Model-Based Process-Oriented Human Operator Cognitive Performance Measurement for the Evaluation of Human-Machine Systems

Von der Fakultät für Ingenieurwissenschaften,

Abteilung Maschinenbau und Verfahrenstechnik der

Universität Duisburg-Essen

zur Erlangung des akademischen Grades

eines

Doktors der Ingenieurwissenschaften

Dr.-Ing.

genehmigte Dissertation

von

Andreas Florian Hasselberg
aus
Dinslaken

Gutachter: Univ.-Prof. Dr.-Ing. Dirk Söffker
Univ.-Prof. Dr.-Ing. Dirk Kügler
Tag der mündlichen Prüfung: 28. Juli 2014

Contents

List of Figures

List of Tables

List of Abbreviations

ACC Area Control Center

AED Automated Error Detection

AMAN Arrival Manager

APP Approach Control

ATC Air Traffic Control

ATCO Air Traffic Control Officer

BARS Behaviorally Anchored Rating Scale

CDA Continuous Descent Approach

CPN Coloured Petri Net

CPS Complex Problem Solving

DFS DFS Deutsche Flugsicherung GmbH

DLR German Aerospace Center (Deutsches Zentrum für Luft- und Raumfahrt)

FAGI Future Air Ground Integration

FIR Flight Information Region

EC Executive Controller

ETMA Extended Terminal Maneuvering Area

FMS Flight Management System

GEMS Generic Error Modeling System

GUI Graphical User Interface

HMI Human-Machine Interaction

HMR Human Model of References

HMS Human-Machine System

ILS Instrument Landing System

KYO Know Your Options

LMP Late-Merging-Point

LoA Level of Automation

MAGIE Micro Air Ground Integration Environment

NDM Naturalistic Decision Making

OE Operational Error

PC Planning Controller

POWER Performance and Objective Workload Evaluation

ROC Receiver Operator Characteristic

SA Situation Awareness

SDT Signal Detection Theory

SID Standard Instrument Departure Route

SLM Step Ladder Model

SMAN Surface Manager / Surface management system

SME Subject Matter Expert

SRK Skill-Rule-Knowledge framework

SOM Situation-Operator-Modeling

STAR Standard Terminal Arrival Route

TMA Terminal Maneuvering Area

TTA Target-Time of Arrival

TWR Tower Control

UAC Upper Area Control Center

UIR Upper Flight Information Regions

List of Formulas

Normative Decision Making (introduced in section 2.1.5)

$Pr(\omega\|x)$	Probability of ω as a consequence of x
$U(x)$	Utility function
X	Set of alternative actions
x, y	Actions
Ω	Set of consequences
ω	Consequence

SDT (introduced in section 2.3.1)

d'	Sensitivity

Time Window (introduced in section 2.3.1)

b	Action
$I_w^1(b)$	Boolean indicator function for b meeting situation specified in w
$I_w^2(b)$	Boolean indicator function for b being relevant towards w
i	Index of time windows
j	Index of actions
m	Total of actions
n	Total of time windows
w	Time window

Performance Measurement in Microworlds (introduced in section 2.3.3)

$F(x_N)$	Objective function
$F^*(x_n; n_s)$	*how much is still possible function*
k	Month
N	Total of months simulated (12)
$u(k)$	Input vector
x_k	State of the microworld
ΔP_k	Not used potential

Coloured Petri Nets (introduced in section 3.2.1)

Σ	Set of defined color sets
A	Set of arcs
C	Color set function
E	Arc expression assignment
$EXPR$	Set of all expressions
$EXPR_v$	Set of all expressions with $v \in V$
e	Arc expression

G	Guard assignment
g	Guard
i	Index of places
j	Index of Transitions
m	Marking
P	Set of places
p	Places
T	Set of transitions
t	Transitions
V	Set of variables
v	Variable

Situation-Operator-Modeling (introduced in section 3.2.2)

C	Characteristic
eA	Explicit assumptions
F	Function (of operator)
iA	Implicit assumptions
O	Operator
p	Parameter
R	Relation
S	Situation

Modeling of Uncertainty (introduced in section 4.3)

A	Area of triangle
a,b	Limits of triangular distribution
c	Mode of triangular distribution
d_0	Initial value of mode and limits
d_{max},d_{min}	Upper and lower boundary
e	Imprecision parameter
$f()$	Probability density function
$F()$	Cumulative density function
h	Height of triangle
l	Length of triangle
N	Total of rays
n	Index of rays
$p()$	Probability
r	Change rate

Result Evaluation Criteria (introduced in section 5.1)

Γ	Fulfillment of the objective *constraints*
Γ_{ab}	Fulfillment of the objective *constraints* (aircraft-based approach)
Γ_c	Fulfillment of the objective *constraints* (combined approach)
Γ_{cb}	Fulfillment of the objective *constraints* (constraint-based approach)
$\Gamma_{FAGI/FG}$	Fulfillment of the objective *constraints* in FAGI and FlexiGuide

γ_i	Fulfillment of the objective *constraints* by aircraft i
$\gamma_{i,altitude}$	Fulfillment of altitude constraints by aircraft i
$\gamma_{i,speed}$	Fulfillment of speed constraints by aircraft i
Δ	Fulfillment of the objective *separation*
Δ_{ab}	Fulfillment of the objective *separation* (aircraft-based approach)
Δ_{cb}	Fulfillment of the objective *separation* (conflict-based approach)
Δ_{FAGI}	Fulfillment of the objective *separation* in the project FAGI
Δ_{FG}	Fulfillment of the objective *separation* in the project FlexiGuide
δ_i	Fulfillment of the objective *separation* by aircraft i
Θ	Fulfillment of the objective *throughput*
θ_i	Fulfillment of the objective *throughput* by aircraft i
A	Duration of active constraints
A_0	Duration of active constraints in the reference sequence
$a_{l,i}$	Duration of restriction l being active for aircraft i
$a_{0,l,i}$	Duration of restriction l being active for aircraft i in the reference sequence
a_i	Duration of active constraints for aircraft i
a_l	Duration of restriction l being active
C	Total of conflicts
D	Duration of all conflicts
D_{max}	Theoretical maximal duration of all conflicts
d_j	Duration of conflict j
i	Index of unequipped aircraft
j	Index of conflicts
L	Total of constraints
l	Index of constraints
K	Duration in which aircraft violate the minimum separation
K_{max}	Theoretical maximal value of K
k_i	Duration in which aircraft i violated the minimum separation to at least one other aircraft
P	Flight duration of all aircraft in the path-stretching area
P_0	Flight duration of all aircraft in the path-stretching area in the reference sequence
p_i	Flight duration of aircraft i in the path-stretching area
$p_{0,i}$	Flight duration of aircraft i in the path-stretching area in the reference sequence
T	Flight duration of all aircraft
T_0	Flight duration of all aircraft in the reference sequence
t_i	Flight duration of aircraft i
$t_{0,i}$	Flight duration of aircraft i in the reference sequence
U	Total of unequipped aircraft
W	Duration of violated constraints
$w_{l,i}$	Duration of constraint l being violated by aircraft i
w_i	Duration of violated constraints by aircraft i

$w_{i,alt}$	Duration of violated altitude constraints by aircraft i
$w_{i,speed}$	Duration of violated speed constraints by aircraft i
w_l	Duration of constraint l being violated

Evaluation of Decisions (introduced in section 5.2)

Γ^-	Missed *constraints* performance
Δ^-	Missed *separation* performance
Θ^-	Missed *throughput* performance
Σ^-	Missed overall performance
A_x^*	Planned actions for earlier situations
E_x^*	Executed actions identified as candidates for erroneous actions
E_x	Executed actions
L_x^*	Planned actions for later intervals
P_x^*	Planned actions identified as candidates for erroneous actions
P_x	Planned actions
S_x	Pre-situation
S_{x+1}	Post-situation

Scenario (introduced in section 5.5)

$t_{s(min)}$	Minimum timely separation
$t_{s(2E)}$	Planned timely separation between two equipped aircraft

1. Introduction

In this thesis, a novel approach is developed to measure the cognitive task performance of human operators interacting with complex dynamic systems. To measure performance of human operators normally the results of interaction sequences are evaluated. Not considered are possible outcomes of the interaction which would have been reached if the human operator had acted differently. This is a serious drawback of existing performance measures.

The approach to measure human operator performance developed in this thesis considers explicitly the alternative actions human operators could have selected. Therefore, a cognitive planning model is implemented, which analyzes available actions and their consequences represented in a discrete state space. The plans generated by this model are used as criteria for the evaluation of actions of human operators. As the developed method measures performance during the interaction, it is called process-oriented. To identify the impact of mental prediction uncertainty on human operators' cognitive task performance, a concept to represent uncertainty in state spaces is developed. Subsequently, the developed model-based process-oriented human operator cognitive task performance measurement is demonstrated and validated.

The developed performance measure makes a more detailed assessment of the human operators possible and thereby facilitates the improvement of **Human-Machine Systems (HMSs)**. The motivation for the development of this performance measure is given below. Afterward, the remaining chapters of this thesis are briefly outlined.

1.1. Motivation

Human operators controlling complex dynamic technical systems play a central role in many domains. Examples are manufacturing plants, refineries, power plants, flight decks, and air traffic control. In these domains, automation increased during the last decades. On the one hand, automation led to relieving the work burden of human operators. On the other hand, new tasks were assigned to human operators and automation caused the work of human operators to become more complex. However, human operators continue to be a key component of automation concepts if their task cannot be automated or in order to incorporate their creativity, flexibility, and knowledge into the combined HMS.

Especially in safety critical domains like industrial process control, power plant control, or air traffic control, the consequences and implications of newly developed computer-based assistance or procedures have to be evaluated before the innovations are transferred into the field. New computer-based assistance or procedures for HMSs are validated in human-in-the-loop simulators [MS05]. The measurement of human operators' task performance is an important aspect of these validations.

For this reason, it is important to examine both how human errors arise, and how they can be prevented. It is important to know not only when erroneous actions occur but especially why the operator selected an action. Of particular importance is the identification of the difference between the real system and the human operators' mental models of the system and why the wrong action made sense for the operator as a consequence of this difference [Dek06].

The identification of actions, which lead often to high task performance, and actions, which often cause a negative task performance and a high risk for erroneous actions, allows for the provision of tailored assistance and for the improvement of the working conditions of human operators. The more precise the human task performance can be measured and the more specific need for assistance can be identified during the evaluation, the more the further improvement of the system is facilitated.

The measurement of performance and the detecting of erroneous actions when dealing with extreme situations is essential also in human-in-the-loop simulations conducted for the purpose of human operator training.

Performance measurement is also important in so-called adaptive automation [KR99, PCV09], or in the supervision of human operators [GOS07]. In adaptive automation, an implemented allocation authority changes the **Level of Automation (LoA)** dynamically according the current needs of the operator [PBD+92]. Amongst others, the human performance can be used as indicator for the need of additional assistance [KR99, PCV09]. A similar application is the supervision of human operators by technical systems. Such systems detect human erroneous actions, issue warnings, or correct the consequences of erroneous actions if necessary. To realize this concept, human operators' actions are evaluated continuously according to their goal-directedness to intervene or takeover in the negative case [GOS07].

Measures of human performance can be distinguished between process measures and product measures [HS09, CPS97, HR84]. Product measures look at the result of a task and assess the outcome in relation to the goals (e.g. time and cost). Process measures look at the activities and individual steps executed during a task and assess how the result was reached (e.g. deviations from the optimal trajectory).

In complex dynamic environment, often several different options exist to reach similar results. Hence, the steps executed during a task cannot be inferred from the reached result. In contrast, the result can be inferred if the steps are known. Consequently, product measures disregard some information and a process measure is necessary to indicate steps causing deviations from the optimal solution. Thus situations in which assistance could help are identified.

Therefore, an objective process measure of human task performance which is applicable for the evaluation of HMSs but also for adaptive automation and human supervision would allow a more detailed analysis of the system respectively a more specific support of the human operator. If an objective process measure is available for the evaluation of HMSs, additionally the detection of inefficient behavior and human erroneous actions as well as identification of the exact situation in which they occurred becomes possible. This information can afterward be used to improve the evaluated HMS.

Measured values cannot be interpreted if no criteria exist which define what should or what could be reached [YLVC02]. Consequently, criteria are necessary or otherwise measured values are useless. The necessity of a situational criterion becomes evident if it is considers that leading a dynamic system into a critical situation can only be blamed to a human operator if there was actually a choice to avoid it. It should be asked, up to what point options were available to prevent a critical situations. In this context it is important that human operators are often able to recognize their errors and correct them, if sufficient time is available and opportunities exists.

Besides avoiding critical situations, a common objective is to perform a process as efficiently as possible. Consequently, a reduction of efficiency can be considered as a reduction of performance and as an erroneous action. Also in this case, it is important to know which options existed and if the operator had the choice to avoid a loss of efficiency.

The performance measure developed in this thesis adopts the general idea of [OHS11]. In [OHS11] the available options were calculated and compared to the options judged as feasible by the human operators. The available options denote the set of all actions an operator can perform in a specific situation. In [OHS11], single actions are considered as options. In this thesis, the term options is slightly modified and denotes all interaction sequences an operator can perform following to a specific situation. In other words, the term is extended from single actions to multiple actions.

The performance measure developed in this thesis applies the consequences of the available options as criteria to measure the human operators' performance. The consequences are compared to the options finally selected by the human operators. In order to assess not only the direct consequences of decisions but also the long-term consequences, available interaction sequences are considered as options. To make an objective performance measure possible, the goal-directed interaction sequences indicating the best existing option in each situation are used as criteria. By comparing the consequences of decisions to the consequences of the goal-directed interaction sequences, each decision can be evaluated. Thus, not only the result of an interaction but also the process—considered as a sequence of decisions—can be evaluated.

A cognitive model of planning is developed in this thesis to generate goal-directed interaction sequences which represent the best available option. This model analyzes a state spaces representing the available actions and their consequences. The developed performance measure should not be limited to discrete applications as former approaches of state spaces analyzes for the evaluation of HMSs. Consequently, existing approaches are extended in this thesis to be able to scope with time-depended dynamics.

According to [End95a], "situation awareness, as such, incorporates an operator's understanding of the situation as whole, forming a basis for decision making". Furthermore, situation awareness, as defined in [End95a], consists of the three levels "Perception", "Comprehension", and "Projection". Consequently, situation awareness can be incomplete if the situation is not completely perceived, not correctly understood, or its future development not correctly predicted. In particular, the prediction of future states poses a challenge [Dör80]. Thus, actions of human operators can be based on uncertain knowledge about future states of the system. If human operators are uncertain about the consequences of possible actions, they might tend to choose a solution which has a low probability of severe consequences but a high probability of a deviation from the optimum. This uncertainty is an inevitable aspect of HMSs. It can be the crucial difference between a human operator's mental model of the system and the real system. The impact of uncertainty can be quantified if it is possible to use interaction sequences a human operator affected by uncertainty would select as criteria to measure the performance.

1.2. Aims and Strategy of This Work

The overall aim of this thesis is the development and validation of a process-oriented measure of human operators' cognitive task performance. The measure should be applicable to complex dynamic task environments that can be formalized and in which rational decisions are possible. Furthermore, the developed performance measure should also be applicable if the decisions are effected by uncertainty.

To achieve the main aim, the following associated aims are must be reached. As reasoned above, goal-directed interaction sequences are used as criteria, which are regarded as a plan which could be implemented by human operators. To generate such interaction sequences, a cognitive model of planning is realized in this thesis. The model takes up the idea to formally describe the action spaces as a discrete state space but extends existing approaches by considering time-depended dynamics.

To make the identification of the impact of uncertainty on decision making possible, uncertainty in assisted HMSs is analyzed and an appropriate modeling approach is chosen, implemented, and integrated into the cognitive planning model.

Finally, the applicability of the developed method is demonstrated in a microworld simulating a simplified **Air Traffic Control (ATC)** task.

To summarize shortly, the aims of the work described in this thesis are

- to develop a process-oriented measure of human operators' cognitive task performance,
- to use goal-directed interaction sequences as criteria for the measurement,
- to develop a cognitive model of planning used to identify goal-directed interaction sequences,
- to extend state-space based methods by time depending dynamics,
- to identify and to model human operators' uncertainty in assisted HMSs,
- to integrate a representation of uncertainty into state spaces,
- to validate the developed measure of human operators' cognitive task performance in a study, and
- to demonstrate the benefits of this measure for the improvement of HMSs.

1.3. Organization of This Work

This thesis is structured as follows and depicted in Fig. 1.1. At first, the background of this thesis is presented in chapter 2. In this chapter, the role of human operators in assisted HMSs is described, the ATC domain is introduced, and existing approaches for the measurement of human cognitive task performance are analyzed.

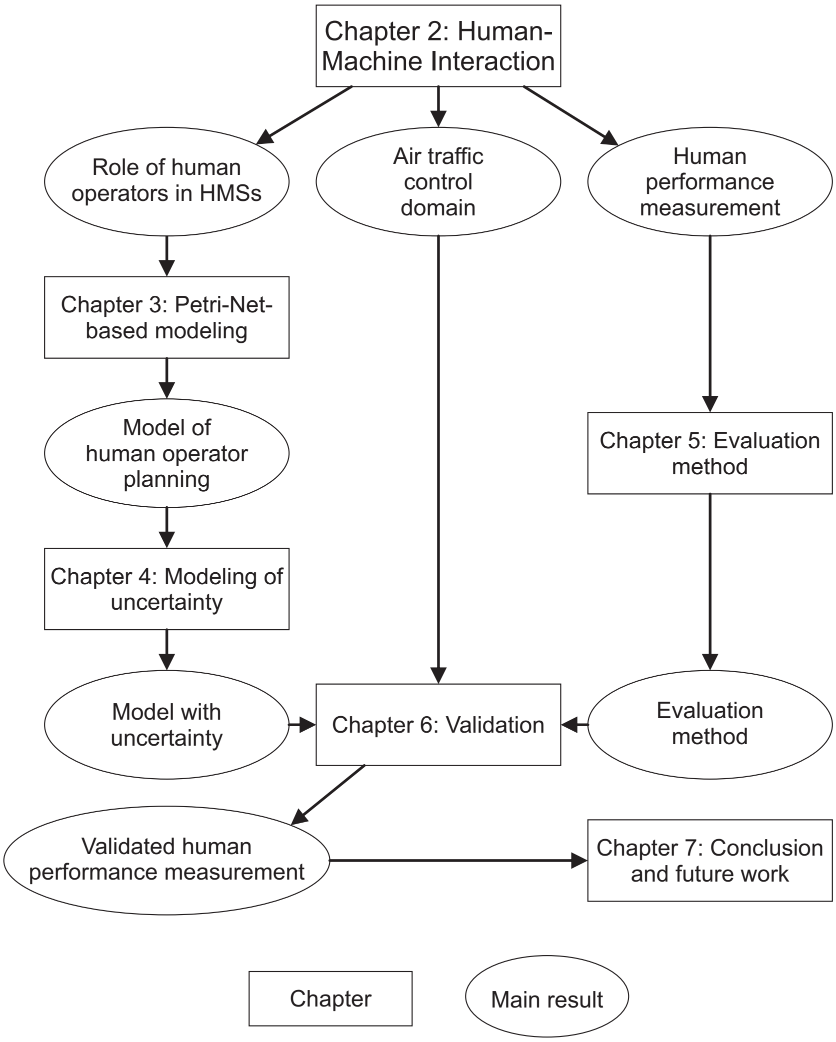


Figure 1.1.: Organization of work

In chapter 3 the applied modeling techniques are introduced first. After that, the example application **Micro Air Ground Integration Environment (MAGIE)** is presented. Finally, the cognitive planning model for the application example is realized. Based on an analysis and classification of uncertainty in assisted HMSs, the planning model is extended by uncertainty connected with prediction of future system states in chapter 4. The procedure for the evaluation of human operators' decisions is developed in chapter 5. Additionally, the functions to evaluate the consequences of actions in the example application are introduced in this chapter. Moreover, the application of the developed procedure is demonstrated as an example. Subsequently, the developed measurement of human operators' cognitive task performance is validated in chapter 6. Finally, conclusions are drawn and future research directions are indicated in chapter 7.

2. Human-Machine Interaction with Complex Dynamic Systems

This chapter explains models, theories, terms, concepts, and task environments which are referred to later in this thesis. The chapter starts with an introduction of models and theories describing the role and tasks of human operators in assisted **Human-Machine Interaction (HMI)** in section 2.1. As the performance measure developed in this thesis is demonstrated with a simplified ATC approach task, this domain is presented in section 2.2. Section 2.3 is dedicated to the measurement of human (cognitive) task performance and gives an overview of measurement methods and constructs used in HMI in general, in the ATC domain, and in microworlds. Finally, in section 2.4, the method developed in this thesis is compared to existing measurements of human task performance and the advantages and disadvantages are discussed.

2.1. The Human Role in Automated Systems

This section is intended to give a short overview of the role and tasks of human operators. As the distribution of tasks and responsibilities between human, machine, and an additional technical assistance is of fundamental importance, it is discussed first (see section 2.1.1). Then models of human information processing are presented (see section 2.1.2) and selected information processing steps are detailed.

Some descriptive models of planning are presented in section 2.1.4 as planning of human operators is of particular importance for the developed cognitive model. Different theories of decision making are briefly presents in section 2.1.5 to analyze possible explanations for deviations of human operator's decision from the best available solution. As the field of human error research provides another point of view for explaining these deviations, a short introduction is given in 2.1.6.

2.1.1. Computer-based Assisted Human-Machine Interaction

The interaction between human operators and machines is a characteristic element of many working domains. In the last decades, automation increased and interaction changed from mainly direct control to knowledge-based supervisory control. In other words, human operators do not influence an output by appropriate manual control actions anymore [Söf03] [Cac98]. Instead, the task of human operators is to monitor the relevant information sources, to interpret this information, to make decisions by connecting knowledge with the interpreted information, and finally to implement these decisions.

Human operators can be assisted in performing these cognitive tasks. Due to the tasks' complexity, this assistance is commonly based on computers. Technical progress allows developing computer-based assistance even for very complex systems in which human operators formerly were unsupported. Computer-based assistance is often regarded as automation in the literature. For example Parasuraman et al. define automation as “a device or system that accomplishes (partially or fully) a function that was previously, or conceivable could be, carried out (partially or fully) by a human operator” [PSW00]. Following this definition, devices taking over only small functions from human operator, for example alarm systems, are called automation, even though they do not act on the machine/controlled systems. To avoid misunderstandings, the term “computer-based assistance” will be used in this thesis to describe such devices. Nevertheless, if such a device replaces the human completely, the task is fully automated as the process of automation is completed. A task cannot only be fully automated (the control loop is closed by an implemented controller) or fully manual (the control loop is closed by a human operator); it also can be partially automated, for example, if a technical assistance senses and interprets the system state and suggests actions. In this case, human operator and technical assistance jointly control the task.

A lot of models and concepts - also this thesis itself - focus on human operators of HMSs but are not limited to the interaction between humans and machines. Models and concepts can often be extended to interactions of human operators with complex environments, for example medical application (interactions with biological system) and emergency management. Indeed, the example application of this thesis - the air traffic control task - is not a solely technical system. In this context, system refers to the air traffic including aircraft and pilots. Therefore, the more general term “controlled system” is used instead of the term “machine” in the following.

Systems controlled by human operators are often attributed complex and dynamic. These attributes are valid not only for machines but for all types of systems with which human operators interact. For this extended view, a dynamic systems was defined by Brehmer as a system whose states change in real time autonomously and as a consequence of the operators’ actions [Bre92]. Furthermore, the complexity of a system was defined as a result from its structure and the relation between the input and outputs which may be nonlinear or noisy [Osm10].

When designing new concepts for HMSs, the crucial question is, weather tasks are handed over to the computer-based assistance or remain under the control of the human operator. If a task is carried out by the human operator, the actions are flexible and limited only by operator’s cognitive capabilities. If the task is transferred to the assistance, the actions are limited to the functions implemented by the designer but they are carried out with high precision and constant quality.

An early definition of rules which helped in deciding what tasks to hand over from the human operator to technical assistance was Fitts’ list [Fit51]. On the list, the capabilities of humans and technical assistance are compared in different task abilities. Tasks which are better performed by technical assistance should be done by technical assistance. Examples are tasks which require quick reactions to control signals or repetitive routine tasks. Otherwise tasks should be assigned to humans operators, if their abilities surpass machines. Examples are task which require inductive reasoning and flexible procedures.

Human-Machine Systems are often designed following such a technology-driven approach. Every task is considered separately and shifted from the human operator to the technical assistance if a technical device can be developed which outperforms the human operator. Only tasks, where technology is not able to deliver better performance, are assigned to human operators.

The opposite of such technology-driven approaches are human-centered approaches. One example is the concept of **Levels of Automation (LoAs)**. This approach gives guidelines to decide which task should be handed over to an assistance and which task should remain under the control of the human operator. The assignment of tasks is independent from the technical possibilities.

For example, Sheridan and Verplanck [SV78] developed a taxonomy with ten LoAs. The taxonomy is shown in Fig. 2.1. On the first level, the human operator makes decisions but the decisions are implemented by the assistance. At level 2 to 5, the assistance supports in determining the options and suggest one options (level 3 to 4) or asks for the human operators approval to implement it (level 5). At level 6, the assistance implements an option when the human operator does not intervene. At level 7 to 10, the assistance makes decisions and implements them and tells the human operator later what it did.

Level	Description
1.	Human does the whole job up to the point of turning it over to the computer to implement.
2.	Computer helps by determining options.
3.	Computer helps determine options and suggests one, which human need not follow.
4.	Computer selects action and human may or may not do it.
5.	Computer selects action and implements it if human approves.
6.	Computer selects action, informs human in plenty of time to stop it.
7.	Computer does whole job and necessarily tells human what it did.
8.	Computer does whole job and tells human what it did only if human explicitly asks.
9.	Computer does whole job and tells human what it did and it, the computer, decides he should be told.
10.	Computer does whole job if it decides it should be done, and if so tells human, if it decides he should be told.

Table 2.1.: Levels of automation in decision making by Sheridan and Verplank [SV78]

Another taxonomy was developed by [End87] which included five different role allocations between human operator and assistance. The classification was later extend to ten levels of automaton [End99]. In this extended taxonomy, a task is divided into four functions: monitoring, generating (options), selecting (an option), and implementing. In each LoA, each task is either assigned to the human, to the assistance or to both. In the latter case, both fulfill the function jointly but the sharing of responsibility is not defined.

While former classifications are mainly based on making and implementing decisions and, thus, on the output functions of a system [PSW00], this classification includes also the input functions, which are monitoring and generating options. A similar approach is followed by the classification of [PSW00], which distinguishes between types and levels of automation. Here, automation is divided into four types according to stages of human information processing. Similar to [End99], these stages are information acquisition, information analysis, decision selection, and action implementation. The main difference to [End99] is that a separate LoA (from fully manual to fully automatic) is possible for each type of automation.

A lot of research was carried out to identify the influence of types and levels of automation on HMS performance during normal operations and in failure conditions, on workload, and situation awareness. For a recently conducted meta-analysis see [WLS+10]. This research identified some benefits of higher levels of automation. First of all, some studies showed that the overall performance of HMS can be increased under normal operating conditions if more functions are assigned to the computer-based assistance (e.g. [MRO12, MP05]). Furthermore, higher LoAs often reduce operators' workload (e.g. [MP05, PCV09]). Both effects are the main reasons for automation of specific tasks formerly carried out by human operators.

As a consequence of higher LoAs and taking human operators out of the control loop, human operators' **Situation Awareness (SA)** decreases (e.g. [EK95, KE04]). This can cause out-of-the-loop problems [KOE00] with the consequence that human operators may not be aware of the actual state of the controlled systems. Consequently, they may not be able to make decisions as fast as needed after taking over if the assistance fails. This is assumed to be one reason for reduced performance if the assistance is not reliable [End99].

By changing the allocation of task and handing over more task to the assistance, completely new problems can arise, for example, when the assistance's and human operator's goal are in conflict [Bai83].

Following the human-centered approach of LoAs, not every task should be handed over to the computer-based assistance. Instead a medium LoA should be selected (e. g. [PSW00]) as higher LoAs, resulting from technology-centered approaches, have serious drawbacks as described above [End99]. Thus, human operators play a central role in many domains despite the technical progress and the increasing automation.

2.1.2. Models of Human Information Processing

Controlling a complex system with accessing knowledge can be divided into several steps. As already mentioned, the classification scheme of types and levels of automation [PSW00] defines four different types of automation according to four human information processing steps, namely sensory processing, perception/working memory, decision making, and response selection (s. Fig 2.1).

The first step of human information processing—sensory processing—includes registering different sources of information, focusing the attention, and pre-processing the data. The next step—perception—includes the conscious perception of data and its manipulation and combination in working memory. The decision is made at the third step—decision making—and a corresponding action is carried out in the fourth step—response selection.

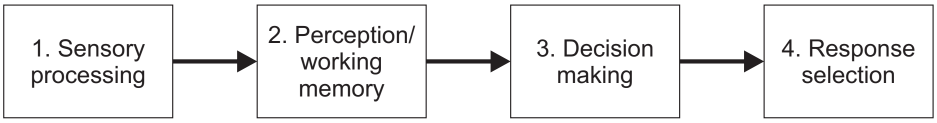


Figure 2.1.: Four steps of human information processing [PMH00]

Similar steps are described by the **Human Model of References (HMR)** [Cac98], which also models human information processing (s. Fig 2.2). In the HMR, the four steps to get from stimuli to response are "perception", "interpretation", "planning" and "execution", which are called cognitive functions and summarized under the term PIPE. This model further contains two cognitive processes which are "memory/knowledge base" and "allocation of resources". The first process is connected to all cognitive functions to allow them to access the stored knowledge. The second cognitive process is connected to all functions and to the memory and defines the sequence of cognitive functions.

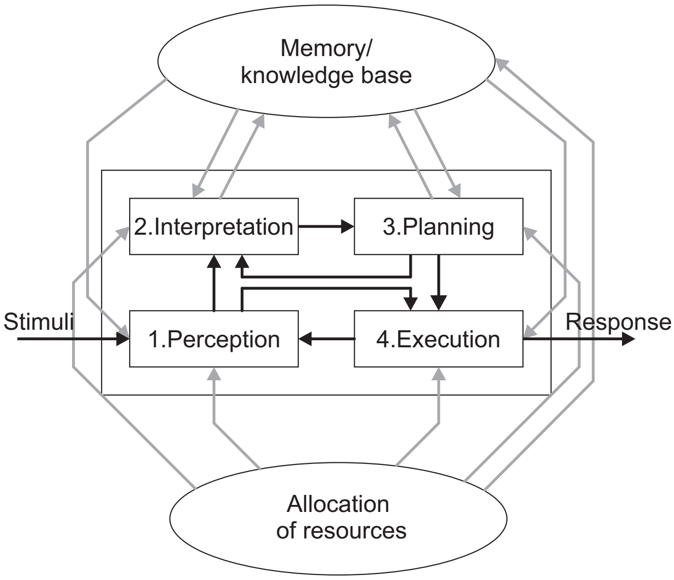


Figure 2.2.: HMR/PIPE Model of human information processing [Cac98]

Both models of human information processing consist of four steps but have two main differences. Although, information is perceived and interpreted in the first and second step in both models, the distribution of the task into both steps differs. In the HMR, the perception is completed in the first step so that the second step includes only the interpretation. In the model shown in Fig. 2.1, the first steps includes only the sensory processing so that perception and interpretation are carried out in the second step. The second main difference between both models is that the third step is called "decision making" in the model depicted in Fig. 2.1 and "planning" in the HMR. Both steps describe the same task, but the name is used to put the focus on the result (decision) or on the process (planning).

A more detailed model of human information processing is the **Step Ladder Model (SLM)** [Ras86] depicted in Fig. 2.3. The model describes a procedural behavior, beginning with activation by the detection of a problem and ending with the execution of an action. Between these two steps are the observation, identification, assessment of the consequences, and the choice of tasks and procedures. Noteworthy is that this model also describes shortcuts between these steps, so that not all of them have to be executed.

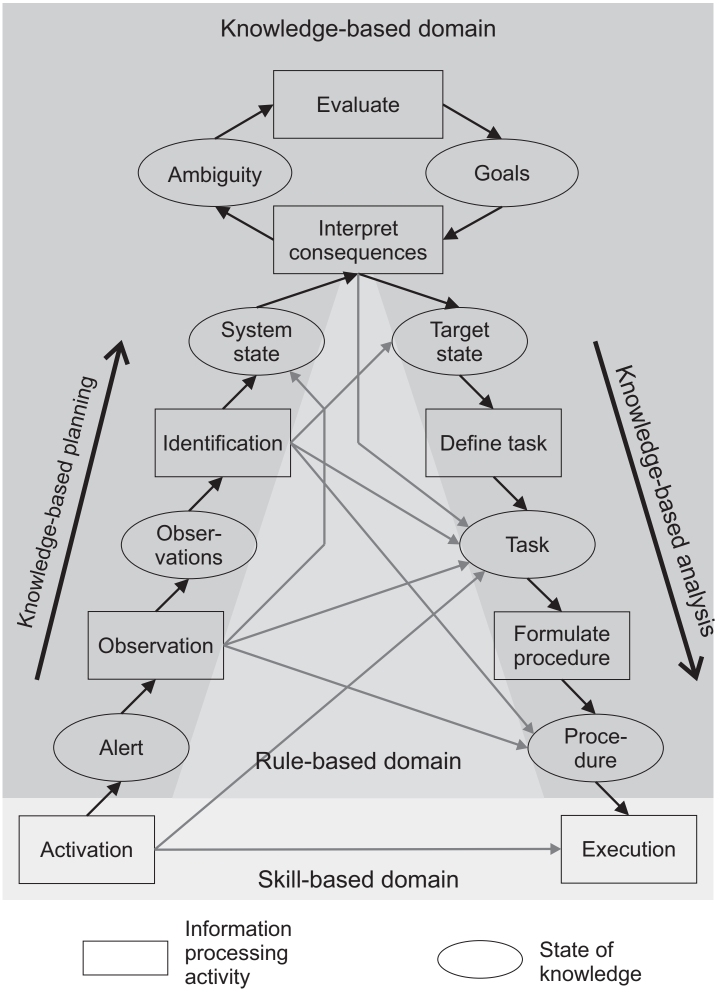


Figure 2.3.: Step ladder model (SLM) [Ras83]

The SLM contains eight information processing activities depicted as rectangles and eight resulting states of knowledge depicted as ovals. The eight information processing activities describe the same procedure as the four cognitive functions of the HMR but the SLM has a higher granularity.

The SLM can be combined with the **Skill-Rule-Knowledge framework (SRK)** [Ras79, Ras83]. According to the SRK, human operators perform actions on different cognitive levels. Skill-based behavior is the lowest level and describes the direct connection of activation and execution. This behavior is highly automated (in the human mind) and takes place without conscious control. At the medium level—the rule-based level—the behavior is consciously controlled by stored rules with an if-then scheme. These rules may be instructed or derived from experience. The knowledge-based behavior—the third level—is only necessary if the human has to solve unknown situations. Only on this level, the knowledge about the system has to be utilized.

The activities of the SLM can be assigned to these three different levels of behavior as illustrated in (s. Fig 2.3) [Ras86, p. 104]. The skill-based behavior allows a direct connection from activation to execution. A multitude of shortcuts exist in the SLM between particular processing steps. These shortcuts are activated by rules and belong to the rule-based level. The complete process from activation to execution belongs to the knowledge-based level is entered. This level is only used as a matter of last resort when no rules or skills can be applied.

The remainder of this section presents models and theories focusing on selected steps of human information processing in detail.

2.1.3. Situation Awareness

A common characteristic of the models of human information processing described in the previous section is the inclusion of perception and interpretation of sensory inputs at the beginning of the process. Perception and interpretation is a pre-requisite for decision making. Without knowing what is going on, the human operator is not able to make appropriate decisions.

The result of the perception and interpretation process can be described by **Situation Awareness (SA)** [End95a]. As proposed by the SLM, especially well trained operators are able to apply rules, which allow them to proceed with action selection immediately after they classified the situation. Consequently, the correct classification of the situation is of utmost importance to activate the correct rule and, hence, to make the right decision.

To clarify the construct, a definition of SA is necessary. The following common definition as a state of knowledge was given by Endsley [End88,End95a]: "Situation awareness is the perception of the elements in the environment within a volume of time and space, the comprehension of their meaning, and the projection of their status in the near future."

Following this definition, SA has three hierarchical levels (see the top of Fig. 2.4). The first level describes the perceived status and dynamics of the relevant elements in the environment. It includes the awareness of the elements of the environment without knowing about their identity, meaning, or relations. On this level, a non-expert may reach the same SA as an expert because the first level of situation awareness does not depend on expert knowledge. The second level characterizes the result after the recognition of patterns and the understanding of the elements perceived in relation to goals. Thus, it depends on the first level of SA and on the knowledge of the human operator. The third level contains the prediction of future states and actions of the elements. The human operators forecast of the future development of the situation

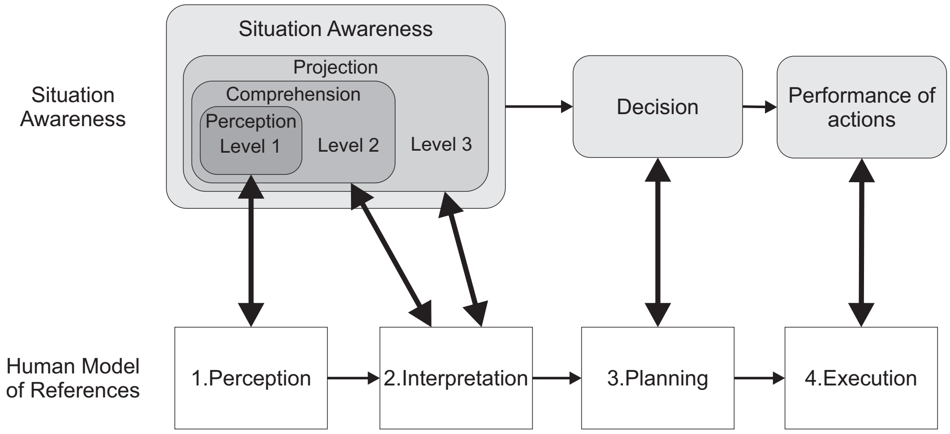


Figure 2.4.: Three hierarchical levels of situation awareness as basis for decisions [End95a] at the top and their mapping to the steps of the HMR at the bottom

is also based on the interpretation of the current situation and the knowledge of the operator about the elements' behavior. Situation Awareness as a state of knowledge should be distinguished from the process of achieving, acquiring, or maintaining SA, which is termed "situation assessment" [End95a].

The "elements in the environment within a volume of time and space", which are mentioned in the definition, are these elements of a working domain which are relevant to reach the actual goal. As these elements may differ in time, SA takes into account the dynamics of a task and relates the current state of the environment to the past and to the future.

The SA model can be mapped on the function of the HMR as illustrated in Fig. 2.4. The first two functions of the HMR can be found in the process of achieving SA. The result of the first step of the HMR (perception) can be mapped directly onto the first level of SA. The second step of the HMR—the interpretation—is split into the processes to acquire the second and third level of SA. The SA model emphasizes the importance, not only to understand the current situations, but also to predict its implications for the future.

The SA model further contains the steps "decision" and "performance of actions" which are executed after achieving a sufficient SA. These steps correspond to the last two steps of the HMR, with the already mentioned difference between decision making and planning. Planning is not explicitly considered in the SA model. The third level of SA includes only the projection of the future states of the environment and not the influence of the operator actions explicitly.

2.1.4. Planning in Human-Machine Systems

Decisions can be made directly after a situation is assessed. This corresponds to the skill-based level in the SRK framework. However, the effects of actions have to be simulated mentally prior to decision making in complex cases. This mental simulation of different actions can be considered as planning. Thus, planning can be a part of decision making and describes the mental simulation of the consequences of actions prior to making a decision. Planning may aim to set a course of actions leading to a desired future state [Söf03, WJM06]. However, it may be the case that the aim is not to reach specific states but to avoid critical states. In this case, planning can also be used to avoid specific states. After the definition of van Wezel et al. [WJM06], planning comprises the decisions of future actions. Thus, planning can include the simulation of not only single actions, but of a sequence of actions.

The planning process results in a plan which contains the future actions. However, a plan may include only the objectives to be achieved. This is obvious if the plan is generated and executed by different agents (e.g. human operator, cognitive technical system). In this case, the executing agent has to choose the actions to reach the objectives. Also human operators of a complex dynamic system may have plans including only objectives if the actions to reach these objectives are clear and do not have to be specified beforehand. Thus, a plan can include the objectives to be reached and the actions to be executed independently of each other.

Anticipation is needed to generate a plan [Dör89,Söf03]. To be able to anticipate future states, the planning agent needs an internal model of the environment. To predict the consequences of the actions regarding the agent itself, the agent needs additionally an internal model of itself.

Different models were proposed to describe the generation of plans. Following the model of Mumford et al. [MSD01], mental planning can be divided into the generation phase, the refinement phase, and the execution phase. In the first phase (generation), a plan is generated which can still be a prototype. In the second phase (refinement), the plan can be revised and completed and details can be worked out. In the third phase (execution), the plan is executed. Depending on the task, the phases can overlap in time. Especially when operating with complex dynamic systems, the refinement phase and the execution phase are alternating. While a plan is implemented, new opportunities arise and previously considered alternatives disappear. Crucial here is the process speed which determines the amount of time that can be used for the refinement of a plan. As planning and execution are carried out in parallel during operations with complex dynamic systems, a clear distinction between the generation phase and the execution phase is not possible [Hoc06].

In the model of Hayes-Roth [HRHR79] an opportunistic approach is followed. In this model, the planning process is not structured but it is a hybrid process that can also explain seeming unstructured planning. In this model, a mental plan is created by so-called specialists, which work together. The specialists access common knowledge and the already generated parts of the plan. The generated parts are shared among the specialists and are referred to as blackboard. If the preconditions required by a specialist are satisfied, that means the information it needs is on the blackboard, it can execute its actions and add further elements to the plan. Switching between various points of the plan during the planning process can be explained with this modal. In addition, this model can also describe a change between different levels of detail. For example, it can be used to explain that a detailed part of a plan is generated prior to an abstract part.

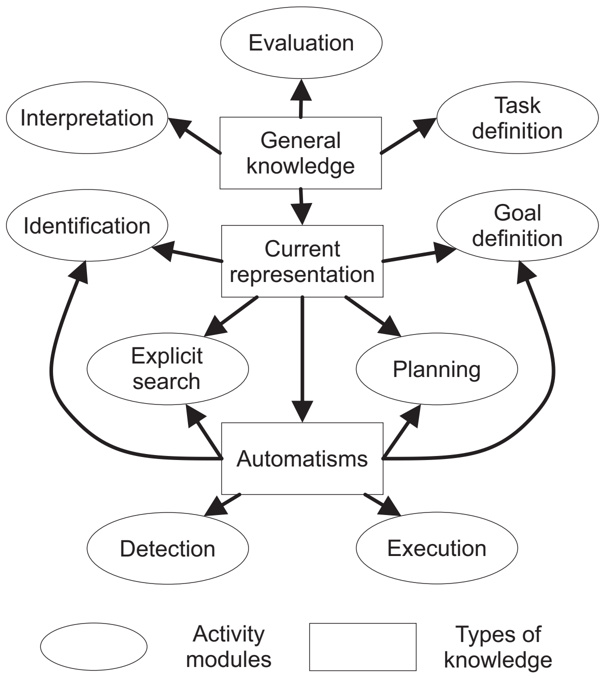


Figure 2.5.: The Dynamic Situation Management architecture by Hoc [Hoc06]

Another approach is the "case-based planning" by Hammond [Ham90]. This approach assumes that plans are created based on previously applied and remembered plans. Plans are stored in memory together with the objectives they can achieve and the problems they can prevent. Before a plan is recalled, the objectives to be reached have to be defined and the expected problems have to be predicted. If it turns out that a remembered plan contains an error, a correction of the plan is carried out. If an incorrect prediction leads to the error, an improved rule for the prediction of problems will be stored in addition to the improved plan. This approach emphasizes that the knowledge and the experience play an important role in planning.

In addition, the SLM (see section 2.1.1) was frequently used to describe diagnosis and planning in dynamic tasks. However, the SLM is criticized for being too procedural (e.g. [Hoc06, Bai97]). In particular, it is criticized that the model takes no feedback from the environment into account, that the model primarily operates on a reactive basis, and that no time is considered in the model. The Dynamic Situation Management architecture is proposed [Hoc06] as an alternative (s. Fig 2.5). It has an representation of the current situation as core. Modules similar to these in the SLM are activated by the representation of the situation. Thus, the steps in this model are less procedural and are led by a mental model. While the SLM concentrates on an external activation through observation of the environment, the Dynamic Situation Management architecture concentrates on internal activation by a mental model. In contrast to both models, the opportunistic model of planning allows both types of activation.

The analysis of these models of planning shows that planning is always associated with application of knowledge in the various models. It is also important to note, that different planning phases can be distinguish and that planning alternates between the refinement phase and the execution phase in dynamic systems.

2.1.5. Decision Making

Decision making is defined in [Sha99] as “the process of choosing a preferred option or course of action from among a set of alternatives”. Included in the process of decision making are information-gathering, likelihood estimation and choosing [Sha99]. This definition of decision making overlaps with situation assessment and planning. Here, the step of “information-gathering” is attributed to generating situational awareness and the likelihood estimation is considered as planning. Only the subsequent step of choosing an action is deemed to be decision-making.

Decision making can be analyzed by the normative, descriptive or prescriptive approach [Sha99]. The normative approach describes a rational behavior. Thus, it describes a way, how one should act to find the optimal choice. The descriptive approach analyzes human behavior based on empirical observations and tries to identify the factors that influence decision making. Research led to the conclusion that the behavior of humans strongly differs to normative behavior (e.g. [KT79]). Reducing this difference by influencing decision making and making it congruent with normative behavior is the central topic of the prescriptive approach [Sha99].

In this thesis, first, the normative approach is important as it can be used as a criterion for evaluating the decisions actually made by human operators. Second, the descriptive approach is important as this approach can describe the actual behavior and, hence, can help to explain the differences to the normative behavior. Therefore, both approaches are briefly described below.

Normative Decision Making

The rational decision making theory (e.g. [Doy99]) describes an ideal decision making, which is rarely observed but useful as contrast to the observed behavior. The following mathematical description is taken from [Doy99]. The theory of rational decision making assumes a set of alternative actions X together with their consequences Ω. It may be too extensive to consider all possible actions so that only a subset of all possible actions is considered, which include the important or interesting differences between alternatives. Further a binary relation of preferences is considered by the theory. This is notated as $x \precsim y$, which means that y is weakly preferred to x $(x, y \in X)$. It is required that the relation of all alternatives can be ordered, which means to be reflexive $(x \precsim x)$ and transitive ($x \precsim y$ and $y \precsim z$ imply $x \precsim z$). The maximally desirable alternative x will be chosen among all alternatives, so that $y \precsim x \ \forall \ y \in X$. The preference order can be replaced by a numerical utility function $U : X \to \mathbb{R}$ by assigning a number to each alternative so that $U(x) \precsim U(y)$ if and only if $x \precsim y$. This is described by the utility theory (e.g. [Wel99]). The function U is an ordinal utility functions as it represents only ordering and not magnitude.

To describe decision making in cases in which the same action may lead to different outcomes with known likelihood, the criterion of maximizing expected utility is used. With $Pr(\omega|x)$ giving the probability of the outcome $\omega \in \Omega$ as a consequence of action $x \in X$, the expected utility of the actions x is

$$U(x) = \sum_{\omega \in \Omega} U(\omega) Pr(\omega|x). \tag{2.1}$$

In this case a cardinal utility function has to be used, which indicates order and magnitude, to allow the weighted addition.

To make a decision following the normative approach is very extensive. The required effort is increasing if decisions have to be made in complex or dynamic systems. Therefore human operators are seldom able to follow the normative approach.

Descriptive Decision Making

An early descriptive decision making theory was proposed by Herbert Simon. According to that theory, humans try to find a good-enough solution (statisficing) instead of an optimal one (maximization) [Sim55]. If choosing among alternatives, the decision maker sets an aspiration level and chooses the first alternative that meets this level.

Another example for a descriptive theory of decision making is **Naturalistic Decision Making (NDM)** [Kle93]. This theory was developed from field observations of experienced decision makers. According to this theory, experienced decision makers do not compare different options but recognize typical situations and then know what to do. In more complex cases, they will mentally simulate the action and then either decide to implement or to modify this action. If the situation is not familiar, they will seek more information to detect similarities to known situations. Only if the first action coming to mind is not considered as feasible, another option will be considered. This is done in a serial process. This model of decision making is called Recognition Primed Decision (RPD) model [Kle93]. After Klein, this behavior of decision makers can be found in real-word-settings (which are the opposite of artificial laboratory tasks), if

- the decision maker is experienced,
- the environment is dynamic,
- a real-time reaction is required,
- the decision has to be made under time pressure,
- the task is ill-defined,
- and the decision has significant personal consequences [KK91].

Most of these conditions are fulfilled also in the interaction with technical systems. It should be noted that also an ill-defined task is seen as requirement. In an ill-defined task, the number of alternative actions is not fixed. It may require some effort to determine the set of alternative actions in such a task, or it may even be impossible, when this set is unbounded. However, the tasks of operators of technical systems are often well specified. Additionally, the set of actions is limited by the interface.

Heuristics may also play an important role in decision making [Gig08]. A first reason for the use of heuristics is the trade-off between accuracy and effort, which is often assumed when information costs time or money. As a consequence of this, the cost for information will exceed its benefits at one point, where the search for information should be stopped. However, it also has been shown that heuristics can lead to better decisions than complex strategies [Gig08] even if information is for free. This is especially the case if inferences have to be made based on small samples and the future may change in unexpected ways.

According to these theories, operators will not evaluate all available options and chose the best available; instead they will select an option which provides satisfactory results. Operators of dynamic system cannot collect all information, as they have a limited amount of time for making a decision. As a consequence, the use of heuristics seems necessary to reduce the effort to a manageable amount. When compared to the SLM (see section 2.1.1), the heuristics correspond to the rules, which can be used as shortcuts between the particular processes. Although the rules are not always correct, they nevertheless allow achieving a high performance. However, this performance cannot be guaranteed when relying on heuristics as heuristics do not consider all relevant information.

2.1.6. Human Erroneous Actions

While the creativity and flexibility of human operators is needed to control complex dynamic systems, human operators are also prone to errors. These errors may lead to accidents and may have catastrophic consequences. It is assumed that human erroneous actions are the main cause for accidents. Usually 60% to 90% of all accidents are attributed to human erroneous actions [Hol98, p. 2-3]. Furthermore, there was a tendency over the last decades to attribute an increasing proportion of accidents to human error [Hol98, HW05].

Due to the possible consequences, it is of utmost important to reduce or avoid human erroneous actions. As errors can have a variety of causes, the countermeasures are manifold and include organizational changes, specific training of operators and technical measures. However, developing countermeasures requires first to analyze errors and to understand the conditions and reasons which lead to or facilitate erroneous actions. There are two different approaches heading for this goal, the prediction of human erroneous actions to estimate the probability of accidents and the retrospective analysis of accidents.

The term “human error” commonly used in the literature is used with different meanings. It can denote the cause for an erroneous action, the action itself, or its consequences [Hol98]. In [Hol98] it is suggested to use “erroneous action” to refer to the action itself unambiguously. However, erroneous actions as well as outstanding performance are two sides of the same medal caused by human performance variability. Thus, the question how to differentiate between erroneous actions and normal performance actions has to be answered prior to a definition.

According to [Hol98] there are three aspects to be considered for a definition of human erroneous actions. Firstly, in order to be able to measure a deviation, a performance standard or criterion is necessary (also suggested by [RA03]). Secondly, there must be an event (or action) that results into this deviation. And thirdly, the acting person needs to have a choice to act in a way not considered as erroneous so that no deviation results. Accordingly, human erroneous action can be defined as actions with avoidable and undesirable consequences. Hence, the defined criterion as illustrated in Fig. 2.6 differentiates between acceptable performance and erroneous actions (not acceptable performance). Following this idea, decisions with consequences below the threshold value are classified as errors and it is assumed that they are caused by internal processing errors whereas decisions with consequences above the threshold value (only slightly deviating from the optimal solutions) are an inescapable implication of the working methods of human operators. However, choosing a criterion which does not reflect the optimal consequences seems arbitrary and choosing the optimal consequence as criteria results in many actions classified as "erroneous actions".

