Learning from pilot performance

Christophe Doniat
Universite technologique de Troyes, Laboratoire Tech-CICO
12, rue Marie Curie, BP 2060, 10010 Troyes Cedex, France
doniat@wanadoo.fr

Abstract

This paper presents a systematic flight analysis methodology through the underlying ETHOS model that we evolved. This model provides us a first keystone to understand how the human pilot capture and build her/his environment through complex environment. We will discuss the identified performances and potential deviations and associated situations.

Introduction

Overview

In the aeronautics community, airplanes become increasingly reliable thanks to technological advances however with such advances in machinery we must also regard human, particularly pilot, behaviour. Among others the phenomenon of human error (Reason 1990) in steering activity to cause air accidents is a case in point. One possible cause is that the pilot might discover meaningful strategies/knowledge through her/his experiment rather than by the application of instructions contained within training sessions and training manuals. These 'personal' strategies and knowledge are difficult or even impossible to anticipate on the part of manufacturers. Thus, it behooves us to have several means to uncover real pilots' behaviours (known, unforeseen). The feedback for air safety introduces light and shade which seems to be a relevant approach into the understanding of this phenomenon so that sensible decisions can be made about interventions in Human-Machine-Interaction.

Basically, this approach aims to provide - it is early to talk about a *rule* at this stage - an improved vision (in terms of feedback) on the daily practical steering of the human pilot in order to build models of more accurate standard procedures than those found in the *general steering handbooks* currently available.

To sketch this, we have developed a systematic flight analysis methodology centred on human factors, called S-ETHOS. This tries to draw a scheme of activity of the human pilot after each mission as recorded by the flight recorder. Pilot activity consists of several tasks (motor acts, checking, looking) under physiological, psychological and physical factors.

This methodology (i) simulates standard pilot behaviour (ii) records real flight behaviour, (iii) compares the both, (iv) shows the deviations and finally (v) analyses some of them. Our motivation is to use these analyses to help us to improve procedures and/or modify system devices in the cockpit.

In addition, deviations had tried to show which cognitive processes were involved in the error production during a *decision cycle*[1]. A detailed classification of errors was described in (Reason 1990) covering some of errors types. Some examples of architectures include ACT-R (Byrne 1997), MIDAS (Corker 1993), TacAir Soar (Jones 1999), Pilot's Associate project (Rouse 1988), Copilote Electronique project (Amalberti 1992), ADAPT (Doane 2000). The common features of these systems are: they depict the situation awareness in different ways in the memory which is simulated. Then they detect, either interferences (e.g. decay at least which is a primary function of forgetting), and/or typical behaviours (e.g. latencies or uncertainties management). Broadly, ACT-R model uses a causal induction theory allowing the knowing of some types of knowledge are learned and used when pilots try to control complex systems. MIDAS relies on a problem-solving theory to study the workload. TacAir Soar uses a cooperative learning and explanation-based learning theories to depict chunkings. Pilot's associate model uses knowledge compilation and explanation-based learning theory as well. Copilote Electronique model uses a cooperative learning and a multimodal learning theories to study the learning processes in an organisation that makes safety-critical decisions can be improved by the

[1] This concept depicts the process during which a task is dealt with.

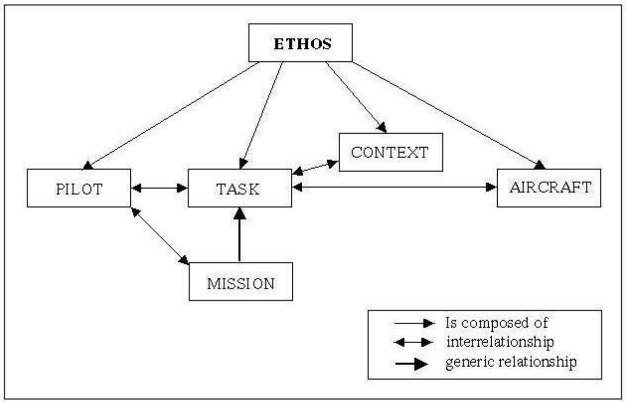


Figure 1: ETHOS model.

careful implementation of decision traceability mechanisms. ADAPT model uses a context-sensitive and an adaptative problem-solving theories.

