Simulating Perceptive Processes of Pilots to Support System Design

Andreas Lüdtke and Jan-Patrick Osterloh

OFFIS Institute for Information Technology, Escherweg 2,
26127 Oldenburg, Germany
{luedtke, osterloh}@offis.de

Abstract. In this paper we present an approach towards supporting the ergonomic design of aircraft cockpits by predicting the probability that pilots might miss relevant information due to routine learning effects combined with non-adequate placement of display instruments. The approach is based on an executable cognitive pilot model. We focus on the cognitive interaction between (1) rule-based processing of flight procedures, (2) the pilot's mental model of the current situation and (3) pilot's attention. The cognitive model is coupled with a formal cockpit design to simulate human-machine interaction during flight procedures. As an example we analyze the perception of automatic flight mode changes.

Keywords. Human modeling for design, human (selective) attention, mental models, human behavior simulation

1 Introduction

Today, human factor analysis of aircraft cockpit systems like autopilot or flight management systems is based on expert judgment and simulator-based tests with human subjects (e.g. test pilots) when first prototypes exist. This is in general a very expensive and time-consuming approach, because a number of subjects have to be hired for the simulation and necessary changes can only be realized with huge effort in the usually late stage of system development. In preceding papers an approach relying on cognitive models as partial substitutes of human subjects has been suggested (e.g. [5], [6]). In this paper latest extensions of the cognitive model are described.

Cognitive models were established in the early eighties as research tools to unify psychological models of particular cognitive processes. These early models neglected mental processes such as multitasking, perception and motor control that are essential for analysis of complex dynamic systems like aircraft cockpits. Models such as ACT-R [1] and SOAR [2] have been extended in this direction, but still have their main focus on processes suitable for static, non-interruptive environments. Other cognitive models like MIDAS [3] and APEX [4] were explicitly motivated by the needs of human-machine interaction. At OFFIS we developed a cognitive model with focus on how pilots adapt their mental knowledge about flight procedures while they gain experience of a particular system [5]. The phenomenon is called Learned Carelessness (LC).

In the EU funded project ISAAC the cognitive model was coupled with STATEMATE system design models to analyze pilot-autopilot interaction by human simulation to identify design structures where LC might have an impact on flight safety [6]. Case studies conducted in ISAAC already demonstrated plausible predictions of a number of erroneous pilot actions and provided insight in potential improvements of the analyzed autopilots. Based on these promising results it was suggested to extend the scope of covered psychological phenomena.

In this paper we present extensions towards simulation of human behavior on the level of perception and how these processes interact with rule-based processing of flight procedures and with the pilot's mental model of the current situation. This extended model shall be used to evaluate the ergonomics of cockpit layouts with regard to characteristics of human attention including a phenomenon called Selective Attention (SA) which may undermine the effect of graphical means used by designers to shift attention to critical events (like flashing annunciations). Both, LC and SA, may induce errors of omission. LC may lead to omitting displayed information, because routinely no relevant information is expected. SA may also lead to omitting displayed information because it is absorbed by other attention capturing displays in the neighborhood, e.g. a flashing warning signal may go undetected if it appears in the context of other dynamic elements.

In the following we first discuss LC and SA in the context of autopilot mode changes and mode annunciations during takeoff maneuvers. Afterwards, we present our model of these phenomena in the context of the OFFIS cognitive architecture. Then the model's operation including the interaction of LC and SA is illustrated based on three simulation run examples. The paper finishes with a brief overview of related work.

2 Monitoring Autopilot Modes

Aircraft cockpits are becoming increasingly information rich. Pilots have to take a lot of information into account in order to monitor the current aircraft status and to plan next actions. In this paper we focus on the autopilot (AP) of the Piper Cheyenne III PA42. The interaction with the AP is highly dependent on the actual AP mode. The modes indicate the current state of the system, what it is doing and what it is going to do. Because modes may change automatically during the flight, pilots are required to monitor the Mode Annunciation (MA). The MA is located in an instrument called the Primary Flight Display (PFD) which primarily indicates pitch and roll of the aircraft in a graphical way. Mode are indicated in form of letter codes, e.g. 'ALT' flashing in Green is an abbreviation for the Altitude Capture mode, in which the AP automatically decreases vertical speed in order to smoothly level off to an altitude previously dialed into the Alerter. 'ALT' flashes for approx. 6 seconds. Afterwards the annunciation is steady. On more modern flight decks a flashing box appears around the letter code and flashes for 10 seconds to attract pilots' attention.

Fig. 1 shows an excerpt of activities pilots have to perform during a takeoff maneuver with a Piper Cheyenne. The horizontal bars indicate what instruments have to be monitored at which time by the pilots according to the normative flight procedure. Parallel bars indicate that several instruments have to be monitored simultaneously.