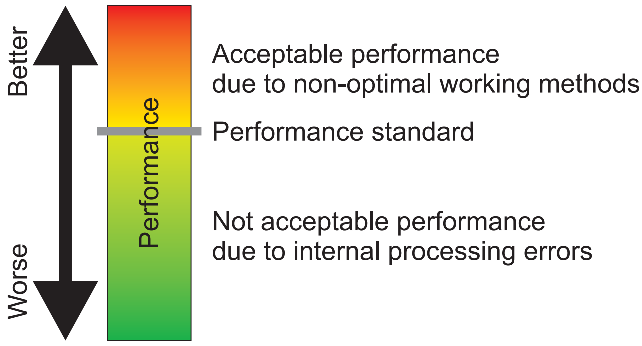


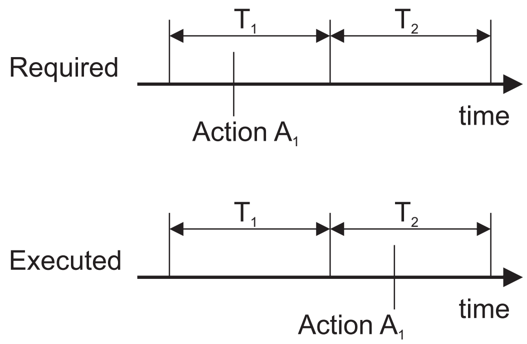
Figure 2.6.: Threshold to differentiate acceptable and not acceptable performance

Two classes of criteria can be applied. On the one hand, externalized verifiable models like system variables can be used as a standard. This class focuses on the consequences of errors. On the other hand, internalized verifiable frameworks can be applied. Here, the criteria are transient intentions and goals of the acting operator. This class concentrates on the action itself as it can classify actions as erroneous before their consequences are known [Hol98].

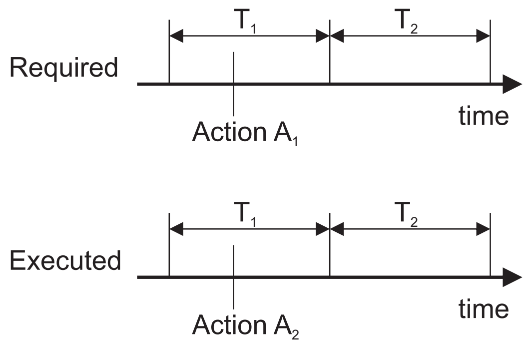
To analyze erroneous actions, several classification schemes were proposed. As these schemes were developed in different field, they are either developed with the aim to explain observed errors or to predict the probability of errors.

A first step in the classification of errors is the differentiation between correctly performed actions, the failure to perform required actions (omission), and the performance of undesired actions (commission). However, while an omission describes a more specific error (not to do something), the term commission subsumes different error modes, as there are many ways to do something wrong. However, to consider only the timing of an action already reveals four different forms of omissions. First, actions can be missing completely, they further can be performed to early or to late, and finally, they can be replaced by another action [Hol98]. It is necessary to define the duration of a the complete event sequence to differentiate between missing and (infinitesimal) delayed actions [Hol98].

To consider only these four error modes already demonstrates that it is often not possible to define an error unambiguously as omission or commission. An example is illustrated on the left of Fig. 2.7. An action which should be performed in the time interval T_1, but actually is performed during the time interval T_2, is an omission during the time interval T_1. However, it is also a commission in the time interval T_2. A similar problem is shown on the right side of Fig. 2.7. The action A_2 is executed while another action A_1 was required in the same time interval. First, skipping A_1 is an omission, but the execution of A_2 is a commission. This shows that every commission is also an omission as the correct action is missing.



(a) Example 1



(b) Example 2

Figure 2.7.: Erroneous actions to illustrate the difficulty of differentiating omissions and commissions

Not only due to the difficulty of assigning an error to the omission or commission class but in particular because the various error modes within a class can have different consequences (e.g. the error modes “to early” or “to late” within the class “commission”), the differentiation between omission and commission is not sufficient but the specific error mode should be indicated, both for the analysis of accidents and for the prediction of errors [Hol00].

The above definition of missing, delayed, or premature actions uses clearly an externalized criterion. Other classification schemes focus on the internal information processing and classify errors by internalized criteria. One example of this is the **Generic Error Modeling System (GEMS)** proposed by Reason [Rea90], which is based on the SLM (see section 2.1.1). The GEMS further builds on the assumption that separate levels of cognitive functions exist as described in the SRK (also see section 2.1.1). Reason defined four basic types of errors in the GEMS. Errors on the skill-based level describe actions which differ from the intention. It can be distinguished between slips, which are execution errors, and lapses which result from wrong recalls of stored information. Errors are called mistakes, if the action and intention are identical but the executed plan is

Table 2.2.: Selected differences between skill-based, rule-based, and knowledge-based errors [Rea90]

	Skill-based slips and lapses	**Rule-based mistakes**	**Knowledge-based mistakes**
Activation	Routine	Problem-solving	
Error Prediction	Mostly predictable error established but wrong (Scheme)	(Rule)	Variable
Error rate	Many possibilities / only few errors		Few errors/ high error rate
Error detection	Quick and efficient detection	Hard, often help from outside needed	

not appropriate. This can happen on the rule- and knowledge-based level. A mistake of applying a rule is called rule-based mistake, whereas a mistake on the knowledge-based level is called a knowledge-based mistake.

Some properties of errors as given by Reason [Rea90] are summarized in Table 2.2. One aspect is the predictability of errors. Errors are predictable, if they are caused by false rules, which corresponds to rule-based errors, and false schemes, which are applied on the skill-based level. Errors on the knowledge-based level cannot be predicted accurately, as errors on this level are a result of complex interactions between knowledge and the mental model.

Also the frequency of errors varies with the level. On the skill- and rule-based level, there are many possibilities for errors, but the relative frequency is low. Since only in a few cases the knowledge-based level is used, only a few possibilities for errors exist on this level but the relative frequency is higher than on the other levels. Additionally, execution errors are usually quickly detected by operators and can be solved, while errors, which are based on a false knowledge and thus appear on the rule- and knowledge-based level, are more difficult to detect by the operator and external help is needed to discover such errors.

Drawing a conclusion from these characteristics, it can be stated that support for error detection would be helpful especially on the rule- and knowledge-based level. The reason is that errors on these levels are difficult to detect by the operators themselves and there is a high error-rate on the knowledge level [Rea90]. On the other hand, the source of errors on the rule-based level are false rules. This allows to predict errors (when the false rules are known) and correct behavior (if the correct rules are used). As the GEMS, like other classifications based on models of human information processing, does not consider contextual factors, it is criticizes due to its lack of ability to explain the exact reason for errors [Hol98].

An—in many respects different—classification scheme of errors was developed by Dörner [Dör89] (also see Schaub [Sch06]). This classification scheme was developed in the field of **Complex Problem Solving (CPS)** (in German: “Komplexes Problemlösen”) and describes the errors of participants dealing with complex dynamic problems in microworlds. This classification scheme contains only errors which can be assigned to the knowledge-based level, which is obviously a consequence of the chosen approach to observe untrained participants trying to solve complex problems. Although the research was not concerned with HMI, the developed classification can be transferred into the field of HMI under the assumption that humans are prone to the same errors solving problems in an artificial microworld and in real world tasks.

As explained in the last section (see section 2.1.5), operators do not optimize but select actions based on rules or heuristics. The SLM and the GEMS illustrate the importance of rules during the interaction with complex dynamic systems. Therefore, human erroneous actions are not caused by overstraining the human information capacity but by selecting inadequate rules. To explain and prevent erroneous actions it is thus important to understand “how peoples’ assessments and actions made sense at the time” [Dek06, p. xi]. As erroneous actions result from a “mismatch between cognition and context” [Hol98, p. 81], they must be interpreted relative to the characteristics of the situation when the error happened. In other words, human erroneous actions result from differences between the reality and the human operators’ mental model of the reality.

2.2. The Air Traffic Control Domain

This section gives a short introduction into the domain of ATC as this task, strictly speaking the approach task, is the example application of this thesis. The domain of ATC is an example for HMSs. The **Air Traffic Control Officer (ATCO)** is the human operator who interacts with a complex dynamic system. Since the controlled system comprises both aircraft and pilots, it is more than a purely technical system.

One challenge in the ATC domain is the expected growth of traffic. Despite the European economic crises, it is assumed that in the years 2013 and 2014 the amount of traffic increases by 1.2% per year [Eur12]. These growth rates are relatively low compared to the rates expected for longer periods. For instance an annual growth of about 3% is expected for the years 2015-1018 [Eur12] and by 2035, about 1.5 times the traffic of 2012 is predicted [Eur13].

Challenges in ATC arise, not only because of the expected growth, but also due to increasing demands. At first, there is the need for environmental sustainability. Therefore, aims were defined to reduce carbon dioxide (CO_2) and noise emission. Moreover, the cost-efficiency of air traffic has to be improved. At the same time, safety must be kept at a high constant level. Despite the increased traffic, the number of accidents should not increase. To scope with this challenges, new procedures and additional support from technical systems will be introduced, which will change the task of air traffic controllers. Improvements in these key performance areas, namely safety, capacity, environment, and cost efficiency, are the four high-level goals of Single European Sky [SES2012].

This section is first intended to reflect briefly the task and responsibilities of ATCOs. As the example application in this thesis is a new approach concept including the merging of two arrival streams supported by an **Arrival Manager (AMAN)**, existing AMANs and the new concept will be presented.

2.2.1. Organization of Air Traffic Control

The most important objectives of ATC are to prevent collisions between aircraft, to support an orderly flow of air traffic, and to provide information useful for safe and efficient flights [ICA13]. Consequently, a minimum separation between the aircraft has to be ensured and the pilots have to be enabled to use efficient flight trajectories and to reduce fuel consumption.

2.2.2. Areas of Responsibility

The air traffic control task is divided into the three areas

- **Tower Control (TWR)** (also aerodrome control),
- **Approach Control (APP)** (also terminal control), and
- Area Control (ACC for Area Control Center, also en-route control) [Bau07,KM12].

Each of these areas will be introduced shortly.

Tower Control (TWR)

The air traffic at an airport—on the ground and in the immediate surroundings—is called Tower Control at it is controlled from the tower. This is the only area in which ATCOs have visual contact with the aircraft. The outside view is the primary source of information for ATCOs.

In Germany, the national air navigation service provider **DFS Deutsche Flugsicherung GmbH (DFS)** is responsible for the runways. The airport is responsible for the traffic at the other traffic areas, but it can delegate the responsibility to an air navigation service provider. As a consequence, the actual division of responsibility varies between airports and the ATCOs can be responsible for the aircraft on the runways, taxiways, aprons, and stands and in the airport's control zone, which is the airspace around the airport.

At busy airports, the task is divided into subtasks. The responsibility for the aircraft, vehicles, and persons on the airport's apron is given to the Ground Control. The tower position is still responsible for the runways and the control zone and gives clearances for take-offs and landings.

Approach Control (APP)

The **Terminal Maneuvering Area (TMA)** (or Terminal Control Area in North America) is about 30 km – 50 km [KM12] around major airports. The ATCOs in the terminal control are working in approach control rooms at radar screens, which indicate the actual position and status of aircraft in (or near) this area. They are communicating with the aircraft via radio. An APP can either conduct the air traffic control for one airport solely or for several airports located close together. It can be located in a control center, as the ATCOs in APP do not require an outside view like in the tower.

The APP controls the approaching and departing aircraft, and fly-overs, which are just crossing the TMA. The task of ATCOs is also to generate the landing sequence while ensuring the minimum separation. Coming from the en-route sector, aircraft often use **Standard Terminal Arrival Routes (STARs)** from the cruise altitude to the initial approach fix. At this point the approach start, which is divided between initial, intermediate, and final approach [KM12]. For the approach, the **Instrument Landing System (ILS)** can be used for the correct alignment and descent. The ATCO can use radar vectors (a heading for the aircraft) to line up the aircraft on the ILS. Similar to the STARs, **Standard Instrument Departure Routes (SIDs)** are used for departing aircraft to guide the aircraft from the runway out of the TMA. The approach control positions can be divided into a pick-up and a feeder sector. The pick-up sector passes the arriving aircraft on to the feeder sector.

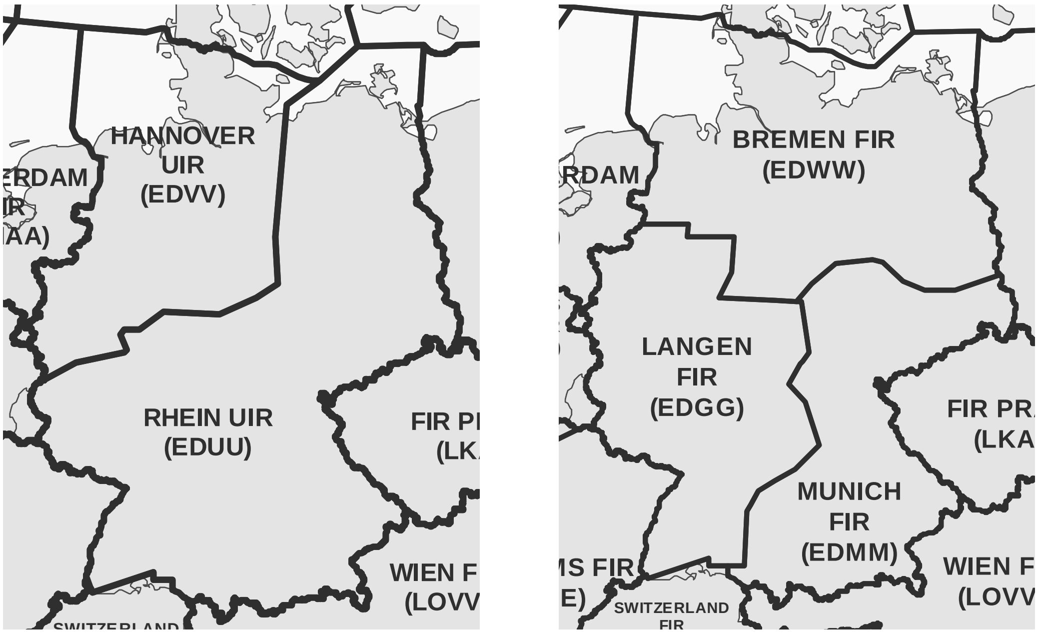
Area Control Center (ACC)

The controlled airspace is divided into different **Flight Information Region (FIR)**, whose borders are often identical to national borders. They have been split up vertically to create **Upper Flight Information Regions (UIR)**. The FIR and UIR in Germany are shown in Fig. 2.8. These Regions are controlled from **Area Control Center (ACC)** or **Upper Area Control Center (UAC)**. The German airspace is divided between three ACCs (Bremen, Langen, and München) and two UACs (Maastricht and Karlsruhe).

Each region is divided into different sectors. A sector is "the smallest area of airspace under specific control" [Bau07]. Normally, each sector is controlled by two ATCOs, one **Executive Controller (EC)** (sometimes called Tactical Controller) and one **Planning Controller (PC)** sitting next to each other. Only under special conditions ATCOs work alone in a sector. To scope with changing amounts of traffic, sectors can be combined or split up to regulate the workload of the controllers. Besides the separation in their own sectors, ATCOs in ACCs have also to ensure the separation of aircraft heading to the same airports when entering APP.

2.2.3. Task of Air Traffic Controllers

On the one hand, ATCOs are bound to stringent rules and procedures. On the other hand, they need to be flexible to solve unknown situations, which they are often con-



(a) UIRs [Eur14U] **(b)** FIRs [Eur14]

Figure 2.8.: Flight Information Regions (FIRs) and Upper Flight Information Regions (UIRs) in Germany

fronted with [Bau07]. Hence, they are working both on the rule-based level but also on the knowledge-based level (see section 2.1.1).

A flight progress strip is given to ATCOs, which depicts the aircraft's identification, waypoints and flight level, some minutes before it enters the sector and appears on the radar screen. The flight progress strip is either paper-based or—to an increasing degree—electronic flight strips are used. When an aircraft appears on the radar screen, the data of the flight progress strip has to be compared to the arriving aircraft. Then the consequences especially for separations have to be assessed. On the radar screen different kinds of information for aircraft can be presented including

- call sign,
- flight level (altitude),
- attitude (change of altitude),
- speed,
- heading, etc.

The two controllers of a sector have different tasks. The task of the PC is to communicate with adjacent sectors, to manage the flight progress strips and to identify trajectories with the least risk of conflicts. The PC is communicating with the pilots.

He has to detect potential conflicts and issues instructions accordingly. The communication between the ATCO and aircraft is mainly voice communication over radio but also datalink is used increasingly. The ATCOs give the aircraft permissions to proceed, called clearances. These clearances can be limited to specific conditions.

A task analysis [KvD99] revealed ten different processes in the en-route controllers' task, one control process, five task processes, which include one or more of four sub-processes. Further work found out that no major differences between the processes of en-route arrival/departure and aerodrome position exist [DKv00]. On overview of the processes is given in Fig. 2.9.

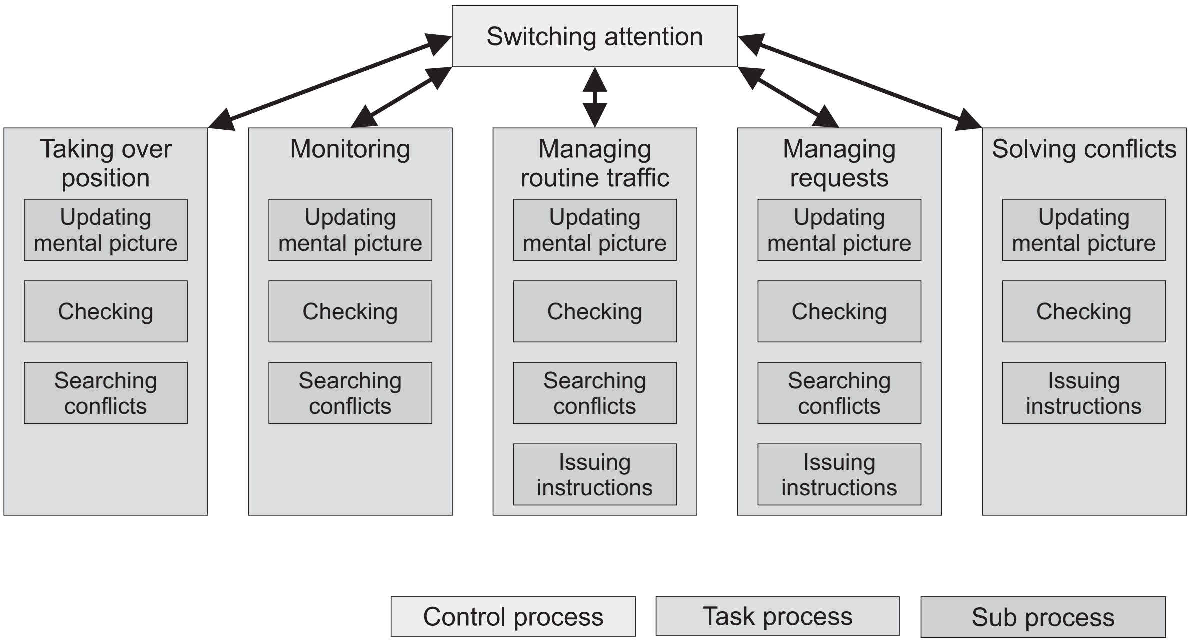


Figure 2.9.: Processes of Air Traffic Controllers [KvD99]

The identified control process is called "switching attention", the four identified sub-processes are called

1. updating mental picture and maintaining situational awareness,
2. checking,
3. searching conflicts, and
4. issuing instructions.

The first three of these sub-processes describe the perception, interpretation, and prediction of information from the environment. This highlights, that ATCOs are mainly concerned with managing information.

The five task processes are called

1. taking over position (includes sub-processes 1, 2, 3),

2. monitoring (includes sub-processes 1, 2, 3),
3. managing routine traffic (includes sub-processes 1, 2, 3, 4),
4. managing requests and assisting pilots (includes sub-processes 1, 2, 3, 4), and
5. solving conflicts (includes sub-processes 1, 2, 4).

After taking over position, the ATCO monitors the traffic. This is the core process and the other processes are activated depending on the monitored situation. The task process 4 is activated externally by requests from pilots, whereas the task process 5 is activated internally after a conflict was detected.

It can be concluded that the task of controllers is mainly cognitive. Their actions are guided by their mental picture of the current traffic situation. Due to the strict rules and procedures, they will work on the rule-based level of the SRK most of the time.

2.2.4. Arrival Manager

Since managing the stream of aircraft arriving at a high-traffic airport is one of the most difficult tasks in ATC, AMANs have been developed to support ATCOs since the early 1980's [Völ90]. AMAN have to meet different - sometimes conflicting - aims, namely to increase the safety, to enhance the efficiency by reducing arrival queuing, to maintain an optimal throughput at the runway, and to decrease the impact on the environment (fuel-burn and noise) [HC10a].

An AMAN has to support the controller in performing his tasks. These tasks of the ATCO managing the arriving traffic, which are illustrated in Fig. 2.10, are

- to build a sequence of the arriving aircraft (sequencing),
- to assign an expected time over waypoints for each aircraft, especially a **Target-Time of Arrival (TTA)** for the runway threshold (metering),
- to plan trajectories, which meet the previously defined times (trajectory generation), and
- to generate clearances from the trajectories and guide the aircraft appropriately (clearance generation) [Obe06].

The development of AMANs began with COMPAS by the **German Aerospace Center (Deutsches Zentrum für Luft- und Raumfahrt) (DLR)**. It used flight plan data and early radar information to build an arriving sequence und calculate TTAs for the runway. The COMPAS system was brought into operation at Frankfurt Airport [Völ90]. The successor of COMPAS is the 4D-Planner which development started in 1996 [ADG+96] in a close cooperation of DLR and DFS. It additionally uses actual radar data for an improved sequence planning and is able to adjust the sequence if the controller's actions deviate from the provided plan. The 4D-Planner further developed by DFS was brought into operation in Frankfurt airport in 2003. Besides Frankfurt, the

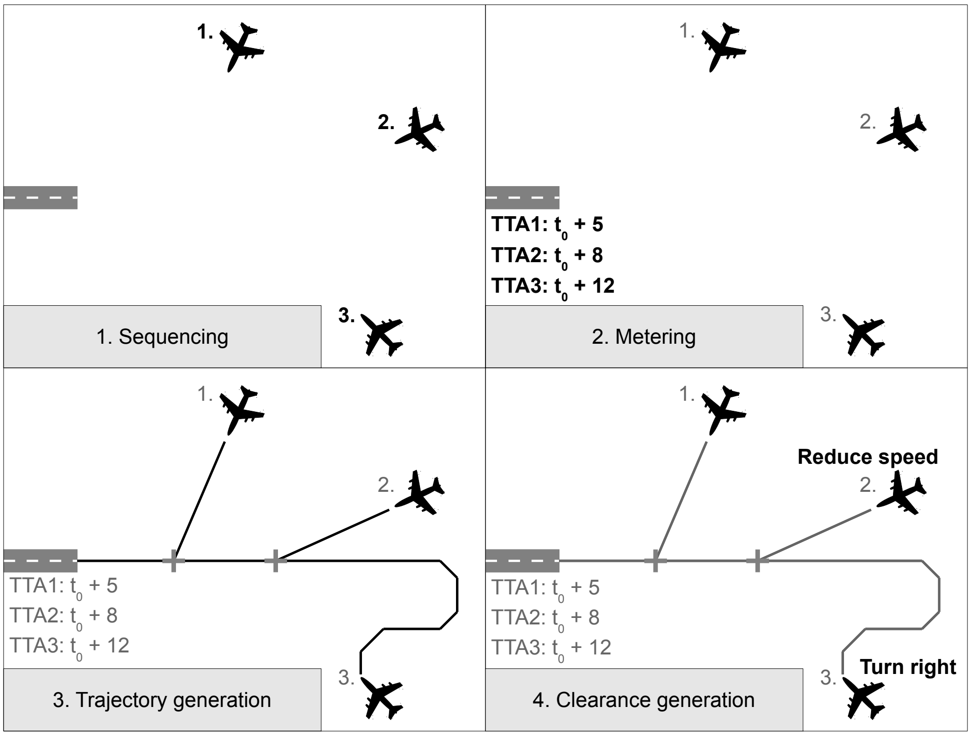


Figure 2.10.: Task of air traffic controllers to establish arrival sequences

4D-Planner is also used at Munich airport [HC10b]. These airports are the only airports in Germany using AMANs. Different commercial AMAN are available and in use today. On overview of the AMAN operating in Europe can be found in [HC10b].

The further development of the 4D-Planner within DLR is named 4D-CARMA. This system is able to calculate trajectories and to generate guidance instructions based on these trajectories. Thus, this AMAN is able to support the controller in the four tasks mentioned above.

2.2.5. Late Merging of Arrival Streams

One of the weaknesses of current approach operations, as analyzed in the **Future Air Ground Integration (FAGI)** Project, are inefficient trajectories [WO10]. They result from merging arriving aircraft early at distances of about 30 NM away from the landing threshold on a limited number of arrival routes. Consequently, they need to use similar speed profiles. This inevitably leads to inefficiency and unnecessary fuel consumption as each type of aircraft has its own optimal speed profile. The late merging of aircraft only 6 NM – 10 NM away from the runway threshold was proposed as a solution (s. Fig 2.11), which allows each aircraft to fly an used-preferred trajectory (preferred speeds and descent rates) [WO10]. However, this procedure increases the complexity and puts a higher demand on ATCOs so that further support is necessary.

The key idea of the concept developed in FAGI is that aircraft use separate routes as long as possible and are merged at a **Late-Merging-Point (LMP)**. Aircraft are

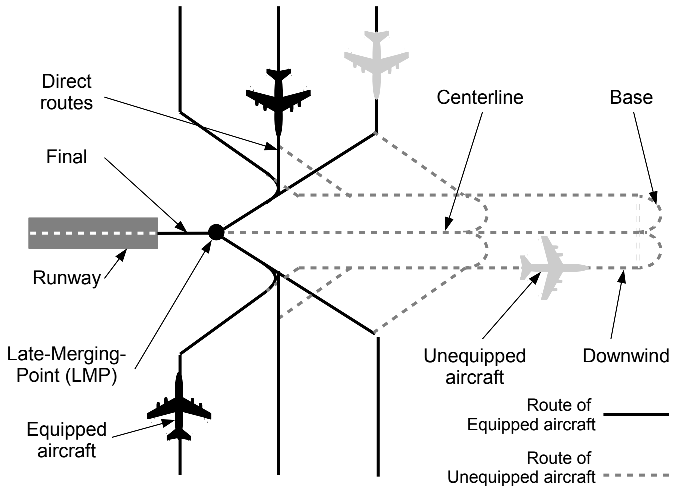


Figure 2.11.: Late-Merging route structure, with direct routes (black lines) and path stretching area (dashed gray lines)

divided into two groups depending on the capability of their **Flight Management System (FMS)**. Aircraft equipped with a 4D-FMS are able to fly a specified route within a high time-precision and are allowed following a direct approach (black aircraft and routes in Fig. 2.11). They negotiate a fixed time with the AMAN and can fly their preferred profile as long as they can meet the time restrictions. The equipped aircraft fly separated routes and are merged at the LMP just 6 NM away from the landing threshold. As these aircraft can fly their preferred profile, they can fly a **Continuous Descent Approach (CDA)**. In a CDA, level segments are avoided, so that aircraft are flying with higher altitudes compared to conventional approaches. Thereby the noise immissions at the ground are reduced. The use of these 4D-FMS capabilities is a key potential for a further increase of efficiency. While CDAs are not implemented in high traffics situations today due to their negative impact on capacity [Erk99] the FAGI-approach aims to enable aircraft equipped with a 4D-FMS to fly CDAs without reducing the airports capacity. As a consequence, the environmental impact in terms of fuel consumption and noise immissions can be reduced.

Aircraft which are not equipped with a 4D-FMS have to be guided in the conventional way manually from the arrival routes over a path-stretching area (called Trombone, gray aircraft and dashed gray routes in Fig. 2.11). The path stretching area consists of a downwind leg, a base leg and an extended centerline. By changing the trombone length, the arrival time of aircraft can be adjusted, so that unequipped aircraft can be merged into the established stream and fill the gaps between equipped aircraft.

A preliminary study showed that the merging of both arrival streams is the most challenging task [WO08]. For this reason, two kinds of visual aids to assist the controller during the late-merging-procedure have been developed. One of this visual aids is ghosting [OWR09]. For each equipped aircraft, a synthetic copy (ghost) is projected onto the runway centerline extension. When the unequipped aircraft have to be lined up, the ghosts indicate positions reserved for equipped aircraft. The ghosts move along the centerline and meet with the symbol of the corresponding real aircraft at the late merging point. The other visual aid developed is Targeting [OWM10]. This visual aid projects a copy of each unequipped aircraft on the centerline. These targets indicate the goal-position of the unequipped aircraft on the centerline, as calculated by the an AMAN. To investigate the effect of the visual aids on the controller under standardized conditions, the simulation environment MAGIE was developed. It applies a simplified route structure proposed for the late-merging operations. This simulation environment is applied as the example application in this thesis.

The 4D-CARMA AMAN was extended to be able to handle the developed concept of integrating aircraft into a stream of 4D equipped aircraft. This capability was validated during the Project FAGI [OGS08, WO10]. The extension of the concepts developed in the FAGI project to a real-world setting, including bad weather conditions and flexible routes, are part of the ongoing project FlexiGuide [TCE+11].

2.3. Conventional Human Performance Measurement

In this section, the term "human performance" is defined and an overview of the current state of the art in measuring human performance is given. At first, the measurement of human performance in HMSs is analyzed. Since the method developed in this thesis is demonstrated with an ATC task, the measurement of human performance in ATC will be discussed in detail. As this thesis uses a microworld as simulation environment, performance measurements in other microworlds, which are common in Cognitive Science, are analyzed. Based on the examination of the various areas where human performance is measured, implications for a new measure of performance are drawn.

2.3.1. Human Performance Measurement in Human-Machine Interaction

Human erroneous actions may lead to serious consequences as mentioned in section 2.1.6. For this reason, the performance of human operators of complex technical systems is of utmost importance. Even in the absence of hazardous results, human actions are a determining factor of system efficiency. Thus, a poor performance of the human operator will at least lead to monetary consequences. Due to these consequences, human performance is a main research issue.

The measurement of human performance is of particular importance during training to evaluate its success and to decide about the amount of training necessary. Furthermore performance measurement is important for the evaluation of new concepts and technology. If new concepts or technology are introduced, this will change the working methods and the performance of the human operators. Consequently, it is essential to demonstrate that new concepts have a positive impact (or at least no negative impact) on the human performance or that a decrease in performance can be avoided by additional assistance [MMFP00, RA03].

The human performance can be defined as the accomplishment of the task by a human operator [Gaw00, MMFP00]. Although this definition sounds simple, there are several problems in the measurement of human performance. Firstly, the accomplishment of the task is the fulfillment of its objectives. However, actually many HMSs lack of clear defined objectives (see section 2.1.5). An example for such an ambiguous objective is the objective to guide a process efficiently. Without giving objectives with exact numbers, it is not possible to make a statement about its fulfillment. Consequently, precisely defined objectives are a prerequisite of human performance measurement.

Validity and Reliability

Valid and reliable measurement of human performance is essential. Validity corresponds to the quality of the results. According to [TD08], four different kinds of validity can be distinguished as illustrated in Fig. 2.12. First it has to be distinguished between cause (e.g. task difficulty) and effect (e.g. task performance) and between the construct (e.g. human task performance) and the measure (e.g. amount of produced units). The measure is the operationalization of the abstract construct. The conclusion validity describes the examined relationship between the measured operationalization of a cause and an effect. Thus, only the measurements are compared. The internal validity is given, if this relationship is a causal one. The construct validity describes how well the operationalization complies with the constructs. In other words, the construct validity is given if the used measurement is appropriate to measure the cause respectively the effect. Finally, the external validity is given, if the results can be generalized.

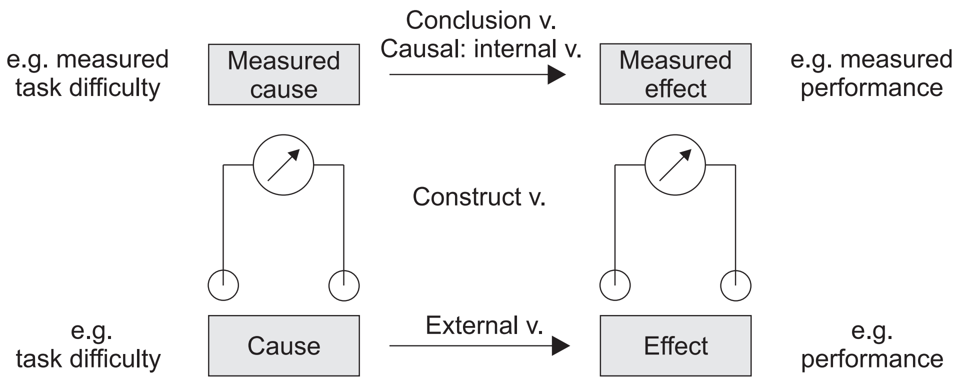
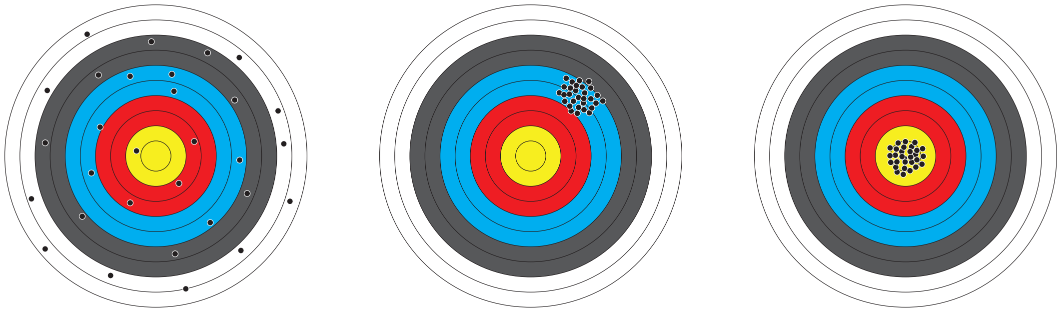


Figure 2.12.: Four types of validity

Another important requirement for a measure is its reliability. Reliability is defined as the variance of the true value of a construct divided by the variance of the measured value. Thus, the reliability is related to stochastic measurement errors. If the measurement error is low, the reliability will tend to 1, whereas a high measurement error will lead to reliability near zero. Consequently, the repetition of measurements with a low reliability will probably produce different results.

The relationship between validity and reliability are illustrated in figure 2.13. The center of the target is the concept that is tried to be measured. The dots are the individual measurements. In the left figure a measurement with a high validity (the mean of the dots is close to the center) but a low reliability is shown. In the figure in the middle a measurement with a low validity but a high reliability (the dots are close together) is shown. A measurement with a high reliability and a high validity is illustrated in the right figure. A performance measure needs—like every other measure—a high reliability and validity.



(a) Valid but not reliable **(b)** Reliable but not valid **(c)** Valid and reliable

Figure 2.13.: Validity and reliability

Types of Performance Measures

Human performance measures can be distinguished according to different characteristics (e.g. [RA03]). The different characteristics of performance measures or summarized in Table 2.3. At first there is the difference between subjective and objective measures. Examples for subjective measures are questionnaires or interviews, which can be used for a self-assessment by the human operators. Another example are expert observers, which may be used to rate the performance and who may be able to identify problems that have remained hidden for human operators. These techniques can also be used to derive objectives measures if a criterion is used as a standard. For example, the rating of an action as excellent is subjective, to confirm that it is adherent with the standards is an objective measure [RA03].

Table 2.3.: Summary of different types of performance measures

Characteristic	Types
Perspective	Subjective (e.g. excellent) vs. Objective (e.g. adherent with standards)
Access	Primary (measured) vs. Secondary (calculated, e.g. max. or average)
Measurability	Direct (measured) vs. Indirect (inferred from direct measures)
Object	Human-Machine performance (e.g. throughput) vs. Human performance
Focus	Result measure (considers exclusively outcome) vs. Process measure (considers additionally intermediate states)

The measurements of human performance can either be direct or indirect [RA03]. If performance is measured indirectly, it must be inferred from direct measures. For example, the performance can be measured directly in detection task (rate of correct detections). Indirect measures have to be used, if the correct decision is not known in each situation. In these cases, variables of the systems are measured, which describe the consequences of decisions. To infer the performance from these variables, they have to reflect the accomplishment of the task. An example for such a variable is the systems' consumption of energy which can be used as an indirect measure of efficiency reached by human operators' decisions. To reach a high construct validity when measuring human performance indirectly, it is crucial to choose a variable, which is appropriate for describing performance (the measured construct). In the case of human performance this variable must reflect the objectives of the tasks.

Another issue is the distinction between primary and secondary measures. Secondary measures are derived from primary measures. The difference to indirect measures is that secondary measures can be derived exactly whereas indirect measures are inferred. Examples for secondary measures are the calculation of maximum or average values.

A difficulty when measuring human performance is to differentiate between the performance of the human operator and the performance of the system respective the performance of the team, if humans are working cooperatively [RA03]. Typical examples of system measures are capacity, throughput, or delays. These measures are greatly influenced by human performance but include also the performance of the technical parts of the HMS. The differentiation between system performance and human performance is of particular importance as the increasing computer-based assistance takes over subtasks from the human operator and it is consequently difficult to allocate the reached performance to the controlled system or the human operator. The confusion between system and human measures leads to problems especially when comparing variants of a system, which differ in the LoA. For example, using the energy consumption of a HMS as an indirect measure allows performance to be compared between human operators coping with the identical problem but it does not allow performance to be compared across different problems (requiring different amounts of energy to be solved) or LoAs. If a required standard of performance or a frame of reference can be defined for each problem or scenario, a comparison between the actual performance and the standard becomes possible [YLVC02]. The resulting deviation of the performance from the standard can finally be used to compare different scenarios.

Another distinction is the difference between process and product measures. Whereas product measures only concentrate on the result or the output, process measure consider how the result was reached. If the performance is measured indirectly, for example the energy consumption, the measure describes the summarized performance of a whole simulation and the performance cannot be broken down to the various actions. This shows that some indirect measures cannot be used as process measures. The difference between process and product measures was already discussed in section 1.1 and the drawbacks of results measures are the main motivation for the development of a process measure of human performance in this thesis.

Signal Detection Theory

If the objective of the process is clearly defined and it is possible to define the right decision in every situation, the performance can be measured by determining the correctness for each action. An example for such a task is the detection of signals or the identification of problems. In this case, the contribution of each decision to the overall performance can be measured by applying the **Signal Detection Theory (SDT)**. The SDT is introduced briefly in the following. For in extensive description see [GS89,MC05].

The SDT assumes two states of the environment (signal/problem present or not) and two human responses (detect signal/problem or not). This results into a 2 x 2 matrix (see Table 2.14). Each decision is classified into one of the four categories of this matrix. The hit rate, which is the amount of hits divided by the total of hits and misses can be calculated out of this matrix. Additionally, the false alarm rate can be calculated by dividing the amount of false alarms by the sum of false alarms and correct rejections.

The SDT can reveal the individual decision criterion and the individual detection performance of human operators. The individual detection performance corresponds to the ability to differentiate between the presence and absence of a signal (or problem). The individual decision criterion is a measure for the human operators' conservativeness. In other word, it can distinguish between human operators preferring a high hit rate at a cost of a high false alarm rate from operators preferring both rates to be lower. The individual detection performance can be calculated using the sensitivity by $d' = z$(hit rate) $- z$(false alarm rate) with z as the inverse of the cumulative Gaussian distribution.

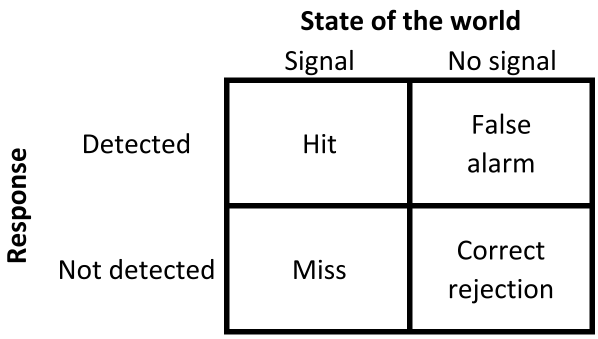


Figure 2.14.: Four categories of performance according to the signal detection theory, depending on the state of the world and the operators response

The decision criterion and the individual detection performance are both represented by the **Receiver Operator Characteristic (ROC)** which is depicted as the ROC-curve. The individual performance results in one point in the ROC plot. In Fig. 2.15 an example of four ROC curves is shown. If the human operator is only guessing, the hit rate and the false alarm rate are equal. This corresponds to a diagonal through the origin in the ROC diagram. A higher individual detection performance will result into a higher hit rate and a lower false alarm rate. The performance corresponds to the distance between the individual point (defined by the measured rates) and the diagonal. The individual decision criterion corresponds to the distance between this point and the axis of ordinate. A more conservative decision criterion will reduce the hit rate and the false alarm rate as well, but the ability to differentiate between the presence and absence of a signal is the same and thus the performance is the same. If the individual decision criterion is modified while the detection performance is constant, the individual point moves along its ROC Curve.

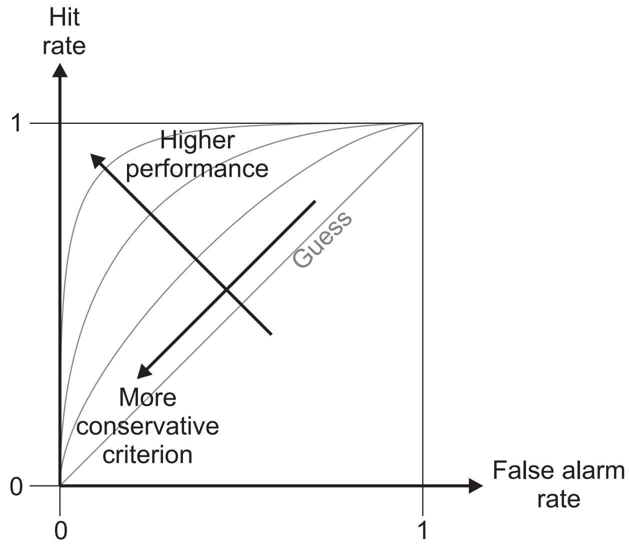


Figure 2.15.: Receiver-Operator-Characteristic (ROC) curves

Time Window

To account for the dynamics of complex systems a time window as a measurement construct was proposed [Rot01, Rot11]. A time window is a relationship between a required situation and a time interval in which this situation is reachable by an action. During a simulation, a time windows is first inactive. It is designated as open, if the situation is currently required and an action to reach the required situation is available at the current point in time. The time window is closed, if the situation is no longer required or is not reachable anymore. According to the time window method, an action of the human operator is considered as a two-tuple consistent of a detectable act and a specific point in time, at which it is performed.

To evaluate human performance two Boolean indicator functions are defined. Therefore m actions are denoted as b_j for $j = 1$ to m and n time windows are denoted as w_i for $i = 1$ to n. The first functions

$$I_w^1(\mathbf{b}) = \begin{cases} 1, & \text{if } \mathbf{b} \text{ meets situation specified in } \mathbf{w} \\ 0, & \text{if } \mathbf{b} \text{ does not meet situation} \end{cases} \tag{2.2}$$

evaluates if an action leads to a required situation, for which a time window is defined.

The second function

$$I_w^2(\mathbf{b}) = \begin{cases} 1, & \text{if } \mathbf{b} \text{ is relevant towards } \mathbf{w} \\ 0, & \text{if } \mathbf{b} \text{ is not relevant towards} \end{cases} \tag{2.3}$$

is used to determine if an action is relevant towards a time window. This function evaluates to 1, if the action does not lead to a required situation but would result in a required situation at a different point in time.

Combining the actions of the human operator and the available time windows, seven different categories result (s. Fig 2.16). The two main dimensions are first the existence of a time window (independent of its state) and a corresponding required situation (state of the environment) and second the response of the operator. This results into a 2 x 2 matrix similar to the SDT. If an action is not relevant to any window, the action is classified as false alarm. In contrast, if no action is relevant to a window, the window is classified as missed. If an action b meets the situation specified in the time windows w, the actions is either on-time, if it is required when the action is performed (time window open), or early or late (time window inactive or closed). An action is incorrect, if it does not lead to the required situation, but would have if it had been executed at another point in time.

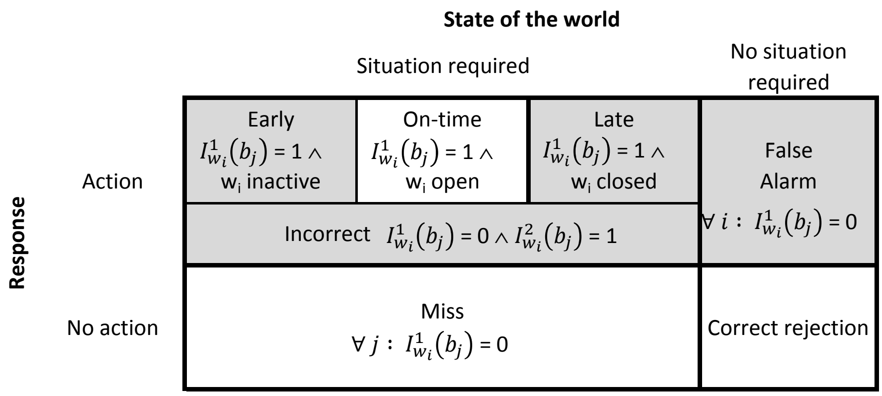


Figure 2.16.: Seven categories of actions defined by the time window method, depending on the state of the world and the operators response [RN11]

Compared to the SDT, the time window method considers the dynamics of a task. The application of time windows gives more information about the operator's actions and thus allows for a more detailed classification. If only on-time actions are considered as hits and all other actions (gray background in Fig. 2.16) are considered as false alarms, the time windows method leads to the same classification as the SDT.

2.3.2. Human Performance Measurement in Air Traffic Control

This section focuses on the measurement of human performance in **Air Traffic Control (ATC)**. As already described in the previous section, the variables used to measure human performance have to depend strongly on the particular application. This section will consider the application of the above described methods to measure human performance in the context of ATC. As the example task in this thesis is an air traffic control approach task supported by an AMAN, this section concentrates particularly on the measures of human performance used for the validation of AMANs.

In the field of ATC simulations are used to evaluate the performance in a controlled environment. ATCOs are interacting with a replica or a prototype of a real system during a simulation. Simulations are done either to assess the training performance or the feasibility of new procedures. Each condition for a simulation run is called a scenario. Simulations allow analyzing scenarios which are too risky to test under operational conditions. This includes scenarios with a very high task demand as well as scenarios which include new technology. A disadvantage of simulations is that after the first action the situations between different simulation runs of the same scenario will not be the same again. Even slight differences in the timing of an action may lead to complete different results and require different further actions [MS05]. As a consequence, the simulation is hard to control and the comparison of different simulation runs is complicated.

The measures used for human performance must reflect the fulfillment of the operators' task. The task of ATCOs is preventing collisions between aircraft and maintaining the safe, orderly and expeditious flow of air traffic [ICA13]. (For a detailed description of the ATCOs subtasks see section 2.2.) As the task of ATCOs is mainly cognitive, most of the activities are not directly observable. That complicates the measurement of performance of ATCOs additionally. As a result many aspects of the performance have to be inferred [HGS99].

Objective Performance Measures

Performance can either be quantified by a subjective opinion or by objective observable variables. An example for subjective measure in ATC are **Subject Matter Experts (SMEs)**, which are often used as observers to rate the human performance. Thereby some instruments like rating scales can be used. The first rating scales in use were vague, for example ranging from poor to excellent performance. One improvement was to introduce **Behaviorally Anchored Rating Scales (BARSs)**, which tie points of the rating scale to observable behavior. These BARSs are completed by the SMEs and are thus depended on their subjective opinion. If SMEs have to go through a checklist to record observed events, the results can be treated as objective because the occurrence of an event should not be arguable if there is a clear definition.. However, the use of SMEs is limited by their availability and the monetary resources to pay them.

Besides from BARSs, objective measures can additionally be derived from recordings of system data. One example is the identification of **Operational Errors (OEs)**. The problem of using OEs as a measure of human performance is that they are very rare. They can be provoked in simulations but only by putting the ATCO under a very high and unrealistic task load. Also this measure lacks to detect aberrations from the normal behavior of ATCOs which lead to near misses of OEs [MS05]. Other examples are System Effectiveness Measures [BDHK83] or **Performance and Objective Workload Evaluation (POWER)** [Ran04]. These collections of possible measures include system variables or measures which are derived from system variables. Examples are the number of controlled aircraft, the control duration, the amount of pairs of aircraft in conflict, etc. These collections were developed to provide objective measure for ATCO' taskload and performance and are easy to obtain.

However, there are some problems associated with these objective measures. First of all, they do not measure the criterions of interest directly so that secondary measures are necessary [RA03] and their interpretation is less clear [MMFP00]. As a consequence, the main problem is not the availability of the data but the construction of valid and reliable measures. Exacerbating the problem is the fact that no criteria are defined for the POWER measures and a lot of measures are used which meaning is not clear [Ran04]. Also it is often not possible to define the difference between system and individual performance [MS05].

Objective measures of performance measures used in ATC were collected by Hadley et al [HGS99] and Rantanen et al [RV07] and organized in databases. Hadley et al listed more the 170 measures and assigned them to different categories. Further, they defined which kind of effect on the system is measured (safety, capacity, efficiency) and in which environment the measure can be applied (en route, tracon (tma), tower, oceanic). They furthermore separated the task of ATCOs in steps (situation assessment, planning & decision making, and implementation) and defined the relevance of each measure for the performance in the individual steps. The literature review of Rantanen et al [Ran04] found 2475 directly and 2344 indirectly measured variables (also including workload measures) in 260 articles. These measures were ordered into 65 classes of direct and 36 classes of indirect measures. They found out that the most used indirect measures for human performance in ATC are the amount of actions, the performance rated by an observer, the response time, durations, the amount of events, and the self-rated performance. 70% of the measurement variables that have been used in the reviewed literature were one of those.

Validation of Arrival Manager

In the following the use of performance measures for the validation of AMANs (see section 2.2.4) is analyzed. For the validation of AMANs, mainly objective performance measures were applied. In the following a few examples are mentioned.

The COMPAS system (see section 2.2.4) was evaluated in real-time simulations and in field test. The simulation compared traffic samples (with COMPAS) against a baseline (without COMPAS) to assess traffic handling performance, controller workload, and acceptance. Direct measures like the observed and planned times, the aircraft's radar tracks and controller activities were recorded during the simulations. This data was used to calculate secondary measures like the sector and TMA flight time of aircraft as measures for performance [Sch98].

During the Eurocontrol project PHARE (Programme for Harmonised Air Traffic Management Research in Europe) conducted in the mid 1990s, which aimed to increase the capacity of European air space through a harmonization and the use of 4D-Trajectories, real time simulations were conducted, to show the benefits of the developed concept.

The objectives of the second major real time simulation exercise PD/2 was to assess the controller workload and performance of arrival traffic handling in the **Extended Terminal Maneuvering Area (ETMA)**. The controller was supported through a computer generated 4D profile and a computer assistance to plan and establish conflict free trajectories. To determine the accomplishment of these objectives, objective measures were calculated like the number of landings per time, the average flight time, inbound delay, the time precision of delivery, and the number of separation violations [RSA+98].

For the 4D-Planner, the successor of COMPAS, it was planned to measure the traffic flow (landing aircraft per time), the demand (possible landings, ignoring minimum separations), delay, and separation during its evaluation in the field [Sch98].

The successor of the 4D-Planner is called 4D-CARMA. This AMAN has been employed in several projects. One example is the project OPTIMAL. The objective of the project was to define and validate approach procedures with reduced noise immissions and increased safety and capacity. A dual threshold approach was validated, in which the use of two thresholds on a runway makes the reduction of the separation minimum possible. During this validation in a real time simulation in December 2007 and January 2008, arrivals per time, average track length, and average flight time [HHUR+09] were analyzed as objective measures.

Another example for the application of 4D-CARMA was the validation of the FAGI-Concept (see section 2.2.5). In the conducted real time simulation, objective performance data was recorded in the categories throughput, separation, conflicts, and flight efficiency. Throughput was measured in terms of landed aircraft. The secondary measures in the category separation were the violation of the separation at the touchdown (in terms of time and distance). Additionally, the average violation and the average additional flight duration and flight distance were considered. The observed violations of the separation were analyzed according to the involved aircraft types and the areas in which they occurred. The flight efficiency was measured in terms of the average flight time and flight distance of all aircraft (respectively all unequipped) [WO10].

These examples show that especially objective data (recorded flight trajectories and events) is used for the validation of AMANs. The use of this objective data is associated with the problems mentioned above. They allow for an assessment of the performance of the complete systems, consistent of human and controlled system. The determination of the respective contributions to the overall performance and thus the determination of the human performance in isolation is not possible. In the provided examples opportunity to analyze the operators' performance more accurately, to identifying specific problems, and to derive possible improvement for to systems is not used.

2.3.3. Human Performance Measurement in Microworlds

This section analyzes the measurement of human performance in microworlds. Microworlds are small-scale computer simulations of real systems [HV98]. Dependent on their purpose they are representative of industrial scale complex HMS or political processes, or, in the brought sense, human-environment interaction (the environment can be a company or a small city). They are designed to be between simple laboratory environments and field studies. Consequently, microworlds are highly controllable and traceable but they lack of a high degree of correspondence with the real system [Gra02, BD93].

