for specifying the "policy" in terms of the value of various partial world states;
2) A user interface for allowing one or more users to input their policies and, if desired, view results of policy application;
3) A computational framework that allows evaluation of a current situation or hypothetical proposed situation against the expressed policy; or
4) An engine that allows application of the policy either to the control of resource application or to a visualization of sensed data about a current situation or projected data about a future or simulated situation.

In the development of a policy-based delegation system for communications resource usage, we were striving to provide a means for commanders to tell an automated network management system their "policies" for how to prioritize the use of communications bandwidth in order to satisfy the most important requests most fully. Note, however, that "most important" was not a static concept, but rather changed across commanders and situations. For this application, we developed a policy representation that allowed commanders to assign, a priori and abstractly, a value to various kinds of communications requests. As communications requests then came in from various field units or operators, they could be matched against the commander's policy statements and a value assigned to each of them. This value was then used by a resource optimizing controller to determine which requests should get network bandwidth with what priority.

This process is conceptually illustrated in Figure 7-13. Each commander's policy is created as a set of statements each of which assigns an importance (or value) function to a defined sub-region in a multidimensional space. Regions might be based on a single dimension ('Requests for weather information [Content] get Importance 0.2') or on a combination of dimensions ('Requests owned by the Zone Reconnaissance task [Owner] for weather information [Content] from Satellite 476B [Source] to 3rd Air Calvary Division [Destination]

get Importance 0.8). If the policy element regions are allowed to overlap, then they must be sequenced (typically from most to least specific) to indicate the order of precedence. In practice, the commander's policy is then used to assign an importance value to any incoming request for communication resources (illustrated conceptually in Figure 7-13). Each incoming request is matched against the sequenced series of policy element statements the commander has made. The first policy element that matches the request determines the importance of that request and informs an automated resource manager about the relative value of satisfying that request.

	Owner	Source	Destination	Content	Importance	
(	{W1}	{S1}	{D1}	{C1}	.9	)
(	{W2}	{S2}	{D2}	{C2}	.8	)
(	{W3}	{S3}	{D3}	{C3}	.5	)
(	*	*	*	*	.1	)

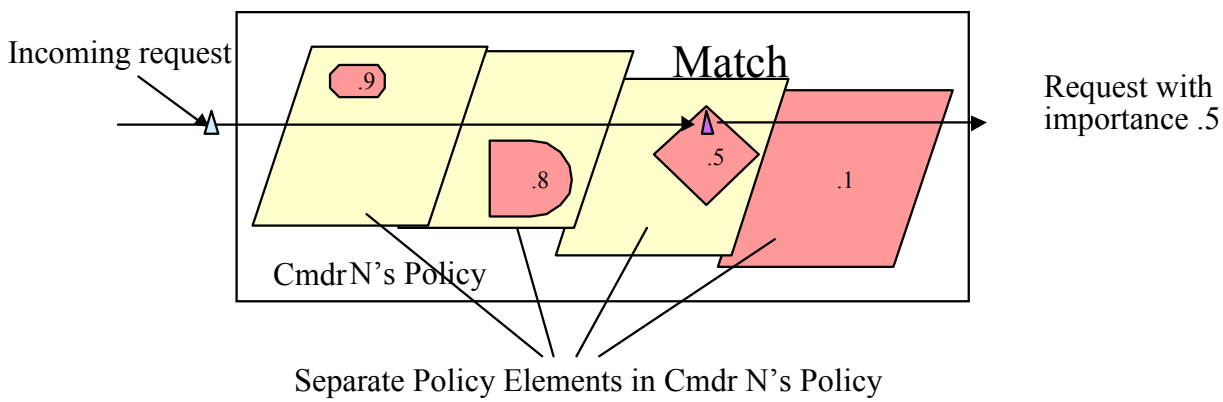

Figure 7-13: Representation of a Policy for Network Bandwidth Prioritization.

While conceptually simple, many useful functions can be performed within this framework. First, it is not necessary that importance be construed as an all-or-nothing value as it is depicted in 7-13. Instead, we have explored more sophisticated representations that allow the requestor to provide a description of how s/he wants the information requested along several dimensions (e.g., freshness, reliability, initiation-time, accuracy, resolution, scope, etc.) Then the resource management system can treat the importance value as a maximum number of value "points" to be awarded for satisfying the request perfectly, while still awarding itself points for partial satisfaction. This permits more sensitive management of resources to be performed.

Second, it is rarely the case that a single commander or supervisor is the only one who may have an interest in dictating policy about how subordinates behave. Rather, each commander must allocate his/her resources in accordance with the policies of those above. We support this requirement (Figure 7-14) by modeling policies that exist at nodes in a command hierarchy. As requests come in, they are matched against the commander's policy that governs them, but must then also be matched against his/her commander's policy – and so on, up the chain of command. We allow each commander to stipulate how this matching policy element should be resolved with the subordinate commander's matching policy element: as a ceiling or floor value, or linear combination of the values. Even when a single well-defined chain of command does not exist the policies of

different "interest groups" may be represented with relative weights on the importance values that each would assign to a potential outcome. We used this approach in representing the many, varied interests which impact the decisions of a commercial airline's dispatch operators (e.g., crew scheduling, maintenance scheduling, marketing, passengers, finance, etc.) [7].

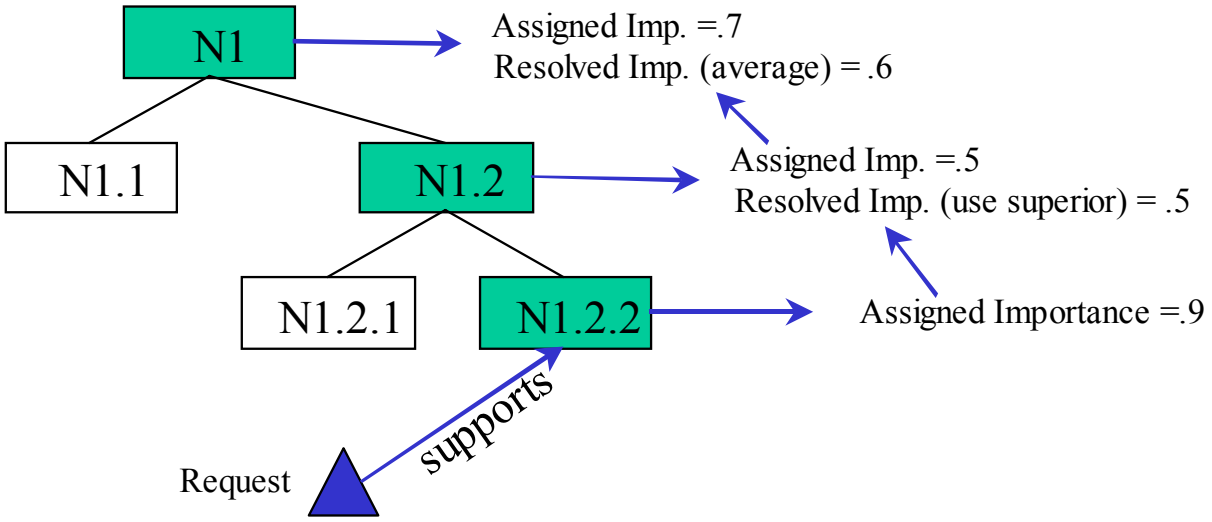

Figure 7-14: Cross Echelon Policy Application and Resolution.

Folding this policy-based form of delegation interaction (method 5) into an overall architecture that includes the other methods is not as difficult as it might first appear. While we have not yet developed a system that accomplishes this, the way forward is clear. Policy is simply an assignment of value or priority to the goal states and tasks in the other delegation interaction types. Priorities for resource usage and the desirability of various outcomes stem, after all, from a superior's goals and plans for subordinates (whether human or machine). If, for example, I task a given unit under my command to perform an Airfield Denial task, and I know that their task is the most important of all concurrent tasks to me, then I have effectively said that giving them the resources they require to perform that task (specific aircraft, munitions, fuel, communications bandwidth, etc.) represents the highest value to me. In other words, delegation interactions that provide specific goals, plans, stipulations and constraints to subordinates carry with them specific policy implications. Whenever a commander can provide more specific delegation instructions, this will generally get him/her closer to the results desired from his/her subordinates, but this will not always be the case. Hence, the ability to stipulate more abstract policies should probably be preserved in a complete delegation system as a means of covering unexpected and unfamiliar situations.

7.6.3 Conclusions and Future Work

While the work described above represents a general framework for delegation interactions suitable for human interaction with smart automation of various kinds and, perhaps uniquely, suitable for the tasking of multiple UMVs, our work has thus far progressed only to the proof of concept stage. As noted above, we have currently implemented only portions of the various methods of delegation that a fully flexibly delegation interface might benefit from, and have done so in disparate systems. Furthermore, our proof of concept implementations have not yet afforded us the opportunity to do rigorous human in the loop evaluations to demonstrate improved performance, if any.

These situations are changing, however. We are currently engaged in exploration of human interaction with Playbook-like interfaces with Parasuraman and Galster [4] and are beginning work on a Playbook interface for real-time interactions with heterogeneous UMV assets by operators who may be concurrently involved in other critical tasks (under a DARPA-IXO SBIR grant). One of the goals of this work will be to develop task libraries and task construction tools and interface concepts to move the delegation interface work along toward implementation and utility.

Of course, anyone who has worked with a poorly trained, or simply mismatched, subordinate is well aware that it is possible for delegation to cause more work than it saves. Our challenge, and that of others who adopt a delegation framework for human interaction with complex and largely autonomous automation, will be to ensure that this does not happen – through judicious use of technology and substantial usability analysis and testing. On the positive side, however, we benefit from the knowledge that delegation approaches to interaction with intelligent yet subordinate actors have worked repeatedly throughout history and, particularly, the history of warfare. As automation in the form of UMVs increasingly takes its place as one of those actors we want to be intelligent, capable and effective yet remain subordinate, we will increasingly need methods for enabling *it* to interact with *us* in the ways that we trust and are familiar with. Since delegation is the primary method we have evolved to meet these requirements, it only makes sense to pursue delegation approaches to human interaction with automation.

7.6.4 Acknowledgements

Many people have contributed to the ideas presented above. A partial list includes Robert Goldman, Harry Funk, Michael Pelican, Dave Musliner, Karen Haigh, Michael Dorneich, Stephen Whitlow, John Allen, and Jim Richardson.

7.6.5 References

[1] Klein, G. (1998). Sources of Power: How People Make Decisions. Cambridge, MA; MIT Press.

[2] Miller, C., Pelican, M. and Goldman, R. (2000). “Tasking” Interfaces for Flexible Interaction with Automation: Keeping the Operator in Control. In: Proceedings of the Conference on Human Interaction with Complex Systems. Urbana-Champaign, Ill. May.

[3] Goldman, R.P., Haigh, K., Musliner, D. and Pelican, M. (2000). MACBeth; A Multi-Agent, Constraint-based Planner. In: Notes of the AAAI Workshop on Constraints and AI Planning, Austin, TX, pp. 1-7.

[4] Parasuraman, R., Galster, S. and Miller, C. (2003). Human Control of Multiple Robots in the RoboFlag Simulation Environment. To appear in Proceedings of the 2003 Meeting of the IEEE Systems, Man and Cybernetics society. October 5-8; Washington, DC.

[5] Sewell, D. and Geddes, N. (1990). A plan and goal based method for computer-human system design. In: D. Diaper (Ed.) Human-Computer Interaction – INTERACT ‘90. Elsevier Science; North-Holland. pp. 283-288.

[6] Funk, H., Miller, C., Johnson, C. and Richardson, J. (2000). Applying Intent-Sensitive Policy to Automated Resource Allocation: Command, Communication and Most Importantly, Control. In: Proceedings of the Conference on Human Interaction with Complex Systems. Urbana-Champaign, Ill. May.

[7] Dorneich, M.C., Whitlow, S.D., Miller, C.A. and Allen, J.A. (2004). "A Superior Tool for Airline Operations," Ergonomics in Design, 12(2). 18-23. Dorneich, M.C., Whitlow, S.D., Miller, C.A. and Allen, J.A. (Submitted). A Decision-Support Tool for Airline Operation's Diversion Management. Submitted to Ergonomics in Design.

7.7 MODELLING MULTI-LAYERED CONTROL: APPLICATION OF THE EXTENDED CONTROL MODEL TO THE ANALYSIS OF UAV SCENARIOS

Automation has often been approached from the bottom up, starting with the system components. This has focused the issues on the merits of humans and machines either as a comparison of attributes and capabilities, or in terms of relative roles as in levels of automation. An alternative is to approach the problem from the top down, using the requirements to joint system performance as a starting point. In this approach the emphasis is on being in control. The controlling system must match the requisite variety of processes involving subsystems with different dynamics, degrees of complexity, and predictability. This requires multiple, simultaneous layers of control. A multi-layered control model provides a good basis for understanding the consequences of automation and the needs of various types of information to support views of the past, present, and future. The top down approach is illustrated by describing the four layers needed to control a vehicle.

The objective of this note is to demonstrate how an UAV scenario can be described using the principles of the Extended Control Model (ECOM). This in turn requires a discussion of the Autonomous Control Level (ACL) framework, which provides a common reference for work on UAVs relating to issues of automation and human factors. This report comprises three parts: a brief introduction to the ECOM, a discussion of the ACL framework, and a description of an UAV scenario using the ECOM as a frame of reference.

7.7.1 Introduction

This chapter considers how to approach the problem of controlling one or more UAVs[2] to accomplish a given mission. This is a problem that is found in military as well as civilian domains, although it is the former that provides the context here. The issue is basically how to define and achieve an effective balance between manual and automatic control, i.e., between having an operator guide the vehicle and having the vehicle guide itself. This issue brings to the fore many of the problems that have been dealt with in the field of industrial automation, but also adds new perspectives and demands.

The chapter first presents a general framework for how to describe this problem, based on the principles of cognitive systems engineering. It then illustrates the principles of the description by considering an example taken from the military domain. The emphasis of the chapter is to present a methodological approach that can be used to make the problems tractable and, hopefully, solvable.

