In: B.B. Borys, G. Johannsen, C. Wittenberg & G. Stätz (eds.): Proceedings of the XVI. European Annual Conference on Human Decision Making and Manual Control, pp. 268–276. Dec. 9-11, 1997, University of Kassel, Germany. (http://www.safety-critical.de/doc/anumanu97.pdf)

Modelling Mental Processes of Experienced Operators during Control of a Dynamic Man Maschine System

Leuchter, S.; Niessen, C.; Eyferth, K and Bierwagen, T.[*]
Berlin University of Technology, Center for Man-Machine Systems Studies
TIB 4/5-3, Gustav-Meyer-Allee 25, D - 13355 Berlin
{leuchter|niessen|eyferth}@zmms.tu-berlin.de
[*] *Deutsche Flugsicherungs GmbH, Entwicklung und Erprobung*
Paul-Ehrlich-Str. 35-37, D - 63225 Langen
bierwagen@se.dfs.de

Abstract

A model of en-route air traffic controllers´ cognitive activities and its implementation in ACT-R, a psychological and implementation framework for cognitive modelling, is presented in this paper. The focus is on issues concerning problems inherent to dynamic situations: dynamic representation and control procedures.

Introduction

The research group ‘En-route Controller´s Representation’ has attempted to model the cognitive activities of experienced operators in a dynamic man-machine system. The group used the air traffic control domain as an example to analyze how operators develop various mechanisms for internally representing the continuously changing traffic situation as a precondition for diagnosis and intervention. A model for this analysis was developed and implemented on the basis of empirical work - interviews, simulation experiments, memory tests with experienced and less experienced en-route air traffic controllers, and theoretical considerations (Bierwagen et al., 1997).

For theoretical and applied reasons, it can be useful to have a computer model of the operator´s cognitive skills (see e.g., Opwis & Spada, 1994). In order to construct a complex psychological model, the implementation of such a model can provide a more detailled and explicit description of every cognitive process involved than a verbal description. Running a model can test a theory by showing if the anticipated effects can be reproduced. A detailled model also serves as a framework for generating hypotheses that support the empirical work. An applied computer model can also be used to analyze and predict the effects of future technological changes on the operator's cognitive activities. These insights into the consequences affecting cognitve performance can be helpful for future system design and training concepts.

The implementation of the model „MoFl“ (*Modell der Fluglotsenleistungen*) is based on a production system in the programming language ACT-R (Adaptive Control of Thought - Rational, Anderson, 1993). As programming environment, ACT-R includes a broad and detailled theoretical framework of human cognition. The basic assumption is that cognitive skills are composed of production rules. A production rule is a modular piece of knowledge. Combining these rules into a sequence represents complex cognitive processes. For the most part, ACT-R is suitable for modelling the cognitive performance of en-route air traffic controllers. But, modelling in ACT-R is limited for some relevant aspects of dynamic situations.

The purpose of this paper is to present the model as a psychological framework and a computer implementation. We will also discuss the two special problems „dynamic representation“ and „control procedures“. This paper is divided into three sections:

- the basic elements of the model
- framework for the implementation: the cognitive architecture ACT-R
- the implementation of the model

The basic elements of the model

In order to discuss the theoretical framework for modelling the cognitive activity of air traffic controllers and its implementation, we first need to outline the task and the basic elements of the cognitive model.