Microworlds are commonly used in the similar domains of dynamic decision making [GVM05] and CPS [Fun98]. Because of their high face validity (they appear to have a high validity), microworld are also used in personnel selection [Fun98]. For this purpose, the real task is not important and the simulated task is only a cover story to increase the subjects' engagement. Furthermore, some microworlds exist which simulate the ATC tasks but for research questions in the field of ATC the terms mid-fidelity simulator or low-fidelity simulator are more common. Examples for such simulators are ATC-Lab Advanced [FLN09], Multitask (used by [KPS+06]), TRACON (used by [Ack92]) and FAirControl [MOW08]. Many studies have been conducted with microworlds or mid/low-fidelity simulators in the field of ATC (e.g. [KPS+06, MP01, RJR+00, JBPD11]).

Characteristics of microworlds

According to Brehmer [Bre92] microworlds used in the domain of dynamic decision making replicate three characteristics of real world decision problems. These are complexity, dynamics, and opaqueness. Complexity results from many interacting elements and from different—possible conflicting—goals. Dynamic in microworlds has four aspects: a series of decision has to be made, decisions are not independent, the world is dynamic, and decisions have to be made in real time. Microworlds additionally do not reveal all their characteristics and relationships and are consequently opaque.

Likewise microworlds are used in CPS. Intransparency, polytely, complexity, connectivity, and dynamics are important characteristics of complex problems according to Dörner [Dör89] and Funke [Fun91]. Intransparency implies a lack of clarity of a situation, which means the current state is not completely observable. The existence of multiple goals is depicted as polytely. The definitions of complexity and dynamics are similar to the definitions in the field of dynamic decision making given above.

The characteristics of microworlds as defined above are often found in real HMS. Consequently, research conducted using microworlds is also of relevance when analyzing problems in HMSs. Additionally CPS has some similarities with NDM (see section 2.1.5). They both concentrate on real-life tasks, which are ill-structures and dynamic, require a sequence of decisions and have multiple goals [Fun01]. While NDM focuses on decision making by experts in real life, CPS concentrates on novices interacting with microworlds.

Microworlds in ATC

As mentioned above, some microworlds were developed which are based on the working environment of ATCOs. One example is the mid-fidelity simulator ATC-lab Advanced [FLN09], which is the successor of ATC-lab and simulates the task environment of en-route ATCOs. Because of the unavoidable tradeoff between realism and experimental control and the different demands of research questions, ATC-lab Advanced allows varying realism and control systematically. Clearances are entered via keyboard and mouse. Values for speed and altitude have to be chosen from a dropdown menu, and a new heading is given by dragging a line to the destination point. ATC-lab Advanced was used in different studies for applied and basics research in which endorsed ATCOs as well as non-experts participated. A screenshot of ATC-lab Advanced is shown in Fig. 2.17.

Another example for a microworld featuring an ATC task is Multitask. Multitask is a PC-based simulation and shows the positions of aircraft on a radar screen. The task also includes two airports near the middle of the screen. The goal of the operator in Multitask is to contact the incoming aircraft and ensure their safe landing. Participants have to change the clearances for aircraft which potentially caused conflicts. The clearances were given by selecting one of eight commands from a control box. A screenshot of multitask is shown in Fig. 2.18.

Performance Measurement in Microworlds

The above applications put high demands on the quality of performance measurement in microworlds. Though, the measurement of performance in microworlds has been criticized in the literature. Since the participants input will determine the course of the process, the intermediate situation and thus the way on which a result was reached vary greatly. Additionally, there are often several very different ways leading to similar results. A performance measure, which considers only the result and not the process, ignores the differences between the chosen solutions [Fun98, HV98]. Moreover, if a microworld is designed without a known best solution, only a relative comparison of the reached score is possible [Fun98]. A further problem is that much data can be collected but that most of it is hard or impossible to interpret [Fun98, HV98]. In addition, a coarse performance measure (like the number of errors) is not suited to differentiate between subjects [HV98]. Also, it is seldom possible to differentiate between the performance of the operator and the system [Klu08]. These problems are very similar to the problems of performance measurement in human machine interaction (see section 2.3.1)

To solve these problems, various alternative performance measures have been proposed. Performance measures, which can be calculated from the recorded log files, have the advantages that they are objective and that they can be determined automatically. In the following, some suggested performance measures are discussed which meet these conditions and furthermore consider the process (process measures) instead of consider only the result (product measures).

Such performance measures were suggested by Howie and Vicente [HV98]. They proposed to analyze the course of variables in the continuous state space to measure performance. The proposed state space consists of measures which indicate the relative achievement of the operator's goals and thus can be considered as a goal space. In their demonstrative example, a process has to be controlled to produce a specified output and to reach a given temperature. Thus the specified output and the temperature are the objectives and the axes of the goal space. The process starts at the origin (0,0) and the point at which both objectives, output and temperature, are reached is defined as (1,1). One proposed measure of performance is the length of this trajectory. The more

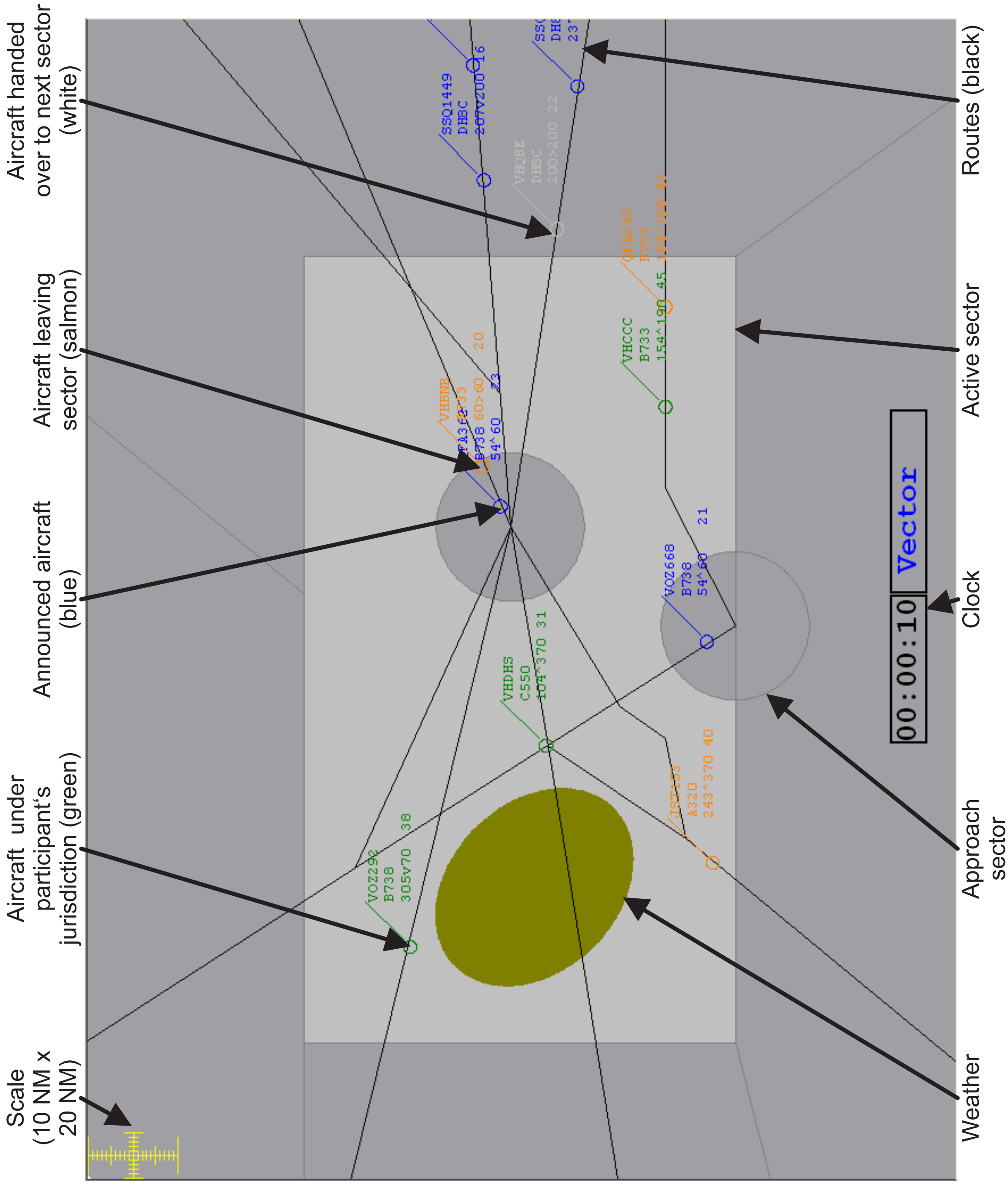


Figure 2.17.: Annotated user interface of ATC-lab Advanced [FLN09, FN08]

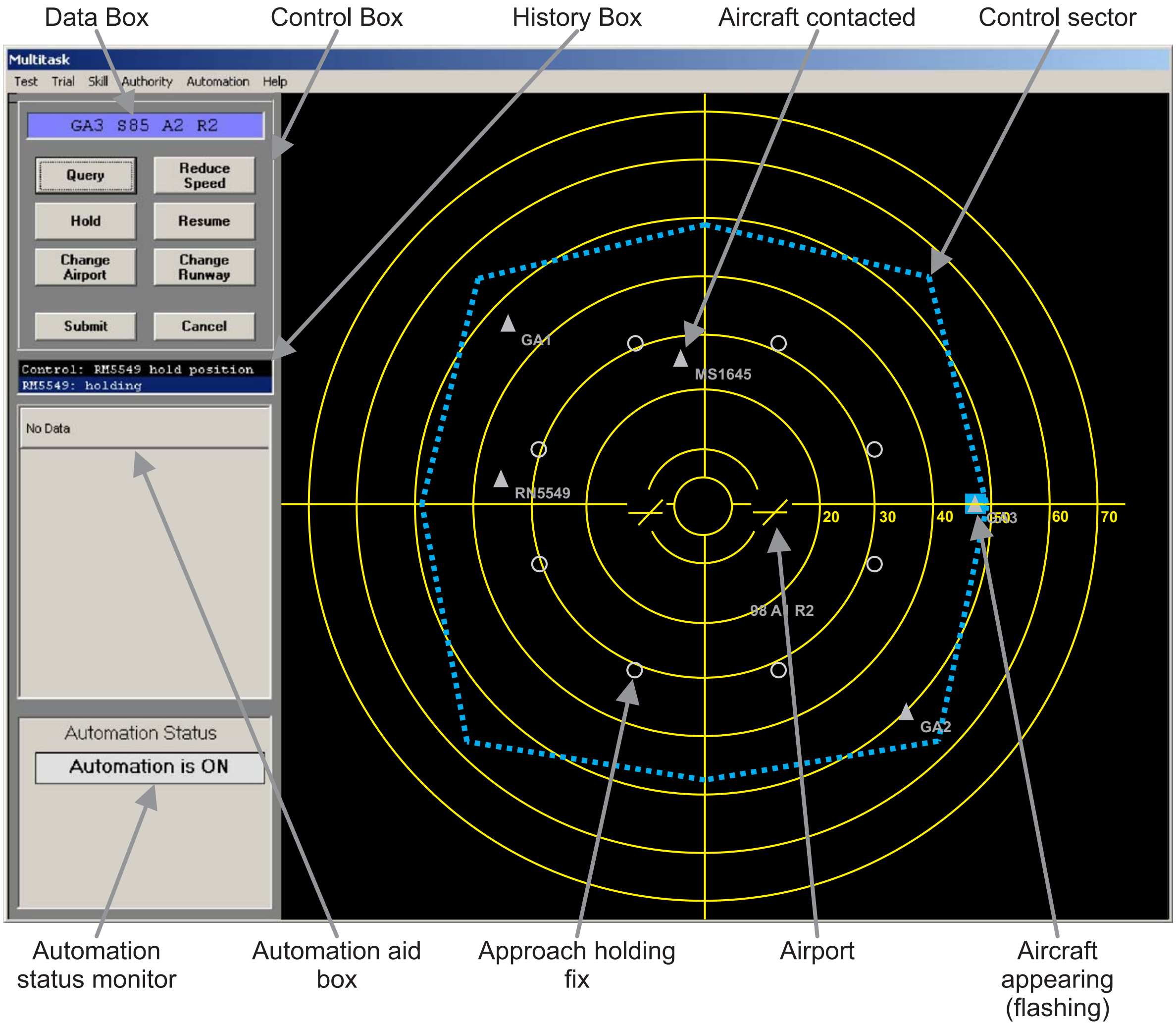


Figure 2.18.: User interface of the microworld Multitask [KPS+06]

directly the operator reaches to goal, the shorter the trajectory. Further the distance to the target in the goal space can be calculated at any time and indicates the approaching of the target point. Another proposed measure of performance is the integral of this distance over time.

One of the disadvantages of the measures proposed by Howie and Vicente is that discrete variables describing the objectives would unavoidable cause jumps in the performance measure. These measures further require a fixed and known target (at least know to the researcher). However, in some tasks with the objective to reach a maximum output the best possible solution is not known. Consequently a measure of relative performance cannot be defined for such tasks and the proposed performance measures cannot be applied.

Sager et al [SBD+11] proposed a performance measure based on optimization to compare the subject's performance to the optimal solutions. The developed method should allow the calculation of an objective performance measures for complex scenarios.

The method was demonstrated with the Tailorshop microworld. In this microworld, participants are in charge of a small company producing shirts for one year. They can make decisions about the infrastructure (machines and workers), the financials, and logistics in every month. Hence the participant's decisions can be described by an input vector for each month $u_k = u(k)$, for $k = 0, \dots, N-1; N = 12$. The state of the microworld is $x_k = x(k)$, for $k = 1, \dots, N$. The objective is to maximize overall balance at the end of the simulation, thus the objective function is

$$F(x_N) = x_N^{overall\ balance}. \tag{2.4}$$

Sager at al used optimization methods to calculate an optimal solution for each month and constructed a so-called *how much is still possible function*

$$F^*(x_N; n_s), \text{ for } n_s = 0, \dots, N-1. \tag{2.5}$$

This function allows the assessments of the still possible performance for every month and thus enables to identify good and bad decisions. They further used Lagrange multipliers to analyze the impact of the components of the input vectors on the *how much is still possible function* which enables to determine the impact of each decision. The *how much is still possible function* is monotonically decreasing and allows inferring the potential that has not been used. The not used potential ΔP_k is calculated by taking the difference between the results of the *how much is still possible function* for two consecutive months

$$\Delta P_k =: F^*(x_N; n_s = k+1) - F^*(x_N; n_s = k). \tag{2.6}$$

As long as the decisions are compared to a global optimum, the function ΔP_k is non-positive and indicates the lost potential for each decision.

This approach allows looking into detail by using a process measure instead of a product measure. However, it is also limited to microworlds, in which a state depends only on its predecessor state and the decision, but not on time. In such microworlds the state of the world does not vary with time continuously but only as a consequence of decisions.

2.4. Concluding Remarks

This section presented some performance measures for human interaction with complex dynamic systems. Examples from HMI in general as well as examples from ATC and microworlds were discussed. It was shown that most performance measures concentrate only on the result and ignore the course of actions and possible alternatives. However, measuring the performance in each situation would allow evaluating each single decision. As a consequence, more detailed insight into the working methods of human operators would result.

A performance measure considering the process was proposed by [SBD$^+$11] for the use in microworld. The optimal solution is calculated with optimization methods. The resulting *how much is still possible function* is the best possible action sequence in each situation and used as a criterion. The method presented to calculate the *how much is still possible function* can be applied if the state of the world depends on the former state and the operators' decision but not if the state also depends on time. In complex dynamic systems, in which the state also varies with time and the exact timing of an action is important, the definition of such a sequence of actions is more complicated. Not only the different actions and different sequences but also the different points in time for each action have to be considered.

This problem will be tackled in this thesis. Consequently, the first main contribution of this thesis is the development of a cognitive planning model to calculate a *how much is still possible function*, which can also be applied to transform a given situation into a goal situation if the state of the environment changes with time (see chapter 3). The interaction sequences calculated by this model are externalized variables used as standard for the measurement of human performance. Human erroneous actions can be detected as deviations from the best available option.

As described in section 2.1, operators do not make the optimal decision in most cases but chose an action which has a satisficing result. Using the optimal performance as criteria causes the excessive detection of human erroneous actions. A differentiation between human error and not optimal performance is not possible. One of the main reasons for a not optimal performance is the uncertainty, both regarding the current situation and especially regarding future states and the consequences of actions. To consider the prediction uncertainty as a reason for not optimal solutions, it is integrated into the criterion by integrating it into the cognitive planning model. The interaction sequences generated by the extended model shall differentiate between small inevitable deviations caused by prediction uncertainty if the human operator is not able to identify a better option and larger deviations if the human operator should be able to select a better option. This extended model calculates action sequences which lead to the goal even if the future is not exactly predicted. Consequently, these sequences deviate from the optimal performance but are a more realistic estimation of human behavior and can be used to identify human errors.

The integration of this uncertainty into the calculation of the action sequence is the second main contribution of this thesis. This will be explained in chapter 4.

If the consequences of operators' decisions are compared to the *how much is still possible function*, exactly the operators' influence is measured and the performance can clearly be assign to the operator. However, to allow evaluating operators' single actions it has to be considered that the results of this action may only show up later. Consequently, the immediate consequences cannot be compared to the *how much is still possible function.* To enable the comparison between the effects of the implemented actions and the *how much is still possible function*, both have to be considered in the same time horizon. Therefore it was proposed [SBD$^+$11] to calculate the difference of the *how much is still possible function* in two consecutive steps to get the lost potential with each implemented action. The observed and implemented actions may be only the first step of a plan the operator has in mind to reach the goal. When this step is evaluated, the further steps the operator could take have to be considered. Hence, to evaluate single decisions, the best solution before the actions is implemented is compared to the decision of the operator, which consists of the measured action and the best possible option.

This method allows not only evaluating actions implemented by the operator but also identifying missing actions. They are indicated by a decrease of the reachable performance (equally to a lost potential) from on situation to the next without an action between. Consequently, the use of a goal-directed sequence as a criterion enables evaluating the human operator's behavior at every point in time. After human operator's decisions are evaluated with this method, the decisions can be assigned to one of four categories similar to those of the SDT (see section 2.3.1). The first dimension is the necessity for action which is given if the sequence of actions includes an immediate action. This action can either be executed by the human operator or not (second dimension). Depending on the position of the executed action in the sequence and its effect, it can also be classified as too late or too early and the extended classification scheme as defined in the time window method can be applied (see section 2.3.1).

The third main contribution of this thesis is the development of a method to assign the measured behavior to the categories of these classification schemes. This will be explained in chapter 5.

The evaluation of actions based on the possible outcome has three advantages.

- First, the characteristics of a situation are compared to the performance in this situation and thus allows detecting differences between the human operators' mental models and the reality.
- Second, the method is objective and reproducible since a model and an algorithm are used to interpret the recorded interaction logs.

- Third, the integration of time-dependent dynamics allows an application to a variety of systems.

3. Petri-Net-Based Modeling of Human Planning

The human operator cognitive task performance measure developed in this thesis uses goal-directed interaction sequences as criteria for the evaluation of human operators' decisions. To generate these interaction sequences, a cognitive planning model is developed. This chapter is dedicated to the development of this model. The developed model consists of three parts. First, it includes a **Coloured Petri Net (CPN)** model of the task environment. The second part of the model is a set of rules describing normative human operators' behavior. The third part is the planning process, which analyzes the state space of the modeled task by applying the set of rules defining the normative behavior. Thereby, the best available option is identified.

The chapter is structured as follows. At first, the concept for the generation of interaction sequences is explained. In particular, the requirements imposed by an application of the method in a complex dynamic task environment as well as their fulfillment will be discussed (see section 3.1). Second, both applied modeling techniques, namely CPNs and **Situation-Operator-Modeling (SOM)**, are introduced. Subsequently, the commonalities of these techniques and their application for the simulation and analysis of HMSs are described in section 3.2. Afterward, the task and the implementation of the utilized microworld simulation environment will be presented in section 3.3. The CPN used for the simulation in this microworld is detailed in section 3.4. Subsequently, the deduction of the set of rules is explained in section 3.5 followed by the description of the implemented planning process in section 3.6. Finally, the developed planning model is demonstrated exemplarily in section 3.7 before some concluding remarks are given in section 3.8.

Previous versions of the cognitive planning model developed in this chapter are already published in [HS13a, HS13b, HS14].

3.1. Requirements and Concept

In this section, the concept for the generation of a planned interaction sequence as a criterion for the evaluation of human operators' decision will be presented. This is based on the analysis of the requirements imposed by an application of the method in a complex dynamic task environment.

3.1.1. Requirements for Performance Criteria in Human-Machine Systems

Several difficulties in dynamic HMSs complicate the definition and generation of goal-directed interaction sequences as criteria. However, to ensure the applicability of the method to a broad range of HMSs, the developed concept has to fulfill the requirements imposed by characteristics commonly found in HMSs and discussed in the following.

- In some cases, the conditions of the system can facilitate its operation, e.g. when only few actions are necessary to reach a certain result. On the other hand, the system can converge to a critical situation without human operators having a change to avoid this situation. In other words, a good result reachable in one situation will probably not be reachable in most other situations. This requires generating a new criterion for each situation.

- Some actions have not only immediate consequences but have also long-term effects. These effects include the gain and loss of options as consequences of actions. The effects have to be considered to decide about goal-directness of an action. This requires knowing an interaction sequence leading the system from the actual situation to a goal situation as a criterion.

- In a dynamic system, the variables can change both as a consequence of operators' interventions as well as with time (e.g. [End95b]). Thus the effects of an action additionally depend on the time of its execution. Consequently, actions or interaction sequences are only goal-oriented in small time windows. Consequently, the options (represented by interaction sequences) are changing with time and a frequent update is necessary.

- Interactivity is a fundamental characteristic of HMSs, as human operators influence the systems and determine the course of events. In addition, inputs can be made in various forms. For example operators can have the choice between several target values. In many cases the systems' states depend on the operators' input so strongly that small differences in input lead to completely different situations which cannot be foreseen. Consequently, interaction sequences as criteria can only be calculated during or after an observed interaction but not previously. Additionally, the complete set of actions possible even in a specific situation cannot be considered due to its enormous amount [OHS11].

- The state of a system is often described by time-dependent continuous variables which are linked for example with differential equations. The operator input, on the other hand, is often event discrete. The modeling technique to describe interaction sequences must consequently allow describing discrete and continuous variables.

- Additionally, there are often multiple and sometime contradictory objectives. For example, the task may be to increase the output of a process and to avoid critical situations in the meantime. Thus, the developed approach must be able to deal with multiple objectives.

3.1.2. Concept of the Developed Planning Model

The developed model to generate interaction sequences is an implementation of the cognitive function planning. However, the information processing of humans consists of several cognitive functions, e.g. after the HMR (see section 2.1.2) of perception, interpretation, planning and execution. Focusing on planning is sufficient as this function generates possible courses of actions out of the available options. As a plans rejection would require an alternative plan to be generated, the future course of actions is eventually determined by the planning function.

In the developed model, planning is realized two-piece. On the one hand, a prediction based on a CPN is used to calculate the possible future states and problems of the controlled system. On the other hand, a set of rules is defined which models operators' behavior. Each rule describes an action added to the plan to prevent detected problems. Consequently, the planed sequence strongly depends on the definition of problems and corresponding actions to avoid these. Therefore, the rules, which combine a problem definition and a suitable solution, are thoroughly and systematically deducted from the objectives and the available action.

A plan represents what is reachable in a given situation. Using a plan as a criterion will consequently allow differentiating between the impact of the human operator and the inevitable effects of the controlled system. To consider also long term effects, plans consist of sequences of actions. To deal with the dynamics and interactivity of HMSs, plans will be generated frequently for the situations measured during an interaction. Out of the many options available in a specific situation, each generated plan represents only one option. This is sufficient only when the plan is goal-directed and represents the best option available. In this case, other options can either be excluded by the defined set of rules or refused during the planning after being identified as inferior. Consequently, the objectives have to be integrated into the set of rules. It is also possible to integrate multiple objectives into the set of rules. Different objectives can be realized by adapting the set. In the developed approach, the continuous variables are discretized in time and each time step is treated as an event. The dynamics of the system are modeled as autonomous changes whereas the inputs of the operator are modeled as external events. Consequently, the developed model fulfills the requirements defined above.

Petri Nets were chosen because they offer numerous possibilities for analyzes as a formal language. Furthermore Coloured Petri Nets can be used for realizing a simulation environment so that real interaction and simulation can be performed using the same model [OGS08, Wer06]. Additionally, CPNs already demonstrated their benefits for the realization of simulation environments and the analysis of human behavior [GOS09, MOW08, HOS09, OHS11]. Another advantage of realizing the concept based on CPNs is that the implementation of the simulation of the operated system and of the planning model can be realized with the same programing language. The benefits of CPN for the holistically modeling of systems and human behavior are also already demonstrated [Wer06, Möh11].

If a plan representing a goal-directed interaction sequence is known, the effects of the human operators' decisions can be compared to the effects of the sequence. However, also human operators' decisions have long term consequences and it is not sufficient to compare the immediate consequences of a decision to the long-term consequences of an interaction sequences. To allow such a comparison, the consequences of the operators' decision and the consequences of the interaction sequence have to be considered in the same time horizon. Therefore, a second interaction sequence is necessary to complete the observed decisions. Comparing both sequences in terms of the quality of the reachable goal allows evaluating the operators' performance. This method is not bounded to evaluate actions (observable decisions). Additionally, missing actions (omissions) can be identified when two interaction sequences of adjacent situations are compared. Missing actions are indicated by a decrease of the reachable performance without an action between. Hence, implementing no action is also considered as an option available for human operators. The details of the application of the developed cognitive planning model for the measurement of human operator performance are given in chapter 5.

3.1.3. Related Work

The developed model is based on the analysis of action spaces, which are modeled with CPN and analyzed with the SOM Technique. Situation-Operator-Modeling was developed inter alia for the formalized description and analysis of HMI [Söf03, Söf04a]. The use of Petri Nets to build executable models based on SOM and the analysis of Petri Nets with SOM have been proposed in former research [GOS09, EGVS10]. One application is the realization of cognitive architectures [GS09b, EGVS10]. Apart from that, SOM and Petri Nets have also been used to enable the comparison of operators' decision making in dynamic task environments and the results from formal state space analysis. For this purpose simulation environment based on CPNs have been developed [MOW08, GOS09, GS09a].

One example for the analysis of human decision making based on a CPN model can be found in [MS10]. A holistic human-machine model was realized as CPN. To analyze heuristics applied by air traffic tower controllers, not only the processes at an airport where modeled (machine model) but additionally the decision making of the controllers (human model). Both models are connected with an interaction model. The model can be coupled with a graphical user interface to constitute a simulation environment. Two heuristics, a hierarchical and a first-come-first-serve heuristic, were implemented with the three-step-principal consisting of an information search rule, a stop rule, and a decision rule. The calculation of the state space of such a holistic human-machine model reveals the decision space of the tower controller and allows analyzing this space.

In this approach, only a small set of inputs can be made by the operator. Furthermore, the effect of time is neglected for the generation of the action space. The approach is thus not able to deal with interactivity (as a complete state space needs to be calculated) and dynamics as the approach presented in this theses. The holistic modeling approach and the cognitive planning model developed in this thesis both consists of a human and a machine model. In the holistic approach, both are integrated into one CPN whereas in this thesis the CPN only models the technical system (machine) and the set of rules models human behavior.

Another CPN-based simulation environment was applied in [GOS09] for the automated detection of formalized human errors (**Automated Error Detection (AED)** approach). The environment simulates the abstract task of an arcade game. The Petri Net is not used as a core of the simulation environment itself. Instead, it models the task and is connected to it to receive information about executed actions and changes in the environment. In this application, multiple partial state spaces are calculated starting from measured intermediate states instead of a complete state space starting from the initial state. This reduces the computational demand. The partial state spaces are limited by a defined maximal amount of nodes. As theses state spaces are based on measured intermediate states, partial state spaces can only be calculated a posteriori. As an example, the human error 'rigidity' as described by [Dör89] and formalized in [Söf04a, Söf04b] is detected if

- an action is not goal-directed,
- the previous action was goal-directed, and
- the action would be directed to the previous goal if no event would have occurred in the meanwhile.

Therefore, the detection of errors requires the identification of action's goal-directedness. An action is goal-directed, if the amount of actions necessary to reach the goal is reduced. As partial state spaces do mostly not contain final goals, intermediate sub-goals were defined and used to determine the goal-directedness.

The similarity between the AED approach and this thesis are the a posteriori generation of partial state spaces to get goal-directed sequences. In the AED approach, the partial state space is first generated and afterward analyzed to determine the goal-directedness. In this thesis, the states are analyzed during the calculation of the partial state space to allow concentrating on a goal-directed sequence. Therefore, this thesis applies a model of human behavior, whereas the AED-approach uses only a machine model. A further difference is that the AED approach can only be applied if changes of the environment can be modeled as single events. It is not able to deal with continuous dynamic.

This thesis is also influenced by the **Know Your Options (KYO)** approach [OHS11]. The KYO approach also analyzed CPNs to evaluate human operators. A two-folded approach is developed to detect difficulties with the interaction of human operators with dynamic environments by contrasting the existing solutions to the perceived solutions. To determine the existing solutions, partial state spaces are calculated a posteriori for measured intermediate states. A simple set of rules to model human behavior is defined to guide the calculation of state spaces. The aim is to determine the goal-directedness of the options defined by the rules.

The main difference between the KYO approach and this thesis is that goal states have to be reachable by the execution of only one action in the KYO approach. In contrast, the planning model developed in this thesis is able to generate sequences of actions to transform an initial state into a goal state. Thus, it can handle tasks requiring more actions to reach goals and consider long-term effects of actions. Furthermore, in the KYO approach, the correct perception of available options was analyzed. In this thesis the execution of actions (and thus the selection and implementation of an available option) is evaluated.

The aim of this thesis is to develop an objective process measure of the operators' task performance. To evaluate the operators' decisions, goal-directed interaction sequences for the operation of complex dynamic systems are used as criteria. As in former approaches, CPNs are used to analyze the consequences of operators' decisions. In contrast to these former approaches interaction sequences instead of single actions are used as a criterion. These are calculated by a newly developed planning model. Furthermore, the developed method can be applied to a wider variety of system, due to its ability to handle actions which effect depends on the time of their implementation.

3.2. Applied Modeling Techniques

In this section, first the concept of CPNs, which is a Higher Petri Net formalism, is described. CPNs are used to model the system controlled by the human operator and are applied in a way that the same model can be used during the interaction and for the purpose of analysis. Next, the SOM technique is introduced. This technique is applied to describe the procedure of the developed planning model. The section will close with a brief comparison of both modeling techniques.

3.2.1. Coloured Petri Nets

Petri Nets and their extension CPNs applied here are mathematical and graphical modeling techniques initially developed by Carl Adam Petri [Pet62]. Petri Nets will be briefly introduced in the following. For an extensive introduction to Petri Nets see [Bau96] or [Kie06]. The following definitions of CPNs are taken from [JK09].

A Petri Net is a bipartite graph which consists of a set of places P and a set of transitions T. They are connected by a set of directed arcs A which are a subset of $(P \times T) \cup (T \times P)$. Places are symbolized by circles (or ovals) whereas transitions are symbolized by rectangles (or bars). If a directed arc connects a place p_i to the transition t_j, p_i is an input place of t_j. If a directed arc connects a transition t_j to a place p_i, the place p_i is an output place of t_j.

Places can store data in terms of tokens (or marks). The set of all tokens in the net are called the marking of the net and define its state respectively the state of the modeled system. In common Petri Nets the tokens are uniform. In contrast, tokens can have a data type like `integer` or `string` in CPNs and thus represent different characteristics. Furthermore, combined data types are possible, which are similar to structures in some programming languages. The data type of each token is called color set and must belong to the set of defined color sets Σ. Each place p can only contain tokens of a fixed color set $C(p)$. This is defined by the color set function $C : P \to \Sigma$. As the color set function is fixed, the data type of the place tokens can not change during run-time.

Transitions model the changes of the system's state and can modify the marking. A transition can fire (is enabled), if the transition's input places contain enough tokens and—if the capacity of places is limited—the output place have enough free capacity. The amount of necessary tokens on the input places respectively the necessary free capacity on the output places is defined by weights assigned to the respective arcs. If

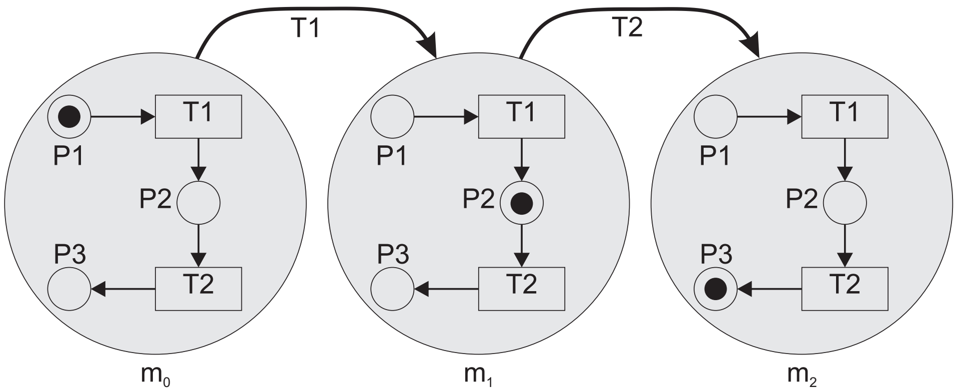


Figure 3.1.: Example of a state space with 3 states (large gray circles). In each state, the corresponding marking of the underlying Petri Net is depicted. The initial marking $m_i = m_0$ is transformed with the firing sequence T1,T2 into the final marking $m_f = m_2$.

a transition fires, tokens are deleted form the input places and are added to the output places corresponding to the weights. Although several transitions can be enabled at the same time, only one transition can fire at once. After firing a transition, the conditions for all transitions are checked again.

To be able to use the extended functionalities of CPNs and to handle more complex marks some variables V have to be defined whereby the type of each variable v has to agree with one of the defined color sets $Type[v] \in \Sigma$ for all $v \in V$. Further, expressions e can be used in CPNs, with $EXPR$ as the set of all expressions. The free variables used in an expression are denoted as $Var[e]$ and the type of the expression is denoted as $Type[e]$. The set of expressions for which all $Var[e] \in V$ are denoted as $EXPR_V$.

One extension of CPNs is that instead of a weight, an arc expression e can be assigned to each arc a by $E : A \to EXPR_V$ to manipulate the token and the data stored in the connected places. Additionally, a function can be assigned to each transition to manipulate the data of all input places or to define the tokens which are added to the output places. This function can be seen as a special kind of arc expression. A further extension is the concept of guards. A guard g can be assigned to each transition t by $G : T \to EXPR_V$ and use the tokens on the input places as input. The type of the guard is required to be a Boolean expression. Additionally to the condition that each input place must contain at least on token, a transition with a guard is only enabled if the guard evaluates to `true`. All reachable markings of a Petri Net constitute its state space or reachability space.

The advantage of Petri Nets is their formal definition and the resulting possibilities for analyzes. One kind of analysis is the reachability. A final marking m_f is reachable from an initial marking m_i if a firing sequence of transitions exists to transform m_i into m_f. An example is illustrated in Fig. 3.1. The developed planning model will use this functionality to determine the reachability of goal states.

3.2.2. Situation-Operator-Modeling

The core of the meta-modeling technique SOM as developed in [Söf03, Söf04a] is that changes of the real world are regarded as a sequence of scenes and actions [Söf03,Söf04a, Söf08, GOS09]. A problem-fixed scene is modeled by a situation while focusing on relevant aspects of the system. A problem-fixed scene can vary with time but the same event always causes the same structural changes of the scene independent from the moment of its occurring. The item operator is used to model effects/actions changing scenes like human actions. The considered aspects of the world are modeled by combining situations and operators to an alternating sequence [Söf08].

The item situation is therefore used to describe the internal structure of the modeled system. The situation S is composed of characteristics C and connecting relations R. The characteristics consists of an unique name (or identifier) describing relevant facts and a variable parameter p (or value) which can have an arbitrary data type. Furthermore, the amount of characteristics is not fixed and may vary from situation to situation. The item characteristic also includes the possibility of representing time-dependent parameters. The relations R are structured like certain modeling techniques (e.g. ODEs, DAEs, etc.) to describe the interdependence of characteristics [Söf08].

Operators are used to model the events and actions changing the scene (changes of situations S). According to [Söf08], a functional perspective is taken to model operators O. The item "operator is an information theoretic term that is defined by its function F (as the output) and the related necessary implicit and explicit assumptions iA and eA as inputs" [Söf08]. The explicit assumptions eA have to be fulfilled to realize a function F. These assumptions have the same quality as the situations' characteristics C [Söf08].

Operators and situations are closely connected due to the identity of the characteristics of the situations and the explicit assumptions of the operators. This allows the double use of operators, as passive operators used to structure situations internally and as active operators to connect consecutive situations externally [Söf08].

Situations are illustrated as oval gray areas and operators as circles connecting two situations. An example is shown in Fig. 3.2. In this example, an operator O_1 transforms the initial situation S_1 into the situation S_2. The situation S_1 consists of the two characteristics C_1 and C_2 (illustrated as small light gray circles), which are connected by the relation R_1 (larger dark gray circle). The situation S_2 consists of two relations (R_1, R_2) and three characteristics (C_1, C_2, C_3). The operator adds the characteristic C_3 and the relation R_2, and therefore describes the changed problem configuration. Other operators may change only the parameters or delete characteristics or relations.

The possible interactions of an agent with its environment can be interpreted as a net of scenes and actions and modeled by a bipartite graph of situations and operators. The root of the graph is the initial situation to which all possible operators have to be applied. Thereupon all possible operators are applied to the resulting situations. This process is repeated until all possible operators have been applied. The graph can contain the same operator a number of times. Such a graph of situations and operators resulting from the execution of possible operators from an initial situation as root is denoted as action space according to [GS10].

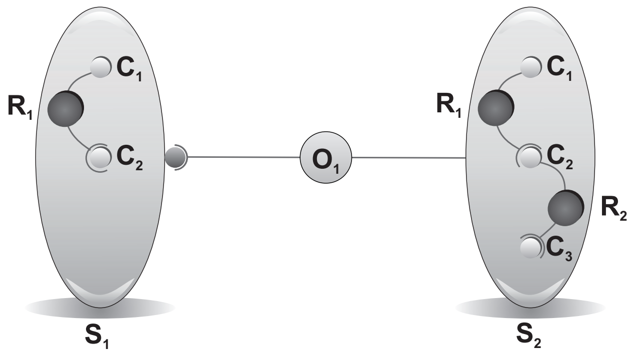


Figure 3.2.: Situation-Operator-Situation sequence. Operator O_1 adds characteristic C_3 and relation R_2 to situation S_1 which results in situation S_2.

If a lot of actions are possible, the graph may become very large or even infinite. Thus, the graph has to be reduced in a suitable manner. An action space containing only relevant actions is denoted as a "partial action space" [GS10]. The presented planning model uses the set of rules describing goal-directed sequential behavior to concentrate on the relevant aspects of the action space. This also reduces the action space to a partial action space.

The SOM approach only gives the frame to model the structure of changeable scenes, and therefore maps the action space using the proposed structural framework to a formalizable representation [Söf08].

3.2.3. Similarities of Coloured Petri Nets and Situation-Operator-Modeling

Coloured Petri Nets and SOM are both bipartite formalisms consisting of active and passive elements. While only one situation is used to model a scene in SOM, a set of places define the actual state of a Petri Nets. Thus, the item situation (including its characteristics) in SOM corresponds to the places of a net (including the tokens). Events in the real world are modeled as operators in SOM and as transitions in Petri Nets. The change of the scene is described by the operator function F and by the arc expression e in CPNs. The explicit assumption eA of an operator correspond to the guard G in CPNs. In this case, each operator is equivalent to one transition. These similarities enable to implement SOM models as Petri Nets and to analyze Petri Nets with SOM. This is also pointed out by [GS09a].

Despite these similarities, there are also differences between both modeling approaches. On difference is that a net of places and transitions only defines the actual state and the possible set of events but a graph of situations and operators describes the action space as defined above. Furthermore, SOM is not bounded to a predefined set of characteristics and relations but the net structure of a Petri Net is fixed. Hence, the SOM formalism extends the possibility of Petri Nets. However, this limitation of classical Petri Nets can be avoided when Reference Nets, a Higher Petri Nets formalism, are used which allow Java objects as tokens [Gam11].

3.2.4. Coloured-Petri-Net-Based Simulation

According to [GS09a], a situation is modeled as a place containing one token. As the data type of this place/token is fixed (in contrast of the structure of a situation in SOM), all characteristics possible during the interaction must be included in the data type. Active operators are modeled as transitions which can modify the token describing the current situation. The simulation environment applied in this contribution follows this approach, but extends it by additionally modeling passive operators and thereby the inner structure of situations as transitions. This extension will be detailed in section 3.4.

Implementations of SOM models as CPNs are used as cognitive architectures [GS09b, EGVS10], for the comparison of operators' decision making in dynamic task environments to the results from formal state space analyzes [GOS09, MS10, OHS11] and for simulation environments based on CPNs [MOW08, GOS09, GS09a].

Coloured Petri Net simulation environments can be used in different ways to analyze the behavior of operators in dynamic task environment. In [MS10] it was used to identify heuristics used by the operators of an air traffic control tower simulation. It was further applied in [GOS09] for the automated detection of formalized human errors. The approach was also used in [OHS11] to compare the consequences of available options as perceived by the operator and the real existing consequences of these actions in a dynamic environment.

3.3. Application Example Microworld MAGIE

The human performance measure applying the cognitive model of planning developed in this chapter is demonstrated with the simulation environment MAGIE. The simulation environment MAGIE was built as a mid-fidelity simulator to evaluate prototypes of new procedures and assistance systems for ATC in an approach sector within a simplified and highly controlled setting (for details see former studies using this simulation environment e.g. [OWR09, OWM10, WFS+10] or section 2.2.5 for a brief introduction of the evaluated procedure). The simulation environment was further extend to be suitable for different task environments like en-route control (as used in [OHS11, PW12]). However, the original approach task will be used in this thesis.

The simulation environment MAGIE consists of a CPN to simulate the aircraft's physical behavior, a **Graphical User Interface (GUI)** which is shown in Fig. 3.3, and a very basic AMAN. Implementing the simulation of MAGIE as a CPN following the design approach presented in section 3.2.4 enables analyzing human behavior by contrasting the Petri Net's state space to the queried operators' perception and to the measured consequences of decisions.

In the new arrival concept implemented in MAGIE, aircraft are divided into two groups depending on the capability of their FMS. Aircraft, which are equipped with a 4D-FMS and are able to fly a specified trajectory within a high time/location-precision, follow a direct approach. They negotiate a fixed time with the implemented AMAN and are allowed flying their preferred profile as long as they can meet the time restrictions at the LMP [OTK+08].

Aircraft, which are not equipped with a 4D-FMS, have to be guided in the conventional way manually from STARs over the path-stretching area consisting of downwind leg, base leg, and extended centerline. They start turning from the downwind leg towards the extended centerline after being instructed. Unequipped aircraft are merged into the stream of equipped aircraft at the LMP [OTK+08]. Both groups of aircraft fly along the final to the runway. To ensure a safe separation between the aircraft (which depends on the wake vortex category), it is crucial to stretch the flight path of the unequipped aircraft in the right amount. This requires predicting the aircraft's trajectory up to the LMP.

Assistance in shape of ghosting and targeting was developed to support the operator in this task during the FAGI project [OWM10] (see section 2.2.5). If ghosting is activated, a copy (ghost) of each equipped aircraft is projected onto the runway centerline extension to indicate positions later occupied by equipped aircraft. If targeting is activated, a copy (target) of each unequipped aircraft is projected onto the runway centerline extension. Targets indicate the goal-positions of unequipped aircraft calculated by an AMAN. Both tools reduce the length of predictions operators have to make. Without assistance, the positions of equipped and unequipped aircraft have to be predicted up to the late merging point. If ghosting is active to indicate positions for equipped aircraft, the positions of unequipped aircraft have be predicted relative to the ghosts. If targeting is active and the goal-position for each unequipped is calculated and displayed, only the change of the difference between actual and goal positions (targets) needs to be predicted.

The simulation environment MAGIE was created to analyze the benefits of new visual assistance following that new concept. Consequently, the task of the operator is derived from that concept. In MAGIE, the operator has to control the unequipped aircraft by issuing clearances, which have to be selected out of a set of available clearances. An unequipped aircraft will fly along its arrival route and into the path-stretching area if it is not instructed otherwise. In contrast to that, the 4D-equipped aircraft and aircraft outside the control zone cannot be influenced. The control zone ranges from short before the downwind leg to short after the late-merging-point (gray zone in Fig. 3.3).

The route structure and the actual position of the aircraft together with their current and cleared speed and altitude are shown in the GUI (see Fig. 3.3). The aircraft equipped with a 4D-FMS have a call sign starting with "A" and are shown in red (light gray in the figure). The unequipped aircraft have a call sign starting with "U" and are shown in yellow (dark gray in the figure). Clearances are given by clicking on an aircraft label, choosing a clearance from the selection window (s. Fig 3.4), and confirming it by another click. In this manner, the operator can change the speed and altitude of the aircraft.

The possible altitude ranges from 3000 ft to FL 80 (flight level). Altitudes above the transition altitude, which is defined at 5000 ft, are given in flight level (1 FL = 100 ft). To simplify the conversion, the local QNH is defined as 1013 hPa so that FL 50 is equal to 5000 ft. The speed ranges from 160 kn to 250 kn. The operator can further instruct the aircraft to start the turn maneuver from the trombone to the extended centerline.

The simplified AMAN integrated in MAGIE generates an arrival sequence, calculates trajectories for each aircraft, and derives the commands necessary for these trajectories. The generated sequence can be displayed in the GUI of MAGIE as a table including call

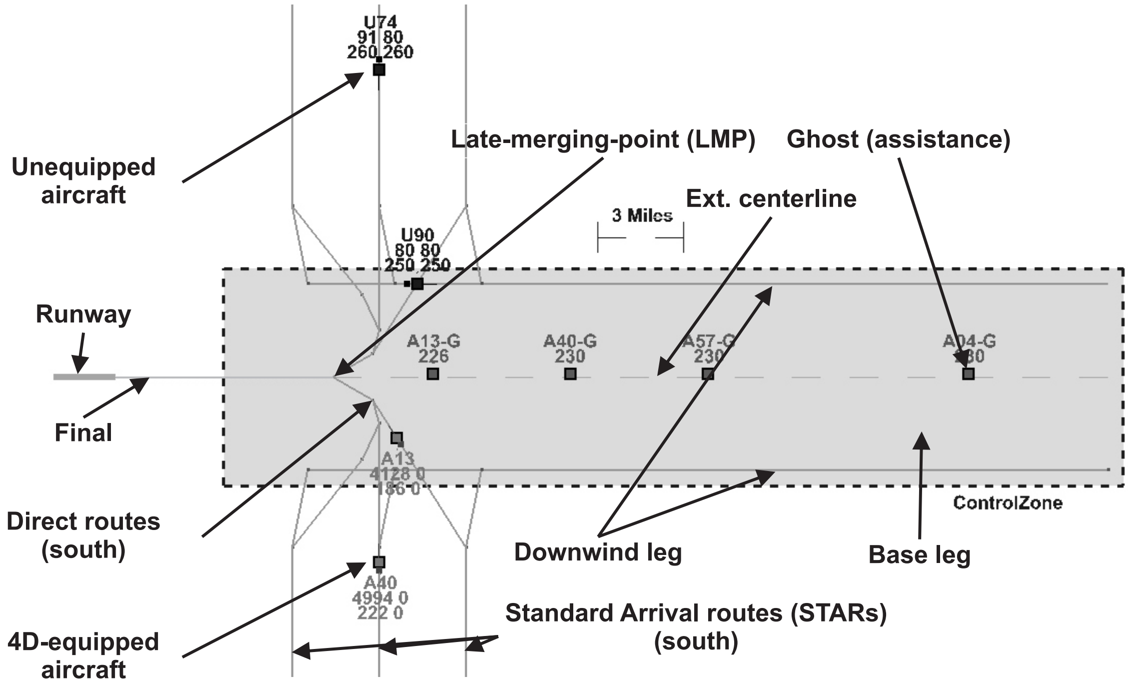


Figure 3.3.: The GUI of the simulation environment MAGIE [HS14].

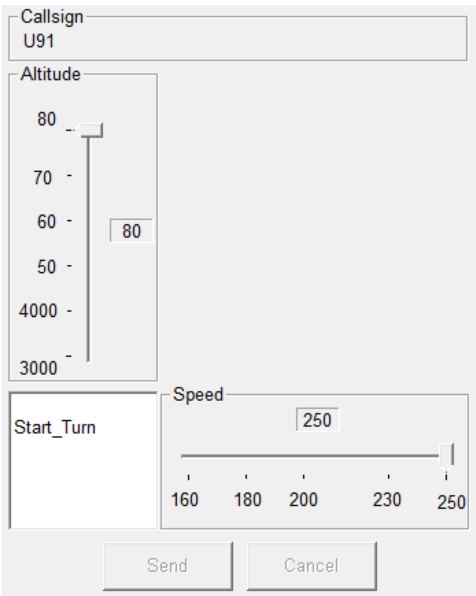


Figure 3.4.: Command window of the MAGIE GUI to change altitude (in the upper left) and speed (in the lower right) and to start the turn maneuver (in the lower left)

sings and arrival times at the LMP sorted by arrival times. Additionally, the concept of ghosts and targets is integrated and virtual aircraft are projected on the extended centerline. The displayed call sign of ghosts consists of the call sign of the corresponding equipped aircraft and a “G”. For targets, a “T” is added to the call sign of the corresponding unequipped aircraft. Moreover, derived commands can be displayed as advisories, either for each aircraft individually at its label or combined in a stack.

The possible set of actions can be modeled by three operators. The operator $O_{spd(a,x)}$ models a speed change of the aircraft a with the target value x. Instructions concerning the altitude of an aircraft are modeled by the operator $O_{alt(a,x)}$. Finally the operator $O_{turn(a)}$ corresponds to the instruction for aircraft a to start the turn maneuver.

The main objective of the human operator is to ensure a safe separation between all aircraft. This is defined as 3 NM for each pair of aircraft (according to a medium/medium combination). The separation is of the utmost importance. The operator further has to ensure an efficient trajectory. This is implemented as constraints for different sections of the aircraft's route (as shown in Table 3.1) and by the objective to guide the aircraft quickly to the airport (of minor importance compared to the other objective). In summary the operator has the three objectives (in order of decreasing priority): *Separation*, *constraints*, and *throughput*.

Table 3.1.: Constraints on route sections requiring reductions of speed and altitude [HS14]

Section	Speed		Altitude	
	Min	Max	Min	Max
Downwind leg	180 kn	250 kn	5000 ft	8000 ft
Base leg	180 kn	250 kn	3000 ft	8000 ft
Ext. centerline	160 kn	230 kn	3000 ft	6000 ft
LMP	160 kn	180 kn	3000 ft	3000 ft
Final[a]	160 kn	180 kn	0 ft	3000 ft
Runway[a]	160 kn	160 kn	0 ft	0 ft

[a] Not to be controlled by the human operator as outside controlled sector

3.4. Coloured-Petri-Net-Based Model of Task Environment

This section presents the CPN model which was realized as simulation engine for the simulation environment MAGIE. It describes how the events and also the continuous dynamics are modeled. The relation of this model and the resulting action space to the SOM approach is discussed for the first time.

The state of the individual aircraft constitute the state of the simulation in MAGIE. This state is changing dynamically as a matter of time and as a consequence of the operators' decisions. The operators' decisions are discrete changes. In contrast, the position of the aircraft changes continuously. Both changes are important for the analysis of the human behavior. The presence of both continuous and discrete dynamics is an important difference to former analyzes of human behavior in Petri-Net-based simulation environments [MOW08, GOS09] were all changes could be reduced to events. In the MAGIE environment the timing of a decision and the actual position of the aircraft play an important role for the comparison of the operators' decisions to possible alternative decision. If a clearance is given too late or too early, this can change the consequences completely, also with regard to the fulfillment of the overall objectives. Also if the separation between aircraft is slightly below or above the threshold value is an important difference. It is thus important to consider the exact position of the aircraft. For this reason, it is unavoidable to simulate the continuous dynamic behavior of the aircraft. Nevertheless, the positions of the aircraft on the radar screen change very slowly. Consequently an update of the position is necessary only each second.

Consistent with the approach discussed in section 3.2.4 the CPN model for the MAGIE simulation consists of only one place $p_{aircraft}$ to model the actual situation. This place can only contain tokens defined by the color set function $C(p_{aircraft}) = aircraft$, which represent the individual aircraft. Such a token is added to the place, when the corresponding aircraft is activated. In short scenarios (up to 10 minutes) all aircraft of the scenario are activated at the beginning. Tokens are deleted from this place when the corresponding aircraft is on the final and leaves the sector. The color set *aircraft* is a combined color set and includes amongst others a variable of the color set *aircraftstate* and a variable of the color set *aircraftclearance*. The color set *aircraftstate* includes the actual position, heading, speed, and altitude of the aircraft; the color set *aircraftclearance* includes the target values for speed, altitude, and heading. Issuing clearances is modeled as a transition $t_{clearance}$ connected to the place $p_{aircraft}$ as input and output place. The transition $t_{clearance}$ realizes active operators modeling the actions of human operators. This active operators transform a situation into a different situation. However, since realized as Petri Net, no characteristics can be added (or removed) from a situation. Instead of modeling several transitions for different types of clearances, only this transition is used, which has the three different types of clearances and their values as parameters. The changes by the given clearances are modeled as arc expressions ($aircraft \rightarrow aircraft$) at the arc connecting the transition $t_{clearance}$ with the place $p_{aircraft}$.