7.7.1.1 Control as a Cognitive Engineering Problem

With the possible exception of a few, unique cases the practical need to control something – a process, a vehicle, or a socio-technical system – always faces two predicaments. One is that control invariably takes place in time and requires time: processes are dynamic and developments are only incompletely predictable. The other is that usually more than one activity is going on, which means that more than one line of development must be kept keep track of. Even in cases where a controller can focus on a single process,

[2] The term UAV – for uninhabited aerial (or airborne) vehicle – is used as a general reference to unmanned vehicles, including land-based and sea-based versions.

that process may have to be considered with multiple time-horizons, hence effectively as being composed of multiple and potentially asynchronous processes.[3]

The consequence of this is that it is inadequate to address the control problem by methods that use decomposition as the main principle. Any system can structurally be described as being composed of a number of parts or subsystems, which in turn may be described as composed of parts or subsystems and so on, until the level of elementary components is reached.[4] Yet while such descriptions are highly suitable for an instantiation of a system's structure, they are not necessarily the best way to describe a system's functions. Here it may be more appropriate to begin by understanding what systems do – or what they are supposed to do – rather than by describing what they are made of. Decomposition is a powerful analytical tool, which has become an almost sacred principle of Western science. Because of that it is too often taken for granted, which means that it sometimes is used when it should not have been. For instance, if we apply the notion of a human-machine system this already embraces the assumption that it is meaningful to make a distinction between humans and machines as two major groups of components. This assumption can, however, easily be questioned. It is, indeed, entirely possible to start an analysis from the notion of goals and means and use that as the guiding principle when a system description is developed. This is a well-established tradition as demonstrated by the classical work of Miller, Galanter and Pribram [1] although it often is disregarded by the mainstream of HCI and HMI.[5]

Discussions of how humans and machines can work together to the benefit of the overall human-machine system can, of course, not avoid the thorny issue of how functions should be allocated or distributed between the parts. Even if the starting point is a functional analysis, it is sooner or later necessary to consider how the resulting design should be implemented. Function allocation also implies that a certain level of automation exists, since without that machines would be incapable of functioning independently, hence to have functions assigned to them. Indeed, the word automation, which is a combination of the Greek words *auto*-, "self" and -*matos*, "willing", means something that acts by itself or of itself. In the modern usage an artefact is said to be automatic if it is self-regulating or able to act or operate in a manner that is determined by the conditions, but which is essentially independent of external control.

To illustrate the differences in how humans and machines can work together, consider an automobile from the 1960s, such as an Austin Mini or a Volkswagen. In these cars very few things were automated – indeed, even the windshield wipers had to be stopped manually in the right position. For such cars there was therefore no need to consider function allocation or to worry about the relation between humans and automation. The driver had to do everything, specifically to control speed and direction while maintaining sufficient safety margins. The situation is completely different for a modern car. At the top of the range, cars are typically equipped with systems such as cruise control – or even adaptive cruise control, anti-locking brakes, electronic stability programs, traction control, and lane departure warnings as well a electronic climate control, automatic transmission, rain sensors, etc. The driver still has to control the direction and speed of the car – or at least to indicate the desired direction and speed, since in many cases the car can then take care of the rest. In short, the situation has changed completely, and the control needed to ensure safe driving is now distributed between the driver and a number of automated systems in the vehicle.[6]

---

[3] It might be more correct to say that the operator focuses on only one goal. Any goal may be considered from different perspectives and using different criteria, hence require multiple processes to be achieved.

[4] As the story of the atom shows, the level of elementary components is never absolute, but depends on the frame of reference.

[5] The basic 'component' in this work was the Test-Operate-Test-Exit (TOTE) unit, which describes a pattern of activity rather than a structure.

[6] Although the gradual transition of control from the driver to the vehicle always is done with the best of intentions, it sooner or later leads to unexpected problems of risk and responsibility.

From a historical perspective, automation has been used either to ensure a more precise performance of a given function or to improve the stability of system performance. After the industrial revolution, a further motivation was to increase the speed of work and production. The net effect of the increasing use of technology was that humans gradually came to be seen as a bottleneck for system performance, thereby strengthening the need of further automation. It is no coincidence that human factors engineering emerged in the late 1940s, when the rapid changes brought about by the scientific and technological developments in the preceding decade had undermined the hitherto peaceful co-existence between humans and machines. These developments opened the field for a new type of engineers, branded rather appropriately by Norbert Wiener as *gadget worshippers* [2, p. 53], defined as people who "regard(ed) with impatience the limitations of mankind, and in particular the limitation consisting in man's undependability and unpredictability." In addition to being seen as a bottleneck for system performance humans were also perceived as a source of unwanted variability that was a primary cause of incidents and accidents. According to the logic of this line of reasoning, it therefore made sense to replace humans by automation as far as possible. Unfortunately, this view has created many of the problems we face today.

7.7.1.2 Bottom-Up Function Allocation

As soon as function allocation was recognised to be a problem, methods or principles were proposed to solve it. Most of these started from the bottom up, i.e., from the components and functions that were seen as constituting the system. Humans and machines were regarded as two separate components with distinct functions, which in the case of humans typically were sensory-motor or mental (cognitive) functions. In a similar manner tasks were decomposed into subtasks and activities, and a component's ability to carry out specific activities became the basis for function allocation and automation. This could be done either by focusing on the activities as such [3] or by comparing humans and machines function by function [4]. The underlying premise was that humans were needed for machines to work, and the problem was to ensure that this could be done in an effective manner – although from a technological rather than psychological perspective. (If humans were not needed for machines to work, the problem would obviously disappear. This condition can, however, only be achieved by fully automated systems.)

The reasons for using the bottom-up approach are not hard to find. Automation began by applying technology to take over relatively simple and well-defined functions, such as James Watt's Governor that controlled the speed of the steam engine. As long as technology was employed to take care of isolated functions, it made perfect sense to consider these alone and to compare humans and machines in terms of their specific capabilities. Once the tradition had been successfully established it was, however, carried over to cases where it should have been used with greater care or not have been used at all. These are typically the cases where it is incorrect simply to decompose a system into its elements.

Common to all bottom-up approaches is that they provide an assessment of the system in terms of component characteristics, rather than in terms of the overall functioning. The best-known example is probably the Fitts list [4], which compares the capabilities of humans and machines by a fixed set of attributes. A more recent example is the scale used to describe degrees or *levels of automation* [5], which proposes different roles or areas of responsibility for humans and machines. Although the direct purpose is to describe – and appraise – the resulting system, rather than to support function allocation the starting point is nevertheless the same: humans and machines seen as separate entities that can be compared by means of a common feature – in this case the degree of autonomy. The description is thus one of *how much* humans and machines do relative to each other, rather than of *what* they do. Since the various possible outcomes furthermore are considered to have different values, this evaluation serves indirectly as a guideline for function allocation. This aspect is even more noticeable in a later version of the scale, cf. Sheridan [6].

7.7.1.3 Top-Down Function Allocation

The alternative to a bottom-up approach is to begin from the top down by considering the requirements to the performance of the overall system. An example of that is the concept of critical functions [7], which has been used for design of displays and operational support in nuclear power plants. While critical functions are common to all systems they differ widely among domains. For instance, maintaining lift is a critical function in aviation, whereas in nuclear power production it is essential to maintain cooling inventory. Two well-known examples of top-down analysis in the behavioural sciences are the General Problem Solver [8] and the previously mentioned Test-Operator-Test-Exit (TOTE) operator [1]. They both represent the more general principle of goals-means analysis.

Although it is possible to define a set of general functions that apply to any system [9], it is sufficient for the present discussion to consider just one function, namely that of maintaining control. This, of course, requires a clarification of "who" is in control of "what". The "what" is usually a dynamic process, i.e., something that is capable of continuous and spontaneous change, activity, or progress.[7] In a similar manner, the "who" refers to the controller or the controlling system. Both process and controller are systems, which means that they can be described as "a set of objects together with relationships between the objects and between their attributes" [10, p. 81]. Since humans are cognitive systems [11,12], and since the controlling system is assumed to comprise both humans and technology, it is here referred to as a joint cognitive system (JCS).

To define the meaning of being in control, it is useful to start by noting how being out of control generally is associated with the occurrence of unwanted conditions. Being in control consequently means having the power or ability to direct and manage the development of events, while not being in control means that this ability is temporarily or permanently lost. A joint system is defined as being in control of a situation either if unexpected conditions do not arise, or if it is possible to avoid unwanted outcomes of such conditions. The former means that the joint system is able to prevent unexpected conditions from occurring; the latter means that the joint system is able effectively to recover from such conditions, should they occur.

In the top-down view, the objective of function allocation is to ensure that control is maintained. It is therefore necessary to consider the effects of function allocation – and of automation – for a variety of conditions over time, rather than for a pre-defined set of steady conditions. It is also necessary to consider which functions are required to maintain control and how they depend upon and support each other, rather than how they compare to each other on specific attributes of humans and machines. The structurally based distinction between system parts, such as humans and machines, is in this way replaced by a differentiation among the functions needed to achieve the overall system objectives.

7.7.1.4 Layers of Control

Depending on the domain, being in control can mean to arrive at a specific destination at a given time, to deliver something at a given place – and also at a certain time, to keep the value of selected process parameters within a certain range, etc. For some processes, such as heating a room or maintaining the water level in a vessel, control is usually uncomplicated and can be accomplished by a simple feedback loop. Most processes, however, comprise several functions or subsystems. The room may, for instance, be part of a larger building complex or a special facility, and the vessel may be a storage tank in a power plant. In these cases the control of a single function is just part of controlling the overall system. Subsystems may have different dynamics (rate of change or rate of response), they may differ in degree of complexity and

[7] It is important to keep in mind that the term process can refer to either a physical, a mental, or a social process. The system can thus be a technological artefact, a human being, or an organisation.

predictability, and specific relations or dependencies may exist among them (cf. the definition of a system given above). Yet effective control must be able to consider all these differences.

This requirement is consistent with the Law of Requisite Variety. The simplest, and perhaps best known, formulation of this is also the title of a seminal paper on the subject: "Every good regulator of a system must be a model of that system" [13]. This basically means that an effective regulator or controller of a system – or a process – must be capable of responding to any situation that can possibly occur. In relation to controlling a vehicle, for instance, it means that the controller must be able to compensate for all disturbances and events such that the vehicle is kept within the envelope of safe and efficient performance.

7.7.1.5 Control and Models

The Law of Requisite Variety is generally interpreted to mean that the controller must have a sufficiently powerful model of the system being controlled. This tradition is especially strong in the field of human-machine studies, where the operator's (mental) model has become a *sine qua non*. The Law of Requisite Variety nevertheless does not say that the regulator or controller must **have** a model of the system, but only that it must **be** a model of the system [14]. The difference between the two interpretations is important and has wide-ranging consequences. In the conventional way of thinking about models the unspoken assumption is that there must be both a model of what needs to be known about the process, and a "mechanism" of some sort to execute or interpret the model. This convention has been established by artificial intelligence and knowledge-based systems (cf. the notion of an inference engine in expert systems), and is possibly a rudiment from Aristotelian logic. The need to keep models and mechanisms separate and distinct is, however, a convention rather than a law of nature. In other words, the variety that the controller must have (to match the requisite variety) need not be represented explicitly by a model or a formal representation, but can be a feature of how the controller functions as such – specifically that the controller can respond adequately to every possible situation.

In the field of human-machine studies, systems that need to be controlled are usually complex and comprise multiple processes that develop with different speeds and which therefore must be described by different time frames. In the example of maintaining the water level in a storage tank, the time frame for that (sub)process is different from the time frame of power production. Filling or emptying a tank is a matter of minutes, rather than hours while power production is a process that develops more slowly, and which must be considered in a time frame of hours or days. The control of the overall process (power production) thus differs from the control of the water level in the tank. The complexity (requisite variety) that this creates must be matched by the controller, which therefore in some important sense must mimic the essential features of the process.

7.7.1.6 Requisite Variety as Layers of Control

As a generic example, consider the control of a transportation process, such as driving a car from home to work or guiding an unmanned vehicle to a destination. In both cases it involves the movement of a vehicle between two points, A and B. To achieve this it is necessary to control the overall progress of the vehicle, i.e., to ensure that it gets nearer to B – which is not always the same as getting further from A. It is also necessary to ensure that the vehicle steers clear of obstacles in the close environment, i.e., that it does not collide with any stationary or moving object, and that it negotiates changes in speed or direction in an effective manner. It is furthermore necessary to ensure that the vehicle has the necessary resources to move, e.g., power. It is finally necessary that the purpose of the transportation is defined, such as the locations of points A and B, the criteria for effective transportation (e.g., speed, power consumption, load, etc.).

In this example four layers of control were required. Other models have often suggested three different types of control, such as strategy, tactics, and operation [15], heading, pursuit, and pre-cognitive control [16], skill-based, rule-based, and knowledge-based performance [17], and strategical (planning), tactical (manoeuvring), and operational control [18]. The reason for proposing four layers rather than three is that for most activities the layer of (operational) control presupposes a further layer where actions are executed. For humans this is exemplified by what we do automatically, without thinking of it or paying attention to it. Examples are walking or running, using tools (as professionals do), maintaining a steady speed and position (in driving), etc. In most activities execution takes place so quickly that we are not aware of it before it has happened, since attention – in the sense of the act of becoming aware of something – requires a certain time.

According to the Law of Requisite Variety, the controller must be able to provide the several types of control listed above, regardless of what the actual number is. The simplest way of doing that is obviously to assume that the functioning of the controller is organised in different layers as well. In this manner the functional architecture of the controller embodies part of the variety needed to control the process, and the controlling system therefore to some extent **is** a model of the process. This obviously needs to be supplemented by representations of the specific characteristics of the domain, the context, and the situation. The process may also require that there multiple goals are pursued at each layer. The result is therefore multiple, simultaneous control functions organised in a number of layers.

7.7.2 Contextual Control Models (COCOM)

In the modelling of cognition, human or otherwise, a distinction can be made between procedural prototype and contextual control models. A procedural prototype model assumes that a pre-defined sequence of (elementary) actions or a procedural pattern exists, and that this represents a more natural way of doing things than others. In a situation, the expected next action can therefore be found by referring to the natural ordering of actions implied by the prototype. Although some steps may be bypassed by taking shortcuts, and although the procedural pattern may be applied recursively, the underlying sequence itself is treated as immutable.