The idea

The present research uses a theory of knowledge acquisition with frame-based representation to study a human pilot's performance during a flight mission. That is, her/his knowledge (e.g. aircraft generic concept, task generic concept, fig 1). We measured her/his performance between a standard performance - customized according to her/his profile - and a real performance. Then, we analysed the different between the both prior performances and obtained a model of expertise according to the application of a generic concept of interference (one of forgetting generic concepts in the working memory studies). Finally, we obtained some primary results shown by the using of this interference.

ETHOS: a model of performance

The ETHOS model (Doniat 1998, 1999), which describes the performance of aircraft pilots in piloting[2] situations, was elaborated through extracting and modelling the knowledge of pilots (38 subjects from Ecole de l'Air[3]) immersed in their environment and considered as experts in aeronautics. We proceeded flight simulations as well on the *take-off* task. The Fig. 1 shows the ETHOS model that contains the concept *Pilot* who can do a *Mission* with an *Aircraft*. *Mission* is composed by a combination of *Task(s)*. Each of task requires *Precondition* and *Postcondition* (*Context*) to be respectively fired and taken place.

[2]Piloting is one of steering activities with at least navigating, using radar.

[3]Ecole de l'Air: school of Air French Army.

These generic concepts shape an application Ontology[4] (Doniat 2002) allowing us to describe pilot performance in any aircraft. A standard task 'take-off' can be described as follows (excerpt):

```
take-off = {
  Look_Position_Heading,Check_Position_Heading,
  ThrottleGear_onPosition_PCMini,
  Look_Position_RPMHaircross,Check_Position_RPMHaircross,
  Look_Position_RPMDigit,Check_Position_RPMDigit,
  Look_Position_EngineTemp,Check_Position_EngineTemp,
  Look_Position_PetrolDigit,Check_Position_PetrolDigit,
  Look_Position_Airspeed,Check_Position_Airspeed,
  Look_Position_Acceleration,Check_Position_Acceleration,
  Look_Position_Airspeed,Check_Position_Airspeed,
  Theta_on_Horizon,...
}
```

We described also formally the ETHOS generic concepts using an object-oriented approach relying on a prototype-based representation called OBJLOG II+ (Faucher 1991). Further detailed representations on the ETHOS model are available in (Doniat 2000).

Performance simulation

In this section, with respect that the ETHOS model is a representation of the pilot's memory, we describe the simulation process of the memory and its related performances. Among our assumptions, we considered the pilot could not suffer any stress or fatigue. The team-training[5] helped us to generate the standard behaviour of a pilot during the take-off task on a flight simulator. The team also explained some part of this performance completing our ETHOS model. The configuration of the aircraft was that given in the handbook and the meteorology conditions were optimal (no disturbance). The experiment generated an activity dataset. The team-training extracted the take-off dataset. We studied with it the pilot's memory, the cycle of treatment of each task, the building of the planned performance through the generic concept Mission and the capacity limitations of the memory.

Memory architecture and relationships

Memory refers to the storage, retention and recall of knowledge. A pilot is instantiated and there is her/his expertise - the whole application ontology - in LTM[6] within precondition, postcondition and task generic concepts to steer a given aircraft. The given aircraft is also instantiated. A pilot's performance will successively bring into play the following activities. The pilot prepares the mission, i.e. defines the tasks to be carried out from the mission description. Then the pilot evolves the planned mission, i.e. the pilot recalls the description of tasks from LTM , fits them to the

[4]Application Ontology as a description of the task or method Ontology which is the shared knowledge between all pilots like a generic expertise.

[5]Team-training was composed of ONERA engineers, instructors, flight officer, air safety officer and training officer.

[6]LTM: Long-Term Memory

context and then carries them out. All this results in having a *'planned mission'* that will be *'carried out'* and will correspond to the *'real mission'*. The STM[7] (middle - right) describes a pilot's STM and is dynamically related to the mission generic concept. Its aim is to record a planned mission for a while corresponding at the duration of the real mission. This generic concept is underlying because it is connected to the LTM to recall a number of task descriptions that will compose the mission. The STM includes the ST-WM[8] in which each task description is implemented as the pilot carries out the mission. It refers to memories which last for a few minutes. ST-WM is of limited capacity, usually 5-9 items (7±2). Beyond this capacity, new information can overlap other items from ST-WM. This is one form of *forgetting*. This is a major issue as each task might be perturbed causing potential dysfunctions. The other part of the working memory is located in the LTM and called the LT-WM[9] allowing to recall the description of each task. The behaviour of the ST-WM (especially the *central executive* concept) is similar as the one described in (Baddeley 1999). At the level of perception, the behaviour of the *conceptual buffer* concept is similar as the one described in (Kintsch 1998). It is also dynamically related to the task generic concept. The ST-WM was limited up to 7 items at the same time.