The continuous dynamics modeled by passive operators as internal relations of situations are discretized and realized also as transitions. These transitions are likewise connected with $p_{aircraft}$ as input and output place and update the state of the aircraft every second. Each firing of a transition updates one aircraft and simulates one second of flight. Four transitions are used to model the different behaviors of aircraft depending on their current flight mode. The transition $t_{equipped}$ is used to updates the equipped aircraft. Unequipped aircraft flying on a route (STAR, downwind, centerline, and final) are update by the transition (t_{route}). The third transition (t_{vector}) is used to update aircraft which are flying a vector. This is the case after the aircraft has been instructed to fly into a specified direction, like during the turn procedure. The fourth transition is used for the intercept process ($t_{intercept}$), which takes place when an aircraft reaches the centerline, leaves its vector, and follows the centerline. Additionally, these transition model changes of the aircraft's flight mode which initiated automatically under specific conditions. For example, the flight mode changes from "intercept" to "route", when an aircraft reaches the centerline. The dynamics modeled by these transitions are implemented as arc expressions ($e_{equipped}, e_{route}, e_{vector}, e_{intercept}$) at the output arc. Each of these four transitions has a guard function ($g_{equipped}, g_{route}, g_{vector}, g_{intercept}$) to enable only one transition for each aircraft. The guards of these functions correspond to the explicit assumption of the operators and the arc expressions realize the operator functions. The guards check the aircraft mode and are mutually exclusive, as an aircraft can only be at one mode at a time.

The Petri Net is modified to simplify the analysis. First, the color set of the place $p_{aircraft}$ is changed to contain a list of aircraft as one token $C(p_{aircraft}) = listaircraft$. Also the four transitions to model the dynamics are replaced by one transition ($t_{dynamics}$). This transition checks the mode for each aircraft and simulates the according behavior. It still realizes the internal relation of a situation and thus passive operators. Additionally, all aircraft are updated at once. The expression at this transition's output arc is a combination of the replaced arc expressions, to ensure the same behavior. It is implemented by

$$e_{dynamics} = \begin{cases} e_{equipped}, & \text{if } g_{equipped} \text{ evaluates to true} \\ e_{route}, & \text{if } g_{route} \text{ evaluates to true} \\ e_{vector}, & \text{if } g_{vector} \text{ evaluates to true} \\ e_{intercept}, & \text{if } g_{intercept} \text{ evaluates to true} \end{cases} . \tag{3.1}$$

The expressions used in the guards are applied to achieve identical calculation. Additionally, the transition $t_{clearance}$ is changed. Instead of simulating the clearances received from the GUI, the transition can generate all possible clearances.

Further elements in the original Petri Net used for the initialization and the communication with the GUI during the simulation were also removed for the purpose of the analysis. As a result of the changes, the CPN used for analysis consists in its core of one place and two transitions.

In Figure 3.5 an exemplarily cutout of a simulation sequence for the aircraft a is shown, from entering to leaving the simulated approach sector. The first situation S_1, describes the aircraft flying along the downwind after entering the sector. The dynamics are described by the arc expression e_{route}, which is shown as an internal relation and implemented by the transition t_{route}. During the downwind the clearance to decrease to 3000 ft is given. This is the manual execution of the operator $O_{alt(a,3000)}$ (as each clearance modeled by the transition $t_{clearance}$), which leads to the next situation S_2. As the operator changes only one parameter, the structure of the situation is unaltered. After that, the aircraft is instructed to turn with the operator $O_{turn(a)}$. This results in situation S_3. In this case the internal relations are exchanged, because the aircraft is in the intercept mode now. A speed clearance (modeled by $O_{spd(a,230)}$) is given during the turn resulting in S_4. This clearance does not alter the structure and just updates the current parameters. Then the aircraft reaches the centerline (S_5). This is modeled by a self-executing operator $O_{mode(a,route)}$. The aircraft is now again in its route mode and the internal relations of the situation are changed accordingly. Later, another speed clearance ($O_{spd(a,180)}$) is given, which again changes only the parameters and leads to S_6. Finally, the aircraft reaches the LMP and leaves the control zone.

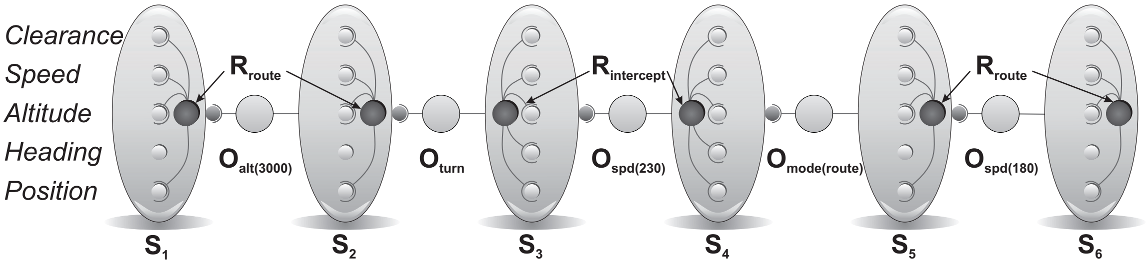


Figure 3.5.: The Situation-Operator sequence to guide an arriving aircraft along the path stretching area to the late-merging-point is depicted. The situation includes the characteristics to model the state of an aircraft. The operators O_{turn} and $O_{mode(route)}$ change the situation's internal relations.

3.5. Deduction of Rules to Model Human Operator Behavior

This section describes how the task specific set of rules is derived which is used by the planning model to generate goal-directed interactions in form of sequences of operators. The rules consist of a problem definition, which indicates the necessity to modify the operator sequence, and a modification which is applied to the already generated part of the interaction sequence. The rules model the goal-directed normative behavior of the human operators.

The problem definitions are deducted from the objectives whereas the modifications to counter the problems or derived from the set of available actions. This general approach is detailed in section 3.5.1. Subsequently, the deduction of the rules for the example application is explained. The definition of the problems based on the objectives of the operator is demonstrated in section 3.5.2. Section 3.5.3 focuses on the derivation of modifications from the available set of actions.

3.5.1. General Approach

In order to make statements about the presence of a problem a clear definition of the considered problems is required. Thus a set of problem functions P_i is needed which needs the current situation (and thus the marking of the Petri Net) as input and returns `true` if a problem i is present. Such a function can easily be constructed for objectives which describe states to be avoided. Similarly, a function can be defined for objectives which define an allowed range of values by inverting the range. If an objective defines a state to be reached (or characteristics of states to be reached) the definition of a problem function requires a greater effort. States (or characteristics), which clearly cannot lead to the desired state, have to be derived logically.

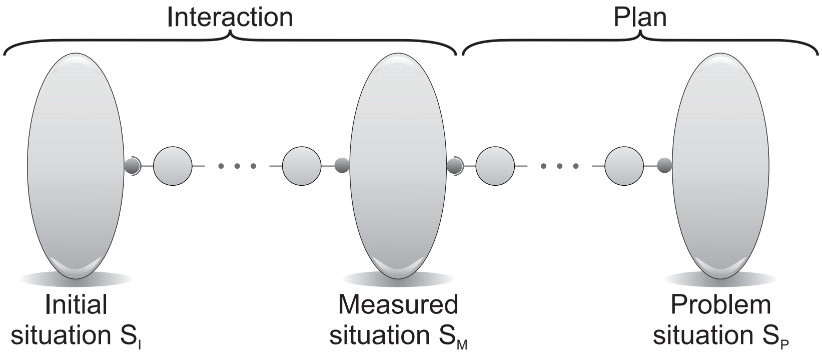


Figure 3.6.: Transformation of the initial situation S_I into the measured situation S_M by the interaction between human operator and system. The measured situation S_M is in turn transformed by the plan generated by the model into the problem situation S_P.

After the problems are derived from the objectives, a modification has to be assigned to each problem. A modification either adds an operator to the already generated interaction sequence or changes an existing operator. Out of the available operators, the operators not influencing the problem or clearly exacerbating it are excluded to find suitable operators, which can solve the problem. Problems are split into more specific problems if the set of suitable operators/modifications can be reduced for each specific problem. In the best case, exactly one modification for every problem remains.

To define problems, it is important to distinguish between the initial, the measured, and the problem situation as illustrated in Fig. 3.6. The initial situation S_I is the situation at the beginning of an interaction. After some time of interaction, the resulting situation can be observed and is called measured situation S_M. The planning model starts with that situation and generates a plan. This transforms S_M into the problem situation S_P. Some problems in the set of rules are related only to the problem situation, others are additionally related to the measured situation. Consequently, the problem functions $P_i(m_M, m_P)$ needs the marking of the CPN model corresponding to the measured situation m_M and the marking corresponding to the problem situation m_P as input.

Problems can have different priorities. Accordingly, a solution for a problem with a high priority is allowed causing problems with lower priority if they are unavoidable. If these lower priority problems are detected later, the operator sequence cannot be changed again as this causes the higher priority problem again. The priority assigned to each operator avoids such circles. The operator sequence is changed only, if the priority of the modification is higher than the priority of all operators affected by that modification.

3.5.2. Deduction of Problem Definition in MAGIE as Example

The definitions of the problems are derived from the objectives of the human operator. In case of MAGIE, these are the objectives *separation*, *constraints*, and *throughput*, which are defined in the instructions. As the problems have not been defined yet, they are derived in the following.

The objective *separation* leads straightforward to the problem function P_S which evaluates to `true` if the distance between two aircraft is lower than the separation minimum of three nautical miles. As the objective *constraints* defines states to be avoided, the allowed ranges for speed and altitude are inverted into forbidden ranges. This results in the problem functions P_{speed} and P_{alt}.

As the objective *throughput* is defined in a positive way, a transformation into an equivalent negative definition is necessary. A high *throughput* can only be reached if the aircraft are as close as allowed when reaching the runway. Consequently, the aircraft should initialize the turn maneuver, as soon as a trajectory fulfilling the other objectives would result. This is equivalent to avoid flying along the downwind ($P_{downwind}$) if possible. Additionally, the aircraft should fly as fast as allowed by the constraints (P_{toSlow}). Another aspect of the objective throughput is that the aircraft have to intercept the centerline east of the LMP . Thus, aircraft which reach a line, orthogonal to the centerline and intersecting the late merging point, without being on the centerline, are also defined as a problem ($P_{missLMP}$). It is effectively to decrease the altitude as early as possible to avoid problems later. Aircraft not satisfying this condition are also considered as a problem (P_{toHigh}). Thus four problems according to the objective throughput are defined: aircraft flying along the downwind, aircraft slower than the maximum speed, aircraft not meeting the LMP, and aircraft not decreasing altitude as early as possible. All defined problems are given in the second column of Table 3.2, which also shows the derived set of modifications detailed in the following.

3.5.3. Deduction of Modifications in MAGIE as Example

Additional to problems, the corresponding modifications have to be derived, which is done in the following. Some characteristics of MAGIE simplify the deduction of modifications, as often only a limited set of action is reasonable to solve a problem. For example, the aircraft's route over ground is only affected by the turn maneuver. Furthermore, the aircraft's speed and altitude do not interact so that decreasing the altitude does not increase the speed. Instead, aircraft's speed changes with a constant deceleration or acceleration and altitude changes with a constant angle and thus is proportional to the vertical length of the trajectory. Additionally, conflicts depend only on the lateral position, which can only be influenced by the turn maneuver or by speed changes. Because of the route structure it is assumed that the vertical position of the aircraft is similar and cannot be used for their separation. Furthermore, the aircraft can be considered separately as the same rules apply to each aircraft.

First, it is important to assign specific priorities to each modification. In agreement with the instructions, modifications made for *separation* have the priority 3, modifica-

Table 3.2.: Complete set of rules deducted from the objectives and the available actions (A less detail version can be found in [HS14])

Objective	Problem		Problem definition			Modification			
			Route	Other AC	Additional condition	Modificator	Operator	Time	Priority
Separation	P_S	P_{S1}	Fixed	Behind	P_{S2} occurred & Spd = spd_{PS2}	Move	$O_{spd(x,spdPS2)}$	Later	3
		P_{S2}	Fixed	Behind		Add	$O_{spd(x,max)}$ + $O_{spd(x,spdPS2)}$	Earliest	4/3
		P_{S3}	Fixed	Ahead	P_{S4} occurred & Cl. spd = spd_{other}	Move	$O_{spd(x,spdother)}$	Later	3
		P_{S4}	Fixed	Ahead	Ongoing conflict	Add	$O_{spd(x,160\,\text{kn})}$ + $O_{spd(x,spdother)}$	Earliest	4/3
		P_{S5}	—	Ahead	On final	Add or move	$O_{spd(x,spdother)}$	≤ Now	3
		P_{S6}	Fixed	Ahead	—	Add or move	$O_{spd(x,spdother)}$	≤ Now	3
		P_{S7}	Variable	—	—	Move	$O_{turn(x)}$	Later	3
		P_{S8}	Downwind	—	P_{S9} occurred & Cl. spd = 250 kn	Move	$O_{spd(x,250\,\text{kn})}$	Later	3
		P_{S9}	Downwind	—	Spd < 250 kn	Add	$O_{spd(x,180\,\text{kn})}$ + $O_{spd(x,250\,\text{kn})}$	Earliest	4/3
			Position		Additional condition				
Constraints	$P_{downwind}$		On downwind		Turn not in sequence	Add	$O_{turn(x)}$	Now	1
	P_{toSlow}	toSlowBD	On base or downwind		Cl. spd < 250 kn	Add	$O_{spd(x,250\,\text{kn})}$	Now	1
		$P_{toSlowc}$	On centerline		Cl. spd < 230 kn	Add	$O_{spd(x,230\,\text{kn})}$	Now	1
	$P_{missLMP}$		North or south of LMP		Variable route	Move	$O_{turn(x)}$	Later	3
	P_{toHigh}	$P_{toHighD}$	On downwind		Cl. alt > 5000 ft	Add	$O_{alt(x,5000\,\text{ft})}$	Now	1
		$P_{toHighBC}$	On base or centerline		Cl. alt > 3000 ft	Add	$O_{alt(x,3000\,\text{ft})}$	Now	1
		$P_{toHighLMP}$	At LMP		Alt > 3000 ft & variable route	Move	$O_{turn(x)}$	Later	3
Throughput	P_{speed}	P_{speedC}	On centerline		Spd > 230 kn	Add or move	$O_{spd(x,230\,\text{kn})}$	≤ Now	2
		P_{speedF}	On final		Spd > 180 kn	Add or move	$O_{spd(x,180\,\text{kn})}$	≤ Now	2
	P_{alt}		On trombone		Cl. alt < 5000 ft	Add	$O_{alt(x,5000\,\text{ft})}$	Now	2

spd = speed, alt = altitude, cl. = cleared

Table 3.3.: The four priority groups for the defined set of rules

Priority	Group
4	Fixed
3	Separation Throughput (only turn)
2	Restrictions
1	Throughput (except turn)

tions for *constraints* have the priority 2 and modifications for *throughput* have the lowest priority 1. Furthermore, the priority of 4 is defined for modifications which are fixed and cannot be affected even by modifications due to *separation*. As modifications due to the objective *constraints* should not be able to modify the turn (it is possible to fulfill the *constraints* independently of the turn), but modifications due to the objective *separation* must be able to, a further category is defined for modifications of the turn to increase the *throughput* with priority 3. An overview of the resulting priorities is given in Table 3.3.

To reduce the set of possible solutions, P_S is split into the nine specific problems P_{S1} to P_{S9}. One important difference concerns the aircraft's route. It is distinguished, whether the aircraft is on the downwind, if it started the turn but its route can be modified, or if the route is fixed and only speed modifications are available. A fixed route results, if the turn maneuver was already implemented in the measured situation, and a variable route results, if the turn maneuver is a part of the plan (s. Fig 3.6). In case of variable routes, the turn maneuver can be modified, in case of fixed routes or if the aircraft is on the downwind, avoiding conflicts is possible only by changing the speed of an aircraft.

The second difference between problems takes advantages of considering the aircraft separately. Thus, the relative position of the considered aircraft to the other aircraft in a conflict can be used to differentiate between problems. Consequently, it is possible to distinguish if increasing the speed may solve the conflicts, if the other aircraft is behind, or reducing the speed may solve the conflict if the other aircraft is in front.

Conflicts of aircraft with a fixed trajectory and another aircraft behind constitute the problems P_{S1} and P_{S2}. These problems can be solved only by increasing the speed. However, aircraft are already as fast as allowed to maximize *throughput*. As conflict avoiding has a higher priority than constraints, increasing the speed above the allowed limit is acceptable here. Consequently, a speed increase will be added to the operator sequence if problem P_{S2} occurs. However, the speed has to be reduced after the conflict is solved and the modifications activated by problems regarding the constraints will not be allowed overriding speed changes added to solve conflicts as conflicts have a higher priority. Therefore, a speed reduction is additionally added to the operator sequence when P_{S2} occurs. If the conflict is not solved by the modification and reoccurs later, which is defined as P_{S1}, this speed reduction is delayed to increase the duration of the aircraft flying at maximum speed. The speed increase has the priority of 4 as it should not be modified again (especially not by P_{S1}), the speed reduction has the priority of 3 as it is related to *separation*.

The next two problems describe conflicts with a fixed trajectory and with the other aircraft ahead. This conflicts can be solved by reducing the speed. When problem P_{S6} occurs, it is sufficient to reduce the speed some time before the conflict occurs to the speed of the other aircraft ahead. However, if a conflict is already present in the measured situation, which is defined in P_{S4}, the speed has to be reduced to the minimum possible speed to increase the distance to the other aircraft as quickly as possible. For the same reasons as in the case of P_{S2}, a second speed change is added after the first occurrence of this problem and is postponed, as long as the problem is not solved. This is defined in P_{S3}. Again, the modification have the priority 4 and 3.

If an aircraft arrives on the final without a conflict, the turn maneuver started at the right moment. Thus, conflicts on the final included in P_{S5} are solved by changing the speed. If the other aircraft is ahead, this is assigned to, otherwise this conflict is classified as P_{S1}. The problem P_{S7} includes aircraft in conflict with a variable trajectory. In this case, the conflict is caused by starting the turn at the wrong time. It can be solved by changing the moment of the turn and thus by stretching the length of the aircraft's route. As the turn is planned as early as possible, the only available option is to delay it.

The last two problems describe conflicts while the considered aircraft is on the downwind. Here the speed is first reduced to the minimum and then increased. Similar to P_{S2}/P_{S1} and P_{S4}/P_{S3}, this problem is split into P_{S9} and P_{S8}.

The two problems $P_{downwind}$ and $P_{missLMP}$ regarding the objective *throughput* can be solved only by changing the route. This in turn can only be archived by adding the turn operator to the planned sequence respectively by delaying it. The other problems resulting from the objective *throughput*, P_{toSlow} and P_{toHigh}, are divided dependent on the position of the aircraft.

To reduce the altitude as early as possible, three rules are defined. One rule decrease the altitude as soon as the aircraft reaches the downwind (and is thus in the control zone) and the second rule decreases the altitude as soon as the aircraft reaches the base-leg (and thus a further reduce is allowed). As the altitude is already decreased as early as possible, the altitude at the late-merging point can only be reduced by increasing the length of the route (which means to delay the turn). This is problem $P_{toHighLMP}$. If an aircraft is not flying with the maximum allowed speed, called P_{toSlow}, the speed has to be increased depending on the current route section of the aircraft. Both modification of the turn, $P_{missLMP}$ and $P_{toHighLMP}$, have the priority 3 so that they can not be affected due to *constraints*.

The problems resulting from the *constraints* are also divided according to the aircraft positions. As the rules resulting from *throughput* are stricter regarding speed increase and altitude reduction, only the opposite rules P_{speedC}, P_{speedF}, and P_{Palt} have to be defined for the objective *constraints*. Thus, speed is reduced if it is above the allowed maximum. Further ,the altitude is increased if it is below the allowed value. As the lowest altitude clearance is 3000 ft which is the minimum on the centerline and during the turn, an altitude below the allowed minimum is only possible on the downwind and only one modification (to climb) has to be defined. The complete set of deducted modification is given in Table 3.2. These set of rules, consisting of a problem condition and a modification, describe the normative behavior of human operators interacting with MAGIE. It can be applied by the cognitive planning model described in the next section to generate interaction sequences to reach goal situations fulfilling the objectives.

3.6. The New Cognitive Planning Model

In this section, the developed cognitive planning model to generate a goal-directed operator sequence to be executed by a human operator interacting with a complex dynamic system is described. As human operators of such systems have many options to interact, innumerable situations may result [MS05]. When the action space of the human operator is analyzed, the state space explosion problem [JK09] occurs. Accordingly a complete action space of a complex dynamic simulation (like MAGIE) cannot be calculated and only partial action space for situations of interest can be analyzed, as proposed in [GOS09]. The aim of the developed planning model is to reduce the action space of a dynamic system to the partial action space by finding the relevant parts.

The steps executed by the developed planning model are illustrated in Fig 3.7. At first, the simulation protocol is loaded which describes the recorded/measured interaction of the human operator as a sequence of states. This is interpreted by the model as a sequence of situations and operators. Then the model focus on specific aspects of a situation, called focused situation, to simplify later steps. A focus situation contains only some characteristics of the original situation. In the example, a focus situation considers only certain aircraft. Focusing only on some characteristics of a situation assumes that no relations connect the focused characteristics to the other characteristics. In the following step, the main planning process starts using the set of rules modeling the operators' behavior and the CPN model of the controlled system. Each planning process loads an initial focused situation and simulates the system behavior (modeled as internal relations of the situation) and the execution of planed actions (modeled as external operators). The planning is repeated for every focused situation of the initial situation to generate individual plans. In every repetition, the focused is extended so that every time more characteristics of the original situation are considered. The sequences generated for former focused situations are considered during the simulation of the planning process in further repetitions, to allow the detection of problems which affect the complete situation. However, the formerly generated sequences are not considered for the application of rules. Thus, they cannot be modified during latter planning processes and these sequences cannot hinder the activation of rules. During the repetition, the focus is extended so that finally an interaction sequence results which can transform the original situation into a goal situation.

The planning model uses the modified Petri Net for this purpose. A direct access to get the actual state and possible successor states is needed. This is realized by the Access/CPN [WK09] interface. Using the direct access to the net, an arbitrary marking (denoting an arbitrary situation) can be defined to start the simulation. Consequently, the marking can be manipulated in a way that only a focused situation is loaded.

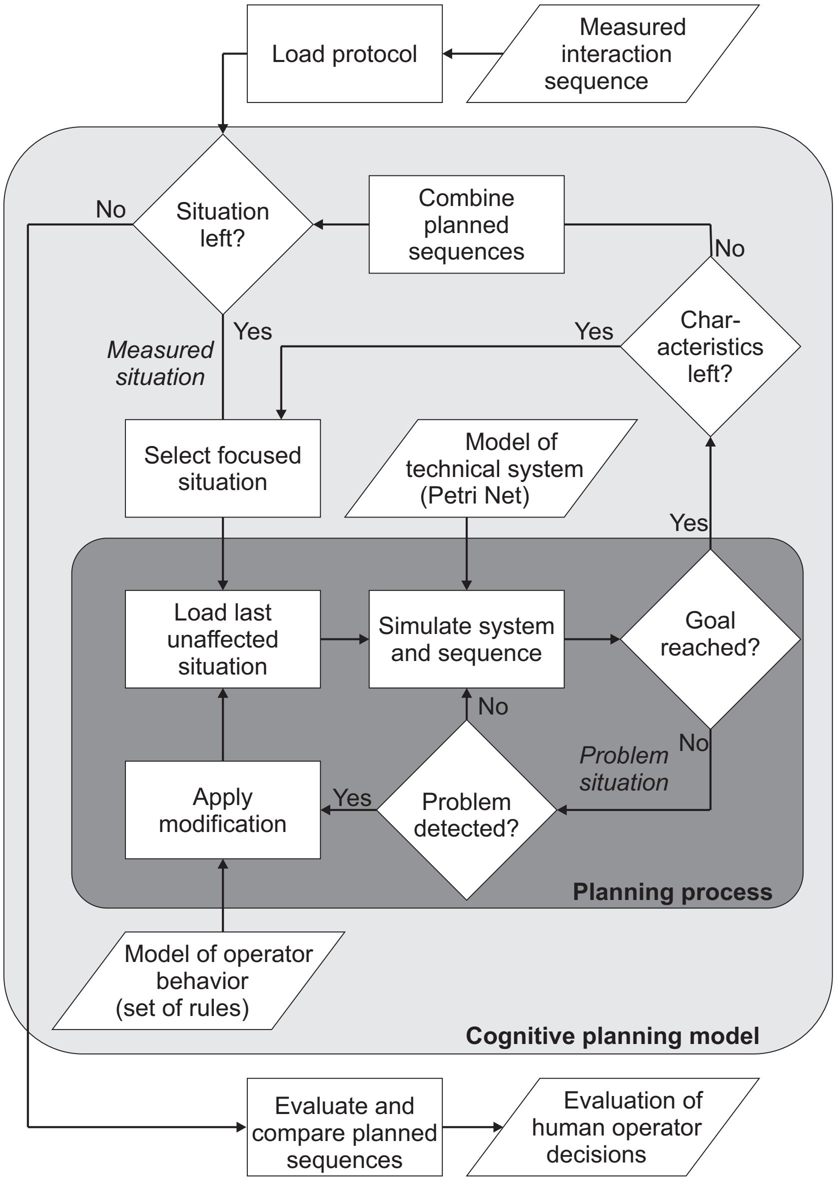


Figure 3.7.: Processes in the cognitive planning model [HS14]

3.6.1. The Planning Process

The planning process generates a goal-directed interaction sequence by adding one operator after the other. This process consists of three steps and uses the Petri Net of the operated system to predict the future states and problems of the systems and the set of rules described in the last section.

The planning process starts with one situation loaded into the Petri Net. The necessary modification of the Petri Nets marking is possible with the direct access provided by Access/CPN. Then the internal dynamics of the system is simulated. The simulation is done by using the direct access to the Petri Net and firing the transitions which implement the relations included in the loaded situation. If a sequence is already generated, the operators in that sequence are executed during the simulation. All states reached during the simulation are checked for problems defined in the set of rules by using the problem functions $P_i(m_M, m_P)$. If the state meets the problem condition of one rule, this rule will be activated.

The activation of a rule will cause a modification of the operator sequence. This is the third step of the planning process. The modification either adds a further operator or changes the execution time of an existing operator. If an operator is modified, the following operators are deleted because the modified operator may change the systems states and the potential problems after its execution. Consequently, the problems which initially caused the deleted operators to be added may not occur after that modification.

After the modification, the simulation is reset to the latest state which is not effected by the modification. This means that the situation is loaded into the Petri Net, to which the added operator is applied. In case an operator is delayed, the situation to which the operator was applied before the modification is loaded and in case an operator is executed earlier, the situation to which the operator is applied after the modification is loaded. These three steps are repeated until an operator sequence is found which is suited to transform the initially loaded situation into a goal situation.

The modifications, which bring an operator forward or postpone it, will be applied step by step until either the problem is solved or the operator will be applied as early as possible. In the latter case, the problems, which require a further modification of this operator, are ignored during the further procedure as they cannot be solved and are unavoidable.

An exemplary planning process to generate a goal-directed sequence is illustrated in detail in Fig. 3.8. In this example, the steps are repeated in four iteration cycles. In the upper part of the figure the state of the Petri Net is shown and in the lower part the corresponding action space is shown represented in SOM symbolic. The planning process starts with the first prediction from state 0 at time $t = 0\,\mathrm{s}$ and simulates the system until a problem is detected in state 4 ($t = 4\,\mathrm{s}$). As the transitions from 0 to 4 are only caused by internal dynamics, they are all described by situation $S1$ and the changes of the state are described by internal relations of the situation. As a problem defined in the set of rules is detected in state 4, the operator sequence is modified. In this example the operator O_1 is added at time $t = 2\,\mathrm{s}$. Consequently state 2, which is just before the execution of operator O_1, is loaded into the Petri Net. During the next run, the simulation starts at state 2 and applies the operator in the calculated interaction sequences at first and then simulates the internal dynamics. The resulting states $(5-9)$ are described by the situation S_2. To solve the next detected problem, a second operator is added to the sequence (O_2). During the third run of the simulation (from $t = 4\,\mathrm{s}$ to $t = 7\,\mathrm{s}$, resulting in the states $10 - 13$), the second operator O_2 is simulated and a third problem is detected in state 13. This is solved by a modification of the second operator. The time of its execution is decreased by one second from $t = 4\,\mathrm{s}$ to $t = 3\,\mathrm{s}$. Consequently, the fourth run will start at $t = 3\,\mathrm{s}$ from state 6. This simulation run leads to state 18, which is a goal state. Thus the resulting operator sequence $O_{1(t=2\,\mathrm{s})}, O_{2(t=5\,\mathrm{s})}$ is a goal-directed sequence.

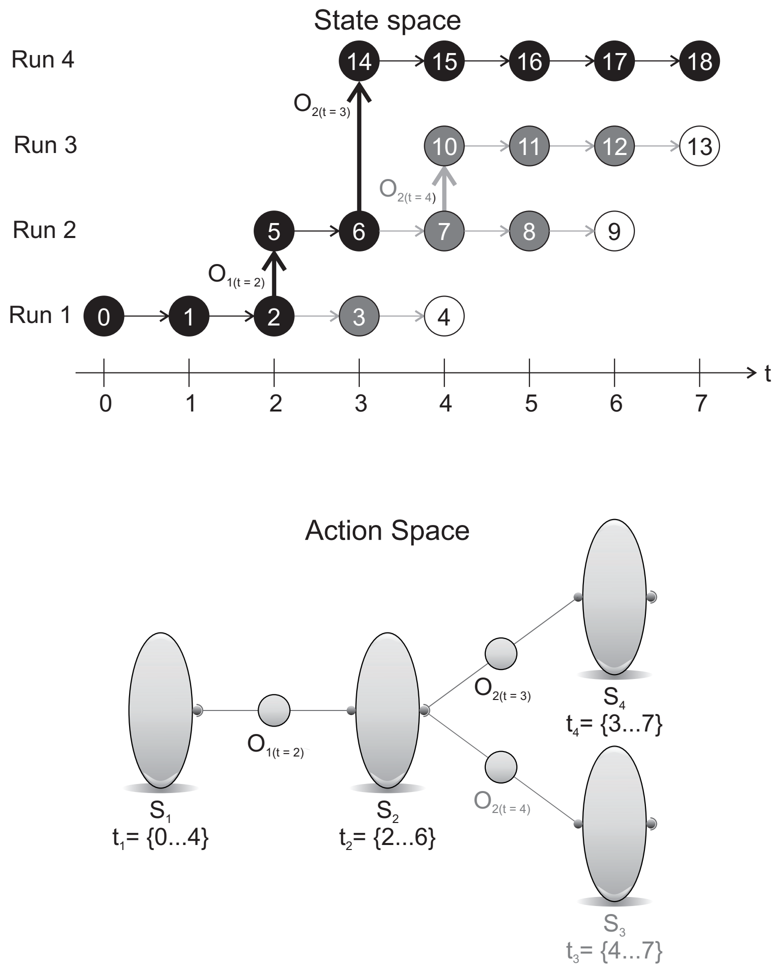


Figure 3.8.: Exemplary result of a planning process. An interaction sequence is calculated (at the bottom) by analyzing the state space (at the top). States and situation included in the final sequence are labeled in black. States and situations withdrawn during the calculation are labeled in gray. White states indicate the presence of a problem [HS14, HS13b].

3.6.2. Application of the Planning Model to MAGIE

In the application example MAGIE, a situation contains all aircraft currently in or near the controlled sector. As there are no relation between the individual aircraft, it is possible to focus on some of the available aircraft. In the first run of the planning process, only one aircraft is focused. When this focused situation is loaded into the Petri Net, the other aircraft are deleted for the list to be placed on $p_{aircraft}$. Consequently, the planning process considers only one aircraft. The planning process is repeated for each unequipped aircraft present in the measured situation. Thereby, the focus is extended in each repetition and one additional aircraft is considered during the planning process.

The system is simulated by firing the transition $t_{dynamics}$. If some operators are already in the planned interaction sequence, the transition $t_{clearance}$ is fired to simulate these. If one of the 19 defined problems is found during simulation, the corresponding modification will be activated. The simulation stops if a goal is reached and all aircraft have arrived at the runway. During each run, only operators for the just added aircraft are planned but the resulting operator sequence contains operators for all focused aircraft.

As the aircraft are not independent of each other aircraft and the sequences of other aircraft are considered during the planning process, the order in which aircraft are added to the focused situation is important. Either a first-come-first-serve order can be used or in the contrary extreme all different orders can be calculated.

If x unequipped aircraft are present during a simulation and the first-come-first-serve rule is applied, one sequence is generated running the planning process x times. Overall $x!$ possible orders exist, so that the planning process has to be executed $x \cdot x!$ times if all orders are analyzed. When 4 unequipped aircraft are present at once, 24 sequences are calculated and the planning process started 96 times. Consequently, the calculation of all possible sequences becomes very extensive if x increases.

3.7. Results

The planning model is demonstrated with the situation illustrated in Fig. 3.9. In this figure a part of the controlled sector including the LMP and its surroundings is shown. In this situation reached at $t = 200\,\mathrm{s}$ after the start of the simulation, the four unequipped aircraft U90, U74, U23, and U48 are present (U48 is on an arrival route in the south a not visible in the shown part of the GUI). A plan will be calculated to transform this situation into a goal situation in which all aircraft have reached the runway. For this example situation 24 different orders of aircraft are calculated in 96 runs of the planning process. Some exemplary sequences are reported to clarify some important aspects of the algorithm.

During the planning process for U90 the following modifications are applied to solve the detected problems. At first the problem $P_{toSlowC}$ is detected immediately, as the aircraft is not flying with the maximum allowed speed of 230 kn. Consequently the operator $O_{spd(U90,230)}$ is added. At $t = 246\,\mathrm{s}$ this aircraft is too close to A57. This problem belongs to category P_{S6}. The speed of U90 is reduced to the speed of A57, which is 170 kn, by $O_{spd(U90,170)}$. The speed reduction is moved backwards step by step to solve the problem. When it is applied at $t = 218\,\mathrm{s}$, the separation will still fall below the minimum, but the speed of A57 has changed to 160 kn in the meanwhile. Accordingly, the operator $O_{spd(U90,160)}$ is added. After this operator has been moved to $t = 215\,\mathrm{s}$, which deletes the operator $O_{spd(U90,170)}$, the problem is solved and U90 reaches the end of the sector as fast as possible (according to the *constraints*) and without a violation of the minimal separation. This final sequence is given in Table 3.4. To generate this sequence, 2501 states are calculated.

Then the operator sequence for U74 is calculated under the assumption, that the above described sequence for U90 is implemented. Here it is detected that U74 can decrease its altitude and thus the according operator $O_{alt(U74,5000)}$ is added. Further the turn-

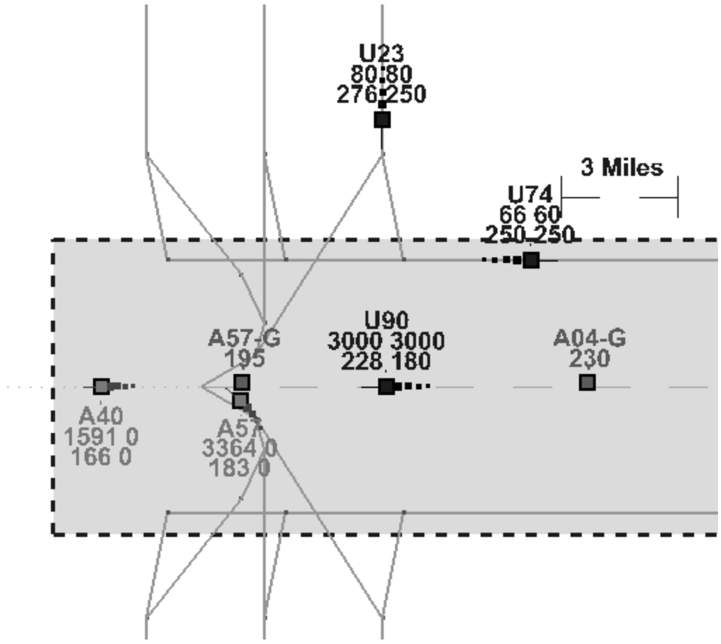


Figure 3.9.: Section of the MAGIE GUI showing the traffic close to the Late-Merging-Point after 200 seconds of interaction

operator $O_{turn(U74)}$ is added to keep the trajectory as short as possible. Additionally, a descent to 3000 ft is given. When U74 reaches the centerline, its speed is above the limit (P_{speedC}). Thus, the operator $O_{spd(U74,230)}$ is added to the sequence sufficiently early. Following this trajectory, U74 is too near to A04. As the trajectory of U74 is not fixed (P_{S7}), the turn is postponed to solve this problem. The operators after the turn are consequently deleted but added again in the further calculations. When $O_{turn(U74)}$ is executed at $t = 210$ s, the U74 and A04 will firstly be too close to each other on the final and thus this is solved by speed modifications. Therefore, a speed reduction to 170 kn is added and moved to $t = 368$ s. Finally, another speed reduction to 160 kn at $t = 420$ s solves the problem and completes the sequence. This sequence is also given in Table 3.4. To generate this sequence, 3075 states were calculated.

This procedure is repeated 96 times to calculate the sequences in all 24 different orders. To generate the combined sequence for the order U90, U74, U23, and U48, additionally the two sequences for U23 and 48 are calculated. Afterwards the four sequences are combined to the result given in Table 3.5. This sequence transform the given situation at $t = 200$ s to a goal situation, in which all aircraft have left the sector and the three objectives *separation*, *constraints*, and *throughput* are fulfilled.

3.8. Concluding Remarks

The developed model of cognitive planning allows generating plans, which are sequences of operators from the Situation-Operator-Modeling point of view and firing sequences from the Petri Net point of view. To generate these sequences, a model of the con-

Table 3.4.: Operator sequences for the unequipped aircraft U90 and U74 calculated individually for both aircraft

Aircraft	Operator		Time
	Characteristic	New value	
U90	Speed	230 kn	201
U90	Speed	160 kn	215
U74	Altitude	5000 ft	201
U74	Start turn		210
U74	Speed	230 kn	284
U74	Speed	170 kn	340
U74	Speed	160 kn	420

Table 3.5.: Combined operator sequence for four unequipped aircraft integrated in the order U96, U54, U92, U81.

Aircraft	Operator		Time
	Characteristic	New value	
U90	Speed	230 kn	201
U74	Altitude	5000 ft	201
U74	Start turn		210
U90	Speed	160 kn	215
U23	Altitude	5000 ft	263
U74	Speed	230 kn	284
U23	Start turn		328
U74	Speed	170 kn	340
U48	Altitude	5000 ft	344
U23	Speed	230 kn	402
U74	Speed	160 kn	420
U48	Start turn		483
U23	Speed	170 kn	486
U48	Speed	230 kn	557
U23	Speed	160 kn	566
U48	Speed	170 kn	624
U48	Speed	160 kn	709

trolled system, realized as CPN, and a model of the normative behavior of the operator, implemented as a set of rules, is applied.

Using the generated interaction sequences as criterion, human operators behavior can be compared to a situation dependent criterion, which allows a more precise assessment of the performance than a fixed criterion. Furthermore, the interaction sequence represents which goals are still reachable and to what extend the objective can be fulfilled. Thus it reflects the possible options of the operator. Consequently, it is possible to evaluate a decision as poor if the operator actually had the opportunity to make a better decision. The other way around, decisions can be evaluated as good even if the fulfillment of the objectives is low if the operator did not have a chance to prevent this. Being able to define those situations in which human operators are more likely to get problems is the first step to improve HMS, procedures, or training in the future.

Former approaches using a Petri-Net-based analysis to evaluate human operators either used fixed interaction sequences in time-dependent dynamic systems or situation-specific interaction sequences in event-discrete systems. This chapter presented a cognitive planning model which allows for the first time generating situation specific goal-directed interaction sequences in time-depended dynamic systems to model the behavior of human operators using a Petri-Net-based analysis.

Additionally, the developed planning model was applied to a dynamic example application. The applicability of the developed model and the correct operation of the implementation could be demonstrated.

As the calculated solution is based on the set of rules, the quality of the solution depends on the completeness of this set. Therefore, much effort was put in the deduction of the rules. However, this set has to be validated in further studies. This could be achieved by comparing the interaction sequences generated by the cognitive planning model to the decisions of the human operator. Decisions of human operators with a higher fulfillment of the objectives than the interaction sequence of the model would indicate an incomplete or incorrect set of rules.

The generated interaction sequences are based on exact predictions and assume a very precise implementation of the planned actions. Human operators, on the other hand, are not able to reach this precision when predicting consequences and implementing actions due to their cognitive limitations. To allow a more precise prediction of human operator behavior, the effect of prediction uncertainty is included in the planning model in the next chapter.

4. Modeling of Uncertainty in Human Operator Planning

The Petri Net model of the system controlled by human operators, which is applied by the cognitive model of human operator planning developed in the previous chapter to predict the consequences of actions, is extended to include prediction uncertainty in this chapter.

Uncertainties are often present when making decisions and influence the performance of human operators. By integrating uncertainty into the cognitive planning model, more realistic predictions of human behavior become possible. The consequences of interaction sequences generated by a model containing uncertainty should less frequently deviate from the measured consequences of human operators' decisions than interaction sequences generated ignoring uncertainty. When fewer or smaller deviations occur, a part of the measured lack of operators' performance can be explained by prediction uncertainty caused by human operators' mental models. Furthermore, sequences generated with prediction uncertainty can be used to differentiate between deviations from the optimal performance caused by prediction uncertainty and erroneous actions due to other reasons.

In this chapter, first the motivation to integrate uncertainty into cognitive models is explained in more detail (see section 4.1). Then the term uncertainty is defined and issues of uncertainty are discussed (see section 4.2.1). Thereupon, the different perspectives of uncertainty in HMI are analyzed as well (see section 4.2.2). These distinction are combined to a classification scheme of uncertainty in section 4.2.3. Subsequently, the different approaches to model uncertainty and their appropriateness for the described problem are examined (see section 4.2.4). Based on this classification, prediction uncertainty is integrated in Colored-Petri-Net-based models (see section 4.3). Some results on predictions modeled with uncertainty are reported in section 4.4. At the end of this chapter, some conclusions are drawn (see section 4.5).

Some ideas regarding the implementation of uncertainty into the developed human operator planning model have already been published in [HS14].

4.1. Need for the Integration of Uncertainty into Human Performance Models

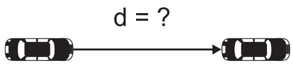
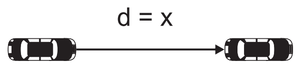
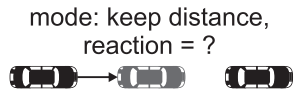
Recently, a lot of effort was made for the development of models of human cognitive performance. The purpose of these models are diverse. One purpose stemming from the engineering science is the evaluation of HMS reducing the need for human-in-the-loop simulations including human operators interacting with these systems. Consequently, the required effort and time for the development of HMS can be reduced. In this field, the aim of the models is to reproduce human performance. Another purpose is to enhance the understanding of human cognition and comes from psychology. These models must not only reproduce human performance but model detailed aspects of human cognition to focus specific cognitive functions. A further example is the evaluation of human cognitive performance. Here the models' predictions are compared to measured operators behavior (e.g. [OHS11]). Consequently, these models do not have to model cognition in detail but must be able to predict human performance. This category includes in particular models of human decision making and applications of the signal detection theory (see section 2.3.1), the lens model (e.g. [Coo96, BKG+00]), and rational decision making (see section 2.1.5).

As detailed in section 2.1.5, humans normally do not make optimal decisions especially when interacting with complex dynamic systems. Making optimal decision is very costly in terms of effort and time, as it requires first to detect all different options, to simulate this options mentally, to evaluate their consequences, and to select the best option. Especially when interacting with dynamic systems, optimal behavior is usually not possible as the time constraints of the system require quick decisions and the available time is not sufficient to follow an optimization procedure. Although, a lot of models use optimal decisions as a standard.

As humans often do not apply optimization methods, a model making optimal decisions is not able to predict the behavior of human operators. Such models should not be used during training or as basis for technical assistance. Human operators would be compared to a formal standard they cannot hold. If they want to conform to the model, they have to adapt their working methods which can result into performance decreases due to insufficient time or cognitive resources for making optimal decisions [Kle01]. If optimal models are used as assistance, the actions advised by the assistance will probably differ to those preferred by the human operator and the operator will not be able to follow the assistance. Consequently, the human operator can either trust the systems without question it decisions. This problem is called overreliance [PR97]. In other cases, the human operator can ignore the systems and make it useless. Further, he can adapt his working methods with the above mentioned consequences. Models of human performance should therefore not provide optimal solutions but realistic solutions taking into account human limitations. As decisions often have to be made under uncertainty (e.g. uncertain information or uncertain predictions) and uncertainty strongly affects decisions, it should be integrated into models of human performance. To make the cognitive planning model developed during the last chapter applicable for training purposes, for example to use the generated interaction sequence in a discussion with a human operator who selected another option, it should not provide an optimal solution in every case but one that is expectable and realistic taking into account the human cognitive limitations and the effect of uncertainty.

Uncertainty changes by integrating technical assistance. In the real world, fallible indicators are often connected to continuous uncertainty, for example, the distance to an obstacle. This uncertainty can often be engineered out by assistance systems, for example, by precise measurements. In contrast to that, often discrete states and precise indicators are present, for example, when a system is in a specific mode [DSK99].

Table 4.1.: Reduction and induction of different types of uncertainty by technical assistance

	Task assignment	
	Human operator	Technical assistance
Uncertainty reduction due to information gathering by technical assistance		
Uncertainty induction due to decision making by technical assistance		

When assistance systems not only take over the task of information gathering and interpretation but also the task of decision making and implementation, uncertainty about consequences of decisions made by technical assistance, and thus uncertainty about future system states, can be induced by the assistance. In other word, even if the mode of operation is indicated precisely, the consequences of this mode and the reaction of the system to specific events can be unclear to the operator.

These different cases are illustrated in Table 4.1 with the example of an automated distance control in a car. In the upper row, the reduction of uncertainty by technical assistance is shown. In the lower row, the induction of uncertainty is depicted. On the left hand side, the human operator has to carry out the tasks, on the right hand side, the tasks are executed by a technical assistance. Consequently, the driver has to estimate the distance to the car ahead in the upper left corner. The uncertainty can be reduced, if the distance is measured more precisely by assistance (upper right corner). When another car drives into the gap, a decision is necessary to avoid a collision (lower left corner). If this is automated, the drivers may not know what behavior of the technical assistance to expect and uncertainty is induced (lower right corner).

The different kinds and forms of uncertainty will be discussed in more detail later. It should be noted here that a model can not ignore these uncertainties nor calculate optimal solutions integrating these uncertainties to make realistic predictions of human cognitive performance.

A frequently chosen approach for modeling human behavior is to provide the required procedural knowledge in form of if-then rules. The development of the human cognitive planning model also follows this approach. Such models aim to mimic human behavior and not to find always the optimal solutions. One widespread example is the cognitive architecture ACT-R which stores the procedural knowledge as production rules [ABB$^+$04]. Similarly, the cognitive planning model develop in this thesis also applies a set of rules to model procedural knowledge. If integrating uncertainty in such models, the arising questions are about the combination of the different kinds of uncertainty with the procedural knowledge in form of if-then rules.

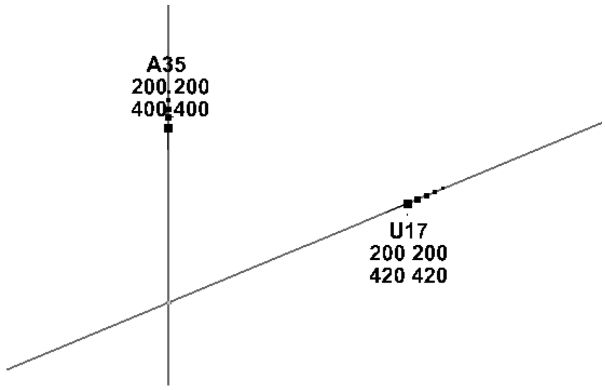


Figure 4.1.: Situation with two approaching aircraft on a radar screen. As the aircraft's routes are intersecting and the aircraft will reach the intersection at the same time, the aircraft are in a conflict.

Similar to other CPNs-based approaches, which calculate the available options with the aim to measure the human operators' task performance by contrasting these options to the human operators' behavior and decisions [OHS11, HS13b], the cognitive planning model developed in the last chapter does not considered uncertainty. This chapter presents an approach to integrate uncertainty into human cognitive performance models, which store the procedural knowledge in form of rules, to achieve more realistic behavior. This approach is used to extend the cognitive planning model developed in the last chapter. The focus lies on prediction uncertainty generated by imprecise predictions of future states by the human operator.

4.2. Classification of Uncertainty

Uncertainty has many aspects and is much more than the probability of a specific outcome. To demonstrate this, different kinds of uncertainty will be analyzed and classified in this section. This will be illustrated with a striking example from the ATC domain.

In ATC, one important task is to ensure a safe separation between aircraft at all time. Aircraft are separated, if the vertical distance between the aircraft is at least 1000 ft or if the horizontal distance is at least 5 NM (see section 2.2). The example used here is a situation with two aircraft on conflicting routes which is shown in Fig. 4.1. In this situation, the task of the ATCO is first to estimate if the minimum separation between the aircraft will be maintained and second to choose the appropriate action in case of violations. The position of the aircraft is given by the corresponding dot on the radar screen. The measured speeds and altitudes of the aircraft are displayed as numbers in the individual aircraft's label.

4.2.1. Issues of Uncertainty

Something is uncertain if it is "not able to be accurately known or predicted" [Har13]. This definition includes two interesting aspects. First, the definition emphasizes that

Table 4.2.: The three different issues of uncertainty

Issue	Uncertainty
Situation	State uncertainty
Options	Selection uncertainty
Consequences	Prediction uncertainty

uncertainty can be related to both knowledge as well as predictions. Secondly, it is highlighted by the insertion of the word "accurately" that uncertainty does not describe that something is completely unknown but only a lack of precision.

In the following the Situation-Operator-Modeling approach (see section 3.2.2) is used to state uncertainty more precisely. Either specific characteristics of situations can be subject to uncertainty or the situation as a whole is subject to uncertainty. For example, a human operator is interacting with a system, which is in a specific operation mode. Thus, the situation modeling the perceived state of the system contains the characteristic “mode”. If the human operator cannot identify the current operating mode of a system, the human operator can guess about the current mode or can consider all possible options as equally probable. In both cases, the characteristic “mode” is uncertain as well as the situation containing the uncertain characteristic. However, the operator may not even know that there are specified modes of operation. Thus the characteristic “mode” is not only inaccurate, but unknown. In contrast to that, the situation which includes all other characteristics of the system is known inaccurately as one characteristic is missing. Thus, the situation is also denoted as uncertain in this case.

An analysis of the sources of uncertainty decision makers have to cope with revealed three different issues of uncertainty, namely situations, alternatives (or options), and outcomes (or consequences) [LS97]. These three different issues can also be found in decision making in human-machine interaction. The three different issues and the corresponding uncertainty are given in Table 4.2.

The first issue is uncertainty about the situation. In this case, the human operator either does not know the exact parameter of one or more characteristics of a situation, or one characteristic of the situation is completely unknown. In the situation shown in Fig. 4.1, one example for an uncertain characteristic of the situation is the distance between the aircraft. Even if the positions of both aircraft are given, the exact distance between them has to be estimated by the operator and usually results as uncertain. Uncertainty about the actual situation will be called state uncertainty in the following.

The second issue is about the options and describes that the operator does not know exactly which actions can be applied. This is especially the case in ill-defined tasks when the set of actions is large. In a highly structured and regulated task domain such as ATC, an experienced and well-trained operator should be aware of possible actions, especially since these are rather limited. Similar to the characteristics of a situation, a single option can be associated with uncertainty or an option can be completely unknown so that the whole set of options is uncertain (not accurately known). In the example given above, a non-expert may not see the option to separate the aircraft vertically and thus the option is unknown. Another example is that the possible climb rate of an aircraft is unknown to the operator. As every possible climb rate corresponds to an option, some options are unknown and thus the set of options is uncertain. Uncertainty about the options will be called selection uncertainty.

The third issue is uncertainty with respect to the consequences. This describes that human operators may not know the exact consequences of their options. In analogy to descriptive uncertainty, predicted situations may either contain uncertain characteristics or characteristics may be missing. Examples for uncertain characteristics are the aircraft's future horizontal position and its altitude after it is cleared to climb. It is noteworthy that the altitude is uncertain during the expected climb procedure, but the longer the prediction horizon, the more certain the aircraft will have reached its cleared altitude. Similar it can be uncertain, if the aircraft reaches the altitude necessary for vertical separation in time. Uncertainty about the consequences can be caused by state uncertainty or by imprecise predictions. For example, if the climb rate of an aircraft is unknown (state uncertainty) the prediction of the aircraft's future altitude can be only imprecise. On the other hand, especially predictions in dynamic systems have been shown to be difficult and therefore to cause errors [Dör89]. Uncertainty related to the consequences will be called prediction uncertainty. The three issues of uncertainty are summarized in Table 4.2.