A contextual control model, on the other hand, implies that actions are determined by the context rather than by an inherent sequential relation between them. In a situation, the next action is therefore determined by the current context and by the competence of the JCS (cf. below). If recurring patterns of actions are found, this can be ascribed to the characteristics of the environment rather than pre-programmed action sequences.

7.7.2.1 Dynamic Control – COCOM

Procedural prototype models are ubiquitous in human factors and behavioural sciences and examples are easy to find.[8] An example of a contextual control model is provided by the COCOM, which describes how a controller or controlling system can maintain control of a dynamic process.[9] In this model the basic principle is that decisions or 'actions' are determined by the current understanding of the situation (called 'construct', cf. Figure 7-15), which includes the anticipation or expectation of what will happen next. The 'events' represent the result of the actions (hence the feedback). If they match the expectations, they reinforce the 'construct'; if there is a mismatch, the 'construct' must be modified. The 'events' can also be completely unexpected, for instance if they are due to disturbances; that, of course, also demands a modification of the

[8] Typical examples are models of decision making or problem solving.

[9] A more detailed description can be found in Hollnagel [19] and Hollnagel and Woods [12]. In the case of the COCOM, the generic and specific names are unfortunately identical.

'construct'. The model can account for the dynamics of a situation, specifically the consequences of lack of time, and for the effects of different levels of control.

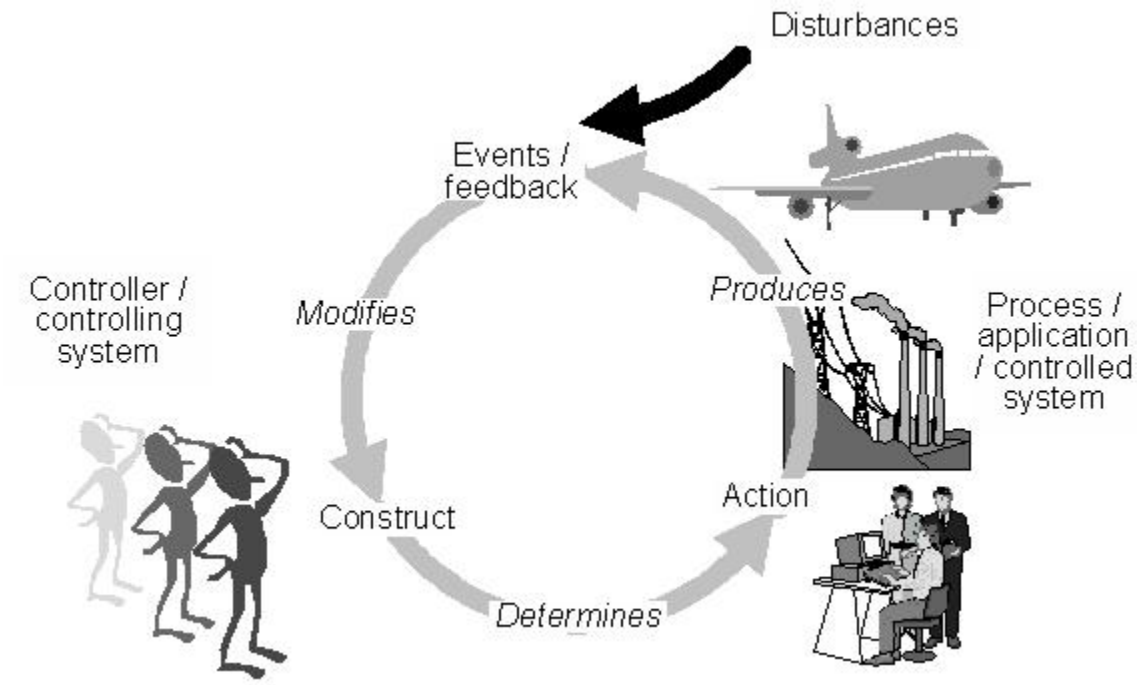

Figure 7-15: The Basic Construct-Action-Event Cycle.

7.7.2.2 Extended Control Model (ECOM)

The ECOM describes how the performance of a joint cognitive system (JCS) takes place on several layers of control simultaneously, using the notion of concurrent control loops. The ECOM follows the same principles of modelling as the COCOM [19] and is built on the latter in the sense that each layer corresponds to the fundamental construct-action-event cycle depicted in Figure 7-15. Some of these are closed (reactive), some are open (anticipatory), and some are mixed. The assumption of multiple layers of activity is crucial for the modelling approach, and it has in practice turned out that four layers are sufficient [20], although there is no accepted theory that determines their number. Although the ECOM has been developed to describe joint cognitive systems that include human operators, it has also been used to describe dynamic systems more generally, both technological and organisational. In the following sections, each of the four layers of activity will briefly be described, going from the lowest to the highest (cf. Figure 7-16).[10]

7.7.2.2.1 Tracking

The tracking layer of Figure 7-16 describes the activities required to keep a JCS inside predetermined performance boundaries, typically in terms of safety or efficiency. Activities at the tracking layer are very much a question of closed-loop control. For the skilled user such activities are performed automatically and therefore with little effort. While activities at the tracking layer usually are performed in an automatic and unattended manner, they may become attended, hence more like regulating, if conditions change.

[10] A more extensive treatment can be found in Hollnagel and Woods [12].

In the ECOM, goals and criteria for activities at the tracking layer are provided by the regulating layer. Most of the tracking activities are readily amenable to technology take-over and automation. If done clumsily this may give rise to automation surprises because the almost complete take-over of closed-loop control makes it difficult for operators to follow what is going on, hence to maintain the situation comprehension that is needed at the other layers of activity.

7.7.2.2.2 Regulating

The regulating layer describes the activities by which a JCS achieves short-term goals, such as specific manoeuvres relative to the environment (which need not be physical space). Regulating is itself basically a closed-loop activity, although anticipatory control may also occur (e.g., [16]). Activities at the regulating layer do not always take place smoothly and automatically, but may require attention and effort. These activities in turn refer to specific plans and objectives that come from the monitoring layer.

Under some conditions, the regulating loop may suspend the tracking loop. It may, for instance, be more important to keep maintain integrity than to follow a path. This can also be expressed as a temporary suspension of one goal (following a path) to the advantage of another (maintaining integrity). Incompatibility between goals can be resolved by changing or adjusting plans.

7.7.2.2.3 Monitoring

Whereas activities at the regulating layer may lead to either direct actions or goals for the tracking layer, activities at the monitoring layer are mainly concerned with setting objectives and activating plans for actions. This can involve monitoring the condition of the vehicle, although this has in most cases been taken over by automation, or the monitoring the state of the environment.

In many domains a distinction between regulating (of position) and monitoring (of location) can be made, although space may not always be physical (or Euclidean). In a vehicle, other activities at the monitoring layer may be related to information sources. Although this is not monitoring of performance *per se*, it may affect the ability to perform, particularly if it is non-trivial. Monitoring does not directly influence location of the vehicle in the sense of closed-loop control and regulation, but rather is concerned with the state of a JCS relative to its environment.

7.7.2.2.4 Targeting

The last type of action occurs at the targeting or goal setting layer. An obvious kind of goal-setting is with regard to the destination. That goal may give rise to several subgoals and activities, some of which can be automated or supported. Other goals have to do with criteria for acceptable performance.

Goal-setting is distinctly an open-loop activity, and is implemented by a nontrivial set of actions that often covers an extended period of time. Assessing the change relative to the goal is not based on simple feedback, but rather on a loose assessment of the situation – for instance, proximity to target. When the assessment is done regularly it may be considered as being a part of monitoring and control. If the assessment is done irregularly, the trigger is usually some unknown factor, perhaps time, perhaps a pre-defined cue or landmark (physical or symbolic), perhaps a background ‘simulation’ or estimation of the general progress (like suddenly feeling uneasy).

Figure 7-16 shows the relations among the four layers in a simplified manner. To avoid graphical clutter, Figure 7-16 includes only the goal dependencies among the layers; other dependencies exist, for instance,

in the propagation of feedback or events. For the same reason each layer is represented only by one construct-action-event cycle, even though there normally will be several concurrent cycles or loops. The arrows at the right-hand side of Figure 7-16 indicate the relative weight of feedback and feedforward control for each layer.

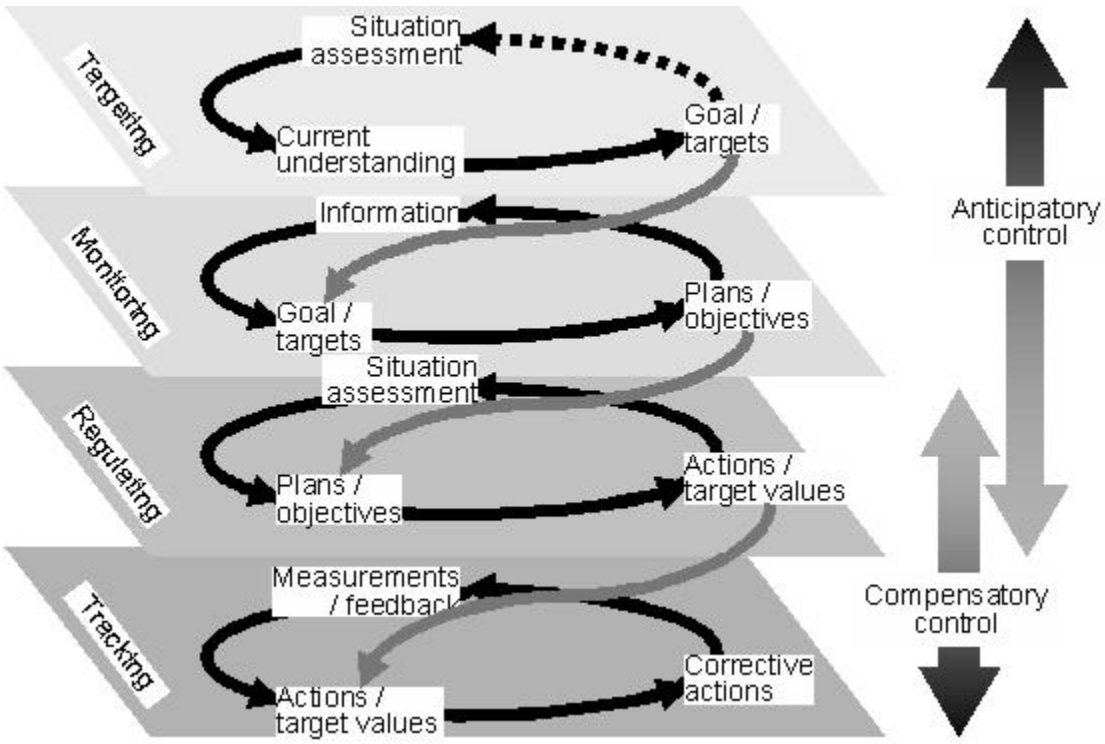

Figure 7-16: The Extended Control Model (ECOM).

7.7.2.3 ECOM Structure and Parameters

The main characteristics of each of the four layers are summarised in Table 7-7. This shows for each layer the type of control involved, the typical demands to attention in the case of a human controller, how often events can be expected to occur, and finally the typical duration of events. Here it is important to note that events on the tracking layer usually are of such short duration that in the case of humans they are pre-attentive. In other words, tracking-type behaviour is equivalent to skills, in the sense that it is something that is done more or less automatically and without attention. The regulating layer comprises actions of a short duration that do require attention, but not for long. The monitoring layer describes actions that go on intermittently as long as the task lasts, although the distribution can be decidedly irregular, depending on demands. Finally, actions on the targeting layer take place every now and then, almost always including the preparation of a task.

Table 7-7: Functional Characteristics of ECOM Layers

	Control Layer			
	Tracking	Regulating	Monitoring	Targeting
Type of control involved	Feedback	Feedforward and feedback	Feedback (condition monitoring)	Feedforward (goal setting)
Demands to attention	None (pre-attentive)	High (uncommon actions); low (common actions)	Low, intermittent	High, concentrated
Frequency	Continuous	Medium to high (context dependent)	Intermittent, but regular	Low (preparations, re-targeting)
Typical duration	< 1 sec (‘instantaneous’)	1 sec – 1 minute (‘short term’)	10 minutes – task duration (‘long term’)	Short (minutes)

The ECOM can be used to describe the interactions between the different layers. The assumption throughout is that all layers are active simultaneously, or rather that goals and objectives corresponding to different layers of control are being pursued simultaneously. One use of the ECOM is therefore to account for the nontrivial dependence between goals and activities among the layers, for instance when the tracking layer is interrupted by an unexpected disturbance. The goals of each control loop can also be temporarily suspended as when a higher-level goal is suspended in lieu of focusing on a lower level one. The bottom line is that controller performance and the effect of partial autonomy can be understood only in the context of the JCS as a whole, and not at the level of individual components or parts.

Returning to the issue of automation, it is clearly possible to consider the possibility of automation at each layer of control. In some cases functions have to be automated because it is impossible for humans to carry them out with sufficient speed, precision, or stability. In other cases functions have to be carried out by humans because the environment is too uncertain. In yet other cases there may be a choice. The top-down approach represented by the layers of control differs from the traditional way of comparing humans and machines because the comparison refers to the demands of the tasks rather than to the capabilities of system components. Table 7-7 can therefore also be used as a way of considering automation. In order to do this it is, however, necessary to enter into some other considerations.

7.7.3 Autonomous Control Level Framework

The concept of Autonomous Control Levels (ACL) has been developed to characterise issues of autonomous control in UAV missions. According to Huang et al. [21] the three main motivations for higher autonomy in unmanned system are:

1) Lack of bandwidth required for extensive tele-operation;
2) Safety for personnel; and
3) That mission effectiveness may be limited by cognitive workload, i.e., that humans are limited in their ability to perform their own tasks within the mission.