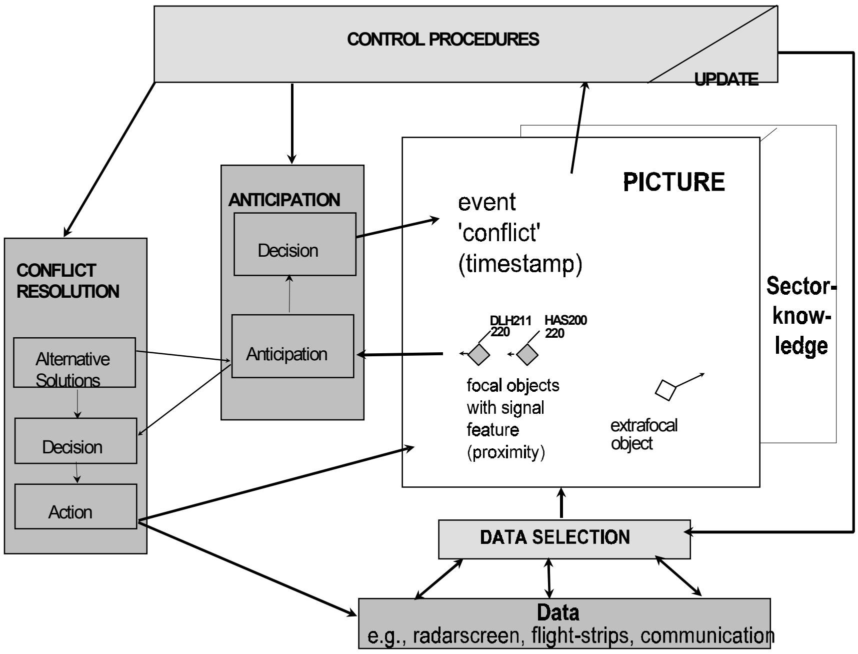
Air traffic controllers have to monitor and to control complex and dynamic traffic situations in order to detect and solve potential conflicts between aircraft. On the basis of different sources of information (e.g., radarscreen, flight strips, head-phone communication with pilots), the controllers have to perceive and comprehend multiple characteristics of many aircraft while new incoming aircraft create new traffic relationships for evaluation. Therefore, a precondition for the diagnosis of potential conflicts between aircraft, planning, and intervention is their development of various mechanisms to internally represent the changing situation. Without an appropriate mental representation, the operator would not be able to anticipate the future state of aircraft in order to detect conflicts, to make responsible decisions, and to organize the whole task. Air traffic controllers express with the term „picture“ (e.g., Whitfield & Jackson, 1982; Falzon, 1982) what is often described as „situation awareness“ (e.g., Endsley, 1995; Flach, 1995): an active mental representation of the current and future traffic situation.

Our purpose was to construct a model which describes this embedment of the controller´s cognitive activities in the dynamic traffic environment.

We developed a model (‘MoFl‘) on the basis of theoretical and experimental work with both experienced and inexperienced air traffic controllers (simulation experiments, interviews, memory tests, and a card sorting task). The experimental work with real time simulation was based on a realistic simulation system of the control task called ‘EnCoRe-PLuS’ (*En-route Controller's Representation - Programmable Airspace Simulation*).

The main components of MoFl are three cycles of information processing, (i.e., monitoring, anticipation, problem resolution) operating on different parts of the situation representation, called the *picture* (see Figure 1). The coordination of these processes is driven by *control procedures*. *Monitoring* and *anticipation* are diagnostic processes (conflict detection), *problem resolution* is the preparatory step for intervention by the controller.

The *monitoring cycle* includes data selection procedures and the regular update of aircraft features. According to our empirical data, the controller selects relevant features of aircraft, especially identification codes, the horizontal and vertical positions of objects, and flight directions. In addition, the controller searches for meaningful signals for the purpose of conflict detection during radar-screening. These are aircraft features like vertical movements, proximity to other aircraft or to points in airspace where conflicts frequently occur.



Because of these signal features, aircraft become 'focal', that means that they are attention demanding objects. Aircraft without such features are 'extrafocal'. In the dynamic environment of air traffic control, objects have to be updated continuously. There is a relationship between the semantics of objects and the frequency of updating: focal, attention-demanding objects demand a higher monitoring frequency than extrafocal objects.

The next step in diagnosis consists of an *anticipation cycle* which operates on the focal part of the picture. For all attention-demanding ('focal') aircraft or aircraft relationships, a future state is anticipated. The goal of the anticipation cycle is to create new cognitive processing information about aircraft. Depending on the results of anticipation, aircraft with signal features can then be represented as 'events'. An event reflects the type of relations between aircraft or relations between aircraft and airspace features in future time and space. The result of an anticipation allows to interpret (*decision*) if the future trajectories of aircraft result either in a conflict, in a safe separation, or the demand for more monitoring. If a conflict is detected, the event *'conflict'* includes an estimation of the time remaining for conflict resolution. Relations which have proved to be safe, are no longer in the focal part of the *picture* and become extrafocal at this time. This indicates that at this time there is almost no demand for cognitive processing, except for updating. If the operator is not sure about the potential conflict, the event '*monitoring*' will be added to the focal part of the representation, indicating both a higher frequency of monitoring and also a high demand for further anticipation.

If conflicts are detected, the *problem resolution cycle* initiates several steps to prevent an impending conflict. The controller has to select the most urgent conflict in order to generate or recall solutions (*alternative solutions*). Next, the operator has to check that the solution does not generate new follow up conflicts (*decision*). We assume that the controller checks by running a mental simulation of the solution (as in the *anticipation cycle*). The results of this model are executed (*action*).

The resulting *picture* is characterized by an analogue representation of objects, events, and objects with reference to other objects, and / or airspace structure. Objects with signal features are represented focally, objects without these features extrafocally. In addition, events which indicate the meaning of aircraft relations in future time and space are represented focally.