4.2.2. Perspectives of Uncertainty in Assisted Human-Machine Interaction

In the context of HMSs, the uncertainty discussed up to now is related to the situation, options, and consequences as perceived by the operator. However, the uncertainty present in the real environment (including the controlled machine) is influenced by technical assistance and interfaces before it can be interpreted by the human operator. Each steps has an impact on uncertainty. Therefore, uncertainty should be considered from the different perspectives of a HMS (environment including machine, technical assistance including interface, and human operator). In order to facilitate the discussion, different terms will be introduced here. These terms are summarized in Table 4.3. First, the uncertainty present in the environment including the machine will be called real uncertainty or ambiguity. One example (s. Fig 4.1) is the reaction time of the pilots. Second, uncertainty can be considered from the perspective of the technical assistance. As explained in section 4.1, technical assistance can reduce or induce uncertainty. Additionally, the interface can also have an effect on the uncertainty depending on the presented information and their illustration. The interface will be seen as a part of the assistance here, as it can comprise some types of assistance, for example information analysis (for types of automation see e.g. [PSW00]). The uncertainty from the perspective of the technical assistance (including the interface) will be called technical uncertainty or inaccuracy. Finally, human operators have to perceived the information and combine it with their knowledge to understand its meaning. This is the uncertainty they have to cope with when making decisions and it will be called mental uncertainty or vagueness. The term uncertainty (without specification) will be used as a general

Table 4.3.: The three different perspectives of uncertainty in Human-Machine Systems

Perspective	**Uncertainty**	**Situation**
Machine/Environment	Real uncertainty/Ambiguity	Real Situation
Assistance (Interface)	Technical uncertainty/Inaccuracy	Technical situation • Measured characteristics • Secondary characteristics • Self characteristics
Human operator	Mental uncertainty/Vagueness	Mental situation • Measured characteristics • Secondary characteristics • Self characteristics

term for all issues and perspectives of uncertainty.

To distinguish between the individual perspectives of HMSs, it is helpful to use three different descriptions for the same scene. At first, the actual state of the machine or the environment controlled by the human operator can be modeled by the real situation. This description includes all relevant characteristics of the current scene and does neither comprise the technical assistance nor the human operator. This situation includes real uncertainty or ambiguity.

Second, the scene can be modeled from the assistance point of view. This situation includes three type of characteristics. First, this situation includes the measured characteristics of the real situation. Moreover, it further contains secondary characteristics which are derived from measured characteristics (e.g. if the aircraft are on conflicting routes). Finally, this situation also includes self characteristics necessary to describe the state of the assistance itself (e.g. working properly or not). This situation is called technical situation and is subject to technical uncertainty or inaccuracy.

Finally, the mental situation models the scene as perceived and interpreted by the human operator. This situation also consists of three different types of characteristics, similarly to the filtered/augmented situation. First, this situation contains the perceived characteristics presented by the interface and perceived by the human operator. Furthermore, the human operator can also extend the perceived situation by secondary characteristics derived by the application of knowledge. Third, the perceived situation can also contain self characteristics describing the state of the human operator. These state can model the human operators self-awareness and include for example the subjective actual workload. In this situation, vagueness is present.

4.2.3. Classification Scheme

The classification of uncertainty regarding the three different perspectives of HMSs can be combined with the three issues discussed above which results into the classification scheme with nine cells in Table 4.4. From left to right, the table depicts the three different perspectives. In the top row, the perspective is illustrated graphically. Top down, this table contains the different issues of uncertainty. In the following, each combination is examined and clarified with the example (s. Fig 4.1).

At first, the state uncertainty is analyzed from the three different perspectives. As the real situation models the state of the environment exactly as it is, no uncertainty for measurements and predictions are considered. Thus, state uncertainty is not present in the real situation.

If these characteristics are modeled in the technical situation, they can be inaccurate as they are based on measurements with limited precision. For example, the position, speed, and altitude of the aircraft in Fig. 4.1 are measured, and thus contain measurement errors and therefore are uncertain. The accuracy can be increased if measurements of different sensors are combined to determine one characteristic. The secondary characteristics of the technical situation calculated by the assistance can be inaccurate as well as they are derived from the measured characteristics. Finally, the self characteristics model the state of the technical assistance and can be inaccurate in cases the assistance cannot determine its status precisely.

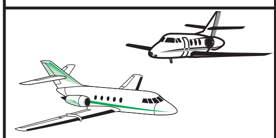
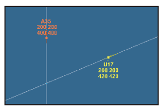
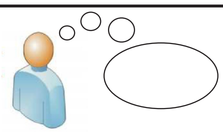
The reasons for vagueness in the mental situation can be manifold. As the human operator perceives the information presented by the interface, the inaccuracy of this information persists also in the perceived characteristics of the mental situation. Additionally, a disagreement between information sources can induce uncertainty [Hen99]. Further, these characteristics are influenced by perception errors, which cause an increase of ambiguity, influenced by the presentation of the information on the interface. In the example, altitude is given as a number and is thus exact. In contrast, positions of aircraft are indicated by dots and the distance between aircraft has to be estimated by the human operator, and thus is subject to vagueness. Moreover, if too much information is presented, the operator may not be able to track all information and to keep the mental representation up to date due to a lack of time or limited cognitive resources.

The secondary characteristics of the mental situation are influenced by the vagueness of the perceived characteristics and the knowledge applied to interpret these characteristics. Knowledge can be applied to reduce vagueness but imprecise or incomplete knowledge can also increase vagueness. For example, if the current mode of an aircraft is not displayed and cannot be perceived directly, the operator can use his knowledge to determine the current mode. The mental situation is related to situation awareness and corresponds to the first and second level of situation awareness as defined in [End95b] as it includes the perception of the current state (position, speed, altitude) as well as its interpretation (conflict).

The second issue of uncertainty, the selection uncertainty is present in each part of HMSs in a different way. At first, there are existing options in the real situation (other options do not exist) and there is no ambiguity inherent in this situation. However, two equivalent options can exist. In this case, there is no option superior to all other options and the best option cannot be defined.

The human operator can chose from the set of options available in the technical situation. This set of available options is in general not identical to the set of existing options. First of all, the options can be restricted. In the ATC example, the options are restricted by rules. Second, the assistance can execute some actions automatically. Thus,

Table 4.4.: Examples of continuous (light gray) and discrete (dark gray) uncertainty for the three issues (situation, options, consequences) and for the three perspectives of assisted Human-Machine Systems (machine, assistance/interface, human operator).

	Perspective							
	Environment / Machine		Assistance / Interface		Human operator		SA Level	Time horizon
								
Issue: Situations - State uncertainty	Real situation		Technical situation		Mental situation		Level 1/2	Present
			• Position/ speed/ altitude	• Mode of aircraft (climb, hold, descend)	• Distance between aircraft	• Conflict • Mode of aircraft		
Options - Selection uncertainty	Existing options		Available options		Perceived options		(Level 2)	Present
	• Equal options (no best option)		• Available range of vertical solutions	• Options involving aircraft A (A on frequency?)	• Possible range of vertical solutions	• Vertical solution unknown (by non-expert)		
Consequences - Prediction uncertainty	Real consequences		Calculated consequences		Predicted consequences		Level 3	Future
	• Reaction time of pilots	• Decisions of pilots • Random effects	• Prediction of future position	• Prediction of loss of separation	• Prediction of future position	• Prediction of loss of separation		

these options are not available to the human operator. Further, there can be additional options available for the human operator for the interaction with the assistance to turn its functions on or off. Selection inaccuracy can be present in the example, if the assistance has no information about the climb rate and the range of vertical solution is uncertain. Another reason for selection inaccuracy is state inaccuracy. For example, the information if an aircraft is already on the radio frequency is missing and it cannot be determined if the aircraft can be influenced by the operator and the set of solutions involving this aircraft can be applied.

The perceived options may also differ to the available options and selection vagueness can be present. This can result from state vagueness and incomplete knowledge. If the human operator is not certain about the actual situation, he cannot be certain if the conditions for an option are fulfilled and the option can be applied. Further, the operator may not know that an option is available or is not sure about the conditions which have to be fulfilled to select an option. This is more likely when a large amount of options exists and the operator is less experienced. In the concept of situation awareness, the perceived options are not explicitly considered. However, a good understanding of the current situation is necessary in order to assess the existing possibilities properly. Therefore, the perceived options are based on the second level of situation awareness (the interpretation of the current situation) but they are not explicitly considered in the definition of situation awareness as given in [End95b].

Finally, the prediction uncertainty is detailed for all perspectives of HMSs. In general, future states are predictable in deterministic environments. However, some aspects cannot be predicted and are probabilistic, for example the probability of technical failures. Also the system may be exposed to unpredictable disturbances making an exact prediction of the system impossible. If a human component is part of the environment, as the pilots in the example, the exact behavior is also not predictable. Consequently, real uncertainty about future states results.

From the technical assistance's point of view, inaccuracy about the consequences exists for two reasons. On the one hand, the technical situation itself may be uncertain and as a consequence, the predictions based on this situation will be inaccurate as well. Thus, state inaccuracy can lead to prediction inaccuracy. On the other hand, models of the environment are used to predict the consequences. Here the consistency of the applied model with the reality is important and the more precise the reality is modeled, the more accurate the predictions will be. For example, if the assistance applies a model of the aircraft performance, which does not correspond exactly to the reality, the predictions about future predictions are inaccurate. Similarly, the human operator makes predictions and the uncertainty of predictions also depends on the state uncertainty in the mental situation and on the applied knowledge. The predictions about future states of the system and the consequences of actions are related to the third level of situation awareness as defined in [End95b].

In [LS97], it was also differentiated between three sources of uncertainty when human operators make decisions. These sources are incomplete information, incomplete understanding, and undifferentiated alternatives. As these are sources of uncertainty the decision maker has to scope with, they are sources of mental uncertainty (vagueness) as

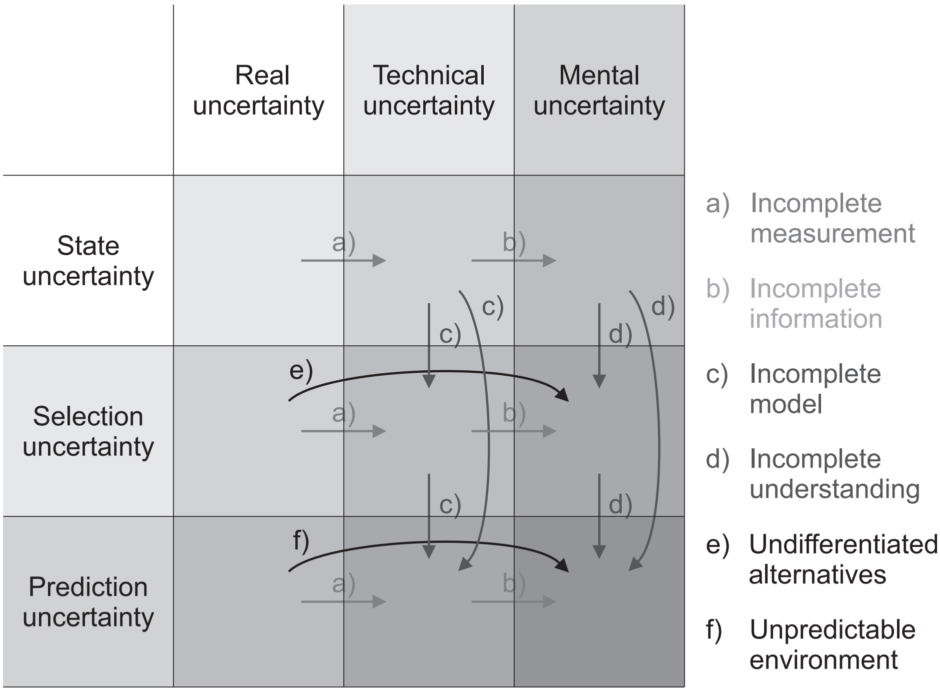


Figure 4.2.: Sources and dependence of the different issues and perspectives of uncertainty

defined above. These three sources can be sorted into the above presented classification scheme. First, incomplete information relates to the information passed on from the assistance to the human operator. Thus, it describes the inaccuracy in all issues of uncertainty from the technical assistance's perspective (e.g. information about the current situation is missing, information about the fulfillment of preconditions for options is missing, or a prediction which would simplify decision making is missing). The incomplete understanding refers to the influence of the human on the presented information. Either the knowledge needed to interpret the information is incomplete or the capacity of the cognitive resources of the human operator are not sufficient to process the amount of presented information. Undifferentiated alternatives describe that no best option exists in the environment. Thus, it describes the ambiguity of the existing options.

The classification scheme reveals some other sources of uncertainty for human operators. One of this sources are unpredictable environments. Furthermore, there are two sources of uncertainty influencing the technical assistance. These sources affect the human operator indirectly. The first of these additional source is incomplete measurement. While incomplete information describes mission information from the human operator's perspective, the incomplete measurement describes that technical assistance has not all necessary information. The other source is an incomplete model and corresponds to an incomplete understanding. The information is available to the technical assistance but this system cannot interpret the data correctly. All six sources of uncertainty are depicted in Fig. 4.2.

4.2.4. Uncertainty in Hybrid Environments

The environments' appearance, which is changed by implementing technical assistance, has a crucial impact on the decision strategy. On the one hand, in the real world, which mainly consist of continuous elements, multiple probabilistic and fallible indicators are used to make intuitive judgments which have to correspond with the reality. For example, the theory of naturalistic decision making [Kle98] is mainly concerned with decision making in such settings. According to that theory an option is chosen directly to the interpretation of a situation and different alternatives are not considered. The consequences of this option are predicted and another option is considered only if these consequences are not satisfying. If the situation is subject to uncertainty, the human decision maker will try to clarify the situation before making a decision. Thus uncertainties about alternatives are of minor importance according to this theory, but both other issues affect the decision process.

In contrast, an artificial environment containing technical assistance has many precise indicators informing about discrete states of the system. When artificial indicators are used as information sources, decisions must be analytical and coherent, which means logical consistent [Mos09]. In such situations, the normative theory of rational decision making [Doy99] specifies that the expected utility of all different options has to be calculated (predicted) and the action with the highest utility has to be chosen. This implies that all possible actions are known. Hence, all issues of uncertainty are important according to this theory in artificial environments. Despite the differences between these two theories of decision making, the prediction, and thus the inseparably connected prediction uncertainty, plays a crucial role.

According to [Ham96], the antipodes of intuitive and analytical decision making are connected to a continuum in the cognitive continuum theory. As a consequence, working domains, in which the human is supported by a variety of assistance systems, can be seen as a hybrid environment in which both intuition and analysis are required [Mos09]. Thus the decision maker has to scope with continuous and discrete characteristics and uncertainty. It is important to distinguish between both types as it determines the modeling approach.

This distinction between continuous and discrete uncertainty is made in the Table 4.4 for the examples given above. Examples of continuous uncertainty are shown with a light gray background, examples of discrete uncertainty are shown with a darker background. Additionally, different uncertainty distributions are possible. The uncertainty of a continuous characteristic can for example have a Gaussian distribution modeled by a mean value and a standard deviation. Otherwise, a distribution could be used which is restricted by defined minimal error, maximal error, or both. For discrete uncertainty, different possibilities could be given with each the same probability or with individual probabilities.

Examples for characteristics with continuous uncertainty are the position and altitude of the aircraft. The position could be described by a distribution which is not limited to an interval. If an aircraft is predicted by an human operator to be 6.5 NM before the next waypoint, it could also be closer to or further away from that waypoint and both can be equally probable. In contrast, a distribution restricted to an interval could be used for the altitude. If an aircraft is advised to climb to flight level 320 and the human operator expects it to need 60 s, the aircraft will probably be at flight level 320 or below after 60 s. It is not likely that the aircraft will be above that flight level (only when the pilot fails to execute the maneuver correctly). Therefore, the upper flight level is a restriction to the probability distribution. An example for a discrete value is aircraft's altitude mode, as aircraft can climb, descend, or hold current altitude. Each mode has its own probability, which varies during climb maneuvers. After the start of a climb maneuver, the aircraft will probably be in climb mode. During the maneuver, the probability of the altitude hold mode will increase as the aircraft may have reached its final altitude already. Finally, altitude hold mode will be the most probable mode.

The alternatives can also be both, discrete and continuous. For example, the human operator can decide to give a clearance to one or the other aircraft and instruct to change the altitude or direction of an aircraft. This is a choice between discrete options. On the other hand, a continuum of options is also possible. For example, different directions or altitudes can be instructed. In this case, a value out of a continuous range can be selected. However, the operating standards often reduce the range to a set of prescribed, discrete values. Regarding the consequences, some values can change continuously, like the position of the aircraft but also some discrete events can happen like the loss of separation.

The presented classification of uncertainty has demonstrated that uncertainty is very diverse and can take various forms. Thus, uncertainty can affect the situation, the alternatives and the consequences and can either be discrete or continuous. In addition, the correct perspective must be selected depending on the purpose of modeling. So either real uncertainties, technical uncertainties of assistance, or mental uncertainty of human operators are modeled. If, for example, the purpose is to find a theoretical optimal solution assuming a correct and complete knowledge of the environment, only the real uncertainty is of interest. In contrast, the second column is of meaning if assistance is developed to find solutions under technical uncertainty. In this case, possible measurement and modeling errors should be taken into account. As the aim of this chapter is to extend the human operator planning model developed in the last chapter by integrating uncertainty for generating more realistic interaction sequences as they could be planned and executed by a human operator, the perceived uncertainty in the third column is of importance.

4.3. Modeling of Uncertainty with Coloured Petri Nets

Existing models of human behavior based on a set of rules and implemented as CPNs [OHS11, HS13b] as well as the human operator planning model developed in the last chapter compare perceptions and decisions of the human operator against optimal solutions, although human operators should be compared to more realistic solutions instead (see section 4.1). For this reason, the aim of this chapter is to enable these models to make more realistic predictions of human behavior. Therefore, an approach to integrate uncertainty into such models is developed. As a prerequisite, the above described classification of uncertainty was developed, which showed that various forms of uncertainty exist in HMSs.

However, it was demonstrated [OHS11] that estimating the consequences of options is difficult even when all required information is available. As the situations were clearly indicated and the participants had enough time to understand them, it can be assumed that only a little perceived state uncertainty was present. The prediction uncertainty found in that study must consequently result from predictions by human operators mental model. Therefore, the effect of predictions seems to be most relevant.

The approach developed in this chapter should fulfill the following requirements to improve existing applications. First, CPNs are able to model discrete uncertainty but have no build-in functions to represent continuous uncertainty. Therefore, this approach should extend the capability of CPNs to represent continuous uncertainty as well. Furthermore, it should be possible to restrict the probability distribution to model upper or lower boundaries. Finally, the probability distribution should assign lower probabilities to larger prediction errors and higher probabilities to smaller prediction error. In the best-case, the probability distribution can be changed without much effort.

In the following, a concept is proposed, which fulfills these requirements. Continuous uncertainty is integrated into existing models of human behavior which are based on a set of rules and implemented as CPN. This concept is demonstrated and integrated into the human operator planning model presented in the last chapter. As a result, aircraft's altitude, speed, and position are calculated with prediction uncertainty.

First, an appropriate modeling approach and a probability distribution are selected (see section 4.3.1). Next, the implementation of this approach with CPNs is described in section 4.3.2. Finally, this implementation is integrated into the human operator planning model for the ATC task as example (see section 4.3.3).

4.3.1. Modeling of Probability Distributions

Uncertainty can be formalized by different methods. The most common used formalism is probability [Hen99]. Probabilities are described by values between 0 (is not true/will not happen) and 1 (is true/will happen). Thereby, the sum of the probability of all possible values/consequences has to be 1. In the objective view, probabilities describe facts and can be calculated (e.g. the probability of throwing a 6 with a dice is 1/6). In the subjective view, probabilities express a degree of belief in a fact or event. In the latter case, the probabilities can be based on knowledge and experiments. The real prediction uncertainty as defined in section 4.2.1 (perspective of the machine/environment) is objective. In contrast, the probabilities of perceived prediction uncertainty (perspective of the human operator) are based on experience. Therefore, these probabilities can also be subjective.

A discrete characteristic can have a parameter a out of the set of possible parameter A. A probability $P(a)$ can be assigned to each possible parameter a by $P(a) = p_a\%$. The sum of all probabilities has to equal 1.

The probability of continuous characteristics can be described by probability distribu-

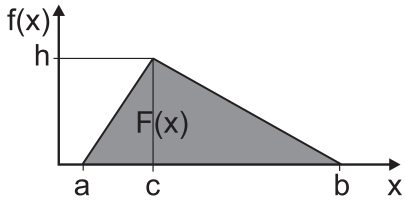


Figure 4.3.: Triangular distribution, defined by the minimum value a, the maximum value b and the value with the highest probability c.

tion functions. Such a function assigns a probability to each element within an interval. The functions can be distinguished depending on intervals size. First, there are functions that assign a probability to each element in the interval $[-\infty, +\infty]$. The most common example is the normal distribution. For example, this distribution can model a measured value including measurement error. Second, some distribution functions are limited to the bounded interval [a,b]. Examples are the beta distribution and the triangular distribution. Furthermore, some functions assign probabilities to elements within an one-sided infinite interval $[0, \infty]$, for example the exponential distribution.

Commonly, CPNs are used to model discrete uncertainty. A possible approach to model discrete uncertainty is to represent all possible consequences with different transitions (or bindings). However, as CPNs do not allow assigning a certain probability to a transitions (all transitions or bindings have the same probability), it would be necessary to keep track of the probability separately, for example with an additional place containing the probability of the actual state. If a transition with the probability $p_a\%$ fires, this marking has to be multiplied by $p_a\%$. This modeling approach is similar to a fault tree analysis where a certain probability is assigned to each possible event. This allows calculating probabilities of faults resulting from a sequence of events. As a consequence of this implementation of uncertainty in CPNs, different firing sequences will be possible from an initial marking. These sequences taken together show up in the state space as branches and reflect all possible predicted consequences. An example for modeling discrete uncertainty affecting predictions can be found in [EGVS10] . In this approach, the action space of an agent is calculated and an interaction sequence is generated. Uncertainty about the consequences is considered by assuming the same probability for each action possible in a situation. The interaction sequence is calculated such that the agents which will keep away from hazardous states.

However, the approach developed here enables CPNs to represent continuous uncertainty as well. To be able to model uncertainty restricted to a limited interval with the higher probabilities in the intervals center and lower probabilities at the interval borders, the triangular distribution—the simplest distribution fulfilling these requirements—is selected and implemented in the following. A triangular distribution is defined by the lower limit a, the upper limit b, and the mode c with $a < b$ and $a \leq c \leq b$. An example is shown in Fig. 4.3. As the base edge of the triangle has the length $l = b - a$ and area of the triangle is $A = 1$, the height (and the probability of mode c) is $h = \frac{2 \cdot A}{l} = \frac{2}{b-a}$.

This distribution can be used to model the prediction uncertainty of a characteristic with a certain prediction error e relative to the magnitude of the characteristic's change. At the beginning of the prediction at t_0, the characteristic is known exactly, thus $a(t_0) = b(t_0) = c(t_0) = d_0$. To assign a higher probability to smaller prediction errors and a lower probability to larger prediction errors, the mode $c(t)$ is calculated exactly and the limits of the distribution $a(t)$ and $b(t)$ are calculated with the maximal possible prediction error e. Consequently, assuming a constant change rate $r > 0$, the triangular distribution after prediction time t can be described by

$$c(t) = d_0 + r \cdot t, \tag{4.1}$$
$$b(t) = d_0 + r \cdot t \cdot (1 + e), \text{ and} \tag{4.2}$$
$$a(t) = d_0 + r \cdot t \cdot (1 - e). \tag{4.3}$$

A prediction modeled like that is shown in Fig. 4.4a. The probability density function $f(x,t)$ of a triangular distribution changing with time is given by

$$f(x,t) = \begin{cases} 0 & \text{for } x < a(t) \\ \frac{2(x-a(t))}{(b(t)-a(t))(c(t)-a(t))} & \text{for } a(t) \leq x \leq c(t) \\ \frac{2(b(t)-x)}{(b(t)-a(t))(b(t)-c(t))} & \text{for } c(t) < x \leq b(t) \\ 0 & \text{for } b(t) < x \end{cases}, \tag{4.4}$$

and the cumulative density function $F(x,t)$ is given by

$$F(x,t) = \begin{cases} 0 & \text{for } x < a(t) \\ \frac{(x-a(t))^2}{(b(t)-a(t))(c(t)-a(t))} & \text{for } a(t) \leq x \leq c(t) \\ 1 - \frac{(b(t)-x)^2}{(b(t)-a(t))(b(t)-c(t))} & \text{for } c(t) < x \leq b(t) \\ 1 & \text{for } b(t) < x \end{cases}. \tag{4.5}$$

Additionally, an upper boundary d_{max} and lower boundary d_{min} of the characteristic can be given. If this is the case, the distribution has to be restricted to values within these boundaries. However, the probability of these boundaries may be larger than 0. For example, with a constant change rate, the limit of the distribution will reach the boundary. After that, the probability of the boundary is increasing. Consequently, just defining d_{max} or d_{min} as the limits of the triangular distribution is not sufficient. Instead, a hidden triangular distribution can be used where a, b, and c are allowed to exceed the boundary and are calculated with the formulas given above. However, when the upper (lower) limit of the distribution exceed the boundary, the probability of the boundary is defined as the cumulative probability of all values in the hidden distribution above (below) the boundary:

$$p(d_{max},t) = \begin{cases} 0 & \text{for } a(t) \leq d_{max} \\ 1 - F(d_{max},t) & \text{for } a(t) > d_{max} \end{cases}, \tag{4.6}$$

$$p(d_{min},t) = \begin{cases} 0 & \text{for } b(t) \geq d_{min}, \\ F(d_{min},t) & \text{for } b(t) < d_{max} \end{cases}. \tag{4.7}$$

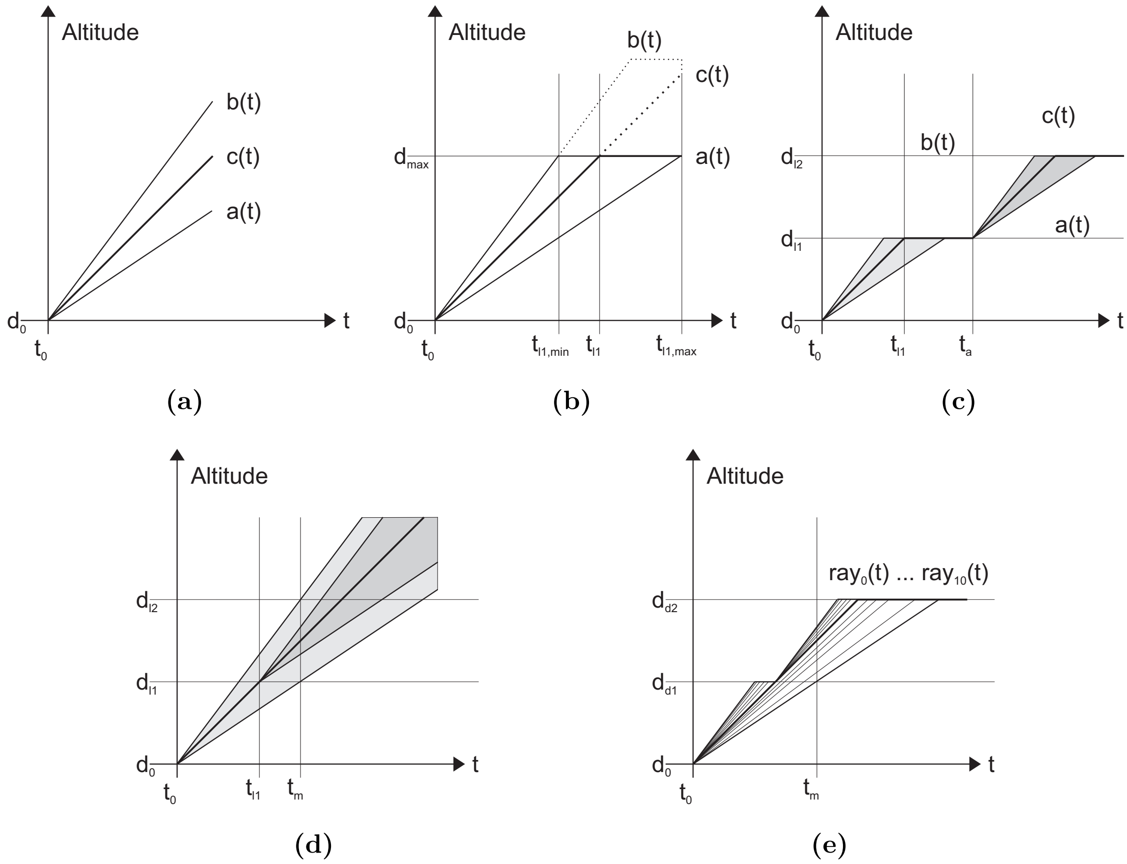


Figure 4.4.: Probability distribution modeled by a triangular distribution using the mode c and the limits a and b (a)-(d) respectively by a bundle of rays (e)

An uncertainty distribution modeled with hidden triangular distribution is shown in Fig. 4.4b. Moreover, the boundaries could also change during the prediction ($d_{max}(t)$ and $d_{min}(t)$). As long the distribution has not reached the boundary, the boundary can change without further consequences. When the boundary has two sections and is given by $d_{max}(t) = d_{l1}$ for $t \leq t_a$ respectively d_2 for $t > t_a$ with the time t_a when the complete distribution has reached the boundary ($a(t_a) = b(t_a) = c(t_a) = d_1$), the triangular distribution can be modeled by

$$c(t) = d_{l1} + r \cdot (t - t_a) \text{for } t > t_a, \tag{4.8}$$

$$b(t) = d_{l1} + r \cdot (t - t_a) \cdot (1 + e) \text{for } t > t_a, \text{ and} \tag{4.9}$$

$$a(t) = d_{l1} + r \cdot (t - t_a) \cdot (1 - e) \text{for } t > t_a. \tag{4.10}$$

This is illustrated in Fig. 4.4c. However, changing the boundary must also be possible when one limit of the distribution has reached a boundary and its probability is greater 0 as illustrated in Fig. 4.4d. In this case, modeling the resulting triangular distribution is more complicated. Two distributions could be used combining both cases above. The first triangular distribution given by equation 4.1 to 4.3 models the probabilities resulting from prediction errors, which corresponding predictions had not reached the boundary. The second triangular given by equation 4.8 to 4.10 distribution would model the probabilities resulting from prediction errors, which corresponding predictions had reached the boundary. Each update of the boundary while a limit reached the boundary would require to consider an additional triangular distribution and further complicate the calculation of uncertainty distributions.

To simplify the representation of the probability distribution, a bundle of rays is calculated instead, with each ray representing a different error magnitude. These rays span the area of the probability distribution. In Fig. 4.4e the distribution resulting from a constant change rate and a changed upper boundary with $N = 11$ calculated rays is shown. Each ray ray_n with $n = 1 \dots N$ is calculated by

$$ray_n(t) = c_0 + r \cdot t \cdot (1 + e \cdot \frac{2n - N - 1}{N - 1}). \tag{4.11}$$

The term $\frac{2n-N-1}{N-1}$ is used to modify the error magnitude and ranges from -1 for $n = 1$ to 1 for $n = N$.

The differences between the individual errors magnitudes are equal, while the differences between the cumulative probabilities are not. The cumulative probability $F(n)$ for each ray n out of N rays can be calculated using the triangular distribution with $a = 1$, $c = \frac{N+1}{2}$, and $b = N$ by

$$F(n) = \begin{cases} 0 & \text{for } n < 1 \\ 2\frac{(n-1)^2}{(N-1)^2} & \text{for } 1 \leq n \leq \frac{N+1}{2} \\ 1 - 2\frac{(N-n)^2}{(N-1)^2} & \text{for } \frac{N+1}{2} < n \leq N \\ 1 & \text{for } N < n \end{cases} .S \tag{4.12}$$

For example, using $N = 11$ rays to span a triangular distribution, the cumulative densities of ray_0 is $F(1) = 0\%$, of ray_1 is $F(2) = 2\%$, and of ray_2 is $F(3) = 8\%$. To calculate the probability distribution between the rays, an interpolation is necessary.

The different approaches to calculate the prediction uncertainty are demonstrated with a climb maneuver of an aircraft in the following. At the beginning, the aircraft is at $c_0 = 10\,000\,\text{ft}$. It has a constant climb rate of $r = 20\,\text{ft}\,\text{s}^{-1}$. The assumed maximal prediction error is $e = 10\%$. After $t_{l1} = 100\,\text{s}$, the aircraft's altitude will be $c(100) = 12\,000\,\text{ft}$, but it will be predicted to be between $a(100) = 11\,800\,\text{ft}$ and $b(100) = 12\,200\,\text{ft}$ (s. Fig 4.4a). As $F(11\,941\,\text{ft}, 100\,\text{s}) = 0.25$ and $F(12\,059\,\text{ft}, 100\,\text{s}) = 0.75$, the aircraft will be predicted to be between 11 941 ft and 12 059 ft with a probability of 50%.

Then a limit of $d_{max} = d_{l1} = 12\,000\,\text{ft}$ is given to model the final altitude. The aircraft will reach this altitude between $t_{l1,min} = 90.9\,\text{s}$ and $t_{l2,max} = 111.1\,\text{s}$ (s. Fig 4.4b). The probability of d_{l1} can be calculated with a hidden triangular distribution by $p(d_{l1}, t) = 1 - F(d_{l1}, t)$. Consequently the probability of $d_{l1} = 12\,000\,\text{ft}$ at $t = 91.1\,\text{s}$ is

$$p(12\,000\,\text{ft}, 97.1\,\text{s}) = \frac{(b(t) - 12000)}{(b(t) - a(t))(b(t) - c(t))} = 0.75 \tag{4.13}$$

with $t_{l1,min} \leq t \leq t_{l1,max}$.

When the second climb maneuver is initiated at $t_{l1} = 100\,\mathrm{s}$ and the limit d_{max} is defined by $d_{max}(t) = d_{l1} = 12\,000\,\mathrm{ft}$ for $t \leq t_a$respectively $d_{l2} = 14\,000\,\mathrm{ft}$ for $t > t_a$, two triangular distributions can be used to model the resulting uncertainty distribution (s. Fig 4.4d). The first triangle is valid for climbing rates which were not sufficient to reach d_{l1} before the second climb started. Consequently, at $t_m = 150\,\mathrm{s}$ the first triangle is defined by $a_1(150) = 12900$, $b_1(150) = 13100$, $c_1(150) = 13000$, and is valid for altitudes below 13 000 ft. The second triangle is described by $a_2(105) = 12950$, $b_2(150) = 13050$, $c_2(150) = 1300$ and valid for altitudes above 13 000 ft.

If a bundle of rays is used to model the uncertainty distribution, the 50% probability range of the altitude at $t_m = 150\,\mathrm{s}$ of the step-climb maneuver can be calculated in the following way. Using $N = 21$ rays, the lower limit of the probability range (F = 25%) lies between ray_7 and ray_8 as $F(ray_7) = 24,5\%$ and $F(ray_8) = 32\%$. Consequently, the altitude of the lower limit will be between $ray_7(150) = 12\,940\,\mathrm{ft}$ and $ray_8(150) = 12\,960\,\mathrm{ft}$. A linear interpolation reveals that the aircraft will be above 12 941.3 ft with a probability of 25% as $12940 + \frac{12960-12940}{32\%-24,5\%} \cdot (25\% - 24,5\%) = 12941.3$.

4.3.2. Implementation of Continuous Uncertainty in CPNs

The approach to represent uncertainty by a bundle of rays developed in the last section is implemented into CPNs realized with the modeling software CPN Tools [JK09] in the following.

First of all, the type of the variable which should be represented including uncertainty has to be replaced, since a single value is not sufficient anymore (commonly of type `INT`). Therefore, a new colorset `colset Bundle = list INT;` is defined which realizes a bundle of rays as a list of integers to describe the uncertainty distribution. As explained above, each of the list's elements represents a prognosis assuming different error magnitudes. Then, a variable with uncertainty can be defined using the new type (e.g. `var characteristic: Bundle;` instead of `var characteristic: INT`).

The functions implemented to process uncertain characteristics with CPNs are summarized in Table 4.5. At first, two functions are defined to transform variables without uncertainty into variables with uncertainty and the other way around. In the first case, a list is generated whereby the number of elements equals the amount of rays N and each element is set to the exact characteristic (function No. 1.1). No uncertainty is generated through the transformation. To transform an uncertain variable into an exact variable, the value in the middle of list is taken, and the rest of the list is rejected (function No. 1.2).

If the characteristic with uncertainty changes, every ray has to be updated. This is realized by three functions (No. 2.1 - 2.3). These functions have in common that they expect the initial bundle `bundle`, the exact change `delta`, which specifies how much the exact characteristic changes and, and the maximal error `e` as input. The most straightforward function out of these is function No. 2.1. This function implements equation 4.11 with $r \cdot t$ given by `delta` and adds `delta` to each ray multiplied by a specific error. After each ray is updated, the first element in the list of rays is always the

Table 4.5.: Developed functions to realize uncertain characteristics with CPNs

No.	Name	Input	Output
1.	**Transformation of uncertain to precise variables and vice versa**		
1.1	`generateBundle (value, N)`	`int * int`	`Bundle`
1.2	`getValue (bundle)`	`Bundle`	`int`
2.	**Update of bundles of rays**		
2.1	`updateBundleDelta (bundle, delta, e)`	`Bundle * int * int`	`Bundle`
2.2	`updateBundleLimit (bundle, boundary, delta, e)`	`Bundle * int * int * int`	`Bundle`
2.3	`updateBundleRange (bundle, delta, range, e)`	`Bundle * int * int * int`	`Bundle`
2.	**Update of hybrid characteristics**		
3.1	`updateBINTDelta (bint, delta, e)`	`BINT * int * int`	`BINT`
3.2	`updateBINTLimit (bint, boundary, delta, e)`	`BINT * int * int * int`	`BINT`
3.3	`updateBINTRange (bint, delta, range, e)`	`BINT * int * int * int`	`BINT`
3.	**Calculation of probabilities**		
4.1	`Pbundle(n, N)`	`int * int`	`int`
4.2	`Plower (bundle,value)`	`Bundle * int`	`int`
4.3	`Vlower (bundle,p)`	`Bundle * int`	`int`

smallest value and represents the lower limit a of the triangular distribution. In other words, if $\texttt{delta} > 0$, $E = \texttt{delta} \cdot \texttt{e}$ is subtracted from the first ray, and if $\texttt{delta} < 0$, E is added to the first ray (Note: E is negative as $\texttt{delta} < 0$).

The second function (No. 2.2) is for uncertainty distributions with boundaries. It additionally expects a boundary as input and is similar to the first function but additionally verifies that no ray exceeds the given boundary. Finally, the third function (No. 2.3) adds uncertainty to each ray relative to an additional value `range`. Its implements a modification of equation 4.11 and updates each ray by.

$$ray_n(t) = c_0 + \texttt{delta} + \texttt{range} \cdot (e \cdot \frac{2n - N - 1}{N - 1}). \tag{4.14}$$

To allow the representation of uncertainty when needed but to be able to represent a characteristic also precisely, a colorset for a hybrid characteristic is defined by `colset BINT = union Int:INT + Bundle:Bundle;` which can either include a precise variable of the type `INT` or an uncertain variable of the type `Bundle`. To manipulate a variable of the type BINT without the necessity to identify if it contains an exact or uncertain value, three functions are defined (function 3.1 - 3.3), which are similar to the above described functions 2.1 - 2.3. The difference is that instead of an uncertain variable (`Bundle`), a hybrid variable (`BINT`) is expected as input. If the hybrid variable contains a precise variable, just the specified delta is added. In the case it contains an uncertain variable, the respective function (2.1 - 2.3) is called.

Furthermore, some functions were implemented to access the probability distribution modeled by the rays. This requires first a function to calculate the cumulative probability for every ray n out of N rays (function 4.1) by implementing equation 4.12. Further, the function 4.2 calculates the cumulative probability for any given value by calculating the cumulative probability for the both nearest rays and returning the interpolation between. Finally the function 4.3 is defined for the opposite query, expecting a bundle and a cumulative probability and returning the corresponding value. This function first searches for those both rays, which cumulative probabilities surrounds the given probability, and returns the interpolation between those rays.

4.3.3. Integration of Prediction Uncertainty into the Planning Model

The implemented data types and functions are integrated into the ATC simulation as part of the human operator planning model developed in the last chapter. The simulated task is explained in section 3.3.

Up to now, an aircraft is modeled as record (aggregation of other types) by: `colset Aircraft = record AI: AcInfo * AS: AcState * ACL: AcClr * APL: AcP timed;` Thus, the type aircraft consists of four values and the type of each of these values is in turn a record. The `colset AcState = record HD: INT * ALT: INT * SPD: INT * PosX: INT * PosY: INT * RnVec:STRING * POR: INT;` is of interest here as it describes the current position of the aircraft (in X-and Y-coordinates, `PosX` and `PosY`), its altitude (`ALT`), and its speed (`SPD`). `RnVec` gives the name of the aircraft's current Route, and `POR` the current position on this route. All continuous variables in this colorset, which have to be predicted, and thus can include uncertainty, are changed to the hybrid variable `BINT`. Thus, the new `colset AcState = record HD: INT * ALT: BINT * SPD: BINT * PosX: BINT * PosY: BINT * RnVec:  STRING * POR: BINT;` results.

To be able to transform aircraft stored in the precise data type without uncertainty into the newly defined hybrid data type and vice versa, two functions are defined, which integrate `generateBundle` and `getValue` to transform an aircraft at once (see function 1.1 and 1.2 in Table 4.5). Because access to aircraft characteristics is encapsulated in get- and set-functions, only these functions have to be adapted to the new hybrid data type. In order to reduce further changes, they return the exact value. If uncertainty has to be handled explicitly the uncertain characteristics of aircraft are accessed directly. This is the case four times: altering the position, altitude, and speed of the unequipped aircraft and altering the position of the equipped aircraft.

If the ghosting assistance is not activated (see section 3.3) in MAGIE, the human operator has to predict the flight trajectories up to the late merging point. As the equipped aircraft follow their route automatically, the progress on this route and consequently the X- and Y-coordinates (X: west–east, Y: south–north) have to be predicted. Additionally, the position of the unequipped aircraft has to be predicted. However, as these aircraft fly east to west during the merging procedure, the Y-coordinate is constant and only the X-coordinate varies. Furthermore, the unequipped aircraft have to comply with the altitude and speed constraints when entering new route sections. Therefore, prognosis of altitude and speed are also of interest. All characteristics of an aircraft which are predicted if the human operator is not assisted depending on the aircraft's equipage are summarized on the left hand side of Table 4.6.

If the ghosting assistance is activated, the operators do not need to predict the aircraft's trajectories up to the late merging point. Instead, it has to be predicted whether the unequipped aircraft fit between the ghosts on the centerline and how the distance between ghost and equipped aircraft will change until the late-merging point is reached. As ghost and unequipped aircraft fly in the same direction with a similar speed on the centerline, it is assumed that the position of the equipped aircraft is predicted according to the position of the ghost. Consequently, not the change of absolute position of the aircraft generates uncertainty, but the change of its distance to the ghosts. Therefore,

Table 4.6.: Variables represented with uncertainty depending on aircraft's equipage and activated assistance

	Ghosting not activated		Ghosting activated	
Characteristic	Equipped	Unequipped	Equipped	Unequipped
X	Uncertain[a]	Uncertain[a]	Exact	Uncertain[b]
Y	Uncertain[a]	Exact	Exact	Exact
Altitude	Exact	Uncertain[a]	Exact	Uncertain[a]
Speed	Exact	Uncertain[a]	Exact	Uncertain[a]

[a] Proportional to change of parameter

[b] Proportional to change of own speed relative to ghost speed

the uncertainty of the X-position of the unequipped aircraft is not implemented relative to this aircraft's absolute speed but relative to the difference between its speed and the ghost's speed. For example, when the unequipped aircraft is on the centerline and flies with the same speed as the ghosts, the uncertainty does not change. However, if an aircraft just started a turn and is flying in the opposite direction compared to the ghosts, the uncertainty increases proportional to the sum of both speeds in X-direction.

As the distance to the ghosts is used as indicator for conflicts if the ghosting functions is active, the position of the equipped aircraft is not of interested and consequently not predicted. On the right hand side in Table 4.6, the characteristics predicted with uncertainty, if ghosting assistance is activated, are given.

4.4. Results of Integrating Prediction Uncertainty into the Planning Model

The integration of prediction uncertainty into an existing an ATC simulation as a demonstration is reported in the following.

The results concentrate on the differences between an exact and an uncertain prediction. The considered situation and the predicted consequences are shown in Fig. 4.6 (ghosting not activated) and Fig. 4.7 (ghosting activated). The black lines in these figures indicate the route structure. All aircraft are flying toward the LMP at $X = 100000, Y = 100000$. Whereas the equipped aircraft fly from the entries in the north and south directly to that point, the unequipped aircraft also entering in the north and south, first fly in east direction and are then instructed to turn west. In the example situation the unequipped aircraft U90 is located at X = 114538, Y = 105994 and is just instructed to start the turn maneuver, it is flying at the altitude of 6049 ft and declining to 3000 ft. The equipped aircraft A57 is arriving from the south. U90 should arrive behind A57 at the LMP and keep a minimum separation distance of 3 NM (= 5558 m). For this situation, two predictions are calculated assuming an uncertainty of 10% and using N = 21 rays. On the one hand, prediction uncertainty is calculated according to deactivated ghosting (s. Fig 4.6), on the other hand, prediction uncertainty is calculated according to activated ghosting (s. Fig 4.7). The uncertain and exact predicted characteristics are given in (see Table 4.6).

In both settings, the altitude of U90 is predicted with uncertainty. The result is show in Fig. 4.5. For the sake of simplicity, only 5 out of 21 calculated rays are shown, including both outer rays (ray_1 and ray_{21}) and the middle ray (ray_{11}). At $t = 103$ s, the first ray reaches 3000 ft. In other words, the aircraft is predicted to reach 3000 ft at $t = 103$ s at the earliest. At $t = 113$ s, also the middle ray reaches 3000 ft. At this time, the probability of the aircraft's altitude being 3000 ft, is exactly 50%. This altitude is reached by the last ray at $t = 126$ s. Now the aircraft is predicted to have reached its dedicated altitude with 100%.

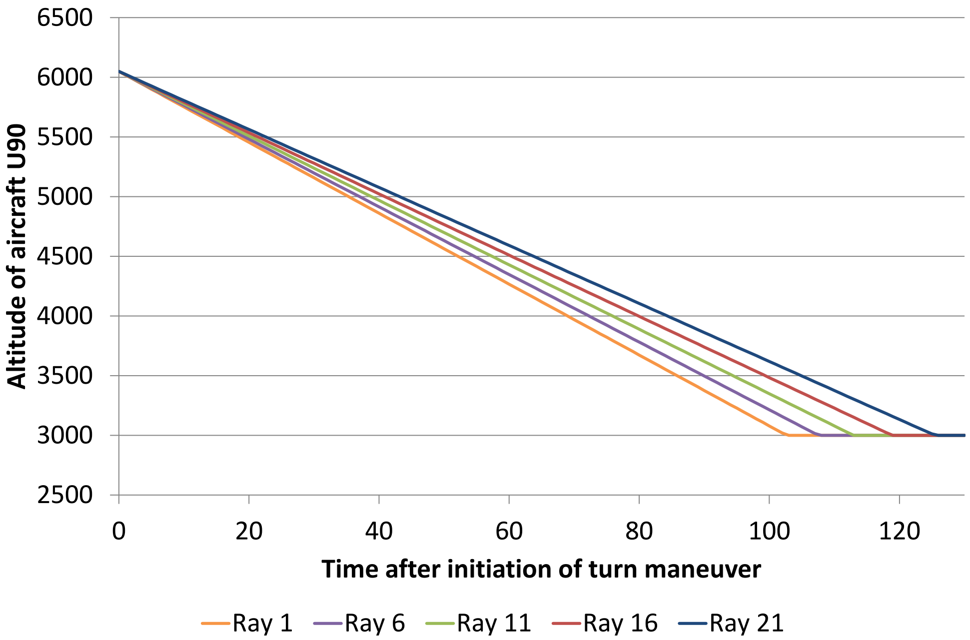


Figure 4.5.: Prediction of the altitude profile of U90, declining from above 6000 ft to 3000 ft. Calculated with an uncertainty of 10%

If ghosting assistance is not activated, the equipped aircraft's progress on route is calculated with uncertainty. Consequently, the X- and Y-coordinates, which are derived from that progress, are uncertain and a distribution of the aircraft positions along the route results. For unequipped aircraft, only their X-coordinate is predicted with uncertainty as they have a constant Y-coordinate ($Y = 100000$) in the critical phase close to the LMP. In Fig. 4.6, the predicted lateral position of aircraft U90 and A57 are shown for specific points in time. In the example, the separation between both aircraft is violated at $t = 107$ s if uncertainty is considered. For this moment, the smallest distance is predicted to be only 5458 m (3 NM = 5558 m) between ray_1 of U90 (ray with lowest x-value) and ray_{16} of A57.

If ghosting assistance is activated, the positions of equipped aircraft are calculated without uncertainty and the uncertainty of the unequipped aircraft's X-coordinate depends on their speed relative to the ghosts speed. The resulting prediction is shown in

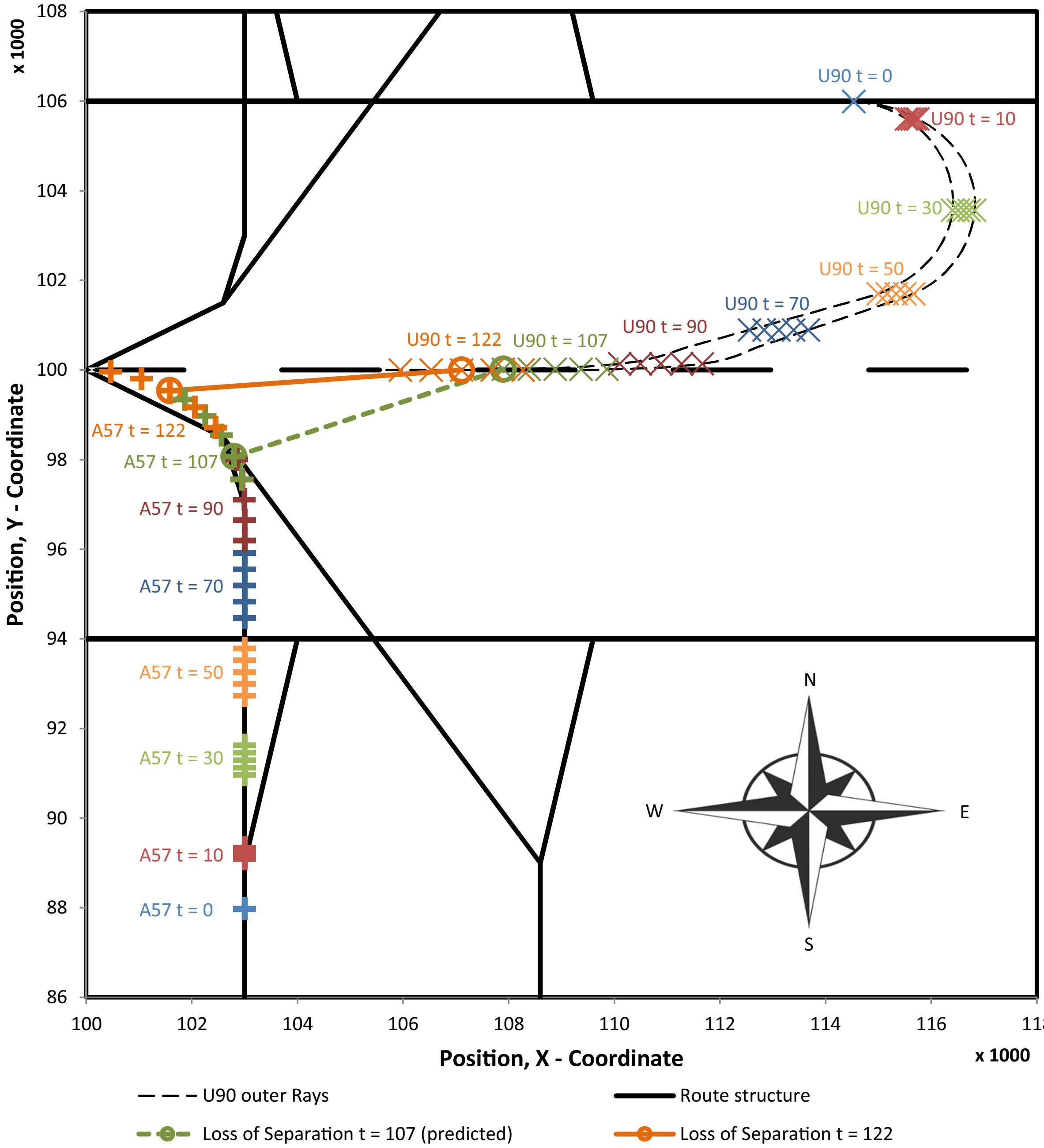


Figure 4.6.: Uncertain prediction of the positions of the aircraft U90 and A57 without ghosting calculated with an uncertainty of 10%

Fig. 4.7. According to the uncertain prediction, separation is violated firstly at $t = 111\,\mathrm{s}$, as the distance between the first ray of U90 and the exact position of A57 is 5480 m at this moment.