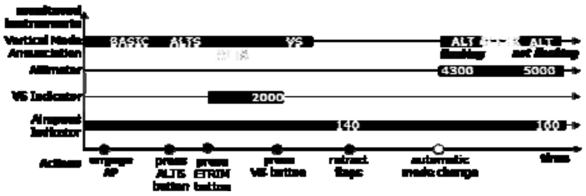


Fig. 1: Pilot activities after liftoff

The diagram refers to three subtasks pilots' have to perform after liftoff. *(1) Automatic climb:* the pilot has to engage and configure the AP in order to initiate an automatic climb to the initial altitude. The ALTS button has to be pressed to activate the target altitude dialed into the Alerter already as part of the preflight takeoff items. The ETRIM button has to be operated to adjust the vertical speed. At a vertical speed of 2000 ft/min the VS button has to be pressed to stabilize the climb. As soon as the aircraft is near to the target altitude the AP mode changes automatically to Altitude Capture mode ('ALT' flashes). *(2) Flaps retraction:* the pilot has to monitor the airspeed in order to retract the flaps at a speed of 140 knots. *(3) Leveloff monitoring:* as soon as the AP mode has changed to Altitude Capture mode the pilot has to monitor that the aircraft actually flies to the initial altitude and no error (for some reason) occurs.

One crucial point in this takeoff phase is that the pilots recognize the automatic mode change in order to begin monitoring the leveloff and because during automatic capture pressing the VS button (for stabilizing the automatic climb) is no longer allowed. Stabilizing the climb makes only sense before the mode change has occurred. If done afterwards it causes an altitude overshoot. The automatic mode change appears between 300 – 900 feet before reaching the target altitude. This paper considers two cognitive processes that may cause pilots to miss mode annunciations: Human (Selective) Attention and Learned Carelessness

3 Human Attention and Learned Carelessness

Human attention can be separated in top-down and bottom-up attention. Top-down attention is a deliberate process that shifts the gaze to current Areas of Interests, e.g. if an action depends on the actual airspeed, the eyes (and head) are moved to the airspeed indicator and attention is shifted accordingly.

Bottom-up attention, often referred to as *Selective Attention*, refers to the possibility that eye (and head) movements and a shift of attention are triggered by the onset of a salient stimulus [7]. Although eye movements and attention shifts can also be caused by acoustic and haptic stimuli, in this paper we investigate only visual stimuli. Thus, a salient stimulus means a discontinuity in space or time in the visual field. A discontinuity in space represents a difference in a static property, like color, brightness, form or orientation, e.g. a green dot in a set of red dots, or a circle in a set of quadrates. In contrast to this, a discontinuity in time (or dynamic discontinuity) denotes a dynamic change, like abrupt onset, flashing or moving of an object. In this paper we focus on dynamic discontinuities (like the flashing letter codes for modes), as this is commonly used by cockpit designers to capture the pilots' attention.

The flashing letter code 'ALT' is intended to take advantage of SA. But recent studies showed that the flashing annunciations have not always the intended effect of shifting the pilots' attention to changing modes. Mumaw, Sarter and Wickens [8] investigated mode changes in Boeing aircraft and found that 30 to 60 percent of pilots did not look at the Flight Mode Annunciation in an appropriate time after the mode had changed. On Boeing and other modern flight decks a flashing box (instead of flashing letter codes) is used to highlight mode changes. A successor study by Nikolic, Orr and Sarter [9] provides a hypothetical explanation: The context of the display, like color or dynamic elements can undermine the effect of attention capturing. The intended effect might be undermined by the data-rich and multiple-dynamic display context. As mentioned above, the PFD not only contains the MA, but, for example, also the artificial horizon which is moving according to the pitch and roll of the aircraft. Due to the colorful and dynamic neighborhood, the probability that the flashing 'ALT' captures the pilots' attention significantly decreases. Nikolic, Orr and Sarter [9] performed further more basic experiments which showed that the detection rate of the flashing box is about 0.649 (standard deviation 0.282), when the stimulus occurs in a colorful and dynamic context; in the control condition (solid black background) the detection rate is significantly higher (0.969, standard deviation 0.057). The results provide evidence that the salience of a stimulus depends on how much it differs from its surroundings.

Top-down and bottom-up attention compete against each other [7], e.g. a salient stimulus might detract the pilot from the task on which (s)he is currently concentrated, which is often intended, e.g. in case of warnings. The other way round, a salient stimulus might go undetected, because top-down attention causes the eyes to move to an Area of Interest where the stimulus is either out of the visual field or still in the visual field but absorbed by a dynamic neighborhood. Top-down processes drive attention while the pilot performs flight procedures. We assume that pilots have mental models of how to interact with the cockpit systems. While human pilots can partially substitute visual perception with other information from other resources, e.g. sense of balance, we assume for our modeling activities that pilots always move their visual attention to an instrument if one of the sub tasks that are currently performed requires the displayed information as an input according to the mental model of the flight procedure. Thus, the mental model drives the top-down attention, and is the main parameter that determines if a certain stimulus (like the flashing 'ALT') is in the visual field or not. Being in the visual field is certainly a precondition for being detected by bottom-up visual attention processes. The missing effect of resource substitution has to be considered in the model validation, e.g. one can conclude from the actions of the pilot that he perceived certain data, and this can be counted additional evidence for perception.

Obviously, considering the mental flight procedure model as the only driver of visual top-down attention is a simplification because it neglects scanning patterns pilots additionally use to get constant updates on the aircraft state. But, studies by Sarter [10] and others have shown that especially the autopilot modes are often not part of these scanning patterns. There is evidence that observing modes is more driven by mode change expectations.