To that might be added time delays associated with communication links, which may hamper feedback control. A further motivation is that unmanned systems currently require control by one or more highly trained human operators; increasing the level of autonomy would clearly reduce this demand. All in all, increased autonomous control is seen as a way of maximising UAV utility.[11]

7.7.3.1 Levels of Autonomy

The ACL framework comprises ten autonomous control levels are shown in Figure 7-17. While the labels for the different levels are evocative, it is difficult to find clear definitions of the terms, even in the Roadmap report [22]. In a discussion of the ACL, Reising [23] pointed out that the levels seemed to be based on a conglomeration of levels of automation, human information processing models, and adaptive automation. While attempts have been made to explain the ACL in terms of concepts such as automation, workload, situation awareness (e.g., [24], but see also [25]), Reising (*op. cit.*) pointed out that this, as well as the original proposal for the ten levels of automation [5], focused on the interaction *between* the operator and the UAV rather than on the autonomy of a vehicle or a JCS considered as a whole. The work of Kaber and Endsley [24] furthermore started from the levels of automation, hence did not provide an explanation of the nature of the levels nor their number. Knowing the development of descriptions of automation and autonomy it is, however, safe to assume that the reason for having ten levels can be found in 1978 proposal, which quickly achieved an almost mythical status – even though their number was later reduced to eight [6].

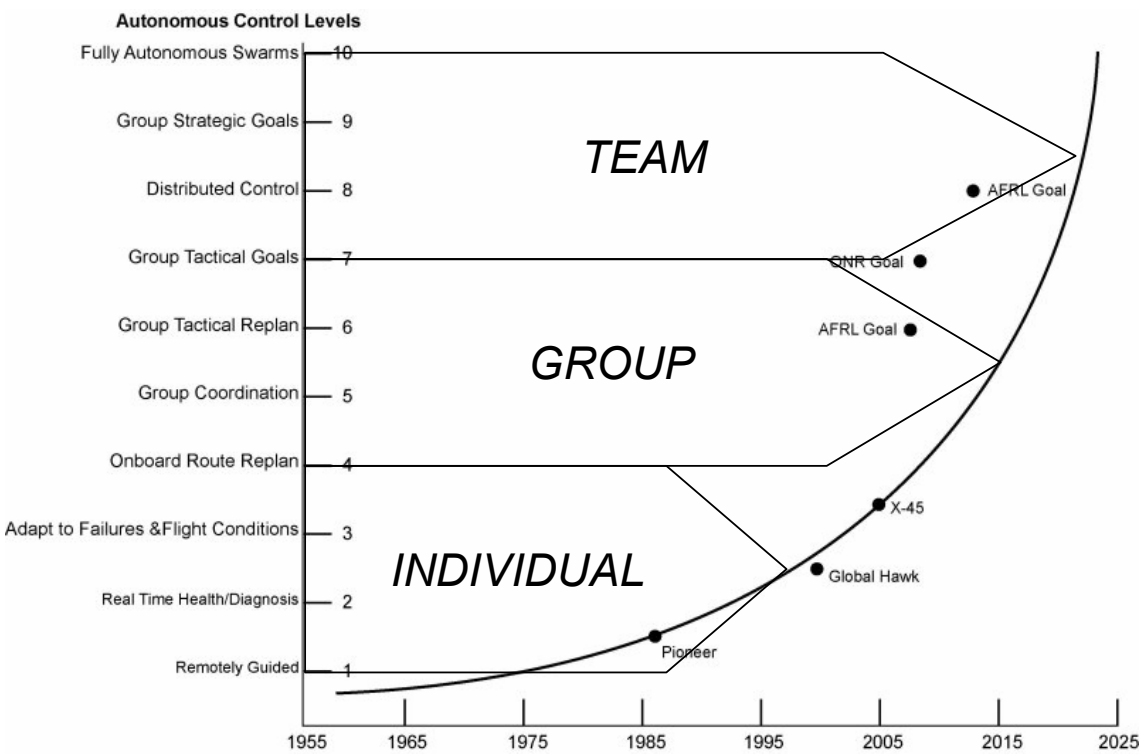

Figure 7-17: DOD UAV Roadmap 2000.

[11] In practice, the issue is one of automation rather than autonomy. An autonomous system is, by definition, completely independent, hence beyond human control. That is neither possible, nor desirable.

Figure 7-17 shows the ten levels of autonomy as developing gradually, beginning with remotely guided vehicles in the late 1950s and ending with fully autonomous swarms of vehicles in 2025.[12] Superimposed on this development is a characterisation of three mission types or configurations, namely individual (single vehicle), group (multiple vehicles), and team or swarm (multiple vehicles performing as a single entity). An important, but unspoken, assumption is that utility increases proportionally with the level of autonomy, hence that a high level of autonomy is more desirable than a low level. The ACL framework also implies bottom-up inheritance, i.e., that capabilities that exist at lower levels of autonomy are carried forth to higher levels.

7.7.3.2 Individuals, Groups, Swarms

In Figure 7-17, the three mission types – individual, group, and team (swarm) – are mapped as corresponding directly to increasing levels of autonomy. On further reflection, however, this need not be so.

As far as the three mission types are concerned, the meaning of an *individual* vehicle is obvious, i.e., it is a mission carried out by a single vehicle. The difference between *group* and *team*, where the latter is taken to be synonymous with *swarm*, is less obvious.[13] As noted by Reising [23], a swarm does not appear to comprise a central controller that tells swarm what to do. As seen in the natural world – schools of fish, flocks of birds, swarms of bees – "… *swarming* itself is a type of emergent behavior, a behavior that isn't explicitly programmed, but results as a natural interaction of multiple entities" [26, p. 1]. A swarm consists of a large number of members that all behave in a similar way, i.e., which have the same functions and capabilities. In particular, the members are replaceable.

On the other hand, a group is defined in social psychology as an assembly of individuals, recognised as being individually different, but treated, for some purpose, as parts of a larger unit. Group members usually have characteristic individual behaviours that in combination enable the group to achieve its purpose. The principal difference between a group and a team/swarm is consequently that group members can exhibit different (specialised) behaviours, while all members of a swarm behave in essentially the same way. In a group, the behaviour of the members must therefore be explicitly coordinated; in a swarm, the members behave collectively. Although the 'intelligence' of a swarm may be 'emergent', it does not necessarily mean that it is more difficult to control.

7.7.3.3 Mission Type and Autonomy

In the original for layers of automation [5], the end points of the scale were complete manual control and complete automation, respectively. In relation to the mission type, it is obvious that an individual vehicle can range from one extreme to the other, specifically that it in principle can be completely automated – but not completely autonomous. The same goes for swarms, although the manual control must address the swarm as a unit rather than the single entities that make up the swarm. In the case of a group, it is probably not feasible to consider manual control, since the coordination demands will be considerable and in a sense lead to the deconstruction of the group. On the other hand, there is nothing that, in principle, makes complete automation impossible. (A group might, in fact, be the only unit that can achieve autonomy.)

[12] The development is also shown as going faster and faster. This may be wishful thinking rather than reality.

[13] Note, however, that the major groupings have no clear correspondence with the categories on the Y-axis. Levels 5-9 refer to the group, while only level 10 refers to the swarm.

These considerations suggest that the ACL should be considered in terms of the two dimensions of mission type and level of automation, rather than in terms of level of autonomy alone. The result of this exercise is shown in Table 7-8, where the level of automation simply is an ordinal number, with no inherent meaning.

Table 7-8: A Possible Two-Dimensional Description of Autonomous Control Levels

Level of Automation	Mission Type		
	Individual	Group	Team/Swarm
7	Strategic goals and replan	Strategic goals and replan	Strategic goals and replan
6	Tactical goals and replan	Tactical goals and replan	Tactical goals and replan
5	Onboard route replan (adapt to route changes)	Onboard route replan (adapt to route changes)	Onboard route replan (adapt to route changes)
4		Coordinated control	Collective control
3	Adapt to failure and flight conditions (“see-and-avoid”)	Adapt to failure and flight conditions (“see-and-avoid”)	Adapt to failure and flight conditions (“see-and-avoid”)
2	Real-time health / diagnosis	Real-time health / diagnosis	Real-time health / diagnosis
1	Manual control (remotely guided)		Manual control (remotely guided)

Although the distinction between strategic and tactical is firmly entrenched in military language, it is from a scientific point relative rather than absolute. The issue is basically one of defining the goals for the system under consideration, whether it is an individual UAV, a group or a team. If a target is set with a long *span-of-foresight* it corresponds to a strategy [15]; if, on the other hand a target is set with a short *span-of-foresight* and/or change frequently, it corresponds to a tactic. (In Shützenberger’s terms, “the optimal strategy is just the simple tactic of attempting to do one’s best on a purely local basis”.) The shortest span-of-foresight is in route revision, which therefore can be seen as a low-level tactic. For group missions, route revision requires coordination; for team/swarm missions, route revision can be achieved by collective control.

The ECOM described four layers of control called targeting, monitoring, regulating and tracking, respectively. If we consider the modified description of the ACL categories shown in Table 7-8, it is possible to propose a mapping between the ACL and the ECOM, as shown in Table 7-9. Here the two highest levels of autonomy – or rather, automation – are seen as corresponding to the targeting layer in the ECOM. The issue is one of defining the goals for the system under consideration, whether it is an individual UAV, a group or a team. It may well be that targeting capabilities will not be considered for individual missions.

Table 7-9: Relations between ECOM and the Revised ACL Description

ECOM Layer	Mission Type		
	Individual	Group	Team/Swarm
Targeting	Goal setting and replan (strategic, tactical)	Goal setting and replan (strategic, tactical)	Goal setting and replan (strategic, tactical)
	Onboard route replan (adapt to route changes)	Onboard route replan (adapt to route changes)	Onboard route replan (adapt to route changes)
Monitoring	Adapt to failure and flight conditions (“see-and-avoid”)	Adapt to failure and flight conditions (“see-and-avoid”)	Adapt to failure and flight conditions (“see-and-avoid”)
	Real-time health / diagnosis	Real-time health / diagnosis	Real-time health / diagnosis
Regulating		Coordinated control	Collective control
	Manual control (remotely guided)		Manual control (remotely guided)
Tracking			

The second part of the targeting layer is on-board route revision, i.e., the ability to generate or select an alternative route. The impetus to do that can come either from the monitoring of flight conditions or from an external source of command (a revision of strategic or tactical goals). Route revision is clearly relevant for single UAVs as well as for multiple UAVs. As regards multiple UAVs, the difference may be whether they are individually controlled (as in a group), or whether they are seen as a larger unit (e.g., a swarm), where individual behaviour is subsumed the behaviour of the swarm. In other words, in a group single UAVs may have different roles and responsibilities, whereas a swarm functions as a collective whole.) On the group level this corresponds to coordination and distributed control, on the team level to collective control.

The next ACL levels, real-time health/diagnosis and adapting to failures and flight conditions (“see-and-avoid”) correspond to the monitoring layer of the ECOM. Monitoring is clearly relevant for all three mission types. Indeed, autonomous monitoring is imperative for missions with multiple vehicles, since human monitoring of simultaneous activities is not reliable. Monitoring can be either of the vehicle itself or of the vehicle’s environment (threats).

As argued above, regulating – in the sense of manual control – is relevant mostly for the individual mission type. To do the same for a group of vehicles will require extensive coordination and therefore not be feasible. It may still be possible for a swarm, since that can be considered as one unit to be regulated. Even so, the efforts involved will not be the same as for an individual vehicle.

In Table 7-9, regulation occurs in two different meanings. In case of an individual vehicle, regulation corresponds to conventional remote guidance or manual control. In the case of groups and teams/swarms low level regulation of this type is not desirable. On the contrary, it is probably a precondition for group and team

missions to be possible that the control of basic manoeuvres is completely automated. There is, however, another level of regulation, which has to do with the proper coordination of group or team members. This coordination is required in order to be able to implement route changes quickly and efficiently, i.e., a sort of guidance at a higher level. In the case of groups and teams, both monitoring and regulating could be extensively automated, leaving targeting in the hands of human operators.

Finally the layer of tracking is assumed to be completely automated, as it already is today. For airborne vehicles tracking (roll, pitch, yaw) must happen so quickly that it exceeds human capability. This layer will therefore not be considered further here.

One consequence of the above considerations is that Figure 7-17 should be revised to show the three mission types in parallel relative to the control levels rather than in series.[14] In other words, they represent three lines of development rather than one. There is, indeed, no good reason why a team should be seen as more difficult to automate than a group. In fact, it may well be the reverse, since a group requires the coordination of several individual behaviours rather than of a "single" behaviour of either an individual or a team/swarm seen as a "collective individual."[15] A suggestion of what this revision might look like is shown in Figure 7-18, which also reverses the ACL ordering of team and group.

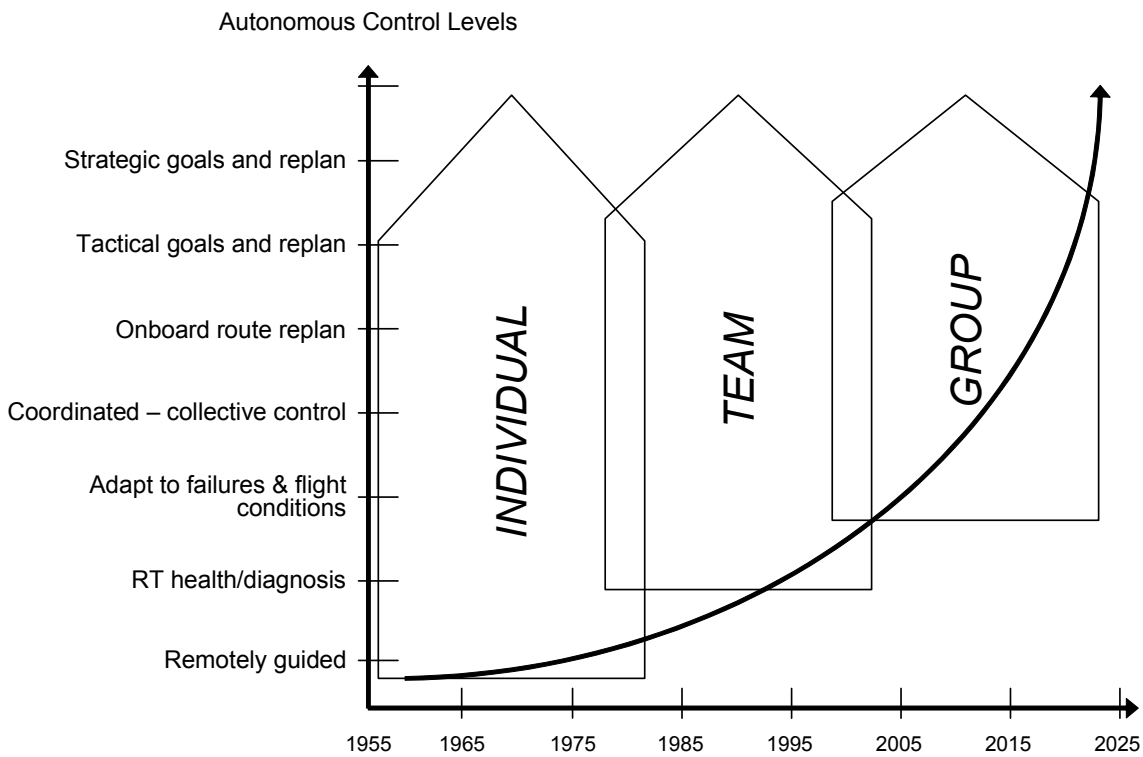

Figure 7-18: Revised ACL Description.