The exact prediction is represented by the middle rays. Consequently, it is not influenced by activating the assistance (and the corresponding exact or uncertain prediction). The distance the middle rays of U90 and A57 is 6484 m at $t = 107\,\mathrm{s}$ respectively 6165 m at $t = 111\,\mathrm{s}$. Consequently, an exact calculation would not predict violations of separations at $t = 107\,\mathrm{s}$ or $t = 111\,\mathrm{s}$. However, at $t = 122\,\mathrm{s}$, the exactly predicted distance between both aircraft has decreased to 5546 m, and thus the separation between both aircraft is violated.

The comparison of exact prediction ($t = 122\,\mathrm{s}$) and uncertain prediction with assistance ($t = 111\,\mathrm{s}$) and without assistance ($t = 107\,\mathrm{s}$), indicates, that the accuracy of conflict prediction increases, when the assistance ghosting is activated. The loss of separation is closer to the exact prediction, when ghosting is activated.

For the generation of human-like interaction sequences, the predictions are used to determine when speed reductions are necessary. Regardless of taking into account the uncertainty, the speed of aircraft U90 needs to be reduced to avoid loss of separation. In this sense, all predictions would lead to the same result. However, if the prediction uncertainty is considered, the separation is violated earlier. Hence, a speed reduction would need to be planned for an earlier point in time. In the example, the speed reduction is predicted to be about 25 seconds earlier with uncertainty. Thus, predictions calculated with uncertainty result into earlier than necessary speed reductions and thus a loss of efficiency. Although, the difference seems small, it has a large impact on the efficiency, as each decision is affected. It is assumed, that uncertain prediction are the reasons for inefficient decisions of human operators and that interaction sequences generated with uncertain predictions produce more human-like interaction sequences.

4.5. Concluding Remarks

The aim of this chapter was to enable models of human behavior based on CPNs to allow more realistic predictions of human operator behavior. This should be achieved by integrating uncertainty into the models. Therefore, a classification of uncertainty in HMSs was developed first, which helped to identify the different issues and perspectives to select and model the kind of uncertainty with the largest impact. The classification supported the identification of the strong impact of prediction uncertainty. Therefore, this uncertainty was selected, modeled, realized in CPN Tools and applied to the example application. The developed classification is further useful to clarify the different forms of uncertainty and can be applied in other cases before an appropriate modeling approach is selected.

The integration of prediction uncertainty enables Petri Net models to make predictions including continuous uncertainty. As the realized data types allow both, exact and uncertain representations, and are accessed by encapsulated functions, integration into the existing planning processes is possible without much additional effort. Modeling

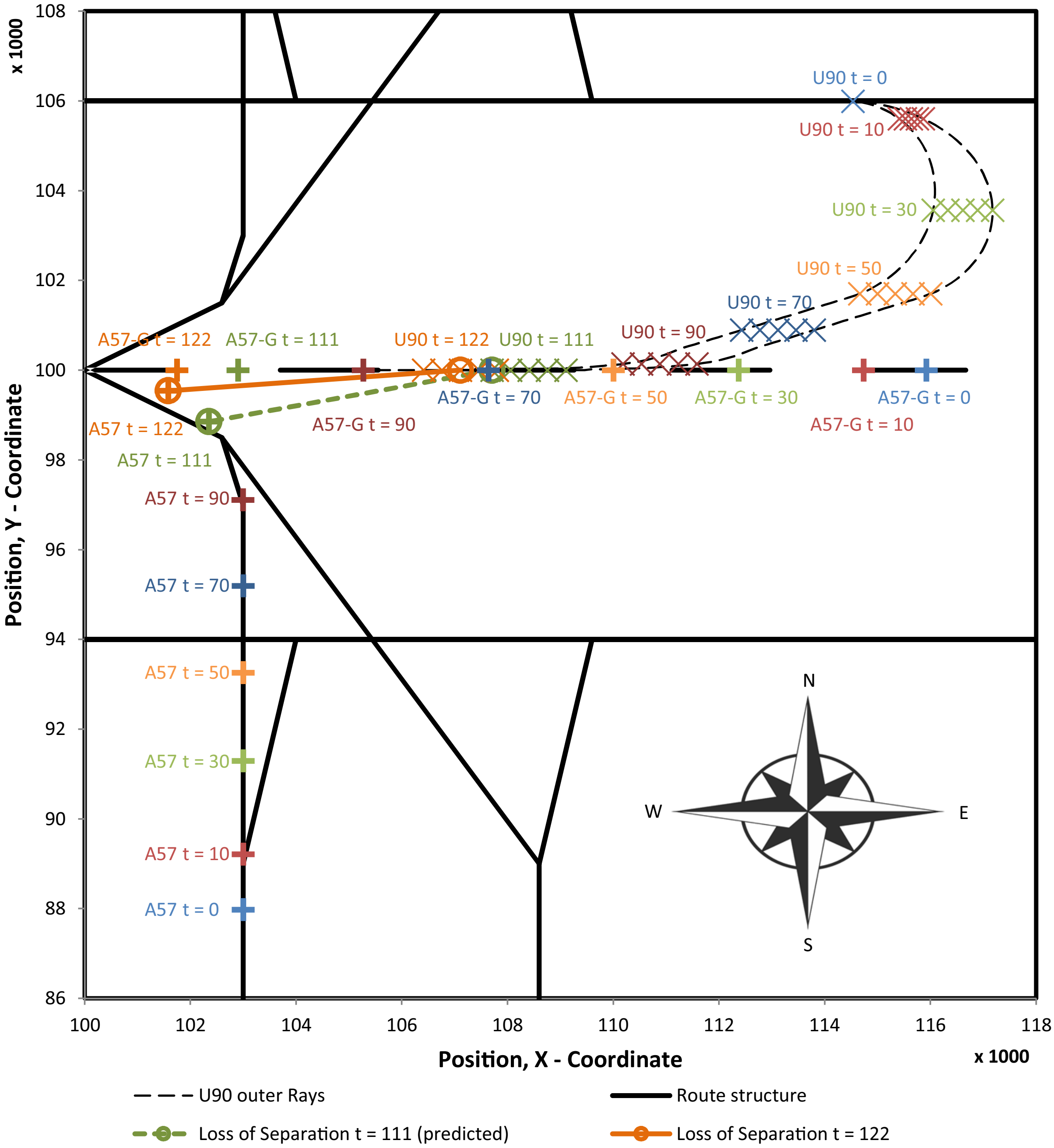


Figure 4.7.: Uncertain prediction of the positions of the aircraft U90 and A57 with ghosting calculated with an uncertainty of 10%

uncertainty as a bundle of rays has, on the one hand, the disadvantage that only an approximation of the assumed probability distribution is possible. On the other hand, switching to other distributions is possible and only requires implementing the corresponding cumulative probability distribution. Furthermore, representing uncertainty as rays allows representing complex distributions resulting from nonlinear predictions, which could not be modeled with a distribution function.

The implementation of uncertainty in the example application indeed resulted into predictions, which lead to inefficient behavior. However, it has to be analyzed how realistic interaction sequences calculated based on uncertain predictions are. Therefore, the parameters, in particular the maximum prediction error, have to be defined. Consequently, it is necessary to compare the generated interaction sequences to the measured behavior of human operators.

5. Model-Based Evaluation of Decisions of Human Operators

The last chapters described the cognitive model of planning developed in this thesis to generate interaction sequences to be used as criteria for the measurement of human operators' cognitive task performance. The application of the generated interaction sequences to evaluate human operators' individual decisions is presented in the following chapter. The generated interaction sequences are compared to the actual decisions of human operators and deviations are detected and evaluated based on their consequences. This method additionally allows determining the involved kind of action and the type of human error.

The evaluation based on the deviations' consequences becomes possible by combining a result measure to evaluate the result of a complete interaction sequence with the approach of event measures which compares the decisions of human operators to facts of the real world. This allows measuring human operators' task performance during an interaction process.

The procedure of the developed method to evaluate single decisions of human operators during an interaction is briefly summarized as follows:

1. Different situations, which occurred during a real measured interaction, are selected as initial states to generate goal-directed interaction sequences with the cognitive model of planning presented in chapter 3 respectively with the extended model described in chapter 4.

2. The measured actions, which were executed by an operator before the selected situation was reached, are combined with the generated sequence starting with the selected situation.

3. Every pair of measured and calculated sequences represents a possible course of interaction and is simulated to determine its consequences.

4. Evaluation functions are applied to assess the effects of these sequences.

5. The evaluation results of two adjacent situations are compared. The difference is attributed to the decisions of the human operator between both situations.

When sufficiently small distances between two situations are used, the evaluation of individual actions becomes possible. By comparing the actions planned to be executed (calculated by the model) and the actions actually implemented (by the human operator) in the interval between two adjacent selected situations, the kind of the relevant action and the type of human error (omission or addition of an action) can be identified. This allows detecting specific problems during observed interaction.

Prerequisite for the application of the method described above is the existence of evaluation criteria. As the simulation environment MAGIE is used as application example, the criteria defined for MAGIE are presented and compared to criteria applied in other studies with the same simulation environment in section 5.1. Subsequently, the procedure of the method is explained in detail in section 5.2. Then, in section 5.3 the exemplary application of this method is described. As reasoned in chapter 4, the human operator should not be compared with an ideal solution but with a realistic and expectable solution. Therefore, the impact of uncertainty, which is inevitably connected with predictions in dynamic environments, is considered. The implemented modifications of the planning model necessary to calculate interaction sequences including uncertainty are presented in section 5.4. Afterwards, the application of the developed method including the uncertainty in predictions is demonstrated in section 5.5. Finally, the results are discussed in section 5.6.

5.1. Result Evaluation Criteria in Example Application

Result evaluation criteria are used to assess the result of an interaction sequence. They transfer the reached result into a number to make the comparison of different interaction sequences possible. The evaluation criteria cannot be chosen arbitrarily as they must reflect the objectives in the analyzed task. In the example application MAGIE, the objectives are instructed to the participants. These are the three objectives *separation*, *constraints*, and *throughput* (see section 3.5.2).

In the written instructions of MAGIE, no formula to calculate the fulfillment of the objectives is given. An exact formula has not been specified because it was not expected that the participants are able to memorize this formula and to use this formula to compare different decisions under time pressure. However, this has the consequence that there are several ways to transfer the objectives described by words into an exact formula. When such a formula is defined, it is important that plausible evaluations result.

However, a ranking of the objectives is given in the instructions. The ranking of the objectives implies that the fulfillment of the higher objective should be the aim in every case, no matter how the lower objectives are affected thereby. Only if several options fulfill the higher objective equally, the lower objective should be considered. Consequently, a weighting of the objectives is not possible and it is necessary to regard all objectives as independent.

Below the development of formulas to measure the fulfillment for the three objectives is described. The defined formulas are justified and compared to these used in other studies applying MAGIE. The formulas are defined such that 1 is the optimum value (objective fulfilled completely) and that lower values indicate a lack of performance. A value of 0 should represent the theoretical minimum (if possible) of the reachable performance.

The variables used for the definitions of the objectives fulfillment are named according to the following rules. Variables describing an individual aircraft or conflict within an interaction sequence are denoted with lower case Latin letters. Variables describing the sum of these individual variables are named with upper case Latin letters. Latin letter are also used as index (lower case) or for totals (upper case). Greek letters are used for the fulfillment of the objectives, lower case letters for the contribution of individual aircraft, upper case letters for the overall fulfillment.

5.1.1. The Objective Separation

The most important objective is *separation*. The aircraft should be separated by at least 3 NM at all time. Consequently, a loss of separation should reduce the fulfillment of the objective.

To measure the fulfillment of the objective *separation* in compliance with the instructions, a formula must be defined which meets the following conditions. First, the calculated value must decrease if the duration of a conflict increases. Further, it must decrease if an additional conflict occurs. Additionally, it must be independent from other effects. Moreover, the criteria should facilitate the identification of the impact of each action.

In general, two approaches are possible to calculate the fulfillment of the objective *separation*. On the one hand, the objective can be based on the sum of the duration of each particular conflicts (conflict-based approach). On the other hand, the objective can be calculated as the sum of the duration of losses of separation for each unequipped aircraft (aircraft-based approach). These two approaches differ, when equipped aircraft are involved in the conflicts. As the equipped aircraft are planned automatically and the separation between two equipped aircraft will never be violated, it is sufficient to concentrate on the manually controlled unequipped aircraft.

Both approaches are compared in the following. In the conflict-based approach, the duration of each conflict d_j with $j = 1, \ldots, C$ are summed up. The variable C is used to denote the total amount of conflicts during an interaction. The overall duration of all conflicts D is accumulated as

$$D = \sum_{j=1}^{C} d_j. \tag{5.1}$$

The fulfillment of the objective *separation* according the conflict-based approach Δ_{cb} (*cb* for conflict-based) can be defined by

$$\Delta_{cb} = 1 - \frac{D}{D_{max}}, \tag{5.2}$$

with D_{max} as the maximal possible value of D so that Δ_{cb} is between 0 and 1. As an aircraft can violate the separation to several other aircraft at the same time, the total amount of aircraft at every moment during the simulation must be considered for the definition of D_{max}.

For the aircraft-based approach, the durations k_i during that aircraft i violates at least one separation are accumulated

$$K = \sum_{i=1}^{U} k_i, \tag{5.3}$$

with the total amount of unequipped aircraft U. The objectives fulfillment Δ_{ab} (*ab* for aircraft-based) is defined by

$$\Delta_{ab} = 1 - \frac{K}{K_{max}}. \tag{5.4}$$

Conflicts between unequipped aircraft have a different impact on this measure following the aircraft-based approach than conflicts between unequipped and equipped aircraft. In contrast, in the conflict-based approach the duration of conflicts is counted independently from the equipage of the involved aircraft. However, an impact depending on the equipage is acceptable as the performance of the human operator should be measured and the conflicts between equipped and unequipped aircraft can be attributed to the technical assistance as well. Furthermore, the amount of conflicts in which the same aircraft is involved at the same time is not considered in the aircraft-based approach. Nevertheless, this additional conflict affects the conflict duration of the third aircraft. Thus, the overall value K increases.

The aircraft-based approach has the advantage that it allows breaking down the overall evaluation to the individual aircraft. Thus, it facilitates the identification of the impact of each action. Consequently the aircraft-based approach is chosen.

As K_{max} is the flight duration of all unequipped aircraft, it could be calculated as the sum of the flight durations t_i of aircraft i. One problem with approach is that an increase of the flight duration would also increase the fulfillment of the objective *separation*. This is not plausible. To solve that issue, the interaction sequence calculated for the initial situation of an interaction is simulated and used as a reference. The resulting flight duration for aircraft i in that reference sequence is called $t_{0,i}$. Consequently, K_{max} can be calculated as

$$K_{max} = T_0 = \sum_{i=1}^{U} t_{0,i} \tag{5.5}$$

and the fulfillment of the objective *separation* is calculated by

$$\Delta = 1 - \frac{K}{T_0}. \tag{5.6}$$

It is now possible to specify the fulfillment of the objective *separation* δ_i for each individual aircraft i by considering only the duration this aircraft is involved in at least of conflict with

$$\delta_i = 1 - \frac{k_i}{T_0}. \tag{5.7}$$

The overall fulfillment can optionally be calculated based on the evaluations of the individual aircraft by

$$\Delta = 1 - u + \sum_{i=1}^{U} \delta_i. \tag{5.8}$$

The objective *separation* was measured similarly in previous studies with MAGIE. In the project FAGI [WO10], the conflict-based approach was used and the duration of all conflicts was accumulated. Further, the fulfillment was not scaled to a value between 0 and 1 so that it was measured by

$$\Delta_{FAGI} = D. \tag{5.9}$$

However, the comparison of interactions with different length is hindered without scaling to a standard range.

In the project FlexiGuide [JCH13], the aircraft-based approach was chosen and the fulfillment was calculated by

$$\Delta_{FG} = 1 - \frac{K}{T}. \tag{5.10}$$

The variable T was used instead of T_0 as the cognitive planning model was not yet available.

5.1.2. The Objective Constraints

In addition to keep the aircraft separated, the human operator must make sure that the eight constraints are respected, namely the limits for both, speed and altitude, defined for the four route sections (downwind, base leg, extended centerline, and final).

The fulfillment of the objective *constraints* Γ should also be between 1 (representing the optimal value) and 0 (representing the theoretical minimal value). To be in compliance with the instructions, the degree of fulfillment must decrease as long as the constraints are violated. Each additional violation must also reduce Γ. Further, there must be no difference between a violation of the altitude or the speed restriction, as no such difference is instructed.

Three different approaches are possible to calculate the degree of fulfillment. For all approaches $a_{l,i}$ gives the duration constraint l being active for aircraft i and $w_{l,i}$ gives the duration aircraft i violating constraint l. L gives the total of constraints and equals 8 (altitude/speed times four route sections). Following the constraint-based approach, the fulfillment is calculated for each constraint separately and afterward the median is calculated. Thus, with

$$a_l = \sum_{i=1}^{U} a_{l,i} \tag{5.11}$$

as the duration constraint l being active and

$$w_l = \sum_{i=1}^{U} w_{l,i} \tag{5.12}$$

as the duration constraint l being violated, Γ_{cb} (cb for constraint-based) can be expressed following the constraint-based approach by

$$\Gamma_{cb} = 1 - \frac{1}{L}\sum_{l=1}^{L}\frac{w_l}{a_l}. \tag{5.13}$$

This approach has the advantage that it helps to identify the constraints which are causing problems, as they are considered separately.

In the aircraft-based approach, first the fulfillment for each aircraft is determined and then the mean value is calculated. Consequently, with the duration of active constraints for aircraft i

$$a_i = \sum_{l=1}^{L} a_{l,i} \tag{5.14}$$

and the duration of their violation by aircraft i

$$w_i = \sum_{l=1}^{L} w_{l,i} \tag{5.15}$$

the degree of fulfillment according to the aircraft-based approach Γ_{ab} (ab for aircraft-based) is defined as

$$\Gamma_{ab} = 1 - \frac{1}{U}\sum_{i=1}^{U}\frac{w_i}{a_i}. \tag{5.16}$$

The advantage of this approach is that it concentrated on the aircraft and helps to identify the individual impact of each aircraft.

The common drawback of both approach is that the constraints respectively the aircraft are weighted equally. Although this sounds plausible, this has the consequence that the same duration of a violation of a constraint can have different impacts depending on the affected aircraft or constraint.

Consequently, the impact of each constraint (or aircraft) should be weighted with its contribution to the overall duration of active limits, which is

$$A = \sum_{i=1}^{U}\sum_{l=1}^{L} a_{l,i}. \tag{5.17}$$

This leads to the combined approach and Γ_c can by calculated by

$$\Gamma_c = 1 - \frac{1}{U}\sum_{i=1}^{U}\frac{w_i}{a_i}\cdot\frac{a_i}{A} \tag{5.18}$$

$$= 1 - \frac{1}{U}\sum_{i=1}^{U}\frac{w_i}{A} \tag{5.19}$$

$$= 1 - \frac{\frac{1}{U}\sum_{i=1}^{U} w_i}{A} \tag{5.20}$$

$$= 1 - \frac{W}{A} \text{ with } W = \frac{1}{U}\sum_{i=1}^{U} w_i \tag{5.21}$$

The transformation of this equation shows that the introduction of such a weighting merges both approaches. This combined approach has the drawback that neither the effect of a limit nor the effect of an aircraft can be calculated without knowing about the overall duration A the limits being active.

Additionally, in all three approaches the problem is that A can vary between two compared options while the overall duration of the violation of all constraints

$$W = \sum_{i=1}^{U} \sum_{l=1}^{L} w_{l,i}. \tag{5.22}$$

is equal. Consequently, the objectives fulfillment would vary while W is constant. Therefore, a similar modification as for the objective *separation* is used and $a_{l,i}$ is replaced by the duration the limits being active in the reference sequence calculated for the initial situation of an interaction called $a_{0,l,i}$. Thus with

$$A_0 = \sum_{i=1}^{U} \sum_{l=1}^{L} a_{0,l,i} \tag{5.23}$$

the degree of fulfillment of the objective *constraints* is defined by

$$\Gamma = 1 - \frac{W}{A_0}. \tag{5.24}$$

The fulfillment can be separated for speed and altitude constraints. With the introduction of A_0, the calculation of the impact of each aircraft respectively limit does not depend on $a_{l,i}$ anymore so that it can be calculated without knowledge about the other aircraft. Consequently, the fulfillment of the objective by aircraft i can be calculated by

$$\gamma_i = 1 - \frac{w_i}{A_0}. \tag{5.25}$$

or for only speed constraints by

$$\gamma_{i,speed} = 1 - \frac{w_{i,speed}}{A_0} \tag{5.26}$$

respectively for only altitude constraints by

$$\gamma_{i,altitude} = 1 - \frac{w_{i,altitude}}{A_0}. \tag{5.27}$$

Similar as for the objective *separation*, the overall fulfillment can be calculated based on the individual evaluations by

$$\Gamma = 1 - U + \sum_{i=1}^{U} \gamma_i \tag{5.28}$$

As in former studies [WO10, JCH13] calculated interaction sequences and $a_{0,l,i}$ were not available, the drawback of the effect of prolonged flight duration on the objective could not be solved. In this studies, the constraint-based approach was chosen and the fulfillment of the objective *constraints* was defined as

$$\Gamma_{FAGI/FG} = 1 - \frac{1}{L}\sum_{l=1}^{L}\frac{w_l}{a_l}. \tag{5.29}$$

5.1.3. The Objective Throughput

The degree of fulfillment of the objective throughput Θ must measure how much the operators succeeded with the task to guide the aircraft as early as possible to the airport. Here the idea is to take the time needed by the operator to guide all aircraft from their entry point to their exit point (T) and to compare it to the time needed in the initial solution (T_0). Consequently, the objective's fulfillment can be defined in a first step as

$$\Theta^* = 1 - \frac{T - T_0}{T_0} = 2 - \frac{T}{T_0}. \tag{5.30}$$

With this definition, Θ^* is equal to 1 if the aircraft are guided as quickly as in the sequence calculated for the initial situation of the interaction ($T = T_0$). As no maximal flight duration can be defined (fuel consumption is not simulated in MAGIE), is it not possible to define Θ such that is equals to 0 in the worst case. However, it is possible to define Θ such that it equals to 2 if this objective is maximized and both other objectives are ignored. Therefore, it is necessary to divide T into the time inevitably required for all aircraft to be guided from the entry to the exit and the time caused by additional guidance over the path-stretching area. Only downwind and extended centerline are considered as additional as the base-leg cannot be avoided. Consequently, with P as the flight duration of all aircraft in the path-stretching area and P_0 to denote the flight duration of all aircraft in the path-stretching area in the reference interaction, Θ can be defined as

$$\Theta = 1 - \frac{P - P_0}{P_0} = 2 - \frac{P}{P_0}. \tag{5.31}$$

The fulfillment of the objective can also be calculated for each aircraft i by

$$\theta_i = 1 - \frac{p_i - p_{0,i}}{p_0} \tag{5.32}$$

with p_i as the flight duration of aircraft i in the path-stretching area, and $p_{i,0}$ as the flight duration of aircraft i in the path-stretching area in the reference interaction sequence generated for the initial situation.

Similar to both other objectives, the overall fulfillment can be calculated based on the individual evaluations by

$$\Theta = 1 - U + \sum_{i=1}^{U}\theta_i \tag{5.33}$$

As no interaction sequences could be calculated in former studies with MAGIE, the comparison of the flight duration with the necessary flight durations was not possible. Furthermore, the interaction sequences were not supplemented by calculated sequences so that not every aircraft landed during the evaluated interaction sequences. Therefore, the amount of aircraft arriving at their exit point varied between simulation runs and this variable was used as a measure [WO10]. In [JCH13], the duration of each interaction was so short that no aircraft reached its exit in most interactions. Therefore, this amount was not suitable. Instead, the amount of aircraft which reached the base-leg was used as a measure for *throughput*.

5.1.4. Interdependence of Objectives

The degree of fulfillment for the objectives *separation* and *constraints* cannot be larger than 1. However this is possible for the objective *throughput* if the routes are shorter than in the interaction sequence calculated for the initial situation. Nevertheless, in the ideal case all objectives have the degree of fulfillment of 1. If shorter routes are advised, *throughput* increases but the separation to other aircraft will be violated and hence the corresponding fulfillment decreases. As *separation* is much more important than *throughput*, *throughput* should not be preferred for the cost of a decrease in *separation*

5.2. Procedure for the Evaluation of Decisions of Human Operators

After evaluation criteria are defined, the procedure of the developed method for the evaluation of human operators' decisions can be applied. The aim of this method is to evaluate each individual action and to identify the contribution of the different kinds of action and types of errors on the overall result of the interaction. The general idea is to evaluate the action depending on what was possible before the action was executed and what is still possible after the action is executed. The procedure developed to reach this aim consists of six steps which are explained in the following. As input to the procedure, a recorded interaction sequence is necessary. In the first step, the cognitive planning model developed in chapter 3, which is based on a CPN model of the controlled system and a set of rules defining the prototypical human operator behavior, is applied to generate interaction sequences for different points during the measured interaction representing what was possible at the respectively situation (step 1).

To compare the results of interaction sequences, all compared interaction sequences have to start with the same situation. Consequently, it is first necessary to combine the measured behavior and the generated interaction sequence (step 2). Then this combined sequence has to be simulated in order to get the results of this interaction (step 3). Finally, these results can be evaluated with the functions defined in the last section (step 4).

After that, an assessment of the actions measured between adjacent combined sequences is possible, as decreases in the objectives' fulfillment indicate erroneous actions (step 5). Finally, in step 6, it is possible to sum up the assessments of individual action and to identity the frequency and impact of specific kinds of actions and types of error.

5.2.1. Generation of Interaction Sequences

The application of the cognitive planning model developed in chapter 3 to generate interaction sequences is the first and most complex and extensive step of the procedure. The execution of this step has three preconditions. First, the process, with which the human operator is interacting, must be modeled as a CPN. Furthermore, prototypical behavior of the human operator for interaction with this process must be defined as a set of rules. Finally, an interaction sequence (interaction protocol) and its results (results protocol) must be recorded. The interaction protocol describes all actions occurred, whereas the results protocol contains all states that resulted from these actions.

Subsequently, some situations have to be selected from the results of the measured interaction sequence. For each selected situation, an interaction sequence is calculated indicating the reachable result. The aim is to evaluate actions of human operator between two selected situations depending on the actions' impact on the reachable result. On the one hand, situations directly before and after the execution of a measured action could be analyzed. This has the advantage that the measured difference can directly be attributed to the surrounded action. However, missing executions of required actions cannot be considered equally with this approach, as they do not take place as events but last for some time. To consider missing actions equivalently, situations are selected with regular intervals.

The size of the interval should be chosen so that in most cases the effect of the actions on the objectives' fulfillment is unambiguous. Thus, first of all, the size of the interval should depend on the frequency of actions. Furthermore, the fact that not every action affects each objective can be used to make a more precise assignment of actions to decreases of the objectives' fulfillment. This allows choosing larger intervals.

The selected situations are loaded into the CPN model of the controlled process and interaction sequences are generated. For details see chapter 3. Interaction sequences are generated for all selected situations which transform the respective situations into goal situations.

5.2.2. Combination of Measured and Generated Interaction Sequences

In the next step, measured and generated sequences are combined. A prerequisite is that both are available, in other words the interaction must be recorded and the first step of this procedure must be applied successfully.

It is necessary to evaluate the same time period of interaction sequences in order to gain comparable measures. For example, two interaction sequences are calculated for the first and last situation of a 10 seconds cutout of an interaction. During the whole cutout, a problem is present (which affects at least one objective). In the interaction sequence calculated for the first situation, the problem is solved after 15 seconds. In the interaction sequence generated for the last situation it is solved after 10 seconds. If only the two

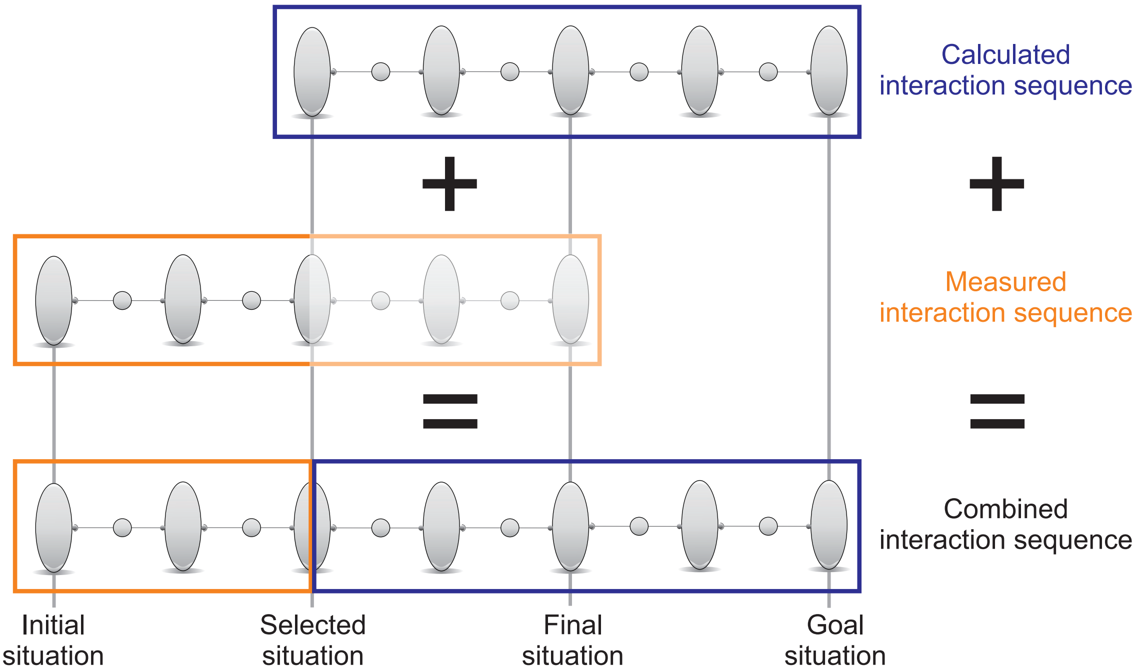


Figure 5.1.: The combination of a measured interaction sequence, which starts with the initial station and ends with the selected situation, and a calculated interaction sequence, which starts with the selected situation end ends with the goal situation

generated sequences are compared, a reduction of the duration from 15 to 10 seconds can be found. Actually, the problems duration is longer according to the seconds sequence as this sequence is generated for a later point in time. Starting from the first situation of the cutout, the problem lasts for 20 seconds. Thus, both interaction sequences can only be compared, if the interaction during the cutout is considered. Hence, to be able to evaluate the existing options for the selected situation, it is necessary to combine calculated and measured interaction sequences. This is illustrated in Fig. 5.1. The actions, which took place between the initial and selected situation, are taken from the recorded interaction protocol and are combined with the calculated sequences which transform the selected situation into a goal situation.

If several action sequences are compared, the starting time of the earliest sequence determines the point after which the measured behavior has to be taken into account. This is the initial situation of the measured interaction sequence. If this situation is selected, the combined interaction sequence contains only calculated actions. If other situations are selected, the combined sequence always contains both measured and calculated actions. This also holds for the final situation. Here the calculated interaction sequence contains the actions necessary to transform this situation into a goal situation.

After this second step of the procedure is executed, as many combined sequences result as situations were selected in the first step.

5.2.3. Simulation of Combined Interaction Sequences

To evaluate the combined interaction sequences, their results must be known. Results are available only for the measured sequences up to now. During the generation of interaction sequences, the effects of different options were determined, but no output was generated containing only the results of the interaction sequence found as solution. In addition, this output would not be sufficient, as the effects of the combined (and not only the calculated) sequences have to be considered.

To determine these effects, a small tool is developed which also uses Access/CPN [WK09] to modify the CPN of the controlled process directly. At first, the initial state is loaded into the Petri Net. In addition to the transitions modeling the dynamic behavior of the process, the transitions corresponding to the actions in the combined sequence are fired. In doing so, a result protocol results for each combined sequence.

5.2.4. Product Evaluation of Simulation Results

After the result protocols of the combined interaction sequences are generated, the protocols are evaluated. As they have the same format as the protocols of the measured behavior of the human operator, both can be analyzed with the same procedures.

The aim of the product evaluation of combined interaction sequences is to generate an output which is sufficient to assess the operators' actions. Therefore, not only the fulfillment of the objectives is calculated for each combined sequence. Additionally, actions are identified which can cause decreases of the objectives' fulfillment during the interval. In this step, each combined sequence is evaluated independently from other sequences.

Executed and missing actions can influence objectives. Consequently, the actions measured in the considers interval x (actions E_x in the measured interaction sequence), the actions planned for that interval by the model (planned actions P_x in the generated interaction sequence), and the difference between planned and given actions (missing actions) are considered. This is illustrated in Fig. 5.2. In this figure, the interval contains one action, which is not necessarily always the case. On the one hand, several actions are possible during an interval. On the other hand, intervals often contain no action.

To reduce the amount of actions to be considered and to allow a precise assignment of actions to decreases of the objectives' fulfillment in the next step of the procedure, the objectives and subsystems are considered separately. For each combination of subsystem and objective, a subset out of the given and planned actions is extracted as candidates for a potential decrease of the objective's fulfillment. As changes of the objective's fulfillment cannot be identified in this step of the procedure yet, all action which can possibly influence an objective come into question as candidates. The planned actions identified as candidates are called $P_x^* \subset P_x$ while the measured actions identified as candidates are called $E_x^* \subset E_x$.

If the objectives of a subsystem can only be influenced by actions related to that subsystem (isolated problems) a large amount of actions can be sorted out. Moreover, objectives and kind of actions can be split up in more specific ones if a more precise assessment becomes possible. Consequently, in addition to its fulfillment, a list of candidates which could cause a decrease during the next interval are generated for each objective.

Some actions concerning a subsystem can cause problems affecting another subsystem's objective. These problems are called common problems. For each of these common problems also a list of candidates is generated. The assignment of common problems to a subsystem's objective is conducted in the next step.

When candidates are identifies, it has to be taken into account that some actions can be undone nearly immediately. The consequences can be illustrated with the following example. In an interval, an aircraft is advised to increase its speed which leads to a violation of the constraints. During the same interval, no speed clearance was planned for that aircraft. This situation can be viewed from two perspectives. Following the first perspective, a wrong action was executed. Following the second perspective, the operator missed to revoke this action so that a missing action is causing the problem. It was decided to follow the first perspective, as the executed action is the primary reason. Hence, when an action is executed while it is not planned, the execution and not the missing withdrawal of this action is interpreted as an error. The example can be slightly modified to illustrate another point. The increase of speed is given while a decrease of speed was planned. In this case, the primary reason for possible problems is the missing action. Hence, when an action was planned and it is not executed in the interval, this missing action is considered as a candidate, even if another action (contrary to the planned action) was given during the interval.

If no candidate can be identified for a combination of subsystem and objective or for a common problem, an "undefined" reason for a decrease of the performance is specified.

The evaluation of a selected situation thus comprises the individual fulfillment of the objectives and candidates for erroneous actions for each objective. Additionally, a description for each common problem including the list of candidates affecting this problem is added. The evaluation of all selected situations is written in one file which is used as input for the evaluation of actions in the next step.

5.2.5. Evaluation of Actions by Comparison of Adjacent Sequences

To evaluate actions, the result evaluations of two adjacent selected situations are compared. This is illustrated in Fig. 5.3. The situation at the beginning of the interval x is called pre-situation (S_x) and the situation at the end of the interval is denoted as post-situation (S_{x+1}). The aim of this step is to generate a list containing all erroneous actions causing a performance reduction together with the corresponding effect on the objectives' fulfillment and additional details about the action. The list of erroneous actions is basically a subset of the list of candidates identified in the last step.

If the fulfillment of an objective decreases during the interval, the human operator lost the option to reach the performance possible in the pre-situation. The actions executed by the operator in the interval (E_x) must deviate from the actions planned by the model P_x. However, this difference between measured and planned actions alone does not

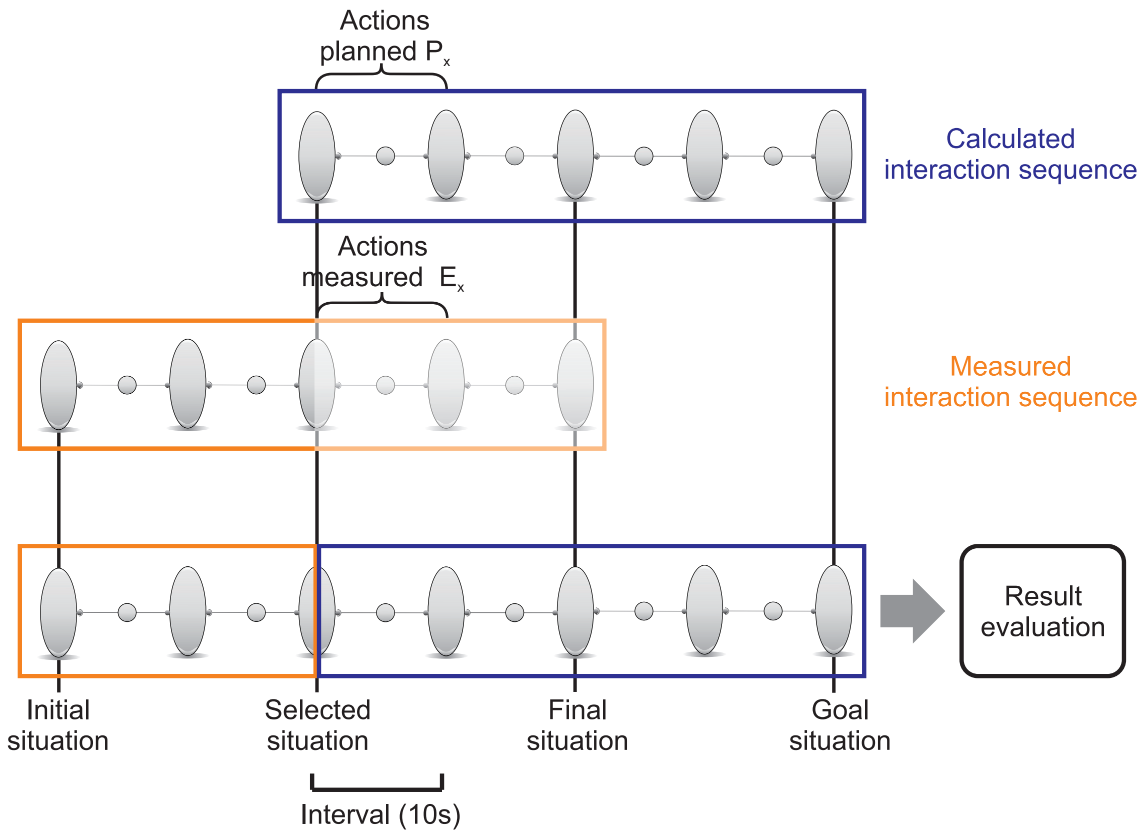


Figure 5.2.: Evaluation of combined sequences

indicate an erroneous action as different options (and thus actions) can lead to the same result. A reduction of the reachable performance must occur at the same time. Only if the objective is decreasing and the measured actions deviate from the planned action, the action implemented by the human operator was not the best option available. Thus, only if a decrease of performance can be assigned to an action identified as candidate in the last step, this action was erroneous. Consequently, these reductions are identified first in this step and are consequently combined with the candidates identified in the last step.

After a performance reduction between two compared interaction sequences is detected, the erroneous actions must be identified out of the candidates generated in the last step. To enable a precise assignment of erroneous actions to performance reductions, the performance as well as the actions are considered for each subsystem separately.

This step is straightforward for objectives of a subsystem, which can only be influenced by actions concerning this subsystem. In this case, the procedure illustrated in Fig. 5.4 is followed. If the set of given actions identified as candidates E_x^* in the last step is not empty, this set of actions is considered as erroneous and added to the list of errors as commission. If no commissions are identified and the set of missing actions identified as candidates in the same combined sequence P_x^* is not empty, these actions are added to the list of errors as omissions.

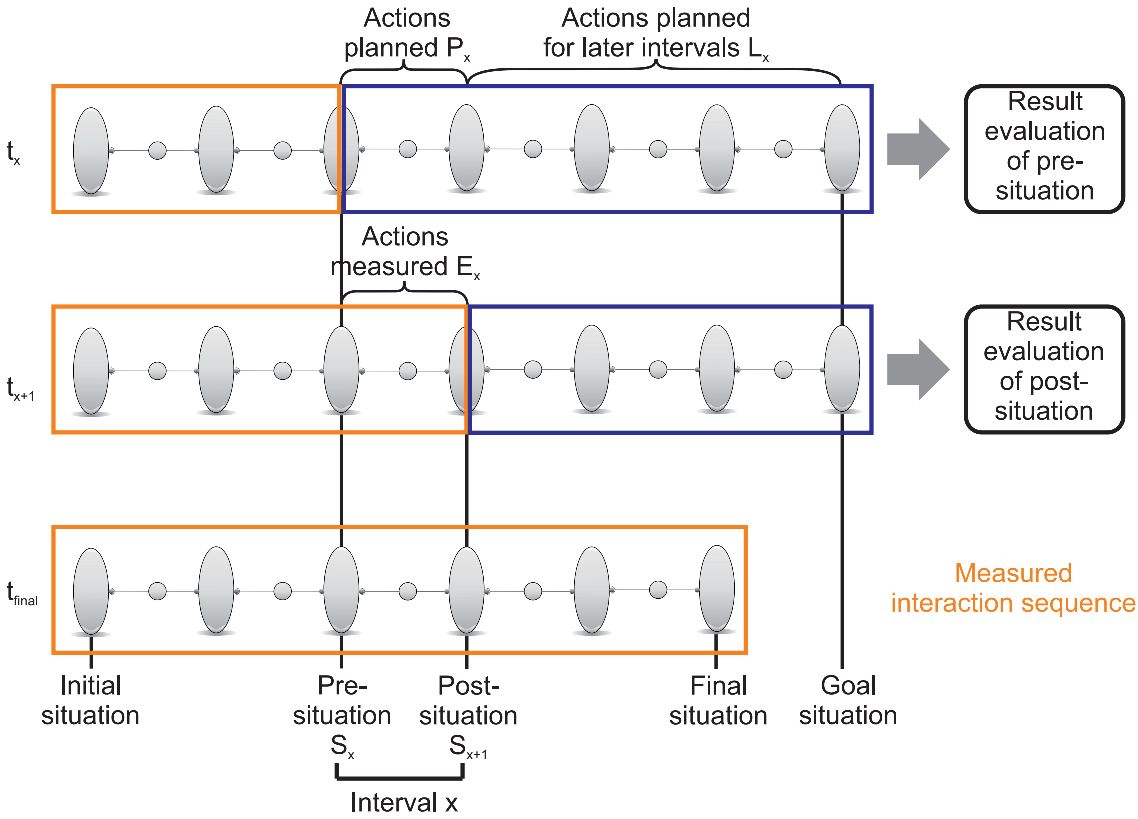


Figure 5.3.: Comparison of combined interaction sequence generated for pre- and post-situations. The actions between both situations in the observed interaction are called observed actions and the actions between both situations in the combined sequence generated for the pre-situation are called planned actions. Note that although situation in various sequences are denoted the same, they differ in general.

Additionally, objectives which can be influenced by common problems have to be considered. Here a list of candidates relevant for a subsystem's objective has to be generated out of the list of candidates for each common problems. Thus, the connection between the fulfillment of a subsystem's objective and the common problems has to be made. Out of the common problems in the combined sequences generated for both the pre- and post-situation, only these problems are of interest, which take place during or after the interval and in which the considered subsystem is involved. Furthermore, problems can only have a negative impact on the objective if they are created or prolonged during the interval. Thus, the common problems in the combined sequences for the pre- and post-situation are compared to identify created and prolonged problems. The candidates assigned to the created or prolonged problems are then summarized to constitute the candidates for a reduction of a subsystem's objective's fulfillment (E^*_x for given actions and P^*_x for missing actions). If no candidates were found for a common problem in the last step (and an "undefined" reason was specified), these reasons are ignored in the combined list. Subsequently, the procedure illustrated in Fig. 5.4 can also be applied for these candidates.

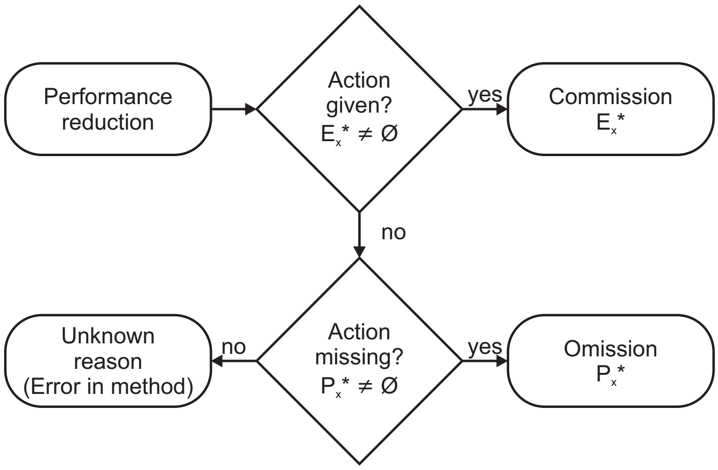


Figure 5.4.: Process to identity errors out of the candidates if a performance reduction is detected

Following this procedure, an error classification similar to this of the signal detection theory (see section 2.3.1) result. The difference between actions required by the environment and actions as responses of the operator are detected.

The list of errors generated this way contains the following details for each action. First, the execution time of an action is given. To be able to handle also periods without measured actions (namely omission errors), the beginning of an interval in which the action is executed or omitted is used. Second, a description of the influenced objective is given. Further, the kind of actions and the type of error is added (omission or commission). Finally, the performance lost due to this erroneous action during the interval is given.

If multiple actions are identified for a decrease of performance, the decrease is divided between all actions equally and it is expected that they all have the same impact. However, this should rarely occur due to the chosen interval size, the consideration of each subsystems separately, and the assignment of actions to objectives.

This classification of actions can be extended so that errors according to the definition in the time window method result (see section 2.3.1). This extension is illustrated in Fig. 5.5. Instead of classifying all given actions E_x^* causing a performance reduction as commission, a more precise assignment follows for each action $e^* \in E_x^*$. At first, it is checked if the action is also a planned action. Here the intersection of given E_x^* and planned actions P_x^* results. As types of actions are considered, this identifies incorrect actions which are planned but executed in a different way than planned. For example, action with an inappropriate intensity are incorrect. Actions not identified as incorrect actions are checked if they are planned for later intervals and element of L_x^* (see Fig. 5.3 for an illustration). This comparison identifies the early actions. Actions neither classified as incorrect or early actions are compared to the actions planned for earlier situations A_x^* in the next step. A_x^* is defined as the union of previously calculated

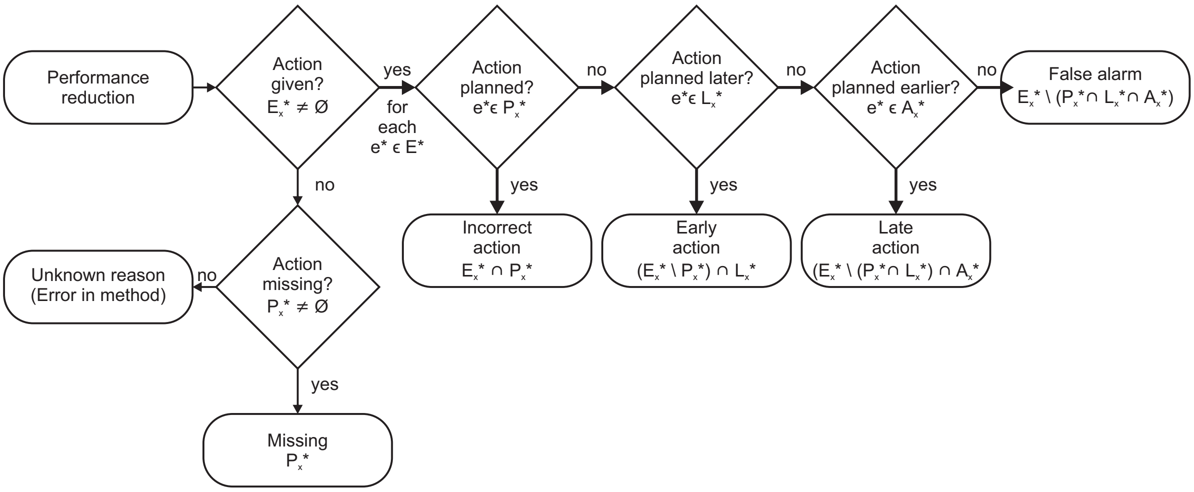


Figure 5.5.: Possible extension of the error identification process according to the time window method

interaction sequences by $A_x^* = P_1^* \cup L_1^* \cup P_2^* \cup L_2^* \cup ... \cup P_{x-1}^* \cup L_{x-1}^*$. Here the late actions are identified. Given actions not part of one of the other considered sets of actions are regarded as false alarms. Following this procedure, all actions are classified according to the complete classification of the time windows method as depicted in Fig. 2.16. However, this extended classification does need not only two adjacent sequences as input but all previous sequences. Due to this increased effort, the extended classification is not applied in this thesis.

5.2.6. Identification of the Impact of Kind of Actions and Types of Error

In the last step of the procedure, the impact of the different kind of actions and types of errors on each objective is determined. Therefore, the list of erroneous actions and their impact created in the last step is used and the elements of this list are combined stepwise. It was decided to combine elements with only one different characteristic at a time. As some information is lost in each step, an output of the results is generated after each step. In each output, the lost performance due to omissions and commissions is given for each possible combination of objective and kind of action.

In the first aggregation level, list items are summarized which are identical expect for the impact on the objective. As subsystems were considered individually, actions, which influenced common problems and thus the fulfillment of the objective *separation* for two subsystems, were added multiple times. These items are combined now.

Then, the same actions with the same effect are combined. Consequently, items with a different time of execution are merged (all other characteristics, expect the impact, must be the same). If objectives were split up to allow a more precise assignment of actions to lost performance, these are reunited in the next step. In a further step, also kinds of actions, which were separated into more detailed kinds, are merged again. For each remaining combination, the effect of both types of errors (omission or additional action) is calculated.

If these results are further summarized, important information will be lost. Either, the kind of action, the type of error, or the objectives would be combined. However, the aim of the whole method was to identify the impact of each action and type of error and therefore the most important additional benefit of the method would be lost. Additionally, when it is no longer differentiated between different actions and errors, similar result could also be achieved by calculating only one interaction sequence to transform the final situation of the measured interaction into a goal situation so that a much smaller effort would be necessary. A combination of the different objectives would require defining a weighting of these objectives.

5.2.7. Application of Procedure

The above described procedure is applied to the microworld MAGIE. First, five different actions are considered, which are to increase and to decrease speed and altitude and the initiation of the turn maneuver. Second, the objective *constraints* is split up in speed and altitude constraints, as violations of speed constraints are caused by changes of the speed and the same holds for altitude constraints. Both allow a more precise assignment of actions to objective. As additionally, the frequency of actions is rather low, an interval size of 10 seconds is chosen.

In MAGIE each aircraft can be considered as a subsystem. The objectives *constraints* and *throughput* are the result of isolated problems for each subsystem whereas the objective *separation* is influenced by common problems. Hence, the product evaluations of the results of interaction sequences in MAGIE, contains data about each aircraft and additionally about each conflict. For the evaluation of an aircraft, the individual fulfillment of the objectives is given, whereby the fulfillment of the objective *constraints* is split up in speed and altitude constraints. To determine the fulfillment of the objectives, the equations 5.7, 5.26, 5.27, and 5.32 are used.

Candidates are determined from given actions, planned actions, and missing actions. Only the objectives' fulfillment and candidates would be necessary, but additionally the call-sign is given to identify the aircraft and some durations are included (flight duration, flight time in compliance with restrictions, flight duration in the path-stretching area) to enable a reconstruction of the objectives' fulfillment. For each conflict, which occurred in the combined interaction sequence, candidates are determined. These are (given or missing) actions concerning one of the involved aircraft. Additionally, details of the conflict, like its duration and the aircraft's call-sign, are determined.

Candidates for a decrease of the performance of the objective *separation* are initiations of turn maneuver and missing and given speed clearances. As the objective *separation* can be influenced by actions concerning both aircraft in a conflict, the candidates concerning each individual aircraft are not exhaustive. For the violation of speed constraints, only missing and given speed clearances are candidates. Missing and given altitude instructions are considered accordingly for altitude constraints. For a reduction of the throughput, several actions come into question. First, missing action which directly influence the throughput are considered, which are missing initiations of the turn maneuver and missing speed and altitude changes. In addition, given initiations of the turn and given speed instructions are considered in subordinate ranking. Regarding the conflicts, actions concerning one of the involved aircraft are candidates. These actions are first initiations of the turn maneuver, then missing actions, and finally other given actions.