While SA refers to the bottom-up aspect of human attention LC refers to the top-down aspect. The mental model of flight procedures is initially formed based on normative flight procedures acquired through handbooks and in simulator sessions. Then during line operation the mental model is modified by cognitive learning processes. LC describes the learning process on which we focused our investigations. The psychological theory "Learned Carelessness" [16] states that humans have a tendency to neglect safety precautions if this has immediate advantages, e.g. it saves time, and allegedly allows keeping the same safety level. In the context of avionics systems safety precautions may be understood as checking the current state or mode of the systems before performing critical actions. LC is characteristic for human nature because we have to implicitly simplify in order to be capable to perform efficiently in a complex environment. Resulting behavior is highly adapted to routine scenarios but, unfortunately, may lead to errors and hazards in non-routine situations. Visual events like flashing indications on cockpit displays might be a countermeasure against the effects of LC but as described above this might be undermined by characteristics of SA. By modeling these two phenomena in the same executable model it is possible to simulate and analyze the interaction of LC and SA in specific scenarios with the goal to optimize normative flight procedures (with their resulting scanning paths) and the ergonomics of display designs.

4 The OFFIS Simulation Platform

Lüdtke and Möbus [5] developed a generic rule based cognitive architecture, which can be used to simulate pilot behavior (including pilot errors). The architecture is based on a flight procedure formalisation in the form of "if-then" rules. The rules formally describe a mental representation of flight procedures. Currently the procedure must contain the tasks of the pilot flying and non-flying, because the cognitive architecture in its present development state does not support task sharing between the two.

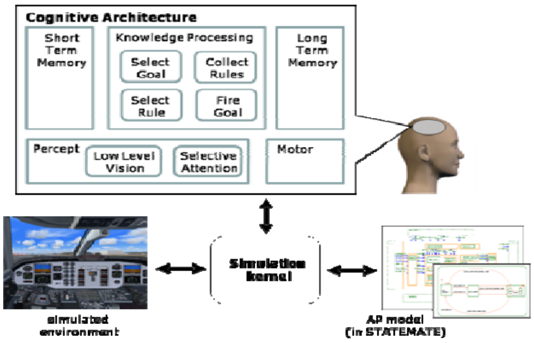


Fig. 2: Simulation platform with cognitive architecture

In order to perform the simulation the flight procedure rules are uploaded to the cognitive architecture (cf Fig. 2). A cognitive architecture with uploaded procedure rules is what we call a pilot model. The cognitive architecture can be understood as an interpreter or executor of formal flight procedure rules.

Within a simulation platform (Fig. 2) the pilot model interacts with a system under investigation (modeled in STATEMATE) and a simulated environment (including the aircraft). A simulation kernel synchronizes the different models and organizes the dataflow.

4.1 Modeling Flight Procedures with Rules

The format of our procedure rules is a Goal-State-Means (GSM) format (Fig. 3). All rules consist of a left-hand side (IF) and a right-hand side (THEN). The left-hand side consists of a goal in the Goal-Part and a State-part specifying Boolean conditions on the current state of the environment. Apart from the condition the State-part contains memory-read items to specify that in order to evaluate the condition the associated variables have to be retrieved from memory. The right-hand side consists of a Means-Part containing motor and percept actions (e.g. hand movements or attention shifts), memory-store items as well as a set of partial ordered sub-goals. In the GSM rule syntax variables are underlined.

The rule in Fig. 3 can be informally be read as “IF the actual goal is to retract the gear and the aircraft has lifted off, THEN pull the gear lever, shift attention to gear annunciation, pursue the goal to check if the gear actually retracts and afterwards pursue the goal to call out the gear state. This rule defines a goal-subgoal relation between GEAR_UP and subgoals CHECK_GEAR_UP, CALLOUT_GEAR_UP. Between the subgoals a temporal order is imposed (by "After").

During simulation the cognitive architecture selects rules based on their left-hand sides and executes the right-hand sides.

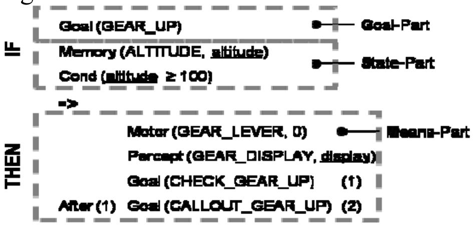
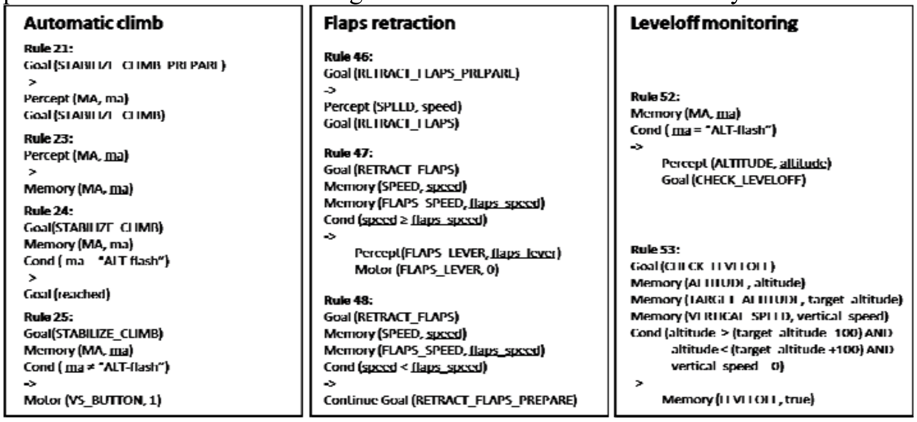