A construction-integration cycle

The ST-WM model - a cyclic mechanism - uses the following steps to take each task from the mission description, evolve it during a C/I cycle[10] and store it in the planned mission.

Compute the whole mission description (intentions) as follows.

Step 1: each task is a stimulus that inputs sequentially in ST-WM.

Step 2: instantiate the current task name.

Step 3: instantiate the current's precondition without factors (temporal, environmental and human) according the parameters of the task.

Step 4: with this instance (from step 3), compute the rule that seeks this knowledge in the LTM through the precondition collection. When it found it, the rule recalls the definition of the associated task in ST-WM and instantiated it partially, i.e. instantiate the precondition with the existing factors and the task itself. If the prior postcondition exists and if it exists a list of factors (temporal, environmental and human or even prior task aspects) then complete or create the current precondition of this task. For example, the ending date of the prior task will become the implementation date of the new task.

Step 5: then, a rule instantiates completely the task, i.e. instantiate the postcondition to conclude the task. According the list of human factors, some of them might be transmitted in the postcondition according their models[11]. After that, the next task might be used them. For example, the physical factor *fatigue* does not vanish after its creating because it holds concurrently. This factor could be creating a *interference* (Baddeley 1999) in the ST-WM. In case of the CT task, if some required outputs states do not have achieve, they are also transmitted in the postcondition to be use before the next task.

Step 6: finally, a rule stores this instantiated task in the planned mission.

Fig. 2 (below) synthesises the standard behaviour with the list of tasks carried out. Column called *Type* contains the type of each task. We grouped together the types BLT and BCT because they carry on the same object, i.e. the pilot looks at this object, then checks the state of this same object with a given reference value. The type BAT describes each motor act on a control. The column called *Stimuli*[12] gives the list of tasks contained in the take-off task. Thus we have the list of tasks, their order, their starting date and their state. The take-off task will be over when the pilot will achieve the tasks giving the following output states from the precondition:

```
list{Throttle(PGS),ThetaOnHorizon(20),LandingGear(OFF),
  Airspeed(300),Height(1500)}
```

We noticed that the pilot inferred a set of conditions through the BLTs, BCTs and BATs to fire the task whose its output state is related with one of the output states to reach. This set is called *situation* by the novice pilot and *recipe*[13] by the expert pilot. For example, line 1 introduces two items in the ST-WM about *Heading* to shape a *situation*. Then the BAT

[7]STM: Short-Term Memory

[8]ST-WM: Short-Term Working-Memory.

[9]LT-WM: Long-Term Working Memory.

[10]We borrowed this concept from (Kintsch 1998).

[11]These models do not exist in this research. However, the ETHOS model is extensible and could embody them.

[12]We use the term Stimuli because each task carried out by the pilot enters the working memory as described above

[13]The expert pilot explains this because I wanted to produce a shape, which could be used to fire other tasks. For example, it might give her/him more time to treat the future goal. The novice pilot explains this because s/he follows the building of the task. Here, the standard pilot rules her/his activity with the situation representation.