Then the result-evaluations of sequences generated for adjacent selected situations are compared to assess the individual actions within the interval. When the evaluation results are summarized, objectives and actions split earlier are combined again. First, it is no longer differentiated between speed and alt constraints. No information should be lost here, as these objectives are influenced by different kind of actions. Further, it is no longer differentiated according to the direction of changes of speed or altitude. Hence, the amount of different actions is reduced from five to three. After that step, the different combinations are reduced to nine, three objectives (*separation*, *constraints*, and *throughput*) and three kinds of actions (initiation of turn, change of alt, a change of speed). For example all changes of speed causing conflicts are combined. To compare the influence of the turn maneuver to the impact of all other actions, additionally speed and altitude changes are combined finally. After this step, the highest level of aggregation results, which contains six different combinations (three objectives and two kinds of actions).

To combine the objectives, a weighting would be necessary. But as already mentioned in section 5.1, the definition of a weighting would be arbitrary, as only a ranking was instructed. Consequently, the objectives are not combined.

The lost performance of the objective *separation* Δ^- is defined as the sum of the impact of all actions reducing the *separation* performance. Similarly, the lost performance is defined for the objectives *constraints* Γ^- and *throughput* Θ^-. The fulfillment of the objectives *constraints* and *throughput* can increase during the interaction (at the cost of a decrease in *separation*) and only decreases of these objectives are considered for the calculation of the lost performance. Therefore, in general: $1-\Gamma^- \neq Gamma(finalSituation)$ and $1-\Theta^- \neq Theta(finalSituation)$. However, as the *separation* performance can not increase during an interaction $1-\Delta^- = \Delta(finalSituation)$.

5.3. Demonstrative Application

To demonstrate the method described in the last section, it is applied to a measured interaction sequence with a duration of 600 s. The results are reported in the following. A step size of 10 s is chosen and 61 sequences are calculated, combined with the measured sequences, simulated and finally evaluated. The results of the evaluation of all combined interaction sequences with respect to the objective *separation* are shown Fig. 5.6. At first, periods in which a conflicts occurred in the measured interaction sequence are highlighted in light red. The first conflict lasted from 160 s – 171 s and the second conflict lasted from 284 s – 297 s. In the figure further the fulfillment of the objective *separation* during the interaction according to equation 5.6 is given as black squares. At the beginning, this objective is fulfilled completely and decreases to 0.9778 during the interaction. The value decreases during two periods, which are each located before both conflicts, while it remains constant otherwise. This shows that the achievable performance is dropping way ahead of a conflict. In other words, once the performance first decreases, the conflict cannot be avoided. The more the fulfillment of the objective decreases, the longer the conflict will inevitably last. Actions causing the conflict or actions that could have prevented the conflict are to be found during the two periods in which the reachable performance is decreasing and not at the time the conflict realizes. This periods can only be identified if the combined interaction sequences of measured and calculated behavior are analyzed. To support this claim, the fulfillment of the objective *separation* is also calculated and shown in this figure (gray triangles), if only the measured interaction sequences are used as an input (and the calculated interaction sequences are ignored completely). In this case, the performance is decreasing only when the conflict finally realizes. This shows that the measured and calculated interaction sequences have to be analyzed to detect possible reasons for a loss of performance.

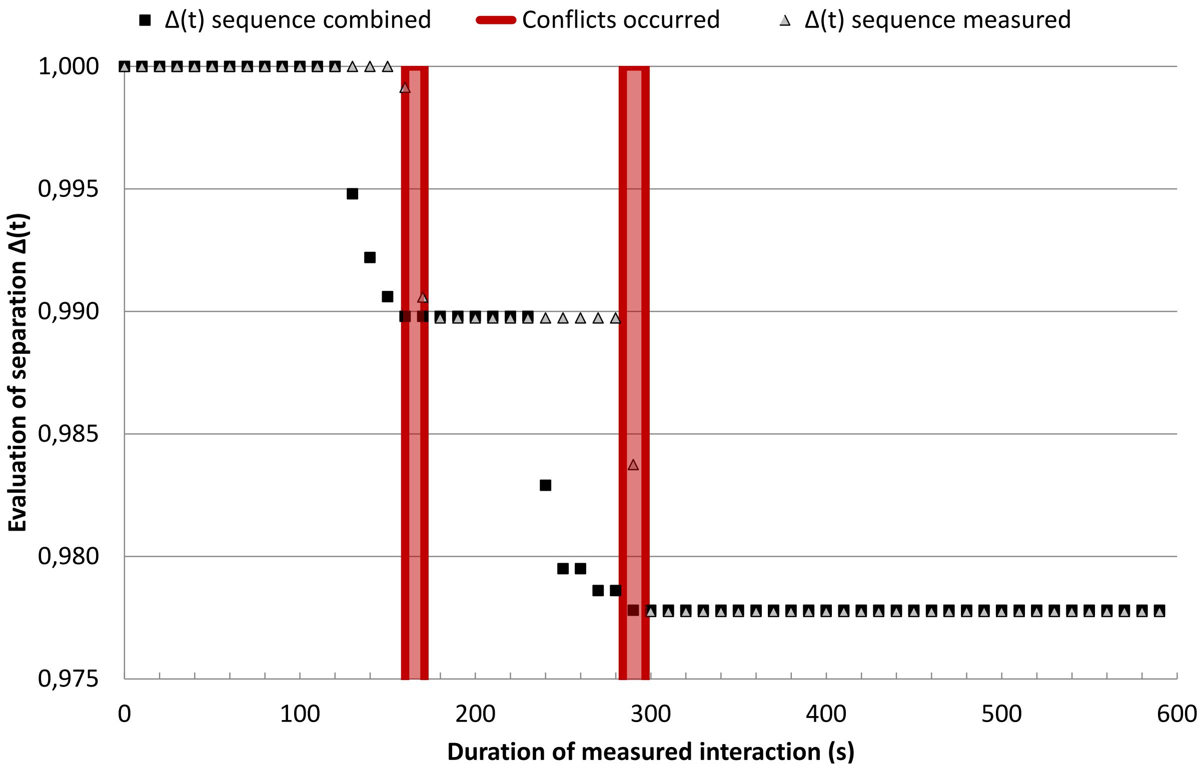


Figure 5.6.: Fulfillment of the objective *separation* by combined interaction sequences (black squares) and only measured interaction sequences (as gray triangles). Durations with conflicts in the measured interaction sequence are highlighted with red (160 s – 171 s and 284 s – 297 s).

The fulfillment of the objective *constraints* during the same interaction sequence is given in Fig. 5.7. Both, the fulfillment of the altitude constraints (as blue squares with crosses) and of the speed constraints (as blue squares with x) are shown separately.

Additionally, the fulfillment of the combined objective is indicated (as blue squares with stars, calculated with equation 5.24). While the speed constraints are never violated (the objective is fulfilled completely at the end of the interaction), a drop in the fulfillment of the altitude constraints indicates a violation. The combined objectives consequently drops by the half amount and reaches 0.9884 finally.

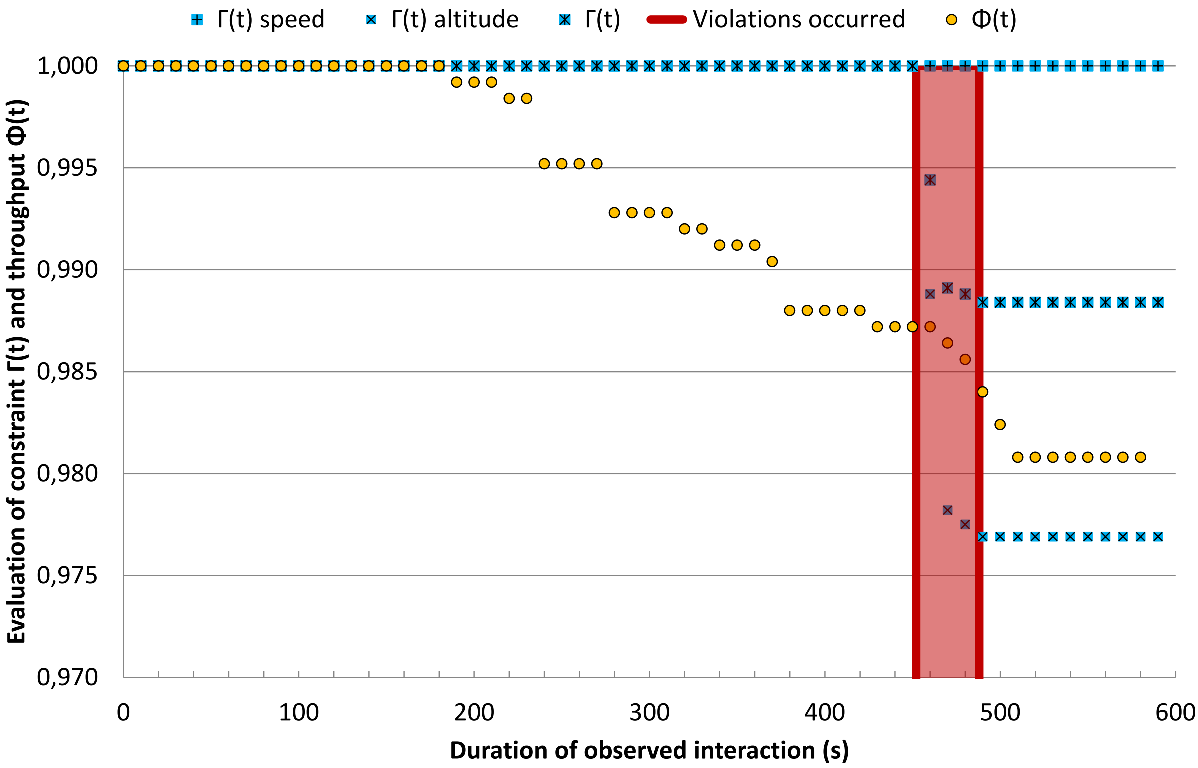


Figure 5.7.: Fulfillment of the objective *constraints* by all constraints (blue squares), by only speed constraints (+), by only altitude constraints (x), and fulfillment of the objective *throughput* (yellow circles). Durations with violations of constraints in the measured interaction sequence are highlighted with red (452 s – 488 s violation of altitude restriction)

The violation causing the drop last from 452 s – 488 s is highlighted in the figure. Here, in contrast to the time curve of the objective *separation*, the objective drops only when the violation occurs although the combined interaction sequence is considered. This indicates that the violation was either caused by an action with direct consequences or was preventable till it finally realized. In fact, the violation of the altitude constraints could be prevented as an aircraft was descending below the minimum allowed altitude and needed to be instructed to stop the descent.

In the figure additionally the objective *throughput* is given as yellow circles. The value of this objective decreases to 0.9808 during the interaction. The aircraft needed longer then necessary to reach the end of the sector. The objective is constant at the beginning and decreases with several smaller steps in the following. The frequent changes indicate, that this objective is affected by many erroneous actions.

Based on the time curve of the objectives, the actions of the human operator are evaluated in the next step. For example, the aircraft U90 and U74 have a predicted conflict at the time point 120. In the following interval (120 s – 130 s), it is planed to accelerate U90 and to decelerate U74. The objective *separation* is decreasing in this interval by 0.026 for each aircraft. As the objective is decreasing for the first time, the conflict becomes inevitable during the interval. Since it is not clear which of both missing changes could prevent the conflict, the loss of performance for each aircraft is assigned to both actions, the missing acceleration of U90 and the missing deceleration of U74. Consequently, four list items are added to the list of erroneous actions.

Later, at time 230 s, U74 and U23 have a predicted conflict. In the following interval, an acceleration of U74 is planned, but this action is not executed and thus missing. Since there are no other executed or missing actions, not only the decrease of the objective *separation* for aircraft U74 by 0.034 is assigned to the missing acceleration, but also the decrease for aircraft U23 by 0.035. (The difference between both changes is cause by rounding errors.)

In the next step the evaluation for each action and aircraft are summarized. First, evaluations for the same time-step and the same kind of action and type of error are combined. In the above example, in which U90 and U70 are in conflict and speed changes were planned for both aircraft in the interval 120 s – 130 s, the decrease of the objective *separation* was divided between both actions so that four elements where added to the list of erroneous actions (two for each aircraft, two for each action). Now the list items for the same action but regarding different aircraft are combined. As each original elements had on impact of 0.013 on the objective, both combined items have an impact of 0.026 each. Both elements can be seen in the first and second line of Table 5.1. In this table the results of the first aggregation with an impact on the objective *separation* are given.

On the highest level of aggregation, Table 5.2 results. This table gives for all six combinations of objectives and actions (turn or other action) the impact of both types of errors, which are the execution of unnecessary additional actions (commissions) and omissions. The results in this table show that no additional actions were executed in the measured interaction sequence. Only omissions caused decreases of the objectives. Furthermore, all decreases can be attributed to changes of altitude and speed. The initiations of turn maneuvers had no influence on the objectives, neither additional initiations (which includes too early initiations) nor omissions (which includes delayed initiations)

5.4. Adaption of the Planning Model to Predict Behavior

As argued in chapter 4, the human operators should not be compared to an optimum which cannot be hold during training or operation. However, the evaluation presented in the last sections uses optimal interaction sequences as criteria. Consequently, it is no surprising result that the measured efficiency is steadily decreasing during scenarios. In the last chapter, the integration of prediction uncertainty into a CPN model was explained. This enables more realistic predictions of human behavior and consequently

Table 5.1.: List of erroneous actions in a measured interaction sequence with impact on the objective separation. The time of execution, the type of error, the kind of actions, and the decrease of the objective's fulfillment due to each action is given

t	Type of error	Kind of action	$[\Delta(t) - \Delta(t+10)] \cdot 10^3$
120	Omission	SpeedUp	26
120	Omission	SpeedDown	26
130	Omission	SpeedUp	12
130	Omission	SpeedDown	12
140	Omission	SpeedUp	8
140	Omission	SpeedDown	8
150	Omission	SpeedUp	4
150	Omission	SpeedDown	4
230	Omission	SpeedUp	69
240	Omission	SpeedUp	34
260	Omission	SpeedUp	8
280	Omission	SpeedUp	4
280	Omission	SpeedDown	2

more realistic measures of human performance. The integration of such an uncertain CPN model into the cognitive planning model (see section 3.6) to generate more realistic predictions of operators behavior is explained in the following of this section. The implemented adaptations of the planning model to integrate uncertainty are already published in [HS14]

The cognitive model of human operators planning must be modified slightly to generate plans under uncertainty. Up to now, the model consists of a set of rules and a CPN of the controlled system. First, this CPN has to be replaced by a net also modeling uncertainty (see chapter 4 for details). Further, some definitions used by the rules have to be modified to scope with the uncertainty. If continuous values, which are associated with uncertainty in the model, have to be kept between certain limits, neither the maximal nor the minimal prediction error should exceed the limit. Thus, the minimal predicted values has to be compared to lower limits and the maximal predicted values has to be compared to upper limits. If this is strongly obeyed, the limits will never be violated.

The rules implemented in MAGIE can be changed in the following way. If the minimum of all calculated distances is below the minimum separation distance, a conflict is detected and the corresponding rule is activated. Furthermore, the rules defined to comply with the constrictions are modified to check both the minimal and maximal predicted value. After these changes are made, the planning model can generate human-operator-like plans including prediction uncertainty.

If the model generates plan without uncertainty, a planned interaction sequences will

Table 5.2.: Most aggregated results of the evaluation of an observed interaction sequence showing the overall impact of a combination of action and error on each objective

		$\Delta^{-} \cdot 10^3$ per type of error	
Objective	Kind of action	Commission	Omission
Separation	Start turn	0	0
	Altitude / speed	0	217
Constraints	Start turn	0	0
	Altitude / speed	0	231
Throughput	Start turn	0	0
	Altitude / speed	0	192

be executed exactly as planned. However, if uncertainty is included, a difference between the plan and the later execution will result. When generating a plan, the decisions about actions and their timing are based on predictions which have the time of the plan's generation as reference point. While the plan is executed it can be adapted continuously. Consequently, decisions can be based on predictions that have the time of the execution of an action as reference point. Thus the prediction horizon is shorter, the uncertainty is lower, and the actions fit more to the real conditions.

If now uncertainty is considered by the model of human operators planning, the generated sequences are planned interaction sequences but not predictions of human behavior. This difference was not present as long as exact plans were generated. Consequently, an additional modification of the model is necessary to generate interaction sequences as predictions for actual behavior. Therefore, it is important to allow an update of a plan when better (more exact) prognoses are possible.

When a plan generated under uncertainty is executed, the accuracy of predictions is steadily increasing allowing updates of the plan. However, often replanning is elaborately and only small increases of accuracy are expected. One possibility is to generate a more precise plan only right before an action is executed to check if this action is still necessary at this point in time or if it should be delayed or canceled. The action is executed only if it is confirmed by the updated plan. Independent from the execution of this action, the plan is updated again right before the next action is planned to be executed (according to the updated plan). This action has to be confirmed or refused by a further update of the plan. This realizes an alternating between an execution and a planning phase as described in the model of Mumford [MSD01] detailed in Section 2.1.4. Instead of an refinement of the plan as explained in this model, the plan is completely regenerated here.

The cognitive planning model is modified in three ways to get a model of human-like interaction. These modifications are a further rule, a variable indicating the execution horizon t_e of an interaction sequence and a function to reset uncertainty. The additional rule specifies that after an interaction sequence was generated, which represents a plan in the first run, the algorithm jumps back to the state right before the execution of the first planned action. The time at this state is stored in the added variable t_e and the uncertainty of this state is removed. In other words, the uncertainty range for each variable is set to zero. Specifically, the type of each variable represented as a bunch of rays (`type bunch`) stays the same but each ray is set to the value of the middle ray which is the exact value. By activating this rule, the execution of a part of the plan is simulated. As the jump to the state right before the execution of an action removes all later actions from the iteration sequences, a new plan has to be generated next. This plan uses prediction with the execution horizon t_e as reference point. The resulting interaction sequence thus contains an expected behavior (before t_e) and a plan (after t_e). Now the added rule is activated again, the algorithm jumps back to the state right before the first action in that part of the interaction sequence representing a plan. This state is transformed into an exact state and the time of this state is stored as new execution horizon. This procedure is repeated until an interaction sequence is generated which contains no action after the execution horizon and hence contains not a plan but an expected behavior.

5.5. Demonstrative Application with Prediction Uncertainty

In the implemented model of human behavior, *separation* and *constraints* must be observed under all predicted consequences. Thus even an interaction sequence generated with simulated projection imprecision must fulfill both objectives to the same extend as an interaction sequence generated with precise predictions. However, an increase of projection uncertainty leads to inefficient actions, as the turn of aircraft may be delayed or speed reductions may be executed earlier than necessary. This results in a decrease of the objective *throughput*.

The projection uncertainty parameter e has to be defined, before interaction sequences can be generated. It indicates how much the predictions deviate from the exact value. The parameter should be chosen in such a way that the operators behavior can be predicted as accurate as possible. In other words, interaction sequences generated for earlier situations during an interaction should fulfill the objective *throughput* as much as measured interaction sequences at later situations. Consequently, the objective *throughput* should remain approximately constant during the interaction.

Several scenarios were generated for MAGIE. They have different demands in terms of operators' guidance precision as the distance between two equipped aircraft is varied systematically between scenarios. In the scenario evaluated in section 5.3, the distance between two equipped aircraft is $t_{s(2E,high)} = 142\,\mathrm{s}$. An unequipped aircraft placed in the gap between those two equipped aircraft has to be guided with a precision of $\pm 3.5\,\mathrm{s}$. In other scenarios, the interval between two equipped aircraft is increased to $t_{s(2E,medium)} = 150\,\mathrm{s}$ respectively $t_{s(2E,low)} = 158\,\mathrm{s}$. The guidance of unequipped aircraft between two equipped aircraft requires a lower precision of $\pm 7.5\,\mathrm{s}$ respectively $\pm 11.5\,\mathrm{s}$.

To determine the value of the projection imprecision parameter e, which leads to an approximately constant fulfillment of the *throughput*, several values are simulated and compared. In Fig. 5.8 the fulfillment of the objective *throughput* is shown for the same scenario and interaction analyzed in section 5.3 with interaction sequences generated with the projection uncertainty parameter set to $e = 0.05$. This scenario requires the highest guidance precision of ± 3.5 s.

Most striking in this figure is the jump at 480 s. While the cognitive model of operator behavior could not find a sequence to guide an unequipped aircraft into a gap between two equipped aircraft, the human operator initiated the turn maneuver. Consequently, the *throughput* increases significantly in this moment, as the aircraft's trajectory will be much shorter than predicted. However, the fulfillment decreases almost continuously during the interactions (with this one exception). This means that the human operator's decisions are less efficient than predicted by the cognitive model of operator behavior most of the time.

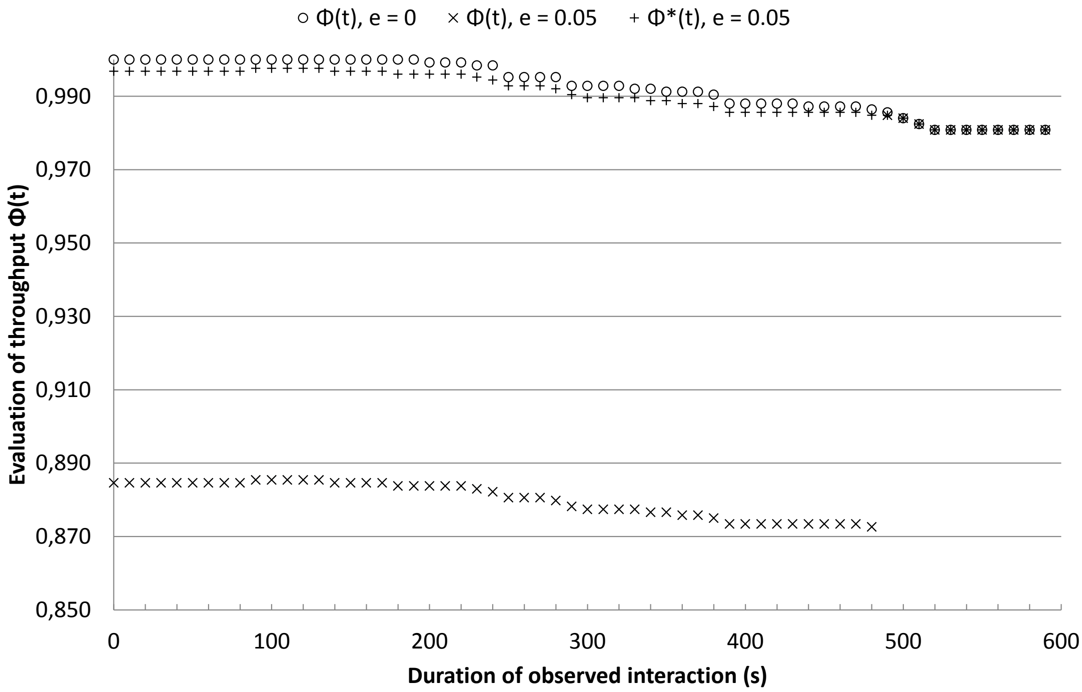


Figure 5.8.: Fulfillment of the objective *throughput* in a high guidance precision scenario (± 3.5 s) with $e = 0$ (circles) and $e = 0.05$ (x). The "+" indicate the throughput performance if the effect of the delayed turn is neglected.

In Fig. 5.9, the fulfillment of the objective *throughput* for an interaction in a scenario with a distance between two equipped aircraft of $t_{s(2E,medium)} = 150$ s and a resulting required medium guidance precision of ± 7.5 s is given. The *throughput* calculated with precise predictions (circles) and with a projection imprecision parameter set to $e = 0.15$ (crosses) is given in the figure. Although the difference between the fulfillment of *throughput* predicted by the model for the initial situation of the interaction and the fulfillment reached by the human operator in the last situation of the interaction is quite small if $e = 0.15$, the fulfillment has three distinctive jumps during the interactions. Besides these jumps, the *throughput* calculated with $e = 0.15$ decreases during the interaction most of the time. However, these jumps are much smaller than the jump of the *throughput* with $e = 0.05$ depicted in Fig. 5.8. They are again related to the initiation of turn maneuvers. Here the human operator starts the maneuvers earlier than predicted by the cognitive planning model which causes a higher than expected *throughput*.

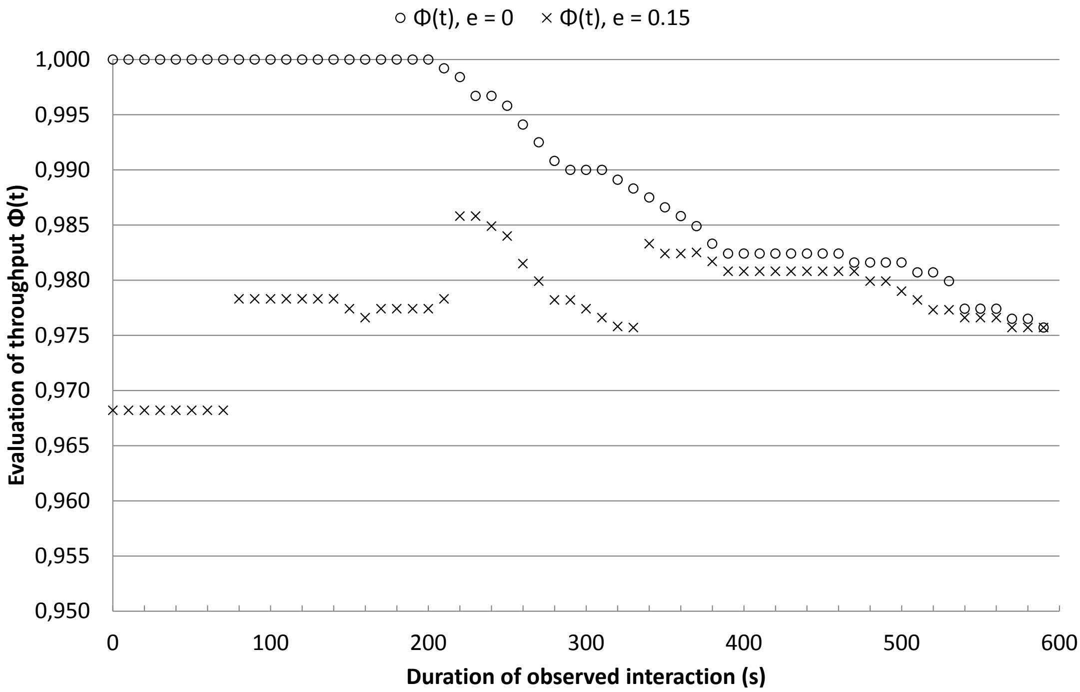


Figure 5.9.: Fulfillment of the objective *throughput* in a medium guidance precision scenario (± 7.5 s) with $e = 0$ (circles) and $e = 0.15$ (x)

In Fig. 5.10, the fulfillment of the objective *throughput* is given for a scenario which requires the lowest guidance precision of ± 11.5 s for unequipped aircraft as the distance between two equipped aircraft is $t_{s(2E,low)} = 158$ s. As a lower guidance precision is required in this scenario, larger values can be chosen for the projection imprecision parameter e without the consequence of some gaps missed by unequipped aircraft. In the figure, the fulfillment of the objective *throughput* is compared for e set to 0, 0.18, and 0.20. The interaction sequence generated by the cognitive planning model with $e = 0.18$ for the initial situation of the interaction reaches a higher throughput than the human operator reached at the end of the interaction indicating that the model overestimates human operators' performance in general. During the interaction, the *throughput* is mainly decreasing which indicates that also in most decisions the performance is overestimated by the model. If the projection imprecision parameter is increased to $e = 0.2$, the prediction underestimates human operator's performance in general as the fulfillment reached at the end of the interaction is higher than the fulfillment predicted at the beginning of the interaction. However, small jumps occur again when the operator initiates turn maneuvers, indicating that these are executed earlier than predicted by the model. Thus, the performance's underestimation is mainly due to these jumps.

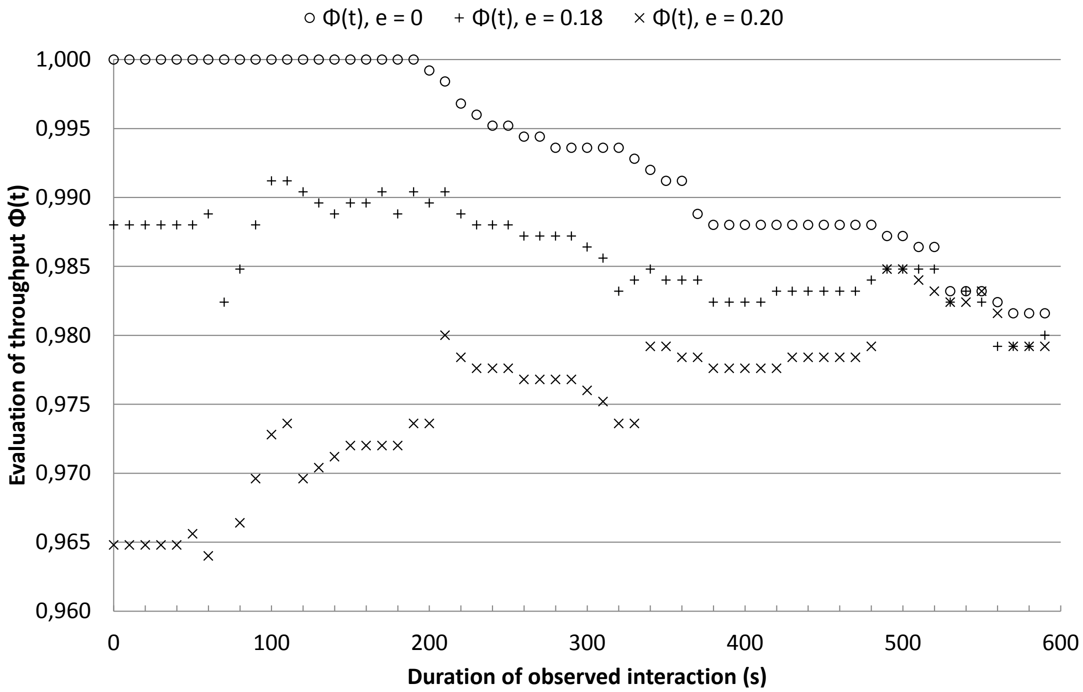


Figure 5.10.: Fulfillment of the objective *throughput* in a low guidance precision scenario ($\pm 11.5\,\mathrm{s}$) with $e = 0$ (circles), $e = 0.18$ (+), and $e = 0.20$ (x)

5.6. Discussion of Prediction Uncertainty

The integration of uncertainty should enable the comparison of human operators to an expected behavior instead of an optimal behavior. Thus interaction sequences calculated under uncertainty should be able to predict the human operators behavior. Various uncertainty parameters were simulated for scenarios, requiring different precisions. The results are ambiguous.

In the first scenario, which requires a very high precision, an assumed uncertainty of 5% leads the model to miss a gap for an unequipped aircraft. The assumed uncertainty of 5% seems to be too large for the required precision. However, apart from omitting a gap, the human operator is less efficient with most of his actions than predicted by the model. From this point of view, the assumed uncertainty of 5% seem to be to low to predict the behavior. This suggest several conclusions. First, the model should assume a larger uncertainty. Second, as a gap is missed, the modeled behavior to choose the right moment to initiate the turn should be changed to fit to the measured behavior. For example, the human operator could decide to initiate a turn even though he is not able to predict that the aircraft will fit into the gap and thus takes a risk. Furthermore, the operator could apply another strategy as modeled by the rules. The strategy used by the operator would then be less efficient than the modeled strategy.

In the second example, the *throughput* reached by the human operator at the end of the interaction can be predicted by the model. However, the jumps in the course of the predicted *throughput* under uncertainty indicate that the human operator made decisions much better than predicted in some cases. These decision are three of four decisions to start the turn maneuver. However, changes of speed are less efficient than predicted. While this model is able to predict the operator's performance in general, it fails to determine the exact causes. An obvious conclusion would be to assume that speed is predicted with a greater uncertainty as assumed, whereas the position is predicted with a high accuracy. However, uncertainty in the prediction of speed are compensated. For example, the speed of an aircraft is decreases earlier than necessary. In this case the reduction of the distance to the aircraft ahead is delayed and a further reduction of speed can also be delayed. Thus, while the first speed reduction is made earlier, the second speed reduction is delayed and uncertainty in the prediction of speed have minimal impact on the objective *throughput*. On the other hand, the set of rules used by the operator could be less efficient then the set used in the model. However, if the uncertainty is assumed correctly, the operator should be able to execute the more efficient strategy.

In the last presented scenario, the *throughput* calculated for parameters of $e = 18\%$ and $e = 20\%$ can predict the operators behavior quite well. As the prediction with $e = 18\%$ in mostly decreasing and thus overestimated the operators performance, the prediction with $e = 20\%$ is increasing most of the time and underestimates the operators performance.

In this case, the modeled strategy seems to be implemented by the human operator. However, this was not the case in other conditions. This indicates that the operator changes the applied strategy depending on the necessary precision.

6. Validation of Process-Oriented Performance Measurement

In this chapter, the model-based human operator performance measurement developed throughout this thesis is validated. Therefore, a study is conducted in which this measure is applied. The model-based human operator performance measure is validated by showing that it is sensitive to the difficulty of the task. Further, the advantages of this process-oriented performance measures are demonstrated. At first, the developed performance measure enables to look into the details of specific situations and reveals the effects of the human operators single decisions. Furthermore, by aggregating the effects of specific actions and errors, this measure additionally enables to evaluate a HMS based on an analysis of actions and errors and gives valuable hints for future improvements. Before the main study is planned, a preliminary study is conducted.

6.1. Preliminary Study

The preliminary study is conducted for two reasons. First, techniques necessary for the main study should be developed. In this preliminary study, Access/CPN is applied for the first time to analyze the state space of a dynamic task environment. Furthermore, the preliminary study is conducted to identify how uncertainty impacts decision making in different simulation environments.

The simulation environment applied in the preliminary study should have the following similarities with the simulation environment applied in the main study (see section 3.3). First, the simulation environment should consist of moving objects. Second, the participants should be required to make decisions based on the predicted position of these objects. Additionally, the possible options to choose from should differ in the difficulty of accessing their consequences. Furthermore, it should be possible to analyze the state space of the task environment like it is possible for the task environment used in the main task to evaluate human operator decisions.

However, there should also be an important difference between the simulation environments. It should be possible to calculate the complete state space of the simulation environment used in the preliminary so that no model of human operator behavior is needed for this task.

Additionally, the simulation environment should be motivating to engage the participants. Based on this requirements, the simulation environment "Pizza Express" was developed by [Sch11]. This simulation environment is—as MAGIE (see section 3.3)—based on a CPN. In contrast to MAGIE, the GUI of the simulation environment is not connected via TCP/IP with the CPN. Instead, they are connected using Access/CPN.

6.1.1. Simulation Environment and Scenarios

The simulation environment "Pizza Express" shows a simple street map, which consists of several squares (s. Fig 6.1). Each square can contain a straight road section, a curve, or a junction. A car is moving along the road sections with a speed of 0.66 squares per second. The speed is constant and cannot be influenced by the participant. Before the car reaches a junction, the participant has to select a direction. This is done by clicking on the arrows shown for each possible directing close to the junction. It is possible to select the direction for several junctions in advance.

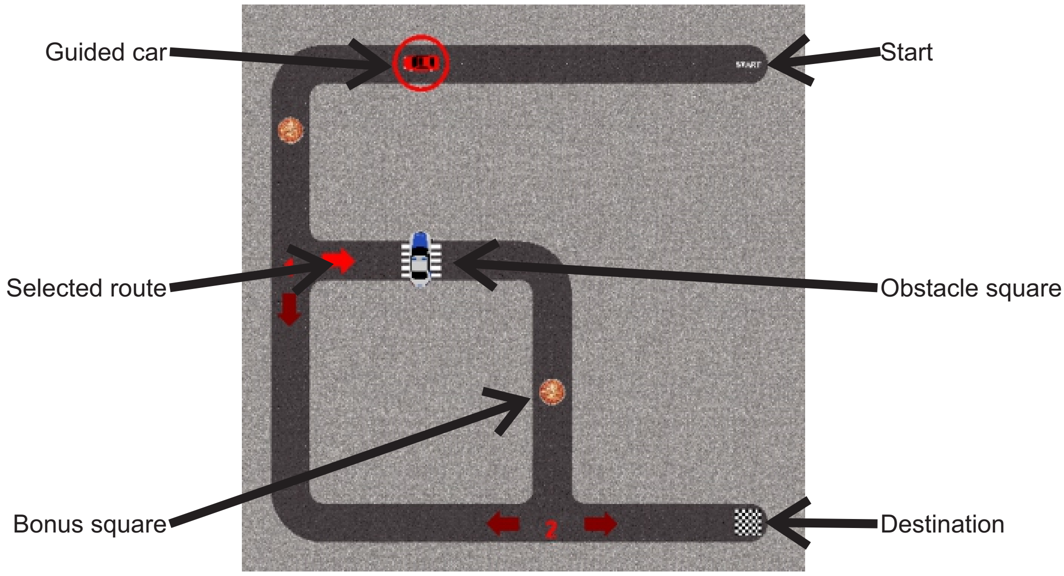


Figure 6.1.: Example for a street map in the simulation environment "Pizza Express"

The car starts at a defined square and it has to be guided to another square marked as destination. There a two other types of squares. First there a bonus squares, marked with a pizza. If the car passes such a square, the participant is reward with 100 points. Each bonus can only be collected once. Furthermore, there are obstacle squares which can either be activated or deactivated. If the obstacle is deactivated, it is marked with a crosswalk. If the obstacle is activated, a police car is additionally shown at this square. The status of obstacles changes with a fixed frequency which differs between obstacles. If the car controlled by the participants passes an activated obstacle, the participant loses 200 points. If the obstacle is deactivated while the car passes that square, no consequences follow. The aim of the participant is to collect as much points as possible in each scenario before finally reaching the destination.

The maps designed for the study contain different variants of a junction at which the participant has to make a decision. A variant of such a situation is shown in Fig. 6.2. If the car reaches the junction, the participant can choose between two alternatives. On the one hand, the route marked with "S" can be selected. There can be bonus squares on this route but there are never obstacles on this route. On the other hand, the participant can select route "A". This route has always one bonus square more than route “S”. Furthermore, route “A” always contains an obstacle square. If the participants selects route” A” and the car will pass the obstacle while it is deactivated, this route has an advantage of 100 points compared to route “S”. However, if the participant passed the obstacle while it is activated, the route “A” had a disadvantage of 100 points compared to route “S”. Consequently the participant has to predict the status of the obstacle and the position of the car to make the right decision at the decision junctions.

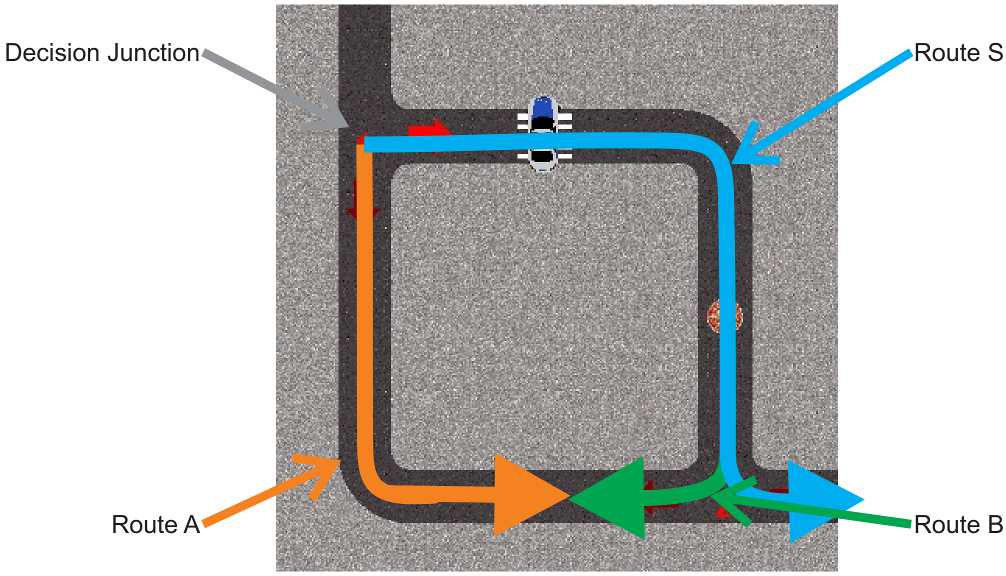


Figure 6.2.: Variant of the decision junctions integrated into the scenarios

To ensure the same conditions for all participants, the status of the obstacle should be the same when reached by the car. As each scenario contains multiple decision junctions, it must be ensured that independent from the decision at the earlier junctions the conditions for the later junctions do not change. This is reached either by designing decision junctions with routes “A” and “S” of equal length, or by extending one route such that the time needed for the extension is an integral multiple of the duration the deactivated/activated change of all following obstacles.

The decision junctions are varied regarding the distance of the obstacles from the junction, the frequency of the obstacles, and the effect of route A. The distance can either be 2, 4, or 6 squares. The frequency can either be $1/5\,\mathrm{s}^{-1}$, $1/6\,\mathrm{s}^{-1}$, $1/7\,\mathrm{s}^{-1}$, or $1/8\,\mathrm{s}^{-1}$. Finally, route “A” can either have an advantage over route “S” of 100 or -100 points. Decision junctions of all 24 ($= 3 \cdot 4 \cdot 2$) possible combinations are integrated in ten scenarios.

Originally, it was planned to reward additional points depending on the time needed to reach the destination. The impact of this time should be so large that much more points would be lost due to guiding the car in circles than could be earned for passing an additional bonus square. As the decision junctions with different length of the routes “A” and “S” would not be comparable to decision junction with equal route length, it was decided to give no points depending on the time needed to reach the destination.

However, this increased the possible benefit of selecting the route marked as “B” in Fig. 6.2.

6.1.2. Experimental Design

The preliminary study is conducted with $N = 25$ participants. The participation is voluntary. The participants receive neither a payment nor any other benefits.

After signing a consent form and a demographic questionnaire, the participants complete four training scenarios. These scenarios are used to familiarize the participants with the simulation environment and how they can control it. Furthermore, they learn the functions of the bonus and obstacle squares. After the training, each participant completes the ten scenarios which together included 24 decision junctions. The order of the scenarios is randomly for each participant.

6.1.3. Results

For analyzing the impact of uncertainty on the decisions of the participants, the complete state space for each scenario is generated. The state space is generated by accessing the CPN used for the simulation directly with Access/CPN and an adapted depth-first-search.

Scenario 2 has to exclude from the analysis due to technical reasons. In cases the participants decides to follow route B (s. Fig 6.2), the conditions of the following decision junction can change. The junctions with changed conditions are also excluded from the analysis. Furthermore, sometimes the participants do not manage to make a decision in time before the car reaches a junction. In this case, the simulation aborts and the following decision situations are not reached. In summary, the participants make the decision given in Table 6.1 separately for each variant of decision situation.

The decisions at the junctions are analyzed with the SDT (see section 2.3.1). Regarding the state of the environment it is differentiated between route “A” being advantageous compared to route “S” and route “A” being not advantageous. Consequently, the decision of the human operator can either be to select route A or not to select route A. The analysis revealed a hit rate of 49% and a false alarm rate of 44%. The sensitivity index is $d' = 0.13$. This is a rather low sensitivity. To identify if predicting the position of the guided car and the status of obstacles varies with the distance of the obstacles or with their frequency, the hit rate, false alarm rate, and the sensitivity index are calculated for each distance respectively frequency separately. The result are given in Table 6.2. The results indicate that the participants were not able to predict the position of the car and the status of the obstacles, if the obstacles was more than two tiles apart from the junction.

The question arises, why the participants have so much difficulty with making the correct decision at the junctions. Consequently, it is further analyzed if the participants do not predict the status of the obstacles but decide based on the status of the obstacles at the moment the car reaches the junction and the decision has to be made. Therefore, the visibility of the obstacle is determined for all decisions. Again a SDT is performed. In

Table 6.1.: Decisions at all variants in the preliminary study

Identifier		Route A	Obstacle		Total decisions	
Scenario	Junctions	bonus	Frequency	Distance	Route A	not Route A
1	1	-100	1/5	4	14	11
1	2	+100	1/6	2	20	5
3	1	+100	1/5	4	8	16
3	2	+100	1/8	4	14	11
4	1	-100	1/7	6	6	17
4	2	+100	1/6	4	14	9
5	1	+100	1/7	6	5	19
5	2	-100	1/6	2	9	15
5	3	-100	1/8	4	14	6
6	1	-100	1/7	4	8	17
6	2	-100	1/6	4	14	11
6	3	+100	1/8	6	2	11
7	1	+100	1/5	2	7	18
7	2	+100	1/7	4	9	17
7	3	-100	1/8	2	5	19
8	1	+100	1/7	2	14	9
8	2	+100	1/6	6	9	14
8	3	+100	1/8	2	20	3
9	1	+100	1/5	6	12	8
9	2	-100	1/8	6	10	10
10	1	-100	1/5	6	11	11
10	2	-100	1/7	4	10	12

Table 6.2.: Results for each condition in the preliminary study

	Total decisions						
Route selected:	A		not A				
Advantageous Route:	A	not A	A	not A			
Condition					Hit rate	False alarm rate	Sensitivity
Distance 2	61	14	35	34	64%	29%	0.89
Distance 4	45	60	51	57	47%	51%	-0.11
Distance 6	28	27	52	38	35%	42%	-0.17
Frequency $1/5\,\mathrm{s}^{-1}$	27	25	42	22	39%	53%	-0.36
Frequency $1/6\,\mathrm{s}^{-1}$	43	23	28	26	61%	47%	0.34
Frequency $1/7\,\mathrm{s}^{-1}$	28	24	43	46	39%	34%	0.14
Frequency $1/8\,\mathrm{s}^{-1}$	36	29	25	35	59%	45%	0.25

this case, the state of the environment is differentiated between activated and deactivated obstacles when the car reaches the junction. The decision is still differentiated between the selection of route “A” and another route. This analysis reveals a hit rate of 43%, a false alarm rate of 51%, and a sensitivity of $d' = -0.21$. This shows that the status of the obstacles was not consistently used as decision criterion. Further, it is analyzed if the consistency of the obstacle’s status—when reaching the junction and when passing the obstacle—facilitates decision making. Here a STD is carried out considering only these cases where the status is consistent. The state of the environment is differentiated between the status activated and the status deactivated. A hit rate of 57% , a false alarm rate of 45%, and a sensitivity of $d' = 0.30$ are revealed. This is a rather low sensibility indicating, that the consistency of the status does not help much to improve decision making.

Additionally, it is analyzed if the participants had not enough training. Therefore, the decisions in the first five scenarios of each participant and in the last five scenarios are analyzed separately. For the first half, a hit rate of 73%, a false alarm rate of 73%, and a sensitivity of $d' = 0.00$ are detected. For the second half, a hit rate of 38%, a false alarm rate of 30%, and a sensitivity of $d' = 0.19$ are detected. This indicates a small learning effect leading to a small increase in the sensitivity. However, more strikingly is the decrease of both rates. This can indicate a decrease of the participants’ engagement in the later scenarios.

In summary, the preliminary study showed the difficulties in predicting future states of the system. Only if the obstacle were only 2 squares apart form the decision junctions, the participants could predict the consequences and tended to make the right decisions. However, the preliminary study facilitated the development of techniques necessary for the implementation of the cognitive planning model described in chapter 3 and the analysis of the main study described in the following.

6.2. Simulation Environment MAGIE

In the conducted main study, the simulation environment MAGIE is applied. The participants of the study take over the role of ATCOs, more precisely the role of feeder in an approach sector, and have to control traffic approaching at an airport. The simulated task is described in detail in section 3.3.

The participants are supported by two assistance functions. First, the arrival sequence planned by the implemented computer-based assistance is displayed. Additionally, the ghosting functionality is activated so that ghosts are projected on the centerline indicating positions which should be reserved for equipped aircraft.

6.3. Experimental Design

The first aim of this study is to validate the model-based human performance measure developed in this thesis, by demonstrating its sensitiveness to variances of the task’s difficulty. To control the difficulty of the task, only the required precision of the task is varied. The amount of aircraft and the traffic mix is fixed. It is assumed that the required precision has the strongest correlation with the difficulty of the task. If the amount of aircraft is varied between scenarios, it would have been difficult to deal with a lot of aircraft as well as to stay concentrated while only a few aircraft are controlled. Also, the traffic mix could be modified. As a result, also the amount of aircraft controlled by the operator would vary with the above described consequences.

In each scenario, the arrival times at the late merging point are defined for all equipped aircraft. These arrival times are chosen in this study such that exactly one equipped aircraft can be merged between two equipped aircraft without violating the minimum separation. Additionally, enough unequipped aircraft are defined in each scenario to fill the gaps between the equipped aircraft. This results into an alternating sequence of equipped and unequipped aircraft in the optimal case.

The distance between two equipped aircraft determines the precision necessary to guide unequipped aircraft into the gaps between the equipped aircraft. To vary the required guidance precision between conditions, the distances between the equipped aircraft are varied. Three levels of difficulty are defined with the conditions easy, medium, and difficult.

To define the planned distance between aircraft, the minimal possible timely distance between two aircraft is considered first. The simulated aircraft have a speed of 160 kn shortly before being handed over to the tower. With a minimal separation of 3 NM, a minimal distance of 67.5 seconds results (s. equation 6.1).

$$\frac{3\,\mathrm{NM}}{160\,\mathrm{kn}} = \frac{3 \cdot 1852\,\mathrm{m}}{160 \cdot 0.514\,44\,\mathrm{m\,s^{-1}}} \approx 67.5\,\mathrm{s}. \tag{6.1}$$

Therefore, two equipped aircraft must be separated by 135 seconds at least to allow an unequipped aircraft to be merged between them.

Second, the distance used in previous experiments is taken into account. In previous experiments a timely distance of 75 seconds was planned. Hence, a distance of 150 seconds results between two equipped aircraft if one unequipped aircraft should be placed between them.

It is defined that the same distance of 75 seconds (150 seconds between two equipped aircraft) as in previous studies is used in the condition *Medium* of this study. In the condition *Difficult*, the distance is reduced by the half of the possible reduction to a planned distance of 71 seconds (142 seconds between two equipped aircraft). In the condition *Easy*, the distance is increased by the same amount as it is reduced in the difficult condition. Therefore, a distance of 78 seconds (166 seconds between two equipped aircraft) results. The conditions are illustrated in Fig. 6.3.

Each of the three conditions is simulated four times. Hence, a within subject 3 (difficulty) times 4 (repetition) repeated measurement design results.

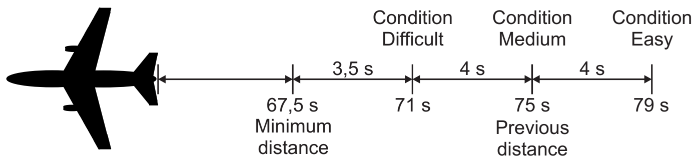


Figure 6.3.: Timely distances between equipped and unequipped aircraft

6.4. Method

In this section, the conduction of the study is explained. The participants, the procedure and the direct measures are described.

6.4.1. Participants

Before the study is conducted, the required amount of participants is calculated using the software G*Power [GPo13, FELB07] (Version 3.1.5). The error of first kind is set to $\alpha = 0.05$, the error of second kind to $\beta = 0.8$. The study should be able to detect effects of medium size, therefore an effect size of $f = 0.25$ is selected. This is defined as a medium effect [Coh88]. For the calculations in G*Power, the corrected effect size is used. For the factor "difficulty" ("repetition"), a corrected effect size of $f' = \sqrt{3} \cdot f = 0.4330$ ($f' = \sqrt{4} \cdot f = 0.5$) and as number of measurements of 4 (3) was used for calculations. The G*Power screenshots are given in the appendix.

The calculations indicate that at least 12 participants are necessary to detect medium size effects with the experimental design.

Finally, 14 (7 female, 7 male) undergraduate students from the Technical University of Braunschweig participated in the study. Their age ranges between 18 and 31 years ($M = 23.9$, $SD = 3.83$). Each participant is paid 25 € as an incentive for participation, which lasts approximately 3 hours.

6.4.2. Procedure

The procedure of the study is illustrated in Fig. 6.4. At first, the participants are informed about the purpose, procedure, and risk of their participation and sign a consent form. After that, they fill out a demographic questionnaire, with questions about their age, subject of study, and previous experience with the simulation environment MAGIE.

Following that, participants read detailed instructions about their task, available assistance functions, and the interface. They are encouraged to ask questions in case of ambiguity. After the instruction, three practical trials have to be completed, each lasting for 10 minutes. Each condition is practiced in one training scenario. The training scenarios are designed to be similar to the experimental scenarios.

After the practice trails, twelve experimental scenarios are absolved. Again each scenario lasts for 10 minutes. The four scenarios for each experimental condition are grouped. The order of the conditions was controlled for. However, the order of the scenarios within each condition was the same for each participant. The order of the condition in the training corresponded to the order of conditions in the experimental trails. After finishing the twelve experimental scenarios, the participants receive their incentive.