Fig. 3: Example GSM rule

Fig. 4 shows a subset of rules for the takeoff maneuver described above (Fig. 1) with its subtasks automatic climb, flaps retraction and leveloff monitoring. *(1) Automatic climb:* With rule 21 the pilot model prepares climb stabilization by looking at the flight mode annunciation and perceiving the displayed value. Rule 23 is a percept rule used directly after the percept action to store the perceived value into memory. Rule 24 and 25 retrieve the current mode from memory, in case it is Altitude Capture mode rule 24 prescribes to perform no action, in case it is not Altitude Capture mode rules 25 prescribes to press the VS button. "ALT-flash" is the symbol for the flashing letter code. *(2) Flaps retraction:* Rule 46 serves to prepare flaps retraction. With rule 47 and 48 the pilot model decides either to operate the flaps lever (rule 47) or to continue to monitor the speed annunciation (rule 48) by using rule 46 again. *(3) Leveloff monitoring:* Rule 52 serves to prepare monitoring the leveloff. It is fired as soon as MA annunciates the transition to Capture mode. With rule 53 the pilot model decides if the aircraft has leveled off correctly. A further rule not shown here prescribes to continue monitoring the leveloff if the altitude has not yet been reached.



Automatic climb

```
Rule 21:
Goal (STABILIZE_CLIMB_PREPARE)
->
Percept (MA, ma)
Goal (STABILIZE_CLIMB)

Rule 23:
Percept (MA, ma)
->
Memory (MA, ma)

Rule 24:
Goal(STABILIZE_CLIMB)
Memory (MA, ma)
Cond ( ma = "ALT flash")
->
Goal (reached)

Rule 25:
Goal(STABILIZE_CLIMB)
Memory (MA, ma)
Cond ( ma ≠ "ALT-flash")
->
Motor (VS_BUTTON, 1)
```

Flaps retraction

```
Rule 46:
Goal (RETRACT_FLAPS_PREPARE)
->
Percept (SPEED, speed)
Goal (RETRACT_FLAPS)

Rule 47:
Goal (RETRACT_FLAPS)
Memory (SPEED, speed)
Memory (FLAPS_SPEED, flaps_speed)
Cond (speed ≥ flaps_speed)
->
      Percept(FLAPS_LEVER, flaps_lever)
      Motor (FLAPS_LEVER, 0)

Rule 48:
Goal (RETRACT_FLAPS)
Memory (SPEED, speed)
Memory (FLAPS_SPEED, flaps_speed)
Cond (speed < flaps_speed)
->
Continue Goal (RETRACT_FLAPS_PREPARE)
```

Leveloff monitoring

```
Rule 52:
Memory (MA, ma)
Cond ( ma = "ALT-flash")
->
      Percept (ALTITUDE, altitude)
      Goal (CHECK_LEVELOFF)

Rule 53:
Goal (CHECK_LEVELOFF)
Memory (ALTITUDE, altitude)
Memory (TARGET_ALTITUDE, target_altitude)
Memory (VERTICAL_SPEED, vertical_speed)
Cond (altitude > (target_altitude - 100) AND
        altitude < (target_altitude +100) AND
        vertical_speed = 0)
->
      Memory (LEVELOFF, true)
```

Fig. 4: Subset of rules for takeoff procedure

4.2 Processing Rules inside a Cognitive Architecture

The human model has been developed in a modular way. It consists of a short-term and a long-term memory, knowledge processing, a learning component and components for perception and motor (Fig. 2). The percept and motor components serve to communicate with the simulated environment and the system model. All data entering the model via perception is stored in the short-term memory. Additionally the short-term memory stores a set of goals which the model has to process (goal agenda). The long-term memory stores procedural knowledge in form of GSM-rules. The following presentation of the cognitive architecture focuses on the data structures and processes that have been added in order to model human perception including Selective Attention (SA).

Simulated environment: The cognitive model relies on a symbolic representation of the simulated world with which it is intended to interact. Most relevant are the topology of the cockpit as well as the ergonomic features and current values of the instruments. Every instrument is represented as an Area of Interest object (*AOI Object*) with the following structure: a name, e.g. "MA", the current value, the 3-dimensional position coordinates, the physical dimensions height, width and depth, the primary colors and a Boolean variable indicating, if the AOI is dynamic in the sense of dynamic discontinuity (like the MA with its flashing letter codes). As described above, dynamic AOIs might cause a shift of attention if their effect is not absorbed by other dynamic AOIs in the neighborhood. Changes of dynamic AOIs (e.g. flashing) are modeled as *Visual Events* that are sent to the cognitive model by the simulation kernel and are processed by the SA mechanism (see Perception below). A Visual Event is always associated to exactly one AOI Object and is furthermore specified by attributes that characterize the associated stimulus, e.g. type of event (e.g.

flashing, moving), frequency of flashing, and duration of event. Visual Events are sent for example by the MA when the mode changes automatically to Altitude Capture mode and 'ALT' flashes and by the PFD when the artificial horizon moves while the aircraft is changing its attitude.