		TYPE	STIMULI	STANDARD BEHAVIOUR			(next)	TYPE	STIMULI	STANDARD BEHAVIOUR		
				Duration: 0.38 mn								
T				Order	Date	Value				Order	Date	Value
A	1	BLT-BCT	Heading	1	3'45"	150	12	BAT	Trigger LG	12	4'01"	0
K	2	BAT	Throttle on PCMini	2	3'45"	1	13	BLT-BCT	Light LG li	13	4'01"	1
E	3	BLT-BCT	RPM haircross	3	3'47"	100	14	BLT-BCT	RPM Hairc	14	4'01"	100
-	4	BLT-BCT	RPM Digit	4	3'47"	100	15	BLT-BCT	RPM Digit	15	4'01"	100
O	5	BLT-BCT	Engine temperature	5	3'48"	294	16	BLT-BCT	Engine ter	16	4'01"	294
F	6	BLT-BCT	Petrol Digit	6	3'48"	280	17	BLT-BCT	Petrol Digi	17	4'01"	280
F	7	BLT-BCT	Airspeed	7	3'50"	70	18	BAT	Throttle on	18	4'09"	0.5
	8	BLT-BCT	Acc.	8	3'50"	0.8	19	BLT-BCT	Light PC u	19	4'09"	0
	9	BLT-BCT	Airspeed	9	3'53"	120	20	BLT-BCT	Airspeed	20	4'25"	300
	10	BAT	Theta on Horizon	10	4'00"	20	21	BLT-BCT	Height	21	4'25"	1500
	11	BLT-BCT	Height	11	4'00"	0						

Figure 2: Experimental results for take-off task.

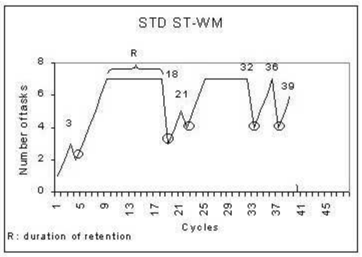


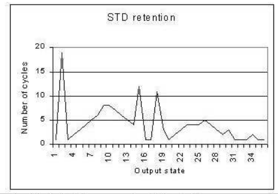
Figure 3: Forgetting cycle in take-off task.

Throttle on PCMini is fired creating the output state *Throttle(PCMini)*. (Notice however that situation is no relevant to fire this BAT and some other output states - no present here - belong to the prior task called *alignment*). Next, lines 3-7 introduce successively 14 items and the ST-WM keeps only up to 7 items at the same time by overlap, which fire the BAT *Theta on horizon* and so forth.

Managing the capacity of ST-WM

The ST-WM model uses the following steps to rule its limited capacity up to 7 items at the same time. This process carries on until the defined output states are reached, i.e. the prior precondition list. Fig. 3 shows the cycles of filling/emptying of the ST-WM after each output state was reached. Numbers (X-axis) depict the cycle when a task took place. Circles depicts another *forgetting* effect by emptying the ST-WM. In fact, each reached output state is an event (from a stimulu) which interferes the ST-WM as suggested in (Baddeley 1999) giving this forgetting. Moreover, **R** depicts an amount of cycles during which the knowledge about a *situation* is hold in ST-WM. This is one element which allows the calculation of the pilot's workload.

Since the pilot build *situations*, s/he must retain several knowledge in ST-WM until each situation is shaped. The *retention* concept is thus underlying. Fig. 4 depicts the number of output states managed during the *take-off* task and the number (max.) of cycles for each of them according its output states distribution.



	Output state		Output state		Output state		Output state
1	heading(150)	11	check_petrol.digit(280)	21	light.LG.lighted(on)	31	light.LG.unlighted(on)
2	check_heading(150)	12	airspeed(70	22	check_light.LG.lighted(on)	32	check_light.LG.unlighted(on)
3	throttle(PCMini)	13	check_airspeed(70)	23	RPM.haicross(100)	33	airspeed(300)
4	RPM.haicross(100)	14	acc(0.8)	24	check_RPM.haircross(100)	34	check_airspeed(300)
5	check.RPM.haircross(100)	15	check_acc(0.8)	25	RPM.Digit(100)	35	height(1500)
6	RPM.Digit(100)	16	Theta.on.horizon(20)	26	check_RPM.digit(100)	36	check_height(1500)
7	check.RPM.Digit(100)	17	height(0)	27	temperature(294)		
8	temperature(294)	18	check_height(0)	28	check_temperature(294)		
9	check_temperature(294)	19	trigger.LG(off)	29	petrol.Digit(280)		
10	petrol.Digit(280)	20	check_trigger.LG(off)	30	check_petrol.Digit(280)		

Figure 4: Retention rates during take-off task. For example, *Throttle* output state is hold for 19 cycles in ST-WM.

Step 1: fill the ST-WM eventually up to 7 items maximum, i.e. either by inserting the output state of the fired current task and by deleting the older and unrelated output state, or if the same output state exists then updating of its value.