The coordination of these processes is driven by *control procedures*. We assume that the different processing components cannot be interrupted. The contoller has to switch between them: for example, between the solution of a conflict and further monitoring. On the basis of the state of the ´picture´, control procedures select the most important and urgent processing demand.

Framework for the implementation: the cognitive architecture ACT-R

A theoretical concept that could frame a model of air traffic controllers described above should address three aspects of the dynamic task environment. 1. The continous changes of the situation. These changes do not allow fixed sequences of cognitive processing, they rather call in a cyclic update of varying relations as a basis of situational awareness. 2. The necessity to predict future states of the situation in order to predict potential conflicts. Such predictions alter the goals of ongoing control activities. 3. The demand to coordinate and to sequence simultanious requirements of the control task.

We could not find any approach that includes all these aspects in a detailled manner. The hitherto proposed concepts for adaptive control of complex task enviroments (e.g., Anderson, 1993; Rasmussen, 1986; Hacker, 1978) concentrate on rather static tasks and on invariant goal structures. For example the cognitive architecture of Anderson's ACT-R does not take into account that in dynamic situations the operator has to continuously update her or his mental representation, even if it is not directly part of the current problem solving process. In addition, such production systems are directed by a fixed goal hierarchy. Because of the changing and complex situation requirements, the controller has to flexibly coordinate cognitive activities. This coordination is context-dependent: it does not follow a pre-defined goal hierarchy.

Despite of these points, ACT- R provides a suitable framework for modelling the task of air traffic controllers: 1. as a psychological framework of human cognition, it also describes an environment for implementation, and 2. ACT-R is based on explicit and very detailled assumptions about cognitive architecture.

ACT-R includes two kinds of knowledge representation: declarative and procedural knowledge. The basic units in declarative memory are so-called working memory elements (WMEs). A WME is an object with identity. It has named slots that can be filled with Lisp objects or references to other WMEs. References to other WMEs can be interpreted as relations, so that a semantic net with WMEs as nodes and references for relations is spread out. ACT-R defines an object-oriented structure for declarative memory. Every node in the net is an object of a certain class. A class is declared by naming all slots an object of this class will have. Subclassing is possible.

The implementation of the controller´s „picture“ occurs within this framework. Aircraft and events are represented as WMEs in working memory. However, from the psychological perspective we assume an analogous non-symbolic mental representation of the situation. There is some empirical evidence that experienced controllers anticipate future states of aircraft without calculating the trajectories. This can indicate that they build up a non-metric, analogous representation of the situation. When we think about this analogous representation, we follow Craik´s (1943) and Johnson-Laird´s (1983) basic ideas of a functional internal model that parallels processes of the external world. The „picture“

- is understood as an active knowledge-based construction of meaningful relations between elements of a situation, and not as an addition of perceptions.
- is incomplete with regard to the content of information and is temporary. The representation is build up by schemata in order to serve current functions, and is not stored in long term memory.
- can be manipulated by drawing inferences, by making predictions, by understanding phenomena, by deciding what further processing or action to take, and by controlling the execution.

Despite this, the implementation emulates the „picture“ as the totality of the cognitively available objects at a given time, their features, and their perceived and infered relations in actual and future time and space in terms of WMEs (see section 4).

The multitude of represented objects, relations, and features within the „picture“ demand that the controllers prioritize the processing at any one time. In ACT-R, Anderson postulates a hierarchical goal structure as a direct reflection of the task dependency in the environment. To model this hierarchy of goals, several WMEs can be pushed onto the goalstack, a special structure within working memory. Processing is controlled by the current goal, which is the first element of the goalstack. The current goal spreads activation among its neighbors in the semantic net. The system focusses only on this top goal at this time. But, because of the dynamic task environment of air traffic control, there is no fixed hierarchical goal structure. Therefore, the continuously changing situation demands another prioritizing of the processing of simultaniously on-going events at any particular time. In addition, time contraints in this context force a flexible and appropriate selection of the most relevant demand for processing. In order to model this contextualized scheduling of processing, we had to postulate a different concept. Our assumption is that the scheduling of processing is determined by the state of the whole mental representation of the traffic situation. In section 4 this concept is described more detailled.

Production rules are the procedural part of memory. They consist of a condition and an action part. Conditons and actions refer to WMEs. The application of a production rule is realized by a simple pattern-matching mechanism. In order to support goal-directed performance, the first condition of every production rule must match the current goal. If all conditions of a production rule are true, then the action part is executed. Possible actions are: manipulation of the goalstack (push and pop), creation and deletion of WMEs, and modification of the slots of already retrieved WMEs. An ACT-R run consists of the continous application of production rules.