Memory: The memory component of the cognitive model stores the mental image of the current environmental state. Consequently, there is a corresponding Memory Object for every AOI Object. Memory Objects store a subset of the AOI Object attributes including name and the current value.

Rules: Additionally to the GSM-rules described above we added a second rule type, called reactive rules. Rule 52 in Fig. 4 is an example of this rule type. The only difference is that reactive rules have no Goal-Part. While GSM-rules represent deliberate behavior and are selected by the knowledge processing component during the execution of a flight procedure, reactive rules (State-Means (SM) rules) represent immediate or reactive behavior which is triggered by visual events in the environment (bottom-up perception).

Knowledge Processing: The knowledge processing component executes a four step cognitive cycle typical for production systems: (KP1) a goal is selected from the goal agenda in short-term memory, (KP2) all rules containing the selected goal in their goal-part are collected and a request for retrieving the current state of the variables contained in the Boolean conditions in the state-parts of the rules is sent to the memory component, (KP3) after the request has been answered one of the collected rules is selected by evaluating the condition-part, finally (KP4) the selected rule is fired, which means that the motor and percept actions are sent to the motor and percept component respectively, the subgoals are added to the goal agenda (together with the partial temporal order) and the values contained in memory items are sent to the memory component.

The cycle time is 50 ms like in ACT-R. This time may be prolonged depending on the memory retrieval in KP2. In KP2 all variables contained on the left-hand sides of the collected rules have to be retrieved from memory. The retrieval time is influenced by the number of variables.

In KP1 the goal agenda may contain several goals that are currently applicable according to the temporal order. We modeled a simple task switching mechanism in order to achieve an alternating between tasks. Our mechanism is similar to the multitasking general executive of Salvucci [**Erreur ! Source du renvoi introuvable.**] and selects goals, based on a most-recently-used criterion. In KP2 reactive rules may be added to the set of collected rules if new values for the variables contained in the State-Part have been added to the memory component (by the percept component). In KP3, reactive rules are always preferred to non-reactive rules.

Vision: When modeling visual perception, the main focus is on *what can be perceived based on the visual constraints*, and *how much time is needed* to perceive something. In order to answer this, one needs to model visual focus, visual field, human attention, as well as head- and eye-movements. In our model we assume a visual field of 170 degree horizontal and 110 degree vertical around an optical axis (defined by the gaze direction of the eye). The focus is modeled with an expansion of seven degree around the optical axis. Eye-movements can be initiated either by the knowledge processing component or by Visual Events in the environment. Currently we do not distinguish between moving eyes and moving attention. We assume that if eyes are moved also the attention is moved. This is of course a simplification which does for example not allow simulating the phenomenon "seeing without noticing". For a validation of the model with human pilots, an extensive debriefing of the human pilots is necessary in order to identify seeing without noticing, and to enhance gaze data.

The visual component is split into *Low Level Vision* (LLV) and S*A*. LLV implements all basic functions of human vision: eye- and also head-movements (including focus and visual field). SA is modeled as a mechanism that computes for each Visual Event if it is detected or not. The two processes are performed in parallel. LLV is triggered either by the knowledge processing component when percept actions are sent to the percept component during rule firing in KP4 (top-down perception as explained above) or by the percept component itself if SA detects a salient stimulus (bottom-up perception). In both cases LLV performs the necessary steps to move the eyes and head to an Area of Interest (AOI). For top-down perception this is done in three (LLV1-3) and for bottom-up perception in two (LLV2, 3) steps:

- LLV1: For top-down perception the variable (e.g. ALTITUDE) contained in the percept action has to be transformed into an AOI by retrieving the position of the corresponding instrument from memory. Then the position of the instrument is transformed into the coordinate system of the eyes, so that the angle α between the actual optical axis of the eye V1 (e.g. pointing towards the MA) and the desired optical axis V2 (e.g. pointing toward the Airspeed Indicator, ASI) can be computed:

$$\alpha(V_1, V_2) = arcos\left(\frac{x_{MA} \cdot x_{ASI} + y_{MA} \cdot y_{ASI} + z_{MA} \cdot z_{ASI}}{\sqrt{x_{MA}^2 + y_{MA}^2 + z_{MA}^2} \cdot \sqrt{x_{ASI}^2 + y_{ASI}^2 + z_{ASI}^2}}\right) \quad \textbf{(1)}$$

  This preparatory step takes about 140 to 200 ms normally distributed around 170 ms. For bottom-up perception this step is skipped because the AOI is defined by a visual stimulus directly in the cockpit at a certain instrument position. Thus, no position has to be retrieved from memory.