Step 2: if a fired task gives an output state that is related with one of the output states to reach then deleting of the output states that are not related with the output states to reach and keep the other output states already created.

We described the execution of the standard pilot's performance which provided valuable information for the assessment of the ETHOS model. That is, the building of *situations* before to fire the tasks towards the expected output states and the used cycles of the ST-WM. The performance simulation has been done without any human factors since we did not have any models and then the experiment was optimal. However, the ETHOS model is extensible by taking into account these models through the precondition/postcondition generic concepts.

Experimentation: modelling and evaluation

Modelling

We have tested and checked the model relevance in the study of real pilot's performance. We used three experiments already executed by the participants earlier on Mirage M2000D flight simulator. The study presented here shows results from comparison between the real pilot performance during the take-off task and the standard take-off task described in the prior section. Our study took one month to make analyzes and simulations. The pilots could not suffer any stress and

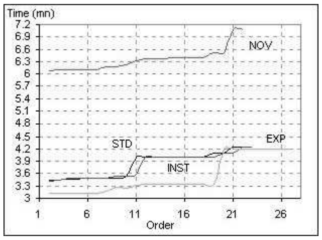


Figure 5: Four different performances of the take-off task.

		TYPE	STIMULI	BEHAVIOURS											
				STANDARD			NOVICE			INSTRUCTOR			EXPERT		
				Order	Date	Value	Order	Date	Value	Order	Date	Value	Order	Date	Value
T	1	BLT-BCT	Heading	1	3'45"	150	6	6'10"	150	1	3'41"	150	1	3'11"	150
A	2	BAT	Throttle on PCMini	2	3'45"	1	1	6'08"	1	2	3'44"	1	2	3'11"	1
K	3	BLT-BCT	RPM haircross	3	3'47"	100	2	6'09"	100	3	3'48"	100	3	3'11"	100
E	4	BLT-BCT	RPM Digit	4	3'47"	100	3	6'09"	100	4	3'48"	100	4	3'11"	100
-	5	BLT-BCT	Engine temperature	5	3'48"	294	4	6'09"	294	5	3'48"	294	5	3'11"	294
O	6	BLT-BCT	Petrol Digit	6	3'48"	280	5	6'09"	282	6	3'48"	165	6	3'11"	164
F	7	BLT-BCT	Airspeed	7	3'50"	70	7	6'17"	70	7	3'50"	70	7	3'18"	70
F	8	BLT-BCT	Acc.	8	3'50"	0.8	8	6'18"	0.42	8	3'53"	0.91	8	3'27"	0.52
	9	BLT-BCT	Airspeed	9	3'53"	120	9	6'23"	127	9	3'54"	132	9	3'25"	123
T	10	BAT	Theta on Horizon	10	4'00"	20	11	6'36"	17.72	1	4'00"	15.66	10	3'31"	20
A	11	BAT	Theta on Horizon							12	4'00"	16.32	18	3'35"	23
S	12	BLT-BCT	Height	11	4'00"	0	10	6'33'	0	10	3'59"	0	11	3'34"	0
K	13	BAT	Trigger LG on off	12	4'01"	0	12	6'37"	0	13	4'00"	0	12	3'34"	0
	14	BLT-BCT	Light LG lighted	13	4'01"	1	13	6'37"	1	14	4'00"	1	13	3'34"	1
	15	BLT-BCT	RPM Haircross	14	4'01"	100	14	6'40"	100	15	4'01"	100	14	3'34"	100
	16	BLT-BCT	RPM Digit	15	4'01"	100	15	6'40"	100	16	4'01"	100	15	3'34"	100
	17	BLT-BCT	Engine temperature	16	4'01"	294	16	6'40"	294	17	4'01"	294	16	3'34"	294
	18	BLT-BCT	Petrol Digit	17	4'01"	280	17	6'40"	282	18	4'01"	281	17	3'34"	272
	19	BAT	Throttle on PGS	18	4'09"	0.5	18	6'52	0.53	19	4'09"	0.52	19	4'19"	0.52
	20	**BLT-BCT**	**Light PC unlighted**	19	4'09"	0	19	6'52"	0	20	4'09"	0	24	4'19"	0
	21	**BLT-BCT**	**Airspeed**	20	4'25"	300	20	7'09"	333	21	4'25"	411	25	4'19"	347
	22	**BLT-BCT**	**Height**	21	4'25"	1500	21	7'09"	1500	22	4'25"	1500	26	4'19"	1500
N	23	BLT-BCT	RPM haircross										20	4'19"	97
E	24	BLT-BCT	RPM Digit		0.38 mn			0.52 mn			0.34 mn		21	4'19"	97
X	25	BLT-BCT	Engine temperature										22	4'19"	272
T	26	**BLT-BCT**	**Petrol Digit**										23	4'19"	78
-	27	...													
T														1.07 mn	
A															
S															
K															