[14] On the other hand, this means that they will appear sequentially with respect to time, rather than as being developed in parallel. That may be a not entirely unwanted side-effect.

[15] All this assumes, of course, that the intentions of the roadmap have been correctly interpreted.

7.7.4 Analysis of UAV Scenarios

The use of the concept of layers of control will be illustrated by describing an UAV scenario. The description will address three characteristic levels of autonomy defined by the 'Roadmap', namely ACL1, ACL6, and ACL9. In terms of the descriptions given above, this corresponds to a remotely guided, single vehicle mission; a group mission; and a team mission.

Table 7-10 shows the scenario that is used. It divides the mission into the three phases of 'ingress', 'over target', and 'egress', where each phase is described further in terms of a number of tasks or functions.

Table 7-10: Overall Description of UAV Scenario

INGRESS	Control, Guidance, Navigation (own ship and attack aircraft)
	Replan
	Communication (C2 MC, attack aircraft, other UAV UCS)
	System management (+contingencies)
	Self defence
	Target location
OVER TARGET	Target registration, identification, verification, designation (for attack aircraft)
	Control, Guidance, Navigation
	System management (+contingencies)
	Communication (C2 MC, attack aircraft, other UAV UCS)
	Sensor management
	Self defence
	Rules of engagement
	Battle damage assessment
EGRESS	Control, Guidance, Navigation
	Communication (C2 MC, attack aircraft, other UAV UCS)
	System management (+contingencies)
	Self defence

One thing to notice about this scenario description is that a number of tasks are common to all three phases. These are: (1) Control, Guidance, Navigation (own ship and attack aircraft); (2) Communication (C2 MC, attack aircraft, other UAV UCS); (3) System management (+contingencies); and (4) Self defence. It is therefore reasonable to describe the scenario in terms of specific and common tasks, cf. Table 7-11. This shows that only the 'ingress' and 'over target' phases comprise specific tasks. It would be possible to analyse the scenario in more detail using a recognised task analysis method, such as hierarchical task analysis

or goals-means task analysis. This would actually be necessary to perform a complete analysis of the command and control demands of the scenario, but it cannot be done on the basis of the available material. Further details, as well as expertise knowledge, would be required. For the present purpose the available material is, however, sufficient to illustrate how the tasks that make up the scenario can be characterised in terms of different control layers, as a basis for anticipating the effects of increased automation (autonomy).

Table 7-11: Common and Specific Tasks in the UAV Scenario

	SPECIFIC TASKS	COMMON TASKS
INGRESS	Replan	Control, Guidance, Navigation (own ship and attack aircraft)
	Target location	Communication (C2 MC, attack aircraft, other UAV UCS)
		System management (+contingencies)
		Self defence
OVER TARGET	Target registration, identification, verification, designation (for attack aircraft)	Control, Guidance, Navigation (own ship and attack aircraft)
	Sensor management	System management (+contingencies)
	Rules of engagement	Communication (C2 MC, attack aircraft, other UAV UCS)
	Battle damage assessment	Self defence
EGRESS		Control, Guidance, Navigation (own ship and attack aircraft)
		Communication (C2 MC, attack aircraft, other UAV UCS)
		System management (+contingencies)
		Self defence

7.7.4.1 Description of Common Tasks

The four common tasks of this scenario are:

- Control, guidance, and navigation (own ship and attack aircraft). This is a composite task, which has at least the three components listed here. We shall further focus on the vehicle itself, since, e.g., guidance of attack aircraft reasonably might be subsumed under communication. In this case 'control' is understood as corresponding to tracking, i.e., keeping yaw, pitch, roll, etc., within design and operational limits. As such it is clearly something that is done automatically by the vehicle, as it

is for all modern aircraft.[16] Guidance and navigation involve the layers of monitoring and regulation. The control inputs to the vehicle will typically be in terms of the course, speed, and altitude needed to execute specific manoeuvres or reach specific goals. The goals themselves are both those that are defined by the mission (pre-planned) and those that arise as a result of contingencies, i.e., system management and self defence.

- Communication (C2 MC, attack aircraft, other UAV UCS). This is understood to comprise maintaining contact with the base as well as communicating with other entities involved in the mission (attack aircraft, etc.). It can therefore be seen as part of the monitoring – and to some extent of the regulating – of the mission as a whole, rather than tasks that specifically are part of what the vehicle does. The mission of the vehicle may indeed be seen as serving the purpose of communication, i.e., providing information as a necessary input to a superordinate task, i.e., that of achieving the larger mission objective.
- System management (+contingencies). These are clearly tasks that correspond to the monitoring layer of the ECOM. As mentioned above, monitoring can be of the vehicle itself as well as of the environment. In that later case it may lead to the formation of new goals, cf. below.
- Self defence. In order for the vehicle's mission to be accomplished, it must obviously be able to maintain its functional integrity, which means the ability to defend itself in various ways. The defence can, for instance, be by evasive manoeuvres or by direct counteraction. In relation to the ECOM, self defence can be seen as a targeting activity, i.e., the formation of situation specific goals – which may possibly supersede pre-defined mission goals. Needless to say, targeting is based on processing of incoming information, hence monitoring. A new goal established in this way will give rise to new loops on the monitoring and regulating layers, cf. the description above.

The relations among the ECOM layers and the common mission tasks is summarised in Table 7-12.

Table 7-12: ECOM Characterisation of Common Mission Tasks

Common Mission Tasks	Corresponding ECOM Layer			
	Targeting	Monitoring	Regulating	Tracking
Control, guidance, navigation		X	X	Autonomous
Communication (C2 MC, attack aircraft, other UAV UCS)		X	(X)	
System management (+ contingencies)		X		
Self defence	X	(X)		

7.7.4.2 Description of Specific Tasks

The six specific tasks are found in the 'ingress' and 'over target' phases.

- Ingress: Replan. In the original task description this appears as a subsidiary task to 'control, guidance, navigation'. It is therefore appropriate to see it as associated with targeting, or rather re-targeting.

[16] Similar developments are found in land-based vehicles, even for civilian purposes, and in maritime vehicles.

It takes place in response to an external command, and as a result leads to changes (i.e., new goals) for both monitoring and regulating.

- Ingress: Target location. This is a specific, composite activity which best can be interpreted as higher-level regulating. It is composite in the sense that it comprises monitoring of the current position and of the environment, in order to recognise the target. This may in turn require specific manoeuvres (regulating) to bring the vehicle sufficiently close to the target. Target location thus entails both monitoring and regulating.
- Over target: Target registration, identification, verification, designation (for attack aircraft). One of the main goals of an UAV is to identify a target and to communicate that information to the mission centre or to, e.g., an attack aircraft. In terms of the ECOM, target registration is mainly a question of monitoring the environment in combination with communication (as a common mission task).
- Over target: Sensor management. This is understood as being the management of the various types of information (sensors, channels), to make the best use of the available information for the current goal/action. It can thus be seen as a kind of internal regulating, not of the vehicle's behaviour, but rather the selection and processing of sensor input to be optimally responsive to the needs of the current task. In terms of the ECOM, sensor management is a special kind of regulating.
- Over target: Rules of engagement. The rules of engagement (ROE) provide guidance governing the use of force consistent with mission accomplishment [27]. The ROE can therefore be seen as defining part of the performance envelope for the vehicle, specifically the behaviour that under normal circumstances is prohibited. In order to meet the ROE it is therefore necessary to monitor the situation and to abandon specific actions if they are in conflict with the ROE. Seen as a task, 'rules of engagement' therefore entail both monitoring and regulating.
- Over target: Battle damage assessment. This is a straightforward case of monitoring for specific changes. It may produce specific manoeuvres as a way of getting the recording the needed data. Battle damage assessment thus comprises monitoring and possibly regulating.

The relations among the ECOM layers and the specific mission tasks is summarised in Table 7-13.

Table 7-13: ECOM Characterisation of Specific Mission Tasks

Specific Mission Tasks	Corresponding ECOM Layer			
	Targeting	Monitoring	Regulating	Tracking
Replan	X			
Target location		X	X	
Target registration, identification, verification, designation (for attack aircraft)		X		
Sensor management			X (internal)	
Rules of engagement		X	X	
Battle damage assessment		X	(X)	

7.7.5 Control, Automation and Views

Designing for control can be viewed in different ways, ranging from graphical interface design to intelligent, autonomous agents or even virtual reality avatars. Within cognitive systems engineering, understanding the nature of control is a prerequisite for the design of the elements of control such as interfaces and information presentation. As mentioned above, the essence of control is that unwanted variability in the process, which may lead to deviations from the desired or prescribed course of development, is reduced to a minimum or does not occur at all. Effective control therefore requires the ability to detect such variability and to respond to it, which in most cases also means the ability to anticipate large fluctuations and prevent them from happening. A system for effective control must therefore support three different types of view: a view of what has happened (the past), a view of what happens here and now (the present), and a view of what may possibly happen (the future).

If the notion of the three different views is applied to the four layers of control, it is evident that not all views are important at each layer. Targeting, for instance, is concerned with developing plans for future actions, while tracking is more about responding to changes in the situation. Indeed, the differences in the type of control (feedback, feedforward) involved at each layer give rise to different demands for data and information. A proposal for the relative importance of the three different views is shown in Table 7-14.

Table 7-14: Relations between Control Layers and Process Views

View	Control Layers			
	Tracking (feedback control)	Regulating (feedforward + feedback control)	Monitoring (feedback control)	Targeting (feedforward control)
Past			Important	Very important
Present	Very important	Very important (Feedback)	Very important	
Future		Very important (Feedforward)		Very important

Getting the information provided by each view involves a cost, most obviously in terms of time. Making use of the information – processing it, understanding it – also takes time, which generally increases with the complexity of the information. Since the process that is controlled usually cannot wait, the management of time is a critical issue in the design of control systems [28,12]. This is certainly the case for the control of a vehicle, not least if it is uninhabited.

7.7.5.1 Control and Time

At the tracking and regulating layers, time is very short – and may in some cases be so short that humans hardly can be involved. A driver can steer a car (tracking layer control) when (s)he is in it, mainly because (s)he is able directly to perceive the environment – at least during daylight. It would be very hard to do this by remote control, since important cues such as motion feedback are missing, even in a full-blown synthetic sensory environment. At the regulating layer human control may be possible depending on the speed of the vehicle (a ship, a land vehicle, a flying drone), although in most cases it is still impractical. At both layers

automation may therefore be necessary to achieve sufficiently fast performance; modern cars, especially at the high-end, are a good example of that. At the monitoring and targeting layers the demands to rapid responses are smaller, but automation may be needed for other reasons, e.g., to achieve steadiness and regularity of functions or to be able to address multiple, simultaneous processes.

While most of the sources of information comprising the different views are external to the controller, for instance sets of local and remote sensors, some information comes from the controlling system itself. The regulating layer needs information about what happens at the tracking layer, the monitoring layer needs information about what happens – and has happened – at the regulating layer, and so on. Even a partial loss of information may lead to a degradation of control and possibly destabilise the dynamic equilibrium among the different layers. Since automation often leads to a loss of information, the principle of layers of control described by the ECOM provides a potentially powerful tool for analysing and understanding how such effects may arise and how they may affect how well the controlling system performs.

7.7.6 Conclusions

A first analysis of the selected UAV scenario shows that it comprises a number of common and specific tasks. The specific tasks can be seen as constituting one line of activity, which serves the primary purpose of the scenario, i.e., a successful mission. The common tasks can be seen as constituting four parallel lines of activity, which must be carried out more or less continuously in order to ensure that the specific tasks can be successfully accomplished. As a result, the scenario presents itself as comprising five parallel lines of activity rather than just one.

7.7.6.1 Effects of Automation on Common and Specific Tasks

The ECOM characterisation of the common and specific tasks provides a basis for considering the feasibility of automation for various mission types. In general, tasks comprising regulating are the easiest to automate from both a technical and a human factors point of view. For an individual mission, this involves the automation of remote guidance in the sense that the execution of actual manoeuvres is done autonomously. This is feasible to the extent that all possible manoeuvres can be anticipated or synthesised. Otherwise it falls prey to the ‘ironies of automation’ [29]. For a team/swarm mission, regulating corresponds to the remote collective control of the swarm, which, as noted above, can be considered as one unit. Given that the automation of a single vehicle is possible, it should therefore also be possible to automate the regulating of a swarm.

The situation is somewhat different for a group, since the individual members of a group need not behave in a uniform manner (cf., the above distinction between a group and a swarm discussed above). This makes the regulating (as automated remote control) of a group more complex since it requires the coordinated regulating of a number of individual vehicles. It may be possible to the extent that the coordination complies with a pre-defined pattern or patterns, but if that is not the case, the coordination will require targeting and monitoring functionality that makes it unsuitable for automation.

Tasks comprising monitoring are in general also well-suited for automation. Regular or routine monitoring has traditionally been one of the earliest functions to succumb to automation, since it is a function that machines perform far better than humans. In terms of the descriptions given above, this applies to all three types of missions. Among the common tasks, monitoring associated to ‘control, guidance, navigation’, ‘system management’ and ‘self defence’ can probably be automated for all types as well. Among the specific tasks, the same goes for ‘target registration’, ‘rules of engagement’, and ‘battle damage assessment’, but probably

not for 'target location', as the latter may vary considerably with the context. Again, the problem may be more difficult for a group type of mission, since it involves differentiated roles or functions of individual vehicles.