- LLV2: Eyes and head are moved in order to focus the new AOI. Based on concepts of Freedman [11] we developed a model for a combined movement of head and eyes. The contribution of head and eye, respectively, to the total change α is calculated as follows:

$$EYE_{contrib}(\alpha) = \begin{cases} \alpha, & \alpha \leq 20 \\ \min(0.31 \cdot \alpha + 13.6,\ 40), & \alpha > 20 \end{cases} \quad \textbf{(2)}$$

$$HEAD_{contrib}(\alpha) = \alpha - EYE_{contrib}(\alpha) \quad \textbf{(3)}$$

  The eye contribution is limited to 40 degree, due to the physical design of the human eyes. The modeled speed of the eye and head are approximated functions from Freedman as well.

- LLV3: The AOI is fixated until LLV is triggered again. If fixation can be kept for at least 200 ms the actual value of the AOI Object is sent to the memory component and stored in the corresponding Memory Object. A shift of attention might be interrupted in LLV3 by a new attention shift request as soon as the centration of the eyes in the head is finished. Thus, it may happen that LLV3 is interrupted before the needed fixation time of 200 ms has passed. In such a case the AOI value is not written into memory.

Each Visual Event sent by the simulated environment is processed by the SA mechanism which is divided in three steps:

- SA1: Based on the actual eye and AOI position it is determined if the AOI to which the currently processed event belongs lies within the current focus or at least in current visual field. To be in the visual field the AOI must be within 85 degrees of the eye position. If the AOI is in the visual focus, then the associated event is marked as recognized and SA3 is started, skipping SA2. If it is outside the visual field the associated event is marked as unrecognized, the next event is retrieved and SA1 is restarted. If it is within the visual field but not in focus SA2 is initiated in order to determine recognition.
- SA2: It is determined if in a neighborhood of 15 degree around the AOI other Visual Events have occurred. The 15 degree neighborhood has been derived from the experimental setup of Nikolic, Orr and Sarter [9], so that the probabilities of their study can be used in our model. If the event indeed has a dynamic neighborhood, a probabilistic choice is computed to determine if the event is recognized or not: the probability to detect an event within a dynamic neighborhood is 0.649 with a standard deviation of 0.282. If the event is recognized SA3 is started else the next event is retrieved and SA1 is restarted.
- SA3: The Visual Event is sent to the memory component to be stored. LLV reacts to this event as soon as step LLV3 is entered. The other LLV steps cannot be interrupted.

Learning Component: The learning component implements our LC mechanism. Rule 25 in Fig. 4 specifies that the vertical speed must be stabilized as long as the target altitude has not been captured (MA ≠ 'ALT-flash'). Using rule 21 the current value of MA is perceived from the associated cockpit instrument. Rule 23 stores the perceived value into the memory. Most of the times when the pilot tries to press the VS button the mode change has not occurred, because the distance to the target altitude is long enough. Thus most of the time the percept action delivers 'ALTS' which indicates that the current mode is Altitude Select and not Altitude Capture. We hold the hypotheses that due to this regularity a pilot would simplify his mental model of the procedure into a version, where the MA value is no longer perceived from the cockpit instrument but is just retrieved from memory. This is modelled by melting two rules into one rule by means of *rule composition* [14]. A precondition for composing rules is that firing of the first rule has evoked the second rule, or more exact, the first rule derives a subgoal that is contained in the Goal-Part of the second rule. Melting the rules means building a *composite* rule by combing the left-hand sides of both rules and also combing both right-hand sides.

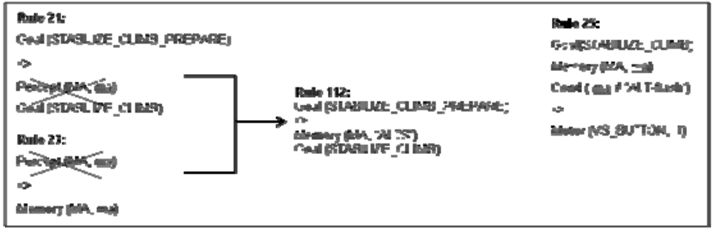


Fig. 5: Simplified rule

The crucial point is that in this process elements that are contained on the right-hand side of the first and also on the left hand side of the second rule are eliminated.

This process cuts off intermediate knowledge processing steps. Fig. 5**Erreur ! Source du renvoi introuvable.** shows the composite rule 112 that was formed by composition of rule 21 and 23. The percept action has been eliminated and the composite always stores the value 'ALTS' in memory. Rule 112 is appropriate in scenarios that are similar to those in which the rule has been learned (MA does not indicate Altitude Capture mode). In deviating scenarios (MA does indicates Altitude Capture mode) applying rule 112 results in careless behaviour: pressing the VS button independent from the current mode annunciation. At the beginning of the simulation all procedure rules in the long-term memory component are normative, meaning that the application of these rules does not lead to an error.

5 Simulation Run Examples

In this section we present three simulation runs which we observed on the simulation platform. These runs serve to illustrate the interaction between the cognitive model and the Piper Cheyenne cockpit systems with a focus on the autopilot. All scenarios refer to the rules presented in Fig. 4 and thus to the takeoff phase after liftoff. In all scenarios the simulated pilot is careless with regard to the Altitude Capture mode annunciation because he has learned rule 112 (Fig. 5) during the preceding simulation runs. Simulation shall show if the annunciation of the automatic mode change is effective in capturing the pilot's attention and to alleviate the effect of LC.