Figure 6: Experimental results of performances.

the fatigue could not be analyzed because the pilots knew they were in the flight simulator.

Evaluation

The team-training helped us to understand different performances during the take-off task performing by the pilots. The pilots also had brought the comments about their activity.

Fig. 5 shows four different performances of *Take-off* task in an aircraft of the Mirage 2000 family. The first simulation allowed simulating and generating automatically the standard performance (STD) as explained earlier. This performance is the reference for comparison. Then, three series of simulations with human pilots (novices (NOV), instructors (INST) and experts (EXP)) generated data sets for given parameters.

Fig. 6 synthesizes these performances with the list of tasks carried out compared to time and to order. The legend is the same as described in section 5.2. We compare each performance carried out by human pilots with the standard performance. This mission began at the same time in each performance. The configuration of the aircraft was the same for each simulation. The meteorology conditions were also the same.

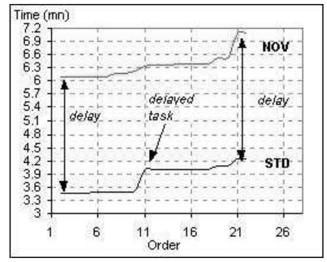


Figure 7: Comparison standard and novice performances.

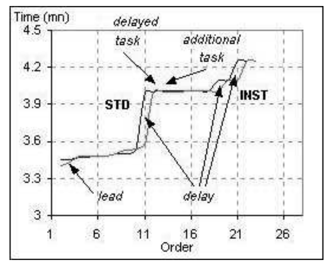


Figure 8: Comparison standard and instructor performances.

Fig. 7 shows standard performance (STD) and novice one (NOV). Note that the novice pilot begins his task later. S/he explains this because s/he spent time to check the running of her/his aircraft, especially the engine. The amount of delay has caused some deviations on some parameters used. For instance, in Fig. 6, attitude is 17.72 instead of 20, Petrol Digit is 282 instead of 280, Acc. is 0.42 instead of 0.8, Airspeed is 127 instead of 120. S/he explained that s/he checked Heading later because s/he had checked this parameter in the previous composite task called *Alignment*. Also, the novice took more time to finish the composite task Take-off. This performance is relevant because it shows that the novice pilot follows the *situation* instead a *recipe*.

Fig. 8 shows standard performance (STD) and instructor one (INST). The instructor pilot carried out her/his *take-off* task emulating the standard performance. S/he started her/his task early, the deviation is -0.04, s/he finished at the same time as the standard performance. Note that (see Fig. 6) s/he checked one parameter called *Height* just before the take-off and s/he used an additional task called *Theta on horizon* to display theta. In spite of the overall performance of task, some deviations occurred. For instance, Petrol Digit is 165 instead of 280. However, Acc. is 0.91 instead of 0.8, Airspeed is 132 instead of 120, Attitude

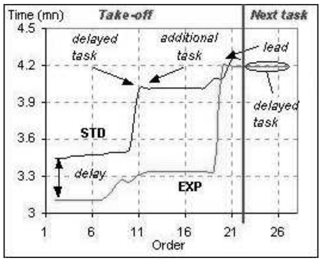


Figure 9: Comparison standard and expert performances.