The prioritizing of processing is controlled by the activation parameter in ACT-R as well as by the current goal. A production is applied if it fires. A rule can fire if all conditions are fulfilled. Typically the fastest production will fire. The speed of application is mainly computed by the time it takes to retrieve the condition WMEs.

Activation signifies the actual relevance of a WME for the processing of information. Sources of activation are the encoding process, execution of a production (addition of new WMEs), and creation of a goal node. The more activated a WME is, the faster it is retrieved. This means that if various WMEs match the pattern of a production rule, the most activated WME is retrieved. If various production rules can be applied, that production rule fires that retrieves the most activated WMEs. A WME can only get retrieved if its activation is above a certain level. But in the case of air traffic control there are three cases in which an inactive WME also has to be retrieved. In the first case, the controller has to update his mental representation continuously. Empirical work showed that controllers reduce the problem space by paying attention to meaningful signals for conflict detection during radar-screening. Because of these signal features, aircraft become focal. That means that they are attention demanding objects, therefore highly activated. Aircraft without these features are 'extrafocal' (less activated). For these extrafocal aircraft there is no further demand for processing and they become inactive. But, in contrast to ACT-R, these inactive WME´s have to be retrieved in order to update them. Second, activation is increased not only by the encoding process. It is also guided by the encoding of signal features of aircraft. The third case concerns the context-dependend coordination of a goal. The high activation level of a goal that targets the solution of a detected conflict between aircraft can be decreased, it can be put aside for a while if there is enough time remaining for a solution. But at a certain point, activation has to increase suddenly in order to retrieve this WME and to apply the appropriate production rule in order to solve the conflict. Otherwise the both „inactive“ aircraft will collide.

Additional features of ACT-R are learning mechanisms to adjust WME and production parameters, partial matching, and the aggregation of production rules. These features are not used in our model.

Implementation of the model

ACT-R is more suitable for cognitive modelling than other non-specialized environments, such as Prolog or C, because it forces the programmer to use a theory of cognition. It uses knowledge about cognitive structures and mechanisms that would have to be reimplemented in another environment. ACT-R is implemented as an add-on for CommonLisp. Special new forms are inserted into the Lisp language to define production rules and working memory elements (WMEs).

In order to implement the cognitive model of an air traffic controller, we defined a rich representation of the „picture“. It is not possible to use a pictorial representation in working memory. The implementation's picture is a semantic net of airspace objects, anticipated events, and inferenced actions that are represented as WMEs. Some of these objects have spatial positions that make it possible to retrieve them by positions. More sophisticated operations such as retrieval by distance to other airspace objects have to be emulated.

We used the object-oriented features of ACT-R to define the structure of the picture. Every airspace object has a position on the radar screen. Derived classes are „airways“, „sector boundaries“, and „aircraft“ which have additional slots including callsign, speed, and altitude. Aircraft are specialized to „incoming“, „changing altitude“, and „near to another airspace object“ (proximity). For every class, instances are generated and modified as WMEs in working memory by data selecting productions during the monitoring cycle. Events represent inferenced knowledge about aircraft. All events refer to aircraft objects. Instances are generated by production rules in the anticipation and conflict resolution module. They belong to the event-subclasses: „monitoring“, „conflict“, and „resolution“. Conflicts can be „crossing“ or „chain“. Conflict events have an additional slot that holds a reference to the conflict partner.

The implementation of the model works in a simulation environment, that consists of an airspace simulation and a radar screen. It communicates with aircraft in the airspace simulation, when it gives commands to the simulated pilots, and with the radar screen, when it selects data to update its picture. The incorporation of new knowledge is done by Lisp callback-functions that are applied between production cycles, if new aircraft have been entering the sector, or answers to requests about aircraft parameters or information about a certain point on the radarscreen are received. This means, that working memory can be altered without control of production rules. The changing representation forces a model without an explicit process flow.

Several tasks are active at every moment. Every task is done by one of the modules „data selection“, „anticipation“,or „conflict resolution“. The superior control procedures module has to build up an ad hoc process flow depending on the current structure of the picture. To achieve this, we assume that the modules cannot be interrupted and are exclusive. The process flow is done by meta productions in the „control procedures“ module that trigger a module with an object or event as parameter. In order to trigger a module and make it not interruptible, we introduced a new class of WMEs. These „control“-WMEs are the only ones that get onto the goalstack.