Scenario 1: LC not alleviated due to other dynamic displays

The pilot model engages the AP and presses the ALTS button (Fig. 6). It looks at the VS Indicator in order to stabilize vertical speed as soon as 2000 feet are reached. In parallel it monitors the Airspeed Indicator to prepare flaps retraction. The automatic mode change occurs while the aircraft starts to fly a turn towards a heading of 50 degrees. The pilot model does not recognize the mode change annunciation. The PFD at this moment is highly dynamic because of the turn. Thus the mode change annunciation appears in a dynamic neighborhood and the probability that the visual event is not recognized is high. The VS button is pressed during Altitude Capture mode and finally the flaps are retracted when the flaps retraction speed is reached. The pilot model does not monitor the leveloff and thus does not recognize that the initial altitude is overshot. In the simulation the scenario ends with a failure as soon as the altitude is 300 feet above the prescribed altitude.

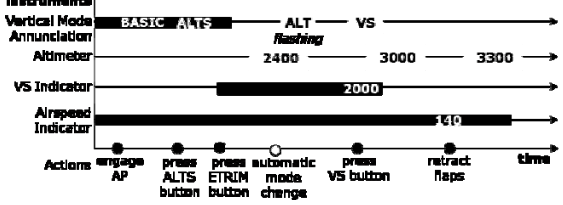


Fig. 6: Mode change not recognized due to dynamic PFD

Scenario 2: LC alleviated by successful SA

Scenario 2 (Fig. 7) is similar to scenario 1 but shortly before the change to Altitude Capture mode the airspeed is already close to flaps speed. When the mode change occurs the pilot model actually monitors the speed annunciation. The other annunciations on the PFD at that time are static thus the probability that the visual mode change event is detected is much higher than in scenario 1. The mode change annunciation is recognized in the visual field and enters the model's memory. The pilot model correctly initiates monitoring the leveloff. Furthermore the mode change information prevents the model from pressing the VS button. The scenario ends with a success because the altitude is reached and maintained.

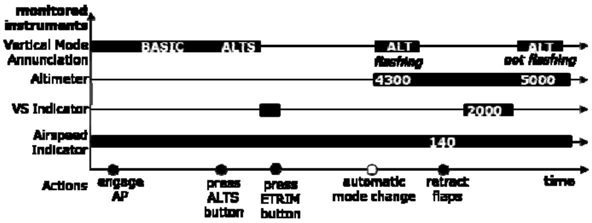


Fig. 7: Mode change recognized due to successful SA

Scenario 3: LC not alleviated due to limited visual field

In this scenario the airspeed of 140 knots is reached when the mode change occurs. The pilot model retracts the flaps and fixates the flaps lever at that time. Thus the mode change Visual Event cannot be recognized because it is out of the visual field. The result is similar to scenario 1: the VS button is pressed during Capture mode and because the leveloff is not monitored the altitude overshot is not recognized. The scenario ends with a failure.

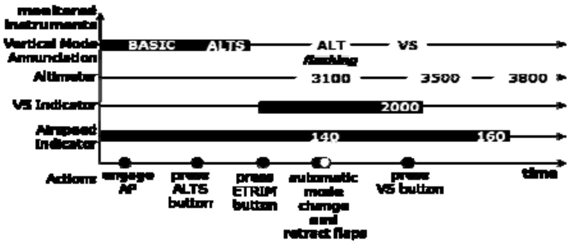


Fig. 8: Mode change annunciation outside visual field

6 Related Work

Processes of visual perception and attention have been modeled in other cognitive models like ACT-R, APEX and MIDAS as well. ACT-R has been extended over the last years to incorporate perception in form of visual focus that can be shifted by production rules (top-down attention [12]). EMMA [15], an ACT-R extension, simulates eye-movements taking into account the distance to the target for calculating eye movement time. MIDAS [13] is equipped with a complete anthropological model.

APEX [4] was designed to model air traffic controllers, and has a detailed temporal model of the eyes. Unique for the OFFIS cognitive model is the consideration of Selective Attention especially in combination with Learned Carelessness. While APEX and MIDAS allow building dedicated rules for SA which reacts to dynamic stimuli, the OFFIS model has an architectural build-in SA model that takes the neighborhood of instruments into account to determine detection rates.

The OFFIS model does not consider visual search mechanisms like scanning cockpit instruments based on their contextual importance. General mechanisms of visual search are a major topic in the ACT-R field (e.g. [17]). Furthermore the OFFIS model does not distinguish between moving the eyes and moving attention. Within human cognition these are separate processes which have not always the same target, e.g. when the eyes of the pilot are fixed on the instrument, but (s)he actually thinks about a communication with the air traffic controller.