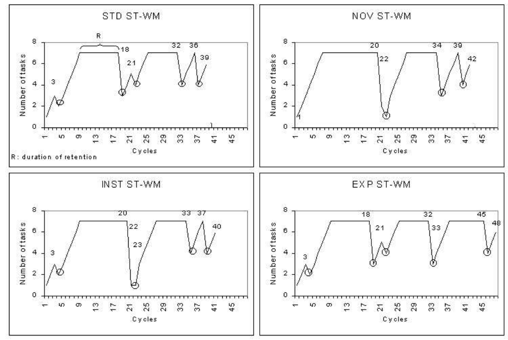


Figure 10: Forgetting cycles of four pilots' behaviours (comparison).

is 15.66, then 16.32 instead of 20.

Fig. 9 shows standard performance (STD) and expert one (EXP). The expert pilot carried out her/his take-off task early and finished early. The starting deviation is -0.34, s/he finished close to the standard performance time (-0.06). S/he explained her/his performance as follows: s/he used more time to check the running of engine, take-off, check the running of engine again, delayed some checks about the ending of take-off in the next composite task. This point can be seen (see Fig. 6) because, s/he used *Theta on horizon* task twice, one to take-off (order 10) and one after having made *Trigger LG on off* and some checks on the running of the engine (order 18). Typically, some deviations occurred. Petrol Digit is 164 instead of 280, Acc. is 0.52 instead of 0.8, Airspeed is 123. S/he respected attitude value at 20 deg. However, s/he used again attitude value at 23 deg. to take off quickly. Finally, s/he delayed the ending of take-off task in the beginning of the next task: note shaded tasks 24, 25, 26 in Fig. 6.

These two performances are relevant since they follow a *recipe* to rule the take-off task.

Categories of deviations

The Fig. 6, 7, 8 and 9 show four different performances. In STD, NOV, INST and EXP as graphs taking data from Fig. 6. In fact, Fig. 5 shows profiles of all four. Fig. 7 shows a comparison between standard and novice profile, Fig. 8 shows a comparison between standard and instructor profile and Fig. 9 between standard and expert profile.

Deviations in activity Fig. 7, 8 and 9 highlight four kinds of deviations: delay, alternate ordering of performance or delayed, additional tasks, lead and alternate value instead of standard value. The delay is when the starting date of the task is later compared to the standard starting date. The alternate ordering of performance or delayed is when the order of the tasks is different compared to the standard order. The additional task is when one task is added into the set of tasks, either different, or repeated. The lead is when the starting date of the task is earlier compared to the standard starting date. The alternate value is when a state of task takes another value rather than that expected. They can be combined together or not. The expert pilot uses delay and/or lead when (i) he is not ready to do the task, (ii) there is an opportunity to do the task in advance. We found that the expert pilot uses several strategies connected with models of dysfunctions to monitor his activity.

Deviations in memory As shown in Fig. 6, STD, NOV, INST and EXP pilots use different ways to manage the *take-off* task according to their expertise. This results in having different behaviours in terms of *forgetting* effect and *retention cycle* of output states in ST-WM.

Fig. 10 shows the four performance of ST-WMs as the forgetting effect is hold. STD one has been described in the section 3.3. NOV, INST and EXP ST-WMs are the different performance during the activity of the pilots. We observed how the pilots have seen each situation and when they obtained the required output states in the *take-off* task. As a result, the updating or adding of an output state interferes the retention of a *situation* in ST-WM.

Broadly, Fig. 11 shows the different retentions rates of the output states in ST-WM during the *take-off* task as explained before in section 3.3. Note that the output state axis does not follow the same distribution that one displayed in section 3.3. This has to be associated with the different *situations* proposed by each pilot.

The study of the pilot turns up two types of performance: *situation-based* and *recipe-based*. These kinds of performances use the mechanisms of forgetting at least to take place.

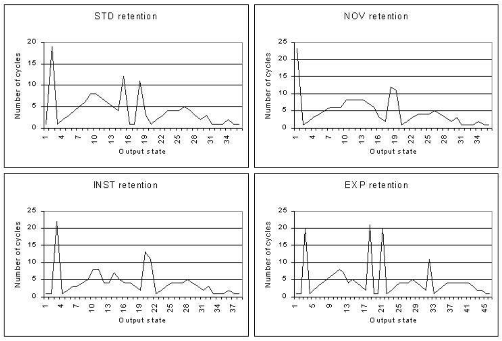


Figure 11: Overall knowledge retention rate during the take-off.