Tasks comprising targeting can, on the whole, not easily be automated – regardless of mission type. This is a simple consequence of the fact that automation is feasible only for situations or conditions that have a high degree of regularity and predictability. The aspirations and promises of artificial intelligence notwithstanding, the take-over by technology of more complex cognitive functions has generally been unsuccessful on a real-life scale (e.g., [12]). For the proposed scenario, it is only the common task of 'self defence' and the specific task of 're-planning' that correspond to the ECOM targeting layer. For 'self defence', autonomous functioning may be possible for common threat scenarios, but in that case it may actually be a question of monitoring-induced regulating rather than of targeting as such. For 're-planning', autonomous functioning is probably neither advisable nor feasible, not even within the time span of the Roadmap.

Each of the tasks can be further characterised in terms of the ECOM, with respect to the control layers that are involved. This analysis shows that the monitoring and regulating layers dominate. The targeting layer occurs a few times, while the tracking layer is assumed already to be completely automated. The characterisation of tasks in terms of layers of control provides a basis for assessing the feasibility of automating specific tasks and functions. The general principle is that regulating tasks usually are amenable to automation, while monitoring tasks are so to the extent that they are regular and predictable. Tasks involving targeting are not considered as likely candidates for automation.

In addition to considering each task on its own, it is also necessary to consider the implications of having five parallel lines of activity. In terms of the ECOM, this corresponds to having five parallel control structures. In order fully to appreciate the consequences of proposed automation, it is necessary to analyse the dependencies among the five lines of activity, in terms of interacting goals, inputs, and outputs, in addition to doing so also within each line of activity. An analysis of this type has not been attempted here, but may be carried out following the same principles or – if risk is an issue – the principles of functional resonance [30].

7.7.6.2 The Way Ahead?

In the top-down approach, concerns about the level of automation are secondary to the ability to stay in control. This means that the discussion changes from a comparison of system components and functions to a description of how well the system is able to accomplish its purpose. Noting, for instance, that "the machine suggests one alternative and executes that suggestion if human approves" does not say very much about the nature of control – except that it must take place at a pace so slow that there is time for such consultation. A high level of automation, or even full automation, is not in itself negative or bad. Neither is a low level of automation inherently desirable. It all depends on what the joint system is required to do.

Describing system performance by means of layers of control makes it easier to consider the consequences of automation, hence to make sensible decisions about function allocation. A joint cognitive system is defined by its ability to modify its pattern of behaviour on the basis of past experience to achieve specific anti-entropic ends, and automation may affect this ability in both a positive and negative direction. By starting from an understanding of control as the ability to prevent unexpected conditions from arising and to recover from them, should they occur, the effects of design decisions are easier to see. This goes both for the dynamic equilibrium among the four different layers of control, and the information needed to sustain the views of past, present and future events. The top-down approach forces a view of the system as a whole, and thereby weakens possible implicit assumptions about what men are better at and what machines are better at. Since it is impossible completely to predetermine the environment in which joint systems must function, every effort

should be made to ensure that they have the capabilities needed successfully to achieve their purpose, and to maintain control even when things do not go as planned.

7.7.7 References

[1] Miller, G.A., Galanter, E. and Pribram, K.H. (1960). Plans and the structure of behavior. New York: Holt, Rinehart & Winston.

[2] Wiener, N. (1964). God & Golem, Inc. A comment on certain points where cybernetics impinges on religion. The MIT Press, Cambridge, Massachusetts.

[3] Taylor, F.W. (1911). The principles of scientific management. New York: Harper.

[4] Fitts, P.M. (Ed). (1951). Human engineering for an effective air navigation and traffic-control system. Ohio State University Research Foundation, Columbus, Ohio.

[5] Sheridan, T.B. and Verplank, W.L. (1978). Human and computer control of undersea teleoperators. Cambridge, MA: MIT Man-machine system laboratory.

[6] Sheridan, T.B. (2002). Humans and automation: System design and research issues. New York: John Wiley & Sons, Inc.

[7] Corcoran, W.R., Portet, N.J., Church, J.F. and Cross, M.T. (1980). The critical safety functions and plant operation. IAEA Conference.

[8] Newell, A. and Simon, H.A. (1972). Human problem solving. Englewood Cliffs, New Jersey: Prentice-Hall, Inc.

[9] Lind, M. and Larsen, M.N. (1995). Planning and the intentionality of dynamic environments. In: J.-M. Hoc, P.C. Cacciabue and E. Hollnagel (Eds.), Expertise and technology: Cognition and human-computer interaction. Hillsdale, N. J. Lawrence Erlbaum Associates.

[10] Hall, A.D. and Fagen, R.E. (1968). Definition of system. In: W. Buckley (Ed.), Modern systems research for the behavioural scientist. Chicago: Aldine Publishing Company.

[11] Hollnagel, E. and Woods, D.D. (1983). Cognitive systems engineering: New wine in new bottles. International Journal of Man-Machine Studies, 18, 583-600.

[12] Hollnagel, E. and Woods, D.D. (2005). Joint cognitive systems: Foundations of cognitive systems engineering. Boca Raton, FL: CRC Press/Taylor & Francis.

[13] Conant, R.C. and Ashby, W.R. (1970). Every good regulator of a system must be a model of that system. International Journal of Systems Science, 1(2), 89-97.

[14] Hollnagel, E. (1999). Modelling the controller of a process. Trans. Inst. MC, 21(4-5), 163-170.

[15] Schützenberger, M.P. (1954). A tentative classification of goal-seeking behaviours. Journal of Mental Science, 100, 97-102.

[16] McRuer, D.T., Allen, R.W., Weir, D.H. and Klein, R.H. (1977). New results in driver steering control. Human Factors, 19, 381-397.

[17] Rasmussen, J. (1986). Information processing and human-machine interaction: An approach to cognitive engineering. New York: North-Holland.

[18] Michon, J.A. (1985). A critical view of driver behavior models. What do we know, What should we do? In: L. Evans and R. Schwing (Eds.), Human behavior and traffic safety (pp. 485-525). New York: Plenum press.

[19] Hollnagel, E. (1993). Human reliability analysis: Context and control. London: Academic Press.

[20] Hollnagel, E., Nåbo, A. and Lau, I. (2003). A systemic model for Driver-in-Control. 2nd International Driving Symposium on Human Factors in Driver Assessment, Training, and Vehicle Design, July 21-24, Park City, Utah.

[21] Huang, H.-M., Messina, E. and Albus, J. (2003). Autonomy Level Specification for Intelligent Autonomous Vehicles: Interim Progress Report. Proceedings of the 2003 Performance Metrics for Intelligent Systems Workshop, National Institute of Standards and Technology (NIST), Gaithersburg, MD, August 16-18.

[22] DOD (2002). Unmanned aerial vehicles roadmap 2002-2027. Washington, DC: Office of the Secretary of Defense.

[23] Reising, J. (2004). Synthesizing Perspective – Supervisory Control and Decision Support Concepts. (From USAF Aircraft: What will the Future Hold? ASC 04-1332 12 May 04).

[24] Kaber, D.B. and Endsley, M.R. (2004). The effects of level of automation and adaptive automation on human performance, situational awareness and workload in a dynamic control task. In: Theoretical Issues in Ergonomics Science, 5(2), 113-153.

[25] Dekker, S.W.A. and Hollnagel, E. (2004). Human factors and folk models. Cognition, Technology & Work, 6, 79-86.

[26] Clough, B. (2002). UAV's swarming? So what are those swarms, what are the implications, and how do we handle them? Proceedings of AUVSI Unmanned Systems, July 2002. Association for Unmanned Vehicle Systems International, 1-15.

[27] CJCSI (2000). Standing rules of engagement for US forces (CJCSI 3121.01A, 15 January 2000). Washington, DC: Chairman of the Joint Chiefs of Staff Instruction.

[28] Hollnagel, E. (2002). Time and time again, Theoretical Issues in Ergonomics Science, 3, 143-158.

[29] Bainbridge, L. (1983). Ironies of automation. Automatica, 19(6), 775-779.

[30] Hollnagel, E. (2004). Barriers and accident prevention. Aldershot: Ashgate.

7.8 SYNTHESIZING PERSPECTIVE – SUPERVISORY CONTROL AND DECISION SUPPORT CONCEPTS

The U.S. Navy is depending very heavily on the versatile F/A-18 Super Hornet as the mainstay of its carrier fighter/attack force in the foreseeable future. In addition, an electronic attack version is also planned to augment the attack force, with deliveries starting in 2009. Aircraft will have either one or two crewstations depending on the version. On the Air Force side, the F/A-22 Raptor and the F-35 Joint Strike Fighter are the latest aircraft. Both will have a single person in the crewstation. The Navy and Marine Corps also plan to purchase the F-35. The first deliveries of the Air Force and Marine Corps versions of the F-35 will be in 2008, with the Navy's first deliveries starting in 2010. The bottom line is that these three aircraft will provide the two Services' fighter/attack force well into the future [1] – but what type of aircraft will we have beyond these? And what type of crewstation will they have?

One of the issues currently being addressed is the role of future long-range bombers within the Air Force. "The Air Force is rethinking long-range strike, a term that used to mean only one thing: big bombers. As the service adjusts to the Pentagon's new capabilities-based strategy and focuses on desired effects rather than the platforms needed to achieve them, the eventual successor to today's bomber fleet remains intentionally unsettled." [2, p. 29]. The various versions being studied include not only conventional bombers as we think of them, but also various types of space planes. Another interesting aspect of these long-range strike vehicles is whether they will have a crew onboard or on the ground. Among the options being considered are systems with no airborne crew which means it may become an Uninhabited Aerial Vehicle (UAV) [3] (The term "uninhabited" was chosen deliberately; we think it is more accurate than the term "unmanned" which implies only a man would not be aboard these vehicles).

7.8 .1 Uninhabited Aerial Vehicles

UAVs have become well-known based on the conflict in Afghanistan. They served to give the command and control authorities continuous pictures of possible targets and also enabled a dramatic reduction in the time from which the target was identified until it could be engaged.

A number of NATO countries are now using UAVs to augment their forces, especially in performing tasks that are dull, dirty, or dangerous. Force augmentation issues relevant to the human operator exist on several levels, including individual UAV control station design, vehicle interoperability by different organizations, and integration of UAVs with manned systems. Human interface issues associated with individual UAV control station design include guaranteeing appropriate situational awareness for the task, minimizing adverse effects of lengthy system time delays, establishing an optimum ratio of operators to vehicles, incorporating flexible levels of autonomy (manual through semi-autonomous to fully automatic) and providing effective information presentation and control strategies. UAV interoperability requires development of a standard set of control station design specifications and procedures to cover the range of potential UAV operators and applications across military services and countries.

Finally, for UAVs to be successful, they must be fully integrated with manned systems so as to enhance the strength of the overall force. Human factors considerations in this area include how manned systems should best collaborate with UAVs, deconfliction concerns, operation with semiautonomous systems, and command and control issues. The essence of this paragraph can be summarized by the following statement: What is the proper role for the operator of UAVs? The operator's role can be defined in terms of three key factors.

7.8.1.1 Factor 1: Advanced UAV Operator Control/Display Interface Technologies

The operators' stations for the U.S. Air Force's Predator and Global Hawk UAVs are mounted in vans with the operators sitting at command and control stations. The ground-based operators of these two vehicles control them quite differently. The Predator, at least in the landing in takeoff phase, uses tele-operation with the operator actually flying the vehicle from a distance. The Global Hawk, on the other hand, takes off and lands automatically and is largely autonomous during its mission. The operator, using supervisory control, "flies" the Global Hawk by using a mouse and keyboard, not stick and throttle. Not all UAV control stations are large enough to be housed in vans, the operator station for the U.S. Marine Corps's Dragon Eye UAV, for example, is the size of a small suitcase which makes it easily transportable (Figure 7-19).

Figure 7-19: Predator Operator Station (left) and Dragon Eye Operator Station (right).

Research efforts with the Predator console have addressed a number of control and display features. Two examples are: head-coupled head-mounted display applications [4] and tactile system alerts [5]. Two additional efforts will be discussed more in detail.

As an example of a display enhancement, Draper, et al. [6] examined four different display formats which would aid the ability Air Vehicle Operator (AVO) and the Sensor Operator (SO) to determine target location. If the AVO located a target in the wide field-of-view camera, it was often difficult to communicate the location to the SO who had a narrow field-of-view camera. Four different formats were examined to improve communication between the two crewmembers (Figure 7-20).

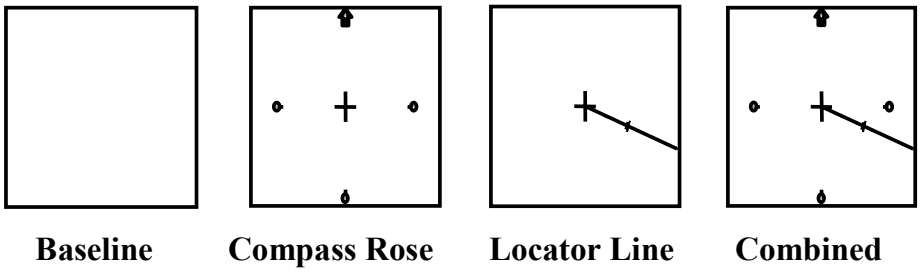

Figure 7-20: Display Concepts.

The results showed that the two formats utilizing the locator line were significantly better than the others. "Time to designate targets was reduced an average of almost 50% using the telestrator [locator line] ..." [6, p. 3-88]. The reason for the superiority of the locator line was that once the AVO designated the target it gave the SO a direct bearing to the target, thereby providing a very efficient means of exchanging information between the two operators.

As an example of control research, Draper et al., [7] compared manual versus speech-based input involving the use of menus to complete data entry tasks. Pilots also performed flight and navigation tasks in addition to the menu tasks. Results showed that speech input was significantly better than manual for all eight different kinds of data entry tasks. The overall reduction was approximately 40% in task time for voice entry when compared with manual input. The operators also rated manual input as more difficult and imposing high workload than the speech method. The reason for the superiority of the voice system was that it enabled the operator to go directly to the proper command without having to manually drill down through a number of menus sublevels in order to find the proper command.

Different types of control modes for operators' consoles were discussed in a recent conference [8]. One recurring theme was a strong desire to move away from tele-operation of the UAVs and progress towards a combination of semiautonomous and fully autonomous operation of these vehicles – regardless of the type of operator console. In order to achieve this goal, a significant amount of automation will be required, especially, when coupled with the desire, in the case of UAVs, to move from a situation where a number of operators control one vehicle to one operator controlling a number of vehicles.