7 Summary and Future Work

We extended the OFFIS cognitive model with a sophisticated perception component. The new visual component models the basic concepts of human perception like visual field and focus, eye- and head movements as well as Selective Attention. This enables the prediction of eye movements and the analysis of the effectiveness of graphical stimuli used to trigger attention shifts in aircraft cockpits. The OFFIS model additionally allows simulating a routine learning process called Learned Carelessness causing pilots to neglect checking the current flight modes before performing actions. The integration of this learning process with the perceptive processes allows analyzing if the graphical attention capturing stimuli are sufficient to alleviate the effects of the routine learning process. Our intention is to use this model to develop a methodology to validate the ergonomics of cockpit layouts. Our future work will concentrate on a detailed validation and improvement of the perceptive processes as well as the model as a whole. The validation requires a complex design of experiments with real pilots. A series of scenarios is needed to induce routine learning effects. Afterwards dedicated scenarios to test the effect of graphical stimuli on attention shifts have to be investigated. This would allow comparing the detection rate of mode annunciations as well as eye-movements of real pilots with the corresponding data of the cognitive model. Such a complex experimental design is foreseen in the European project HUMAN which started in March 2008 in the 7th Framework Program of the EU (cf. www.human.aero).

References

1. Anderson, J.R., Bothell, D., Byrne, M.D., Douglass, S., Lebiere, C., Qin, Y.: An integrated theory of the mind. Psychological Review 111, 4, pp. 1036-1060 (2004)
2. Wray, R., Jones, R.: An introduction to Soar as an agent architecture. In: Sun, R. (ed.), Cognition and Multi-agent Interaction. Cambridge University Press, pp 53-78 (2005)

3. Corker K. M.: Cognitive models and control: Human and system dynamics in advanced airspace operations. In: Sarter, N., Amalberti, R. (eds) Cognitive Engineering in the Aviation Domain. Mahwah, pp. 13–42, Lawrence Erlbaum Associates, NJ (2000)

4. Freed M.: Simulating Human Performance in Complex, Dynamic Environments. PhD thesis, Northwestern University (1998)

5. Lüdtke, A., Möbus, C.: A Cognitive Pilot Model to Predict Learned Carelessness for System Design. In: Pritchett, A., Jackson, A. (eds) Proceedings of the International Conference on Human-Computer Interaction in Aeronautics (HCI-Aero), 29.09.-01.10.2004, Toulouse, France: Cépaduès-Editions. CD-ROM (2004)

6. Lüdtke, A., Cavallo, A., Christophe, L., Cifaldi, M., Fabbri, M., Javaux, D.: Human Error Analysis based on a Cognitive Architecture. In: Reuzeau, F., Corker, K., Boy, G. (eds) Proceedings of the International Conference on Human-Computer Interaction in Aeronautics (HCI-Aero), 20.-22.09.2006, Seattle, USA. Toulouse, pp. 40-47, Cépaduès-Editions, France (2006)

7. Knudsen, E. I.: Fundamental components of Attention. Journal of Annual Review of Neuroscience 30, pp. 57-78 (2007)

8. Mumaw, R. J., Sarter, N. B., Wickens, C. D.: Analysis Of Pilot's Monitoring And Performance On An Automated Flight Deck. In: 11th International Symposium on Aviation Psychology. The Ohio State University (2001)

9. Nikolic, M. I., Orr, J. M., Sarter, N. B.: Why Pilots Miss the Green Box: How Display Context Undermines Attention Capture. The International Journal of Aviation Psychology 14, pp. 39-52, Lawrence Erlbaum Associates Inc. (2004)

10. Sarter, N.B.: Knowing when to look where: Attention Allocation on advanced automated Flight Decks. In: Proceedings of the 5th International Symposium on Aviation Psychology, pp. 239 (1995)

11. Freedman, E. G.: Interactions between eye and head control signals can account for movement kinematics. Biological Cybernetics 84, pp. 453-62 (2001)

12. Byrne, M. D., Anderson, J. R.: Enhancing ACT-R's perceptual-motor abilites. In: Proceedings of the Nineteenth Annual Conference of the Cognitive Science Society , pp. 880 (1997)

13. Hart, S. G., Dahn, D., Atencio, A., Dalal, K. M.: Evaluation and application of MIDAS v2.0. In: Proceedings of the Society of Automotive Engineers (SAE) World Aviation Congress (2001)

14. Möbus, C., Schröder, O., Thole, H.J.: Online Modeling the Novice-Expert Sift in Programming Skills on a Rule-Schema-Case Partial Order. In: Wender, K.F., Schmalhofer, F., Böcker, H.-D. (eds) Cognition and Computer Programming. Norwood, N.J.: Ablex (1995)

15. Salvucci, D. D.: A Model of Eye Movements and Visual Attention. In: Proceedings of the Third International Conference on Cognitive Modeling, pp. 252-259 (2000)

16. Frey, D., Schulz-Hardt, S.: Eine Theorie der Gelernten Sorglosigkeit. In: Mandl, H. (Hrsg.) 40. Kongress der Deutschen Gesellschaft für Psychologie. Hogrefe Verlag für Psychologie Göttingen, pp. 604-611 (1997)

17. Salvucci, D. D., Taatgen, N. A.: Threaded cognition: An integrated theory of concurrent multitasking. Psychological Review 115, 101-130 (2008)

18. Everett, S. P., Byrne, M. D.: Unintended Effects: Varying Icon Spacing Changes Users' Visual Search Strategy. In: Proceedings of ACM CHI '04: Conference on Human Factors in Computing Systems, Vienna, Austria: ACM (2004)