Conclusion and future work

We have presented here the preliminary results about a pilot's performance. First at all, we evolved the basic steps of a methodology for a systematic flight analysis with ETHOS (performance model) and its related simulation. Secondly, we have identified - through one of forgetting mechanisms - an implemented expertise of the pilot. This forgetting is using implicitly by the novice pilot and explicitly by the expert pilot when s/he can do it. We plan to make a further work to find out the landmarks of this interference at least.

References

Amalberti, R., and Deblon, F. 1992. Cognitive modelling of the fighter aircraft's control process; a step toward intelligent on board assistance systems. *International Journal of Man-Machine Studies* 36:639–661.

Baddeley, A.D. 1999. Essential of human memory. Psychology Press. *Taylor and Francis Group.*

Byrne, M.D., and Anderson, J.R. 1997. Perception and action. *Series: The Atomic Components of Thought.* Lawrence Erlbaum Associate, Inc.

Corker, K.M., and Smith, B. 1993. An Architecture and model for cognitive engineering simulation analysis: Application to advanced aviation analysis. *AIAA Conference on Computing in Aerospace.*

Doane, S.M, and Sohn, Y. W. 2000. ADAPT: A Predictive Cognitive Model of User Visual Attention and Action Planning. *User-Modeling and User-Adapted Interaction* 10(1):1–45.

Doniat, C. 1998. Cooperation Conceptual Model of the Flight Pilots behaviour. *Proceeding of 3rd Int. Conf. On the Design of Cooperation Systems, Vol. II.*

Doniat, C. 1999a. Contribution a l'elaboration d'une methodologie d'analyse systematique des vols centree facteur humain: le systeme S-ETHOS. *Ph.D Thesis. Universite de Droit, d'Economie et des Sciences d'Aix-Marseille III. Faculte des Sciences et des Techniques de Saint-Jerome, 98AIX30081, Marseille (France).*

Doniat, C. 1999b. The S-ETHOS System : A Tool and An Investigation Method to Systematic Flight Analysis Centred on Human Factors. *Proceeding of KAW'99: 12th Workshop on Knowledge Acquisition, Modeling and Management.*

Doniat, C. 2000. The S-ETHOS System : A Tool and An Investigation Method to Systematic Flight Analysis Centred on Human Factors. *Proceeding of AVT Symposium: UVA military operations.*

Doniat, C. and Aylett, R. 2002. KA-TOOL and Domain Construction for AI Planning Application. *ES2002: The 22nd SGAI International Conference on Knowledge Based System and Applied Artificial Intelligence.* Cambridge (UK).

Faucher, C. 1991. Elaboration d'un langage extensible fonde sur les schemas : le langage OBJLOG+. *PhD Thesis. Universite de St-Jerome. Aix-Marseille III.*

Jones, R. M., Laird, J. E., Nielsen, P. E., Coulter, K. J., Kenny, and P., Koss, F. V. 1999. Automated Intelligent Pilots for Combat Flight Simulation. Corker, K.M., and Smith, B. 1993. An Architecture and model for cognitive engineering simulation analysis: Application to advanced aviation analysis. *AI Magazine* 20(1):27–41.

Kintsch, W. 1998. Comprehension - A paradigm for cognition. *Cambridge University.* Cambridge University Press.

Reason, J. 1990. Human Error. *Cambridge University.* Cambridge University Press.

Rouse, W. B., Geddes, and N. D., Curry, R. E. 1988. An architecture for Intelligent Interfaces: Outline of an Approach to Supporting Operators of Complex Systems. *Human-Computer Interaction* 3(2):87–122.

Acknowledgements

This research has been carried out during a PhD thesis (Doniat 1999a, 1999b, 2000) (funded by the French Ministry of Defense, especially DGA/STTC[14] to improve the air safety in French Air Force Army (1995)) at ONERA-Salon[15] (French National Aerospace Research Establishment). We are grateful to Professor Ruth Aylett from Centre for Virtual Environments in Salford University for her comments on earlier versions of this article. Our thanks go to ONERA-Salon engineers for help with production of this research. We would like to thank the referees for valuable criticisms.

[14]DGA-STTC: Direction Generale de l'Armement, Service Technique des Technologies Communes

[15]ONERA-Salon and ONERA Human Factors: http://www.onera.fr/dcsd/facteurshumains