The start of every module is a „top level production“. It is triggered by a „top level goal“. This kind of production will push new subgoals onto the goalstack that will trigger other productions of that module. Every production has to clean the goalstack by popping its trigger-WME. When a module is finished the goalstack should then be clean. The productions of the „control procedures“ are triggered by the „control-flow“-goal, which has no parameter. This goal is never popped. Thus when the goalstack is „clean“ it is on top of the goalstack and thus the current goal triggers the „control procedures“-module again.

Processing radio communication when a plane announces that it is going to enter the sector, is the only reason to interupt a module, make a mark in the working memory, and continue the module. The mark has a high priority so that it will be processed soon.

The meta production rules of the „control-flow“-module for the air traffic controller model use this prioritization:

1. if a solution-WME exists in the picture and it is the time to do, then do „action“ on this solution, else
2. if a conflict-WME exists and it is time to do, then „conflict resolution“, else
3. if a monitoring-event or an aircraft-WME with a signal („incoming“, „changing altitude“, or „proximity“) exists in the picture, then do „update“ and „anticipation“ on this WME, else
4. if an aircraft-WME exists, then do „monitoring“ on it.

Every solution-WME and every „conflict“-WME has a slot, where it represents when it is supposed to happen. The control productions use a function, that compares this ideal time with the current time. It fires the appropriate action according to a predefined bias.

If the current goal is „controlflow“, only the meta-productions are able to fire. They match patterns against the picture according to the prioritization scheme listed above. The chosen action will generate a new „control“-WME (CF) of the appropriate subclass. It refers to the detected aircraft-WME or event-WME. The goalstack consists now of („controlflow“,CF). This triggers the toplevel production for CF. It will produce new control-WMEs probably refering to the detected WME, pop CF, and put the new control-WMEs onto the goalstack. They trigger new sublevel productions that all pop their trigger. When the module for CF is finished, the goalstack is („controlflow“), meaning that only the meta-productions are able to fire.

The model deals well with the dynamic environment by using this control scheme. If another task needed interruptible modules, the control procedures would have to be triggered after every production cycle within the module, and the controlflow WMEs would have to be stored in the picture, when they are inactive. The meta productions would then trigger the most important controlflow-WME or generate a new one.

Concluding Remarks

We hope to have shown that the application of a unified theory of cognition as proposed by Newell (1990) or Anderson is appropriate for cognitive modelling. However, even within such a framework it is obvious that the conceptualization and implementation of mental processes in dynamic environments demands additional assumptions.

Acknowledgements

This research project is supported by the Deutsche Forschungsgemeinschaft 'DFG', Ey 4/16-2 , Fr 375/48-2.

References

Anderson, J.R. (1993). *Rules of the Mind*. Hillsdale: Lawrence Erlbaum.

Bierwagen, T., Eyferth, K. & Niessen, C. (1997). Bericht über die Arbeit des DFG-Projektes „Modellierung von Fluglotsenleistungen in der Streckenflugkontrolle" vom 15.9.1994 bis zum 14.9.1996. *ZMMS-Forschungsbericht, 97-2*. Technische Universität Berlin.

Craik, K.J.W. (1943). *The Nature of Explanation*. Cambridge: University Press.

Endsley, M.R. (1995). Toward a Theory of Situation Awareness in Dynamic Systems. *Human Factors, 37(1),* 32-64.

Falzon, P. (1982). Display Structures: Compatibility with the Operator's Mental Representation and Reasoning Process. *Proceedings of the 2nd European Annual Conference on Human Decision Making and Manual Control*, pp 297-305.

Flach, J.M. (1995). Situation Awareness: Proceed with Caution. *Human Factors, 37(1)*, 149-157.

Hacker, W. (1978). *Allgemeine Arbeits- und Ingenieurspsychologie*. Bern: Hans Huber.

Johnson-Laird, P.N. (1983). *Mental Models*. Cambridge, Massachusetts: Harvard University Press.

Newell, A. (1990). *Unified Theories of Cognition*. Cambridge, Massachusetts: Harvard University Press.

Opwis, K. & Spada, H. (1994). Modellierung mit Hilfe wissensbasierter Systeme. In: *Enzyklopädie der Psychologie* (pp 199-248). Göttingen: Hogrefe.

Rasmussen, J. (1986). *Information Processing and Human-Machine Interaction.* New York: North-Holland.

Whitfield, D. & Jackson, A. (1982). The Air Traffic Controller´s Picture As an Example of Mental Model. In: G. Johannsen & J. E. Rijnsdorp (Eds.), *Proceedings of the IFAC Conference on Analysis, Design and, Evaluation of Man-Machine Systems* (pp 45-52). London: Pergamon Press.