Research exploring the issues of one operator controlling multiple vehicles is just beginning. Barbato, Feitshands, Williams and Hughes, [9] examined a number of operator console features that would aid the operator in controlling four Uninhabited Combat Aerial Vehicles (UCAVs). The mission was to carry out a Suppression of Enemy Air Defences. The operator's console contained three liquid crystal displays onto which was presented a situational awareness (SA) map, UCAV status and multifunction information. The SA format presented the overall geographical situation along with, among other information, the flight routes of the four aircraft. The participants were required to manage the flight routes in two ways: manual versus semiautomatic using a route planner. Although the operators were favorable towards the real-time route planner, they did want information regarding what the real-time planner was actually doing (its intent) and they wanted both the original route and the planned route displayed in order to evaluate the two against each other. In essence, the study showed that a single operator could manage four UCAVs – so long as there were not too many unexpected events.

7.8.1.2 Factor 2: Supervisory Control and Decision Support Concepts

In the case of the UAV, the avionics will be partly contained in the flying platform and partly incorporated into the operator's console, whether airborne or ground based. In either case, because of present day capabilities in computers and intelligent agent software, the resulting product can be much closer to a true team. Operator-machine relationships are being created which emulate those occurring between two human crewmembers – mutual support and assistance. A diagram depicting this overall relationship is shown in Figure 7-21.

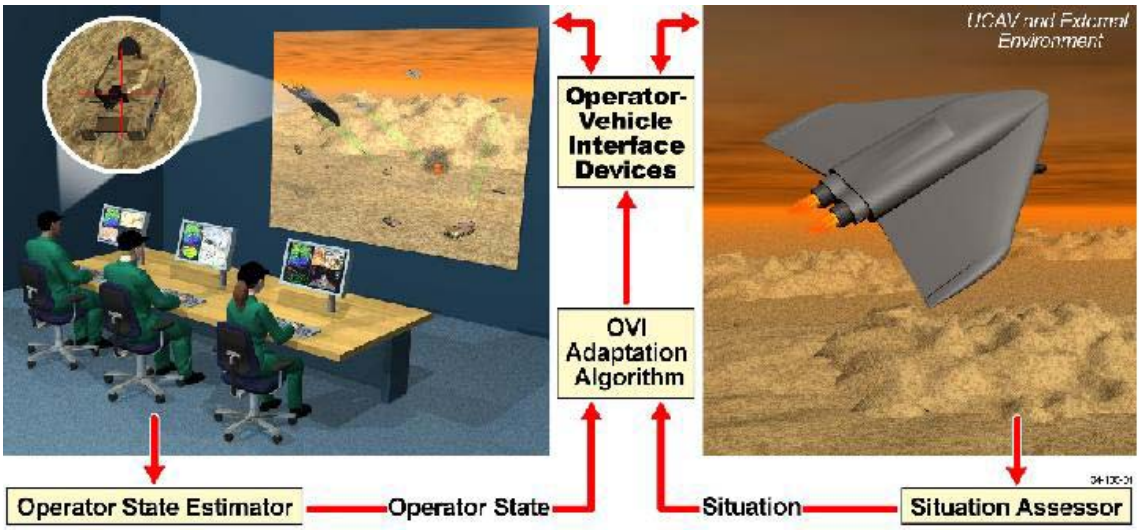

Figure 7-21: Operator-UAV System Diagram.

A major component in achieving this mutual support and assistance is through software entitled associate systems. "Associate systems are computer-based aiding systems that are intended to operate as an associate to the human user" [10, p. 221]. Following from his definition, Geddes goes on to list three very important rules for associate systems and their relationship with the human operator.

- Mixed Initiative – both the human operator and decision aid can take action.
- Bounded Discretion – the human operator is in charge.
- Domain Competency – decision aid has broad competency, but may have less expertise than the human operator.

Because of the mixed initiative aspects of an associate system, function allocation, which assigns roles to the operator and the computer based on their abilities, has to be looked at in an entirely new light. The idea of function allocation has been around since the 1950s and had as its basic premise that the role of operator and the machine (computer), once assigned, would stay relatively constant during the operation of the system. However, this premise does not hold for modern computers since they contain associate systems which can have varying levels of automation at different times during a particular mission; therefore, static function allocation is no longer applicable [11]. Rather, *dynamic* function allocation is a key feature of associate systems with varying levels of automation.

Taylor [12], illustrates how dynamic function allocation changes the working relationship between the Human Operator and the Machine (with associate system based automation); this changing relationship is shown in Figure 7-22.

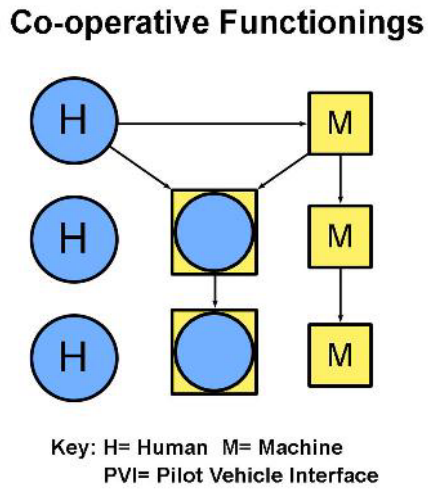

Figure 7-22: Systems Authority Concepts – H = Human; M = Machine (from [12, p. 2-17].

Co-operative Functionings indicates how the operator (H in the figure) and automation (M in the figure) would work together in an associate system. It is quite different from both manual control and supervisory control. In manual control, the human operator specifies the goals and functions to be accomplished and the machine carries out the tasks. In the next level, supervisory control, the human operator still specifies the goals, but the machine carries out both the tasks and functions. In the co-operative functionings (associate system), the human operator and machine interact at all levels, and *either* can perform the goals, functions and tasks. It is through this dynamic sharing of authority that the operator and the associative can begin to operate as a team – an operator and a type of electronic crewmember (EC). However, to function as a team, the operator must trust the EC.

7.8.1.3 Factor 3: Trust and Levels of Automation

One means of establishing operator trust in the EC is to allow the operator to decide how much authority or autonomy, called levels of automation (LOA), to give the EC. "LOA defines a small set ('levels') of system configurations, each configuration specifying the degree of automation or autonomy (an 'operational relationship') at which each particular subfunction performs. The pilot sets or resets the LOA to a particular level as a consequence of mission planning, anticipated contingencies, or inflight needs" [13, p. 124]. While originally conceived for a piloted aircraft, LOAs apply equally well to UAV consoles and their operators. One question that must be answered is how many levels of automation should be assigned to the associate? A number of researchers have examined this issue. The result is as many as 10 [14] and as few as 5 [15].

In order to create an effective team, once the levels are determined, the next task is to determine how they relate to the way humans process information. A further expansion of LOA was proposed by Parasuraman, Sheridan and Wickens [16]; they matched levels of automation with a four stage human information processing model (information acquisition, information analysis, decisions selection, and action implementation). The 10 LOAs are based on a model proposed by Sheridan [14]. They then illustrate how various systems could have different levels of automation across the four portions of the information processing model. This work is very important because it begins to blend levels of automation with human information processing capabilities. The authors realize that the model is not finalized, "We do not claim that our model offers comprehensive design principles but a simple guide." (p. 294) However, it certainly is in the

right direction towards achieving an optimal matching between automation and human capabilities for particular systems.

Using automation levels and having an indication of the information processing workloads of the mission, the operators could establish a "contract" with the EC in the pre-mission phase. They could, through a dialogue at a computer workstation, define what autonomy they wish the EC to have as a function of flight phase and system function. As an example, weapon consent would always remain exclusively the operator's task, but reconfiguration of the UAVs flight control surfaces to get the best flight performance in the event of battle damage would be the exclusive task of the EC.

7.8.2 Adaptive Automation

Although the pre-mission contract with the EC helps to establish roles for it and the human operator, the functions allocated to each crewmember remain static throughout the mission. However, missions are highly dynamic, and, as stated before, it would be desirable to change the function allocation during the mission. This dynamic function allocation is achieved through adaptive automation. "In adaptive automation, the level or mode of automation or the number of systems that are automated can be modified in real-time. Furthermore, both the human and the machine share control over changes and the state of automation" [17, p. 43].

Two of the key aspects of adaptive automation are *when* to trigger the shift and for *how long*. The *when* aspect is discussed by Scerbo, Parasuraman, DiNocera and Prinzel [18, p. 11] who list a number of methods for triggering the shifting tasks between the operator and the automation: critical events, operator modeling, performance measurement, psychophysiological measurement and hybrid methods. A diagram of how many of these allocation methods can be used in a system is shown in Figure 7-23.

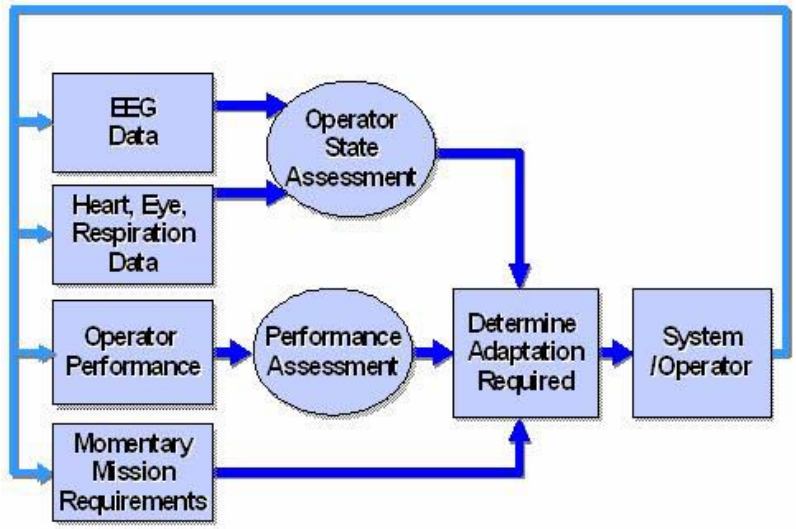

Figure 7-23: Adaptive Automation System Diagram.

As an example of how psychophysiological measurement is used to determine operator state Wilson and Russell [19] required U.S. Air Force air traffic controllers, in a simulation, to manage air traffic around the Los Angeles airport. The task loading was manipulated by the number of aircraft they had to manage

(volume) and the different kinds of aircraft they had to manage (complexity). The tasks were first given to subject matter experts, and the difficulty was increased until they verified that they were in an overload condition and could not effectively handle the traffic. The participants were then given the same type of task and their physiological data was processed by an artificial neural net. The result was the neural net could identify the non-overload condition 99% of the time and the overload condition 96% of the time. These results indicate that psychophysiological measures may be potentially very useful in determining if the operator is overloaded in real-world applications.

Once the state of the operator can be reliably assessed, the next question is, Can the workload be shifted quickly between the operator and the automation? Wilson, Lambert and Russell [20] addressed this question in a study using NASA's Multi Attribute Test Battery (MATB). There are four tasks in the MATB: tracking, systems monitoring, resource management, and communications. As in the air traffic control study previously discussed, pretest conditions were conducted to discover when the operators were overloaded, and the neural nets were used to identify this condition. In one experimental condition, the participants managed all four of the tasks, regardless of difficulty. In the other condition, when the participants reached the overload condition, the systems monitoring and communications tasks were handed off to the automation. The operator continued controlling the tracking and resource management tasks. The results showed that, relative to the manual condition, the adaptive aiding condition resulted in a 44% reduction in tracking error and a 33% error reduction in the resource management task.

The psychophysiological triggering of adaptation appears to be very promising; however, researchers are still very early in applying this technology to real-world settings. "At present, however, there is not enough existing psychophysiological research to provide adequate information on which to base adaptive allocation decisions" Prinzel, Freeman, Scerbo and Mikulka, [21, p. 407]. Although the shifting of tasks from the operator to the automation by psychophysiological methods (the *when* aspect) resulted in successful performance in Wilson et al. [20] study, there doesn't appear to be any general consensus as to *how long* the automation should keep the transferred task in order to optimize overall systems performance. The *how long* aspect has been examined by a number of authors, and the answer appears to be task specific. For example, Scallen and Hancock [22] utilized adaptive automation in a study which required pilots to perform tracking, monitoring, and targeting tasks while flying a simulator. After a target was presented, the tracking task was automated for a 20 second interval, after which it was returned to the pilot. Conversely, in another research effort [23] which looked at three different cycle times between the operator and the automation (15, 30, or 60 seconds), the 15 second switching time resulted in the best tracking performance. However, three of the five pilots who took part in the study reported that the switching back and forth was distracting. As a result, the author states that, "In the case of adaptive allocation systems we propose a moratorium strategy in which there is a minimum frequency with which the system can either assume or relinquish task control." (p. 402)

7.8.3 Putting It Together

Things are getting complicated. We now have levels of automation, human information processing models, and adaptive automation. How do we make sense of all this? Kaber, Prinzel, Wright, and Claman [24] addressed two of the three components in a study which looked at the issue of adaptive automation (AA) relative to the four stages of the information processing model. Besides a manual control condition where there was no AA, it was applied to the all the stages of the four stage model: information acquisition, information analysis, decision making, and action implementation.

The participants used Multitask which created a simulated Air Traffic Control environment. Their task was to provide a landing clearance to various aircraft depicted on the radar scope. The aircraft were flying from the

periphery to the center of the display. An error occurred if the aircraft reached the center of the display, or collided with another aircraft, before the clearance was issued. A gauge monitoring secondary task was also used. If the participant's performance on the secondary task fell below a predetermined level, the primary task would be automated. NASA's Task Load Index (TLX) was used to measure workload.

Although performance utilizing AA was superior to the manual control condition, the results showed that AA was most effective when applied to the information acquisition and action implementation information processing stages. It was not effective in the information analysis and decision making stages. The authors conclude, "All these results suggest that humans are better able to adapt to AA when applied to lower-level sensory and psychomotor functions, such as information acquisition and action implementation, as compared to AA applied to cognitive (analysis and decision making) tasks" [24, p. 23].

The Kaber et al., [24] study began to give some insight into the interaction of two components: information processing and adaptive automation – but as mentioned at the beginning of this section, there are three components, the third being levels of automation. How do they *all* fit together? Kaber and Endsley [25] attempted to show the relationship among all three factors. They constructed 10 levels of automation and an information processing model similar to Parasuraman et al., [16], with the stages being monitoring, generating, selecting, and implementing. In addition, they also incorporated adaptive automation.

They then conducted a study utilizing six levels of automation: manual, action support, batch processing, decision support, supervisory control, and full automation (numbers 1, 2, 3, 5, 9, and 10 in Figure 7-24). Manual and Full Automation are self-explanatory. Action Support is similar to tele-operation. Batch Processing requires the human to create and decide the options to implement and the computer carries these out. Decision Support involves the computer's suggesting options and once the operator selects one of these options (or one self generated), it is then put into operation by the computer. In Supervisory Control the computer generates and carries out the options. The operator monitors and gets involved if necessary. These six levels were then combined with three levels of adaptive automation cycle time (AACT) (20%, 40% and 60%).

	ROLES			
LEVEL OF AUTOMATION	MONITORING	GENERATING	SELECTING	IMPLEMENTING
1. Manual Control	Human	Human	Human	Human
2. Action Support	Human/Computer	Human	Human	Human/Computer
3. Batch Processing	Human/Computer	Human	Human	Computer
4. Shared Control	Human/Computer	Human/Computer	Human	Human/Computer
5. Decision Support	Human/Computer	Human/Computer	Human	Computer
6. Blended Decision Making	Human/Computer	Human/Computer	Human/Computer	Computer
7. Rigid System	Human/Computer	Computer	Human	Computer
8. Automated Decision Making	Human/Computer	Human/Computer	Computer	Computer
9. Supervisory Control	Human/Computer	Computer	Computer	Computer
10. Full Automation	Computer	Computer	Computer	Computer

Figure 7-24: LOA Taxonomy for Human-Computer Performance in Dynamic, Multi-Task Scenarios [25, p. 119].

For example, in a 20 minute trial the task would be allocated to the automation either 4, 8 or 12 minutes. The results showed that, "The best combination of LOA and AACT involved human strategizing combined with computer implementation (Batch processing (LOA 3)) during high automation cycle times (12 min on

cycle and 8-min off cycle)" [25, p. 147]. This result is a big step forward, but also illustrates the difficulty in implementing adaptive automation, levels of automation, and human information processing. If we put this research on a time scale relative to the over 80 years of research in the design of aircraft crewstations, we are just beginning to explore this area. So, we cannot expect instant answers to these very difficult questions. To make matters even more interesting, there are also plans to place varying levels of automation within the airborne platform.

7.8.4 Levels of Automation Within the Air Vehicle

Earlier in this section it was mentioned that there would be intelligent software both in the operator's console as well as within the UAV itself. The airborne computing system enables varying levels of autonomy called autonomous control levels (ACLs) within the UAV. Ten different levels are shown in Figure 7-25. At first glance, it would seem logical to assume that these 10 levels map onto Sheridan's 10 levels of autonomy mentioned in Factor 3: Trust and Levels of Automation. Sheridan's levels deal with the interaction *between* the operator and the UAV. However, these ACLs are referring to autonomy levels *within* the aircraft only and not between the aircraft and the operator. One of the interesting things about this chart is that the lower levels of the chart refer to the ACLs within *each* aircraft in, for example, a flight of four – but from levels five and higher, they referred to how the *entire flight* works together as a group. The ten levels of autonomy in Figure 7-25 range from Level 1: Remotely Guided (tele-operation) to Level 10: Fully Autonomous Swarms where the vehicles are acting in concert with one another to achieve a common goal.

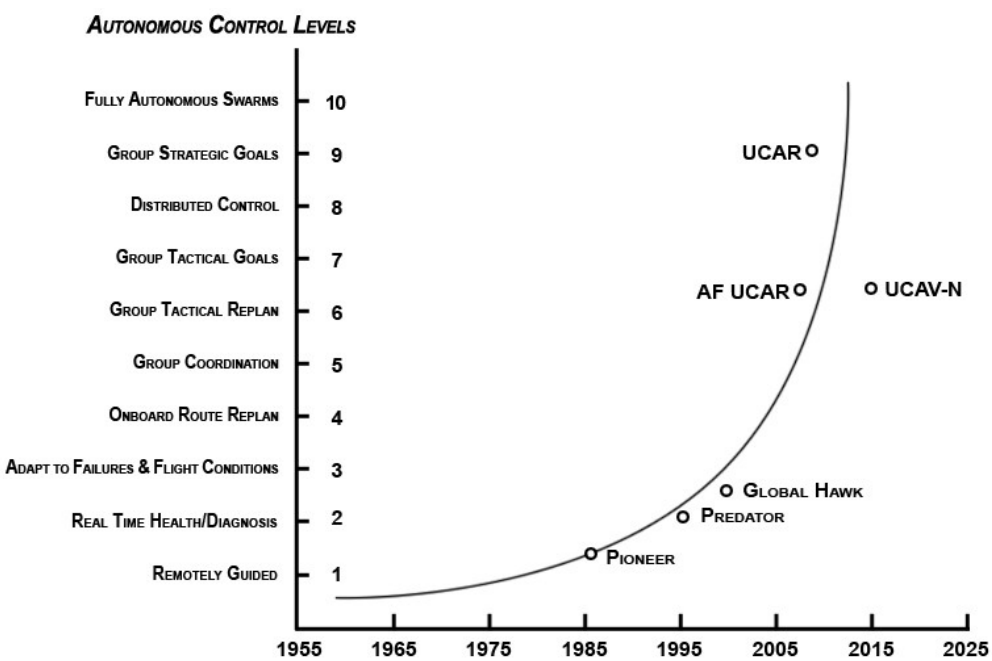

Figure 7-25: Aircraft Control Levels [from 26, p. 84].

Tele-operation has already been discussed in Factor 1: Advanced UAV Operator Control/Display Interface Technologies and will not be further enumerated upon here – but Level 10: Swarms, which offer a whole new level of control both within a group of aircraft and between that group and the operator, will be examined in more detail. The interesting thing about swarms is that there doesn't appear to be any central controller telling

the swarm what to do. If you observe a school (swarm) of fish, they just appear to act as one with no central leader fish giving them directions. The same is true for flocks of birds, groups of ants, and swarms of bees. " 'Swarming' itself is a type of emergent behavior, a behavior that isn't explicitly programmed, but results as a natural interaction of multiple entities" [27, p. 1]. As an example of forming a swarm, consider how ants communicate that they have found a source of food. What happens is that the ants lay down a pheromone trail (chemical markers) that other ants can follow. The strength of the pheromones, however, decays over time; therefore, the ant that finds the closest food supply and returns with it will have the strongest pheromone trail. Other ants will then follow this trail with no central commander ant directing them to do this [28].

So, what does this have to do with UAVs? If a flight of UAVs could act as a swarm, instead of giving them explicit, detailed instructions on the location of surface-to-air missile batteries, for example, they could be directed to just loiter about a certain area of enemy territory and if they come across the missiles they could destroy them. Of course, they would be acting within the level of responsibility given to them by the human operator. Creating digital pheromones for UAVs is one way they could communicate. These types of pheromones are not based on chemicals, but rather on the strength of electrical fields. In a computer-based (constructive) simulation, a UAV swarm using digital pheromones significantly outperformed the non-swarm case [29].

7.8.5 Conclusion

UAVs have a wide range of avionics sophistication, from the relatively basic Dragon Eye to very complex Global Hawks and UCAVs. Many of the UAVs used at the small unit level will have limited automation although, for example, they will be able to plan their own flight route. However, most future aircraft, whether inhabited or not, will contain associate systems that will incorporate varying levels of autonomy and adaptive automation as basic operating principles. These principles will enable the UAV operator and the associate to form a team consisting of two crewmembers – one human and one electronic. In order to function effectively, the operator and the EC must work together as a close-knit team, and the EC may not only supervise one aircraft, but the entire swarm. One essential feature of a successful team is trust in the other partner. As an example, guidelines to create such trust must include specifying the EC's level of autonomy. By using these guidelines, a high-quality, trusting relationship can be achieved between the operator and the EC. This internal trust will, in turn, lead to an efficient and effective team which can operate successfully in a system of systems environment.

7.8.6 References

[1] Schweitzer, R. (2003). Big Bucks for the Best There Is, Armed Forces Journal, June pp. 24-28.

[2] Tirpak, J. (2002). Long Arm of the Air Force, Air Force Magazine, October, 28-34.

[3] Hebert, A. (2003). The Long reach of heavy Bombers, Air Force Magazine, November, pp. 24-29.

[4] Draper, M.H., Ruff, H.A., Fontejon, J.V. and Napier, S. (2002). The effects of head-coupled control and head-mounted displays (HMDs) on large-area search tasks. Proceedings of the Human Factors and Ergonomics Society 46th Annual Meeting, 2139-2143.

[5] Calhoun, G.L., Draper, M.H., Ruff, H.A., Fontejon, J.V. and Guilfoos, B. (2003). Evaluation of tactile alerts for control station operation. Proceedings of the Human Factors and Ergonomics Society 47th Annual Meeting, 2118-2122.

[6] Draper, M., Geiselman, E., Lu, L., Roe, M. and Haas, M. (2000). Display concepts supporting crew communication of target location in unmanned air vehicles. Proceedings of the Human Factors and Ergonomics Society 44th Annual Meeting, 385-388.

[7] Draper, M., Calhoun, G., Ruff, H., Williamson, D. and Barry, T. (2003). Manual versus Speech Input for Unmanned Aerial Vehicle Control Station Operations. Proceedings of the Human Factors and Ergonomics Society 47th Annual Meeting. Denver, CO: HFES.

[8] Association of Unmanned Vehicle Systems International: Unmanned Systems in a New Era, AUVSI Second Annual Ground, Sea and Air Conference (2002), February 12-14, Washington, DC.

[9] Barbato, G., Feitshands, G., Williams, R. and Hughes, T. (2003). Operator Vehicle Interface Laboratory: Unmanned Combat Air Vehicle Controls and Displays for Suppression of Enemy Air Defences, In: Proceedings of the 12th International Symposium on Aviation Psychology, Dayton, Ohio.

[10] Geddes, N. (1997). Associate systems: a framework for human-machine cooperation. In: Smith, M., Salvendy, G. and Koubek, R. (Eds.) Designing of Computing Systems: Social and Ergonomic Considerations. Amsterdam: Elesvier.

[11] Hancock, P.A. and Scallen, S.F. (1996). The future of function allocation. Ergonomics In: Design, Q4, 24-29.

[12] Taylor, R. (1993). Human factors of mission planning systems: Theory and concepts. AGARD LS-192 New Advances in Mission Planning and Rehearsal Systems, 2-1 – 2-22.

[13] Krobusek, R.D., Boys, R.M. and Palko, K.D. (1988). Levels of autonomy in a tactical electronic crewmember. Proceedings of The Human – Electronic Crew: Can They Work Together?, 124-132 (Tech. Rep. WRDC-TR-89-7008). Wright-Patterson Air Force Base, OH: Cockpit Integration Directorate.

[14] Sheridan, T.B. (1980). Computer control and human alienation. Technology Review, October, 61-73.

[15] Endsley, M.R. (1996). Automation and situational awareness. In: R. Parasuraman and M. Mouloua (Eds.), Automation and human performance: Theory and applications (pp. 163-181). Mahwah, NJ: Erlbaum.

[16] 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 – Part a: Systems and Humans, Volume 30, No. 3, May, 286-297.

[17] Scerbo, M.W. (1996). Theoretical Perspectives on Adaptive Automation. In: Parasuraman, R. and Mouloua, M. (Eds.) Automation and Human Performance, 37-63.

[18] Scerbo, M.W., Parasuraman, R., DiNocera, F. and Prinzel, L.J. (2001). The Efficacy of Physiological Measures for Implementing Adaptive Technology, NASA/TP-2001-211018, p. 11.

[19] Wilson, G.F. and Russell, C.A. (2003). Operator Functional State Classification Using Multiple Psychophysiological Features of an Air Traffic Control Task, Human Factors, Fall 45 (3), 381-389.

[20] Wilson, G.F., Lambert, J.D. and Russell, C.A. (2000). Performance Enhancement with Real-time Physiologically Controlled Adaptive Aiding, In: Proceedings of the IEA 2000/HFES 2000 Congress, Human Factors and Ergonomics Society, Santa Monica, California, 3-61 – 3-64.

[21] Prinzel, L.J., Freeman, F.G., Scerbo, M.W. and Mikulka, P.J. (2000). A Closed-loop System for Examining Psychophysiological Measures for Adaptive Task Allocation. The International Journal of Aviation Psychology, 10 (4), 393-410.

[22] Scallen, S.F. and Hancock, P.A. (2001). Implementing Adaptive Function Allocation, The International Journal of Aviation Psychology, 11 (2), 197-221.

[23] Scallen, S.F., Hancock, P.A. and Duley, J.A. (1995). Pilot Performance and Preference for Short Cycles of Automation in Adaptive Function Allocation. Applied Economics Vol. 26, Number 6, 397-403.

[24] Kaber, D.B., Prinzel, L.J., Wright, M.C. and Claman, M.P. (2002). Workload-Matched Adaptive Automation Support of Air Traffic-control or Information Processing Stages, NASA/TP-2002-211932, September 2002.

[25] Kaber, D.B. and Endsley, M.R. (2004). The effects of level of automation and adaptive automation on human performance, situational awareness and workload in a dynamic control task. In: Theoretical Issues in Ergonomics Science. Taylor & Francis Volume 5, Number 2 / March-April, pp. 113-153.

[26] OSD UAV roadmap, 2002-2007, December 2002, p. 84, http://www.acq.osd.mil/usd/uav_roadmap.pdf

[27] Clough, B. (2002). UAV's Swarming? So what are those swarms, what are the implications, and how do we handle them? AUVSI Unmanned Systems 2002 proceedings. Association for Unmanned Vehicle Systems International. July 2002, pp. 1-15.

[28] Bonabeau, E. and Theraulaz, G. (2000). Swarm Smarts. Scientific American, March 2000, pp. 72-79.

[29] Parunak, H., Purcell, M. and O'Connell, R. (2002). Digital Pheromones for Autonomous Coordination of Swarming UAVs. American Institute of Aeronautics and Astronautics, AIAA 2000-3446, pp. 1-9.