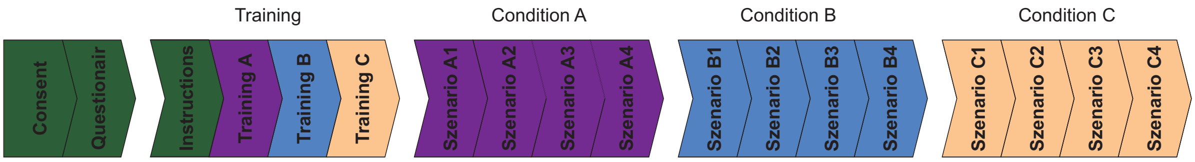


Figure 6.4.: Procedure of the study with training and three conditions

Initially, one scenario was created for each condition. If only the distance between equipped aircraft had been increased between the different conditions and the unequipped aircraft had not been modified, their estimated arrival times at the LMP would have increase. Consequently, they have to stay longer on the downwind. To avoid this, the aircraft are initialized at different points on the arrival route in each condition such that the overall flight duration of all aircraft in the scenarios is approximately the same in all conditions. After defining the first scenario for each condition, the other three scenarios are derived from that by first mirroring the traffic at the centerline (aircraft arriving from north are changed to the south and vice versa). Additionally, aircraft are mirrored at the middle standard arrival route so that aircraft arriving on an east arrival route are changed to arrive on a west arrival route and vice versa. Aircraft on the middle route are not modified in this second step. Additionally, all call sign are change between the scenarios. In doing so, the scenarios are comparable but the participant should not be able to recognize the same basis of all scenarios.

6.4.3. Measures

During each simulation run, states and events are directly measured. At first, the states of all simulated aircraft are recorded for each simulation step which equals one second. As the first and the last state are included, 601 states are recorded for each simulation run. Furthermore, the participants input in form of time and the details of every given clearance are recorded. Although the data are also recorded during the training, only the data recorded during the simulation runs is later used for the performance evaluation.

6.5. Process of Evaluation

To evaluate the participants behavior, the recorded data are analyzed as detailed in section 5.2 and summarized shortly in the following. At first, the cognitive human operator planning model described in chapter 3 is used to generate interaction sequences, using each tenth recorded state as initial states. After that, the recorded interactions sequence prior to a state used as initial state for the model is combined with the interaction sequence generated by the model for this state. In the third step, the combined interaction sequences are simulated. The results of the simulation indicate what was reachable at the selected state. This results are subsequently evaluated using the evaluation criteria given in section 5.1. Then the evaluations of adjacent sequences are compared. This turns the result measure used to evaluate the combined interaction sequences into a process measure. After these comparisons, different kind of actions (initiations of the turn, changes of altitude, and changes of speed) and types of error (omissions and commission) are identified for each evaluated situation and their effects are aggregated for each simulation run.

The developed process-oriented human performance measure is not able to evaluate all simulation runs. Out of 168 simulation runs (14 participants times 12 experimental runs), the results of the evaluation for 13 runs is not plausible. These runs are excluded from further analyses. The capability of the performance measure is limited by the implemented cognitive planning model. This is based on a set of rules which are designed for specific situations. It cannot be guaranteed that the set of rules can be applied to unexpected situations. The (in the design phase) unexpected situation causing most of the problems in the excluded simulation runs is the participant trying to guide two unequipped aircraft into a gap in which only one unequipped aircraft can fit. Due to the limited interaction possibilities of the simulation environment, serious problems are inevitable in this case. However, no rules are implemented to reduce the problems as much as still possible.

The excluded simulation runs fall upon only three participants. Consequently, the simulation runs of eleven participants could be evaluated completely. Those three participants possibly use other strategies during the simulation, leading to unexpected (from the design perspective) situations.

6.6. Validation Results

To validate the developed performance measure, its ability to detect differences between conditions with different difficulty is demonstrated. The calculated missed performance based on the aggregation of the effects of different kind of actions and types of errors for each run is used as input (compare section 5.2.6). The missed performance is given for each objective separately, but for the validation, on overall performance for each run must be used, as the difficulty of the different conditions results from a combination of the objectives. For example, small deviations from the planned positions for unequipped aircraft are more likely to cause conflicts in the condition *Difficult*. Consequently, in the condition *Medium* and *Easy* small deviations are not punished as hard as the condition *Difficult*. Thus, participants might tend to accept larger deviation and consequently reach a lower performance regarding the objective *throughput* in the condition *Medium* and *Easy*.

The objectives are ranked so that no weighting of the objectives can be derived theoretically. However, as the difficulty stems from all objectives together, all of them must be included in the overall performance. Consequently, a weighting is necessary, but its definition is arbitrary. However, as the objectives are ranked, *separation* should have a higher weighting as *constraints*, which in turn should have a higher weighting than *throughput*. It is deiced to use a multiplier of 3 between the ranks so that *throughput* has the weighting of 1, *constraints* has the weighting of 3, and *separation* has a weighting of 9. Consequently, the missed performance is defined by

$$\Sigma^- = \frac{(9 \cdot \Delta^- + 3 \cdot \Gamma^- + \Theta^-)}{13}. \tag{6.2}$$

6.6.1. Graphical Comparison of Conditions

A histogram of the missed performance in all conditions is shown in Fig. 6.5. This shows that in a lot of simulation runs (94), the missed performance was very small (between 0 and 0.01). In contrast, a large missed performance of between 0.1 and 0.15 resulted in 4 simulation runs. To allow for a better differentiation between the simulation runs with only a small missed performance, but to include the simulation runs with a larger missed performance, a logarithmic scaling is used. In Fig. 6.6 the same histogram using logarithmic group sizes is shown. Here, the lost performance between 0.001 and 0.01 is divided into five groups, while the missed performance between 0.1 and 0.15 is summarized to one group.

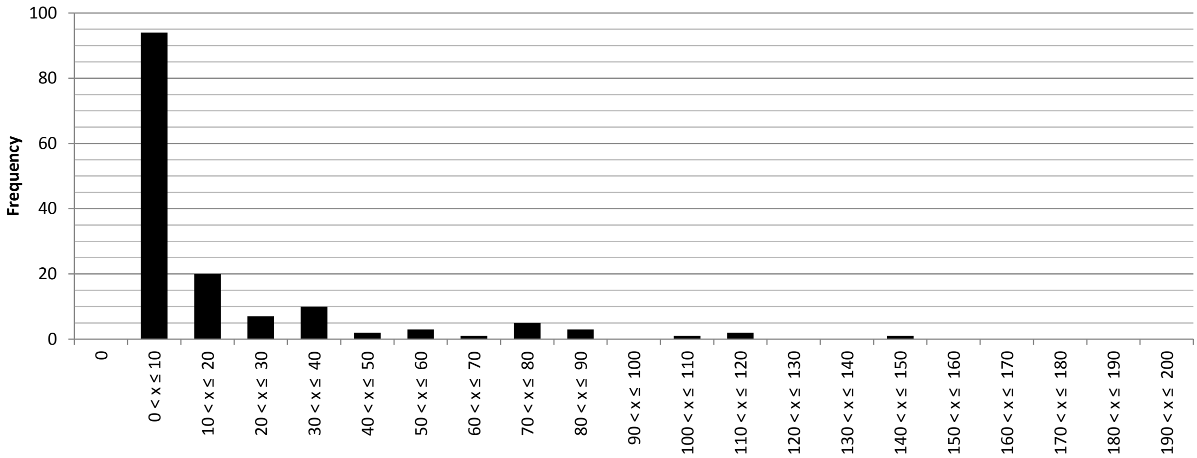


Figure 6.5.: Histogram of missed performance (conditions combined)

In Fig. 6.7 a boxplot is shown, which indicates the minimum, maximum, median and 25- as well as 75-percentile of the missed performance for each condition. Note the logarithmic scaling of this plot. It can be seen, that in the condition *Difficult* the missed performance is larger than in both other conditions. However, in the condition *Easy* the median is even larger than in the condition *Medium*. This reveals, that the median performance was better in the condition *Medium* compared to the condition *Easy*. However, the range of the performance shows that there are more extreme poor performances in condition *Medium* compared to condition *Easy*.

To analyze this further, a histogram is plotted for each condition in 6.8. Also in this diagram, the group sizes are generated logarithmically. The comparison between the

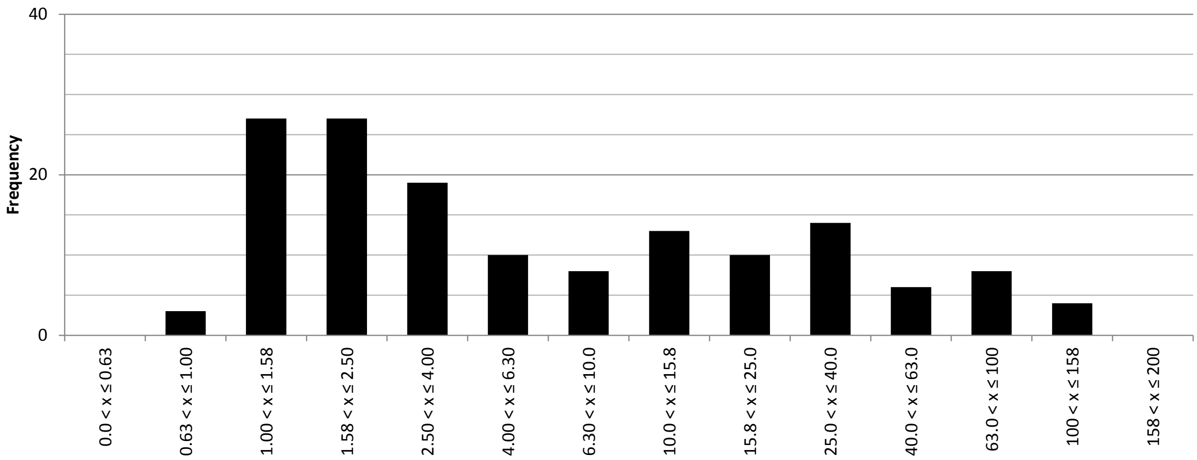


Figure 6.6.: Histogram of missed performance with logarithmic group sizes

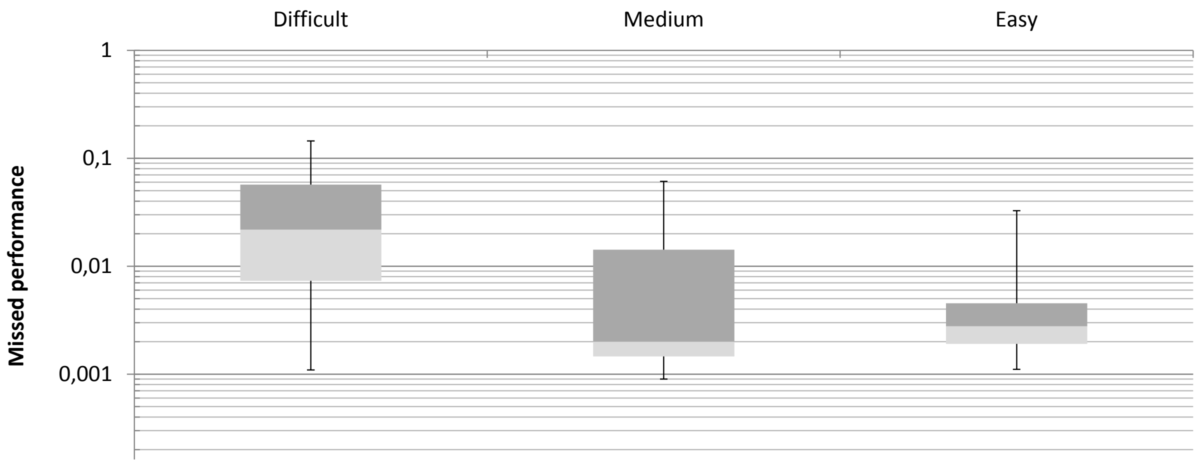


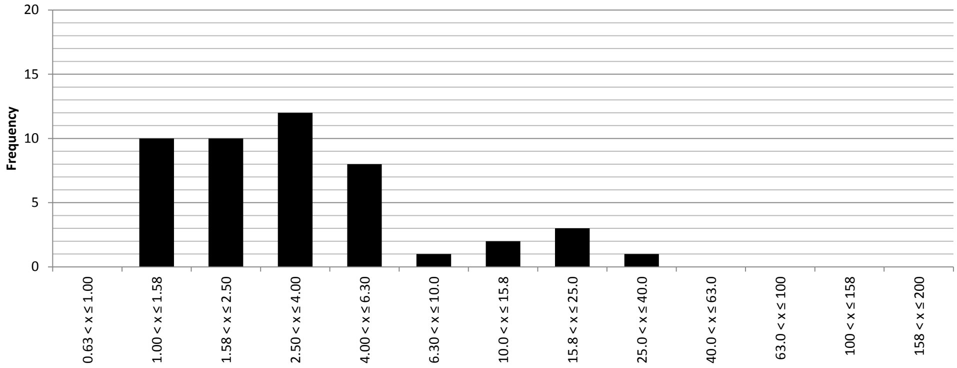
Figure 6.7.: Boxplots of missed performance in all conditions

conditions *Medium* and *Easy* shows that in the condition *Medium* more often only little performance is missed but also that more often a lot of performance is missed compared to *Easy*. In condition *Easy* the measured performance has the lowest range.

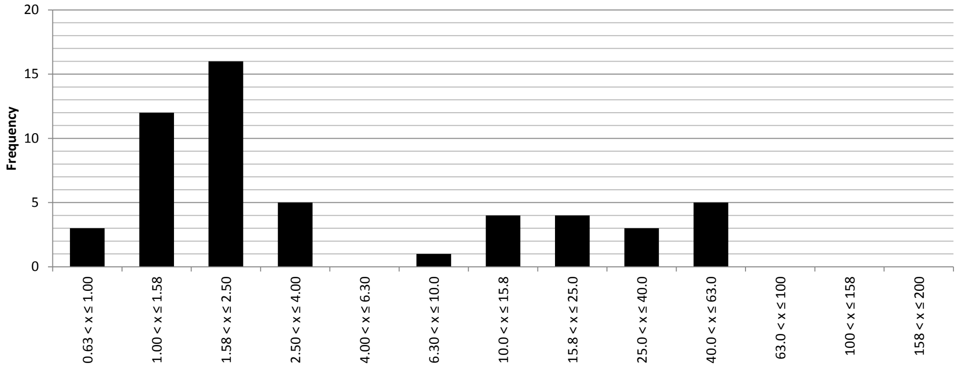
6.6.2. Statistical Evaluation with SPSS

In the following, the data are analyzed with an analysis of variance (ANOVA) with repeated measurement. This method presume a normal distribution of the independent variable, however it is robust to a violation of this assumption. To reduce this violation, the logarithmized measured missed performance is used in the following.

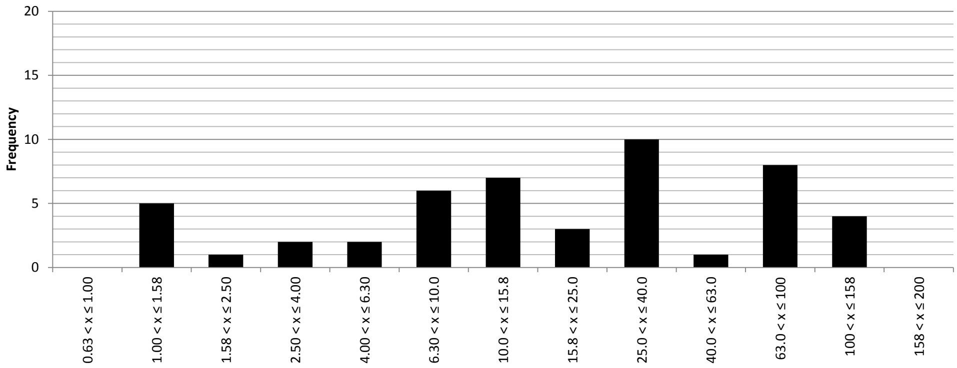
The data are analyzed with SPSS. The repeated measurement of the general linear model is used. The factors *Difficulty*—with 3 levels—and *Repetition*—with 4 levels—are entered as inner subject factors. The error of first kind α is set to 0.05. The three participants, where performance data could not be calculated for every scenario, have to be excluded from the analysis. Thus, it is based on the results from eleven participants.



(a) Easy



(b) Medium



(c) Difficult

Figure 6.8.: Histogram of missed performance with logarithmic groups for the conditions *Easy* (a), *Medium* (b), and *Difficult* (c)

Table 6.3.: Results of repeated measurement analysis by SPSS

	p	p[a]	η_p^2
Difficulty	0.001	0.009	0.51
Scenario	0.045	0.112	0.232

[a] corrected conservatively

Table 6.4.: Pair-by-pair comparison generated by SPSS

p	*Difficult*	*Medium*	*Easy*
Difficult	-	0.025	0.015
Medium	0.025	-	1.000
Easy	0.015	1.000	-

SPSS first executes the Mauchly procedure to test for sphericity. This test is significant for the factor difficulty. It is not significant for the factor repetition or for the interaction of both factors. Consequently, the assumption that sphericity is present in the data should be rejected. Correction factors are calculated and used in the following calculations to compensate for missing sphericity.

The output of SPSS regarding the significance and effect sizes of the two main factors are given in Table 6.3. This output indicates a significance of $p = 0.009$ for the factor difficulty when the conservative correction for sphericity is applied. Without a correction, a significance of $p = 0.001$ results.

Besides the factor *Difficulty*, also the factor *Repetition* is significant ($p = 0.045$) if no correction factors are used. If correction factors are applied, this factors is not significant ($p = 0.112$). The interaction between both factors is not significant. The partial eta square η_p^2, which is a measure of effect size, is 0.51 for the factor *Difficulty* and 0.232 for the factor *Repetition*.

Also a pair-by-pair comparison for the main factors is conducted. For the adaption of the confidence interval, „Bonferroni“ is selected. This comparison for the factor *Difficulty* reveals a difference between the condition *Difficult* and *Medium* as well as between *Difficult* and *Easy*. A difference between *Medium* and *Easy* cannot be detected. This is summarized in Table 6.4.

6.6.3. Discussion of Validation

Comparing the histograms for the three conditions shows a clear difference between the conditions. Most noticeable is the difference between the condition *Difficult* and both other conditions. The differences between the conditions are confirmed by an analysis conducted with SPSS. However, the results of the statistics should be treated with care as the assumption of sphericity is violated - although this was corrected conservatively. The pair-by-pair comparison also confirms the first impression as differences between the conditions *Difficult* and *Easy* as well as between the conditions *Difficult* and *Medium* but no difference between the conditions *Medium* and *Easy* are detected. Thus, it can be concluded that increasing the planned distance between the unequipped aircraft over 150 seconds (condition *Medium*) did not impact the difficulty. However, it can be stated that the model-based human performance measure as developed in this thesis is able to detect the differences in the performance between different conditions in which the required guidance precision is modified. Consequently, the validation is successful.

6.7. Revelations of Details in Selected Situations

The aim for the development of the model-based performance measure was to enable considering the process during an interaction. However, if the developed measure is applied in a way like in the last section, there is no significant gain of information compared to performance measures considering only the result of an interaction. Therefore, the further advantages of the method are demonstrated in the following section by showing the ability of this method to analyze selected situations in detail. Therefore, four simulation runs (cases) are selected, in which one or more interesting periods occurred, and analyzed with the developed process-oriented performance measure.

6.7.1. Case 1: Missed Planned Position in Sequence

In the first selected case, the participant missed to start a turn maneuver in time and caused the aircraft to be moved back in the arrival sequence. For this case, the simulation run of scenario 5 and participant 12 is selected. The expected performance according to the three objectives (with constraints split in speed and altitude constraints) over simulation time is given in Fig. 6.10.

If only the result of the interaction is considered, a relatively low performance for the objective *throughput* can be detected. However, analyzing the process reveals that the performance regarding *throughput* mainly drops during the interval after $t = 420$. Besides the performance over time, the list of reasons for missing performance generated by the method (see section 5.2.6) is used to interpret the interaction. This list indicates that the start of the turn maneuver was forgotten during the interval after $t = 420$. Thus, the participant missed the last moment to catch the originally planned position in the arrival sequence for this aircraft. Additionally, it was not possible to guide another aircraft into this position. If another aircraft could fit into the gap, the decrease of its flight duration would have settled the increase of the flight duration of the delayed aircraft and the *throughput* performance would not have been affected. The other objectives are not affected by the omission of the turn maneuver in the selected case.

The necessary adaption of the planned arrival sequences at $t = 420$ is also evident when considering the planned flight duration in the path stretching area (p_i) over time during this simulation run. The flight duration in the path-stretching area is shown in Fig. 6.9. Following to the situation at $t = 420$, the flight duration of aircraft D increases about 300 seconds. The aircraft will consequently arrive 5 minutes late. As the planned

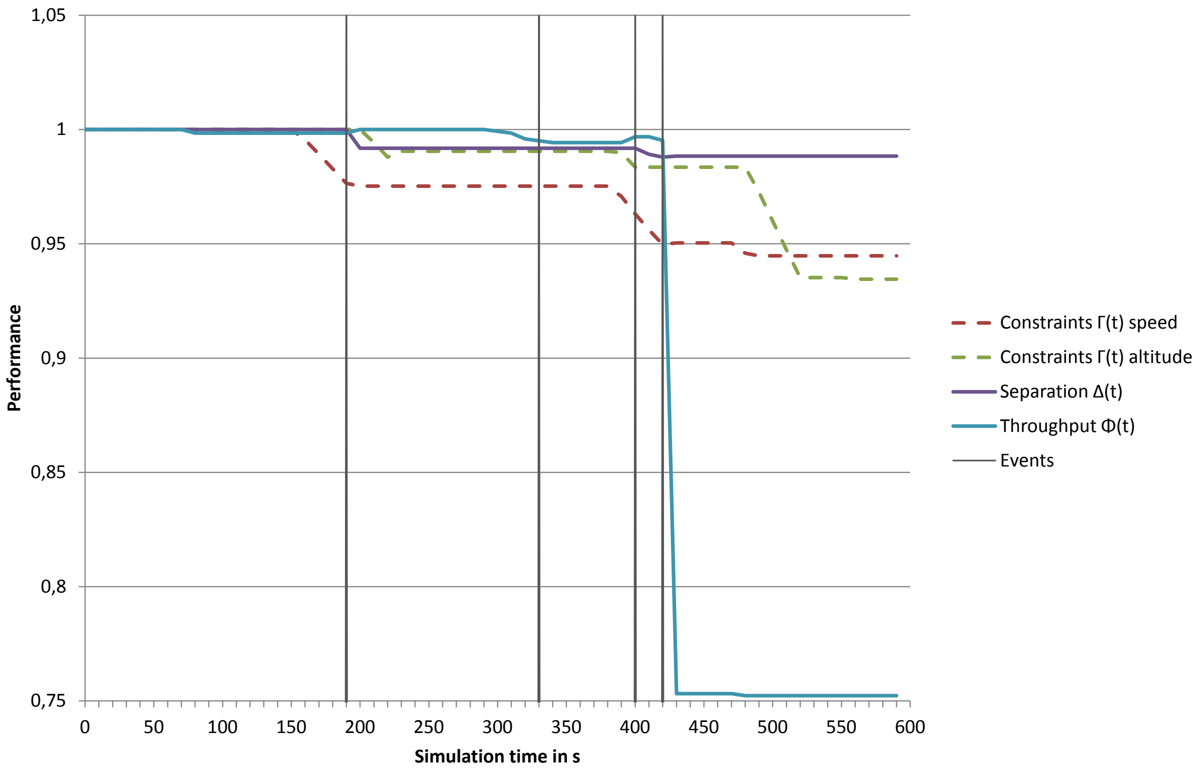


Figure 6.9.: Performance over simulation time for case 1: missing planned position in sequence (participant 12, scenario 5)

distance between to equipped aircraft is 150 s in the selected scenario, the updated sequence includes three additional aircraft prior to aircraft D. The flight duration over time also shows that the participant missed to start the turn for aircraft C at $t = 330$. As it was possible to guide aircraft D into the generated gap instead, this had no direct consequences on the performance. The performance is not affected till aircraft D misses that gap finally at $t = 420$.

This simulation run includes further interesting periods. One example is the situation at $t = 190$. Following to this situation, the performance of the objective *separation* decreases while the performance of the objective *throughput* increases. For this interval, the generated list of errors reveals that a reduction of speed is missing. This reduction of speed would have avoided a conflict but would have increased the flight duration of the aircraft. As consequence of this omission, the objectives *separation* and *throughput* are affected in the observed way.

Furthermore, the performance over time shows that a further error in the interval after $t = 400$ accounts for the finally reached *separation* performance. In fact, a further speed reduction was missing in this interval. This missing action does not only influence *separation* performance but additionally reduced *constraints* performance.

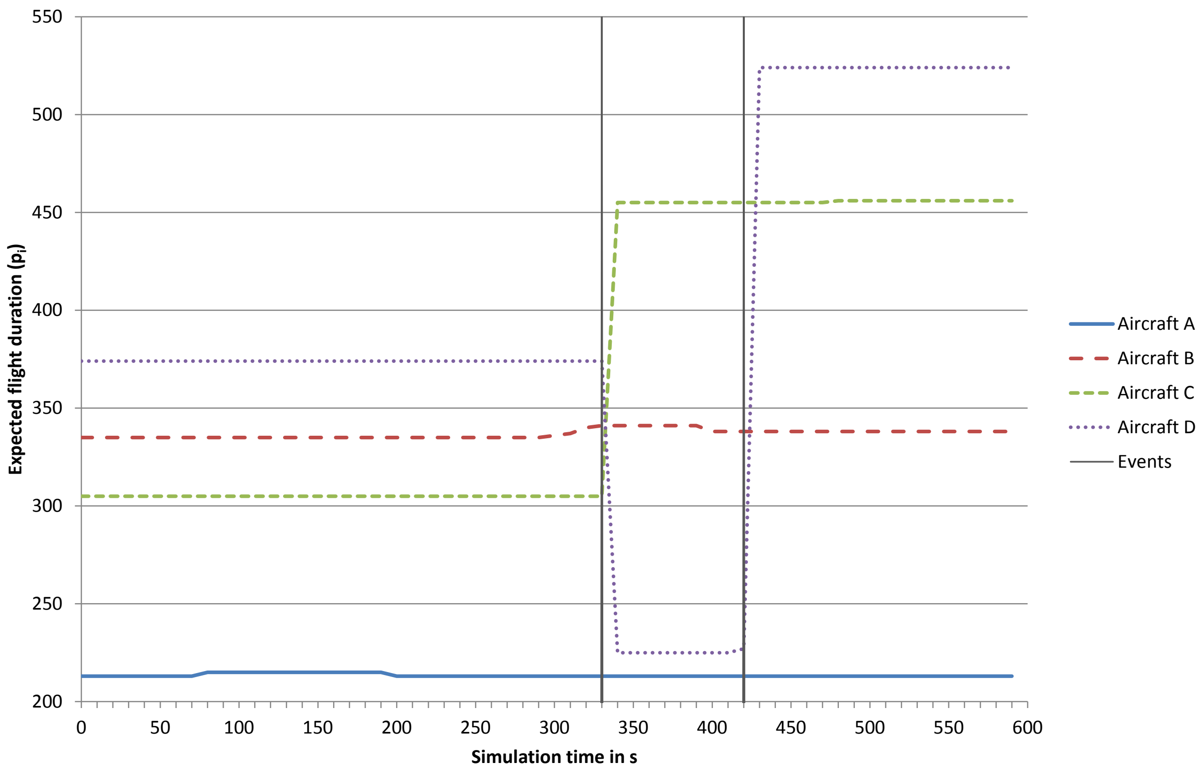


Figure 6.10.: Flight duration in path stretching area over simulation time for case 1: missing planned position in sequence (participant 12, scenario 5)

6.7.2. Case 2: Throughput Performance Increase

In the second case, the participant managed to reach a *throughput* performance larger than one. However, the price for this is a lower *separation* performance. For this case, the simulation run of scenario 6 and participant 12 is considered. The performance according to the three objectives (with constraints split in speed and altitude constraints) over simulation time is given in Fig. 6.11.

In this interaction sequence, conflicts are cased during two periods. The first period is between $t = 180$ and $t = 210$. During this period, the *throughput* performance is increasing and reaches a value larger than 1. Note that 1 is defined as the maximal reachable performance for each objective. However, this holds only when all other objectives are considered but the participant causes a conflict during this period indicated by a decrease of the *separation* performance. Consequently, an increase of the *throughput* performance exceeding 1 is achieved at the cost of a conflict. The generated error list states that a decrease of speed was missed during this period, which causes the problem.

As the considered aircraft started the turn shortly after $t = 70$ already, there was enough time to reduce the speed after starting the turn. However, this was omitted and consequently a conflict was caused.

In the following period from $t = 210$ to $t = 230$, the speed reduction is still missing. However, it does not affect the length of the conflict anymore. Instead, the missed

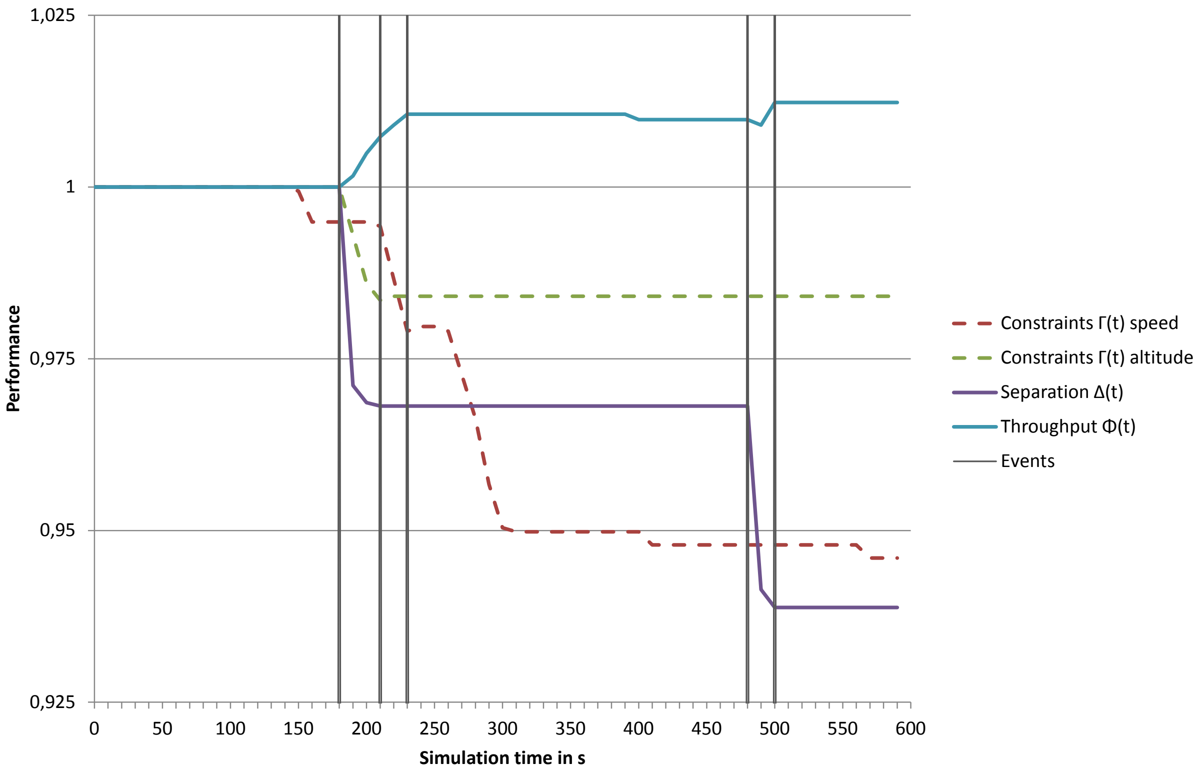


Figure 6.11.: Performance over simulation time for case 2: conflict (participant 12, scenario 6)

speed reduction leads to a violation of the speed restriction and a decrease of *constraints* performance. In contrast, *throughput* performance still increases due to the missing speed reduction.

A similar case can be detected in the second period between $t = 480$ and $t = 500$. A reduction of speed is missed again, which causes a conflict but increases the *throughput* performance. The affected aircraft started the turn shortly after $t = 320$. Again, the conflict was not caused by selecting the wrong moment for the turn procedure but by missing the right moment for a speed reduction.

6.7.3. Case 3: Conflict Between Separation and Constraint Performance

The third selected case includes periods in which conflicts could be avoided for the cost of violating speed constraints. For this case, the simulation run of scenario 1 and participant 9 is considered. The performance according to the three objectives (with constraints split in speed and altitude constraints) over simulation time is given in Fig. 6.12.

When analyzing the reachable performance over simulation time in this case, it can be noticed that the *separation* performance is decreasing almost steadily. Furthermore, a violation of the speed constraints is predicted in the periods from $t = 190$ to $t = 300$ as well as from $t = 340$ to $t = 440$. However, at the end of these intervals, a speed violation

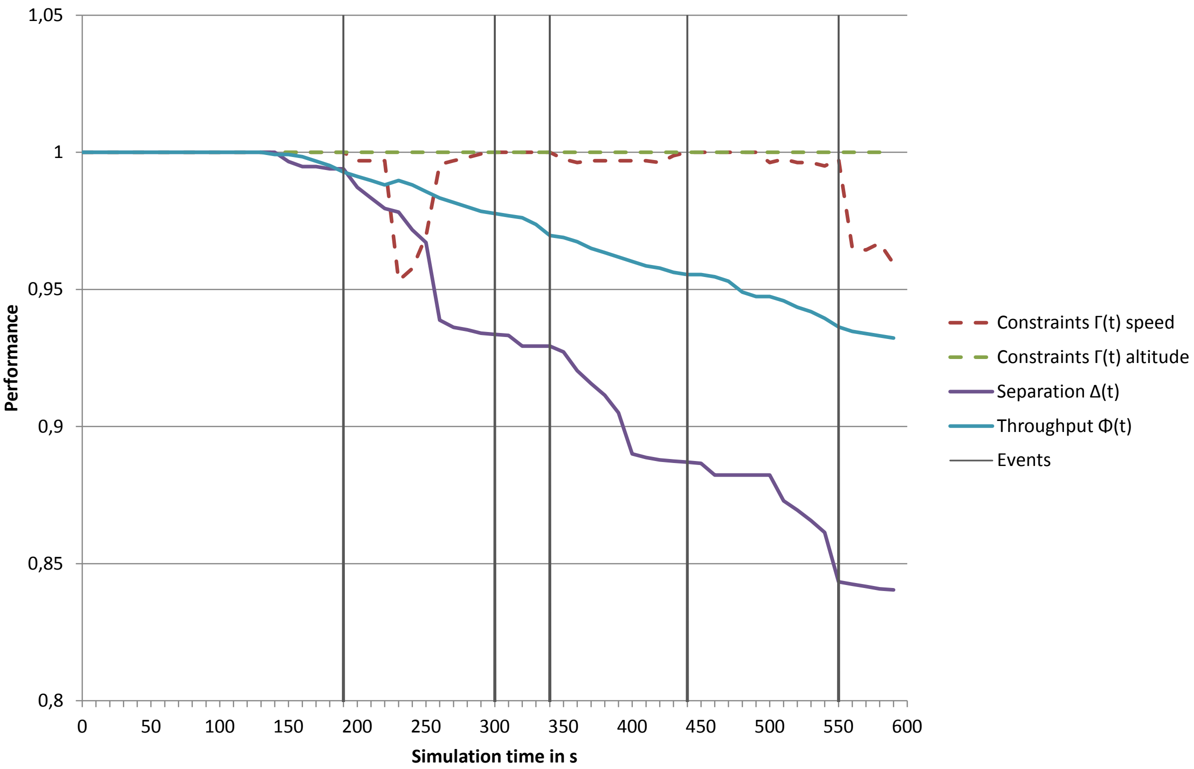


Figure 6.12.: Performance over simulation time for case 3: conflict and speed constraints (participant 9, scenario 1)

is not predicted any longer. Starting from $t = 500$ to the end of the simulation, a violation of speed constraints is predicted again. This implies that this predicted violated did indeed realize during the interaction—in contrast to both violations predicted earlier.

During each of these three periods, the operator model tries to avoid conflicts by increasing the speed above the limit given by the restriction. In doing so, the model accepts a decrease of the *constraints* performance to keep a higher performance in the more important objective *separation*. For this reason, the expected *constraints* performance decreases at first. However, the participant does not implement this increase of speed. Consequently, the violation of the speed restriction does not realize as predicted by the model and the *constraints* performance increases again. However, this has the cost of a more severe conflict and a decreasing *separation* performance. This behavior can be observed during the first ($t = 190$ to $t = 300$) and second ($t = 340$ to $t = 440$) period, with a longer predicted violation of the speed restriction in the first period. In the third period ($t = 500$ to the end), the model again accepts a violation of speed constraints to reduce the severity of a conflict. In this period, the participant follows the model and also accepts this violation in order to reduce a conflict's duration.

6.7.4. Case 4: Separation Performance Reduction Caused by Turn

In the fourth case selected, a turn causes a conflict. For this case, the simulation run of scenario 4 and participant 6 is considered. The performance according to the three objectives (with constraints split in speed and altitude constraints) over simulation time is given in Fig. 6.13.

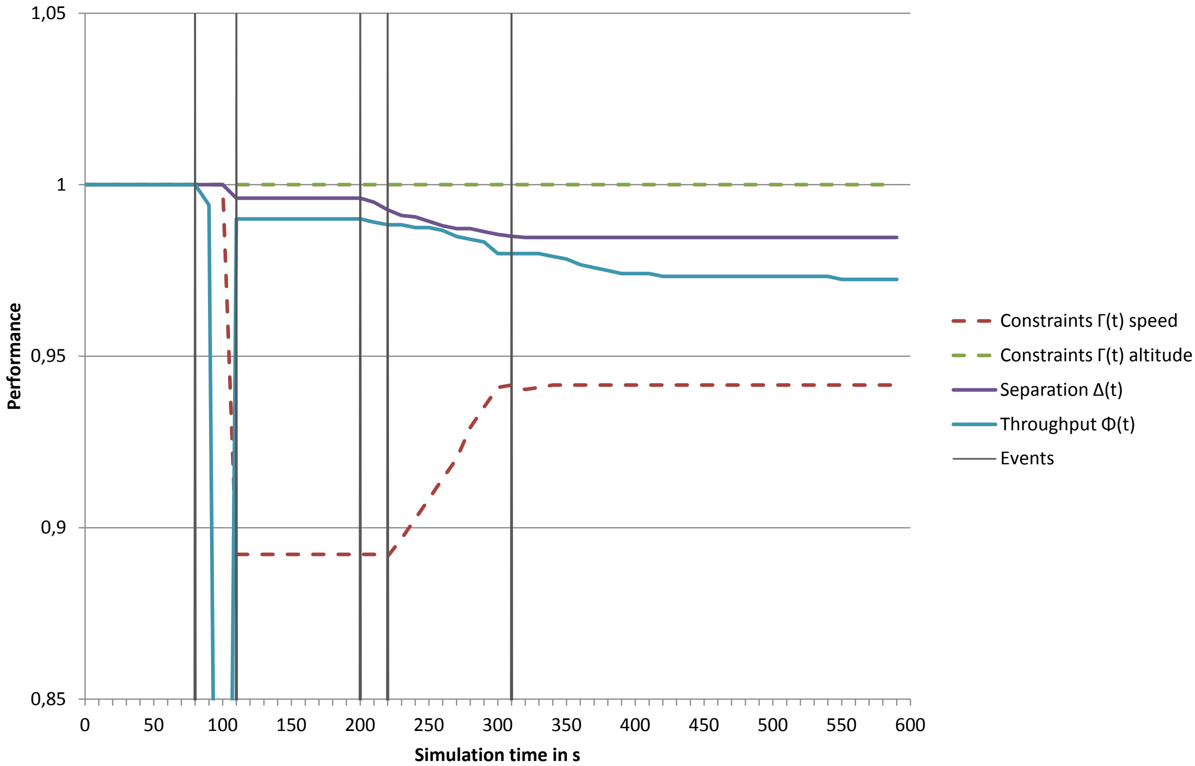


Figure 6.13.: Performance over simulation time for case 4: delayed start of turn (participant 6, scenario 4)

In this selected case, the *throughput* performance is slightly decreasing in the interval after $t = 80$. In the next interval it is dropping to shortly above 0.5 (due to scaling reasons not shown in the figure). For both intervals, an omission to start the turn maneuver is identified as error. In the first interval, the corresponding unequipped aircraft will still fit into the originally planned gap in the arrival sequence between the equipped aircraft but it will be some seconds late. During the second interval, it is not possible to guide the aircraft into that gap without causing a conflict with another aircraft. Consequently, the model changes the arrival sequence. As no other equipped aircraft fits into the opened gap, all following aircraft are delayed causing the extreme *throughput* performance fall-off.

However, in the next situation at $t = 110$, the *throughput* performance is increasing to about 0.99 again. At the same time, the *separation* and *constraints* performance decrease. This is caused by the participant's decision to start the turn maneuver. However, the duration of the conflict caused by the turn maneuver will be short if a violation of the restriction is accepted. This solution is generated by the model and possible through the period from $t = 110$ to $t = 200$. After that period, the *separation* performance is further decreasing. Now, the effect of accepting a violation of the restriction is diminishing and the unavoidable duration of the conflict is increasing. After $t = 220$, the *constraints* performance starts increasing as the possibility of accelerations violating the speed constraints is fading away. For the whole period starting with the turn at $t = 110$, a missing speed increase is identified as error. The *constraints* performance is not increasing to 1 again as the participant does not decelerate the aircraft to comply with the lower constraints at the LMP.

The conflict finally occurring during the interaction is thus partially caused by starting a turn maneuver at the wrong moment and partially by failing to increase the speed above the given constraints. However, if the participant had increased the aircraft's speed, a reduction of the *constraints* performance would have resulted.

6.8. Effects of Kinds of Actions and Types of Errors

Besides the possibility to analyze selected situations in detail, the developed process measure of human operator performance also allows to identify the impact of different kind of actions and types of errors. It is differentiated between the three objectives *separation*, *constraints*, and *throughput* as well as between actions to start the turn maneuver and actions to set new target values for aircraft's altitude and speed. Furthermore, it is differentiated between the execution of an error as reason for missed performance (commission error) and not execution an action planned by the model (omission error).

The evaluation process for each simulation run results in a table including the impact of each objective/action/error combination for each run. To get the overall impact of each combination, these values are just added up over all simulation runs. The results are given in Table 6.5 and illustrated in Fig. 6.14. In this table, the overall performance reduction by all combinations of action, error, and objective is given. Furthermore, the percentage of the objectives' performance reduction caused by each combination of action and error is indicated.

Regarding the objective *separation*, the performance is reduced by $\Delta^{-}_{overall} = 2.0311$ in all 168 simulation runs. Start turn commissions describing the necessary action to start the turn maneuver in an interval where it is not planned by the model only contributed with 6.4 % to this performance reduction. Omissions of the turn have no impact on the *separation* performance. This is expectable due to the fact that the scenario is defined in such a way that no conflicts occurred on the downwind. Consequently, staying on the downwind is always a reliable solution to avoid conflicts and to keep the *separation* performance up. Most of the *separation* performance is reduced due to changes of the aircraft altitude or speed. However, only 0.3 % of the performance reduction is caused by executing such actions. This is due to the fact that it is possible to correct each given action and only the inevitable performance reduction between the wrong execution of an action and its correction is considered. If the correction is not executed, the resulting missed performance is attributed to this missing action. All missing action regarding

Table 6.5.: Overall impact on performance differentiated between each objective, kind of action, and type of error including the relative impact on each objective by each combination of kind of action and type of error

Overall *separation* performance reduction $\Delta^{-}_{overall}$ 2.0311 (100 %)			
Start turn 0.1313 (6.4 %)		Altitude / Speed 1.9001 (93.6 %)	
Commission	Omission	Commission	Omission
0.1313 (6.4 %)	0.0 (0.0 %)	0.0060 (0.3 %)	1.8941 (93.3 %)

Overall *constraints* performance reduction $\Gamma^{-}_{overall}$ 3.0497 (100 %)			
Start turn 0.9779 (32.1 %)		Altitude / Speed 2.0718 (67.9 %)	
Commission	Omission	Commission	Omission
0.9717 (31.9 %)	0.0062 (0.2 %)	0.0851 (2.8 %)	1.9867 (65.1 %)

Overall *throughput* performance reduction $\Theta^{-}_{overall}$ 7.3950 (100 %)			
Start turn 3.9658 (53.6 %)		Altitude / Speed 3.4292 (46.4 %)	
Commission	Omission	Commission	Omission
0.0521 (0.7 %)	3.9137 (52.9 %)	0.0629 (0.9 %)	3.3663 (45.5 %)

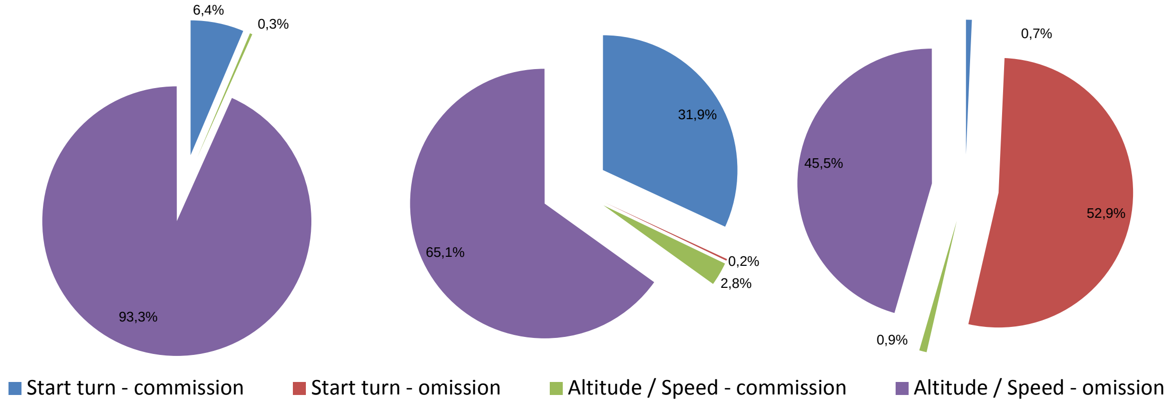


Figure 6.14.: Errors reducing the *separation*, *constraints*, and *throughput* performance, differentiated between kinds of actions and types of errors

speed and altitude (omission) including the missing corrections contribute with 93.3 % to the performance reduction of the objective of separation. Although changes of speed and altitude are combined, it can be assumed that mainly speed changes contribute to the reduction of *separation* performance.

The results show that the largest part of the *separation* performance reduction was not caused by starting the turn at the wrong moment but by not giving the necessary speed (and altitude) changes. This result can be applied for the development of further assistance. The results indicate that an assistance advising only which speed and altitude should be selected and when it should be implemented could help to increase the *separation* performance. In contrast, a further assistance (besides Ghosting) helping the operator to find the correct moment for starting a turn maneuver is only necessary to prevent the remaining 6.4 % of the performance reduction.

However, an assistance helping with the turn maneuver could lead to a larger improvement. If turn maneuvers are executed more precisely to the planned times, less correction of the speed could be necessary and thus the benefit for the separation performance could be larger than 6.4 %. As starting the turn maneuver more precisely is not necessary but giving the required speed instructions is sufficient for a major improvement of the *separation* performance, the development of a turn assistance is also not mandatory but a speed and altitude assistance is enough to expect a higher *separation* performance.

An overall *constraints* reduction of $\Gamma^{-}_{overall} = 3.0497$ was measured during all simulation runs. Regarding this objective, turn actions contributed to about one third to the performance reduction (32.1 %) while altitude and speed changes contributed to about two thirds (67.9 %). In detail, the turn actions nearly completed contributed as commissions to the performance reduction (31.9 %). This includes cases, in which an imprecise turn maneuver is making violations of the constraints necessary in order to avoid a conflict. The influence of omissions of the turn maneuver is very low (0.2 %). This is because only in very rare situations omitting a turn can influence the *constraints* performance. Indeed, the only possible situation is an aircraft on the downwind with an instructed reduction of speed and altitude bellow the allowed value. Additionally, a turn must be planned before the constraints are violated. Omitting this turn leads to a violation. However, larger violations can be prevented in this situation by implementing corrections for speed or altitude.

Altitude and Speed changes mainly contribute to the *constraints* performance reduction as omissions. These are speed and altitude changes planned by the model but not executed by the operator. Again possible corrections for implemented speed or altitude changes causing avoidable problems are included. As an aircraft needs time to change speed and altitude, the omissions probably occurred a long time before the violations they are causing realize.

The conclusions for the development of assistance out of the results concerning the *constraints* performance are similar to the conclusions drawn from the results concerning the *separation* performance. Again, an assisting supporting the selection and implementation of required speed and altitudes commands could lead to a large improvement of missed *constraints* performance. However, concerning *constraints* the percentage would be much smaller (67.9 %).

The result for the objective *throughput* have obvious differences to both other objectives. On the one hand, the impact of turn actions (53.6 %) as well as altitude and speed actions (46.4 %) on the overall *throughput* performance reduction of $\Theta^{-}_{overall} = 7.3950$ is almost equal. On the other hand, the effect of turn actions is mainly caused by omissions (52.9 %) and not by commissions (0.7 %). This result is not surprising as omitting the turn leads to an increase of the flight duration on the path-stretching area and consequently affects the *throughput* performance. If the turn is omitted for several intervals, the planned arrival sequence has to be updated which can have a huge impact on the *throughput* performance. In contrast, the possibilities for commissions to contribute to the missed *throughput* performance are rare. First, an turn must be omitted so that the aircraft cannot hold its planned arrival time at the LMP. Second, it must be possible for another aircraft to meet the arrival time originally planned for the delayed aircraft. In this case, changing the arrival sequence has no impact on the *throughput* performance. If in such a situation the turn of the aircraft is initiated, the arrival sequence is reverted back to the original sequence but the aircraft cannot meet its original arrival time. Consequently, this implementation of the turn maneuver caused a reduction of the *throughput* performance.

The impact of wrongly executed altitude or speed changes (commission) is low again, as these commissions can be corrected. Missing corrections in turn are considered as missing actions (omissions)

Analyzing the results for the *throughput* performance allows to draw the following suggestions for improvements of the system. Both, an assistance for turn maneuvers and an assistance for speed and altitude changes, could contribute to half of the possible *throughput* performance improvements. While a major improvement of the *separation* and *constraints* performance is possible with only a speed and altitude assistance, additionally a turn assistance is necessary to make an increase of the *throughput* performance of more than 50 % possible.

It must be noted that these results are valid only for the used simulation environment. The transferability of this study results to real world applications is limited. Indeed, this study was conduct to demonstrate the benefits of the developed method and not to identify the problems in the selected demonstration environment. However, the method can be transferred. After this demonstration of this method's feasibility in a simplified setting, a real world application becomes possible.

6.9. Concluding Remarks

In this chapter the developed process-oriented model-based human operator cognitive performance measure was validated and its benefits were demonstrated. The method was validating by proving its ability to detect variations of an task difficulty (see section 6.6). However, this is only a minimum requirement as existing result-oriented performance measures are also able to detect these variations.

For this reasons, also the benefits of the developed process-oriented measure, which exceed the possibilities of result-oriented performance measures were demonstrated. The main advantage of the developed process-oriented measure is the possibility to analyze specific situations as show in section 6.7. Furthermore, the method can help to identify conditions in which the participant made decisions resulting into a performance reduction. It thereby can detect situations in which assistance could be needed. Both results can for example be used for a discussion of the simulation run with the participant. As the method allows focusing on the relevant situations, it can support a fruitful discussion.

Further the benefits for the development and improvements for systems was demonstrated in section 6.8. The overall impact of actions and errors can be used to identify the benefits of improvements of further assistance. Consequently, the method can give valuable hints to guide the further goal-oriented development of assistance. Thereby, the method enables making an estimate of the benefits of assistance before this assistance is implemented or even developed.

7. Conclusion and Future Work

In this chapter, a conclusion of this thesis is drawn fist. Subsequently, directions for future research are depicted.

7.1. Conclusion

In this thesis, a model-based process-oriented cognitive performance measure for HMSs was developed and validated for the first time. Instead of measuring the performance by evaluating only the result of an interaction—like existing measures of human operator performance in HMSs—the developed measure evaluates the sill achievable output during the interaction. Thus, it measures the performance constantly during an interaction and enables measuring the performance of each decision. Consequently, the developed performance measure is process-oriented. It allows getting more information especially about situations in which erroneous actions are executed. Additionally, impact of single actions (or their omission) on the reached performance can be identified.

As a basis for this process-oriented performance measure, a cognitive human operator model planning model based on CPNs was developed for the first time. This model completes interaction sequences realized by the human operators and transfers the resulting state into a goal state. Furthermore, the set of rules used to model human operators' behavior was developed during this thesis. The set of rules was derived from the operators' objectives and available actions in the selected example application.

The integration of projection uncertainty into the planning modeled performed in this thesis allows the differentiation between performance possible from a technical perspective and performance expectable from a human operator. It enables explaining a part of the reduced performance by mental prediction uncertainty and to concentrate on other reasons for erroneous actions impacting the performance.

A classification of uncertainty in HMSs was developed in this thesis as a prerequisite for the integration of uncertainty into the model first. This supported selecting precisely the right type of uncertainty to model. The selected approach, to represent uncertain values by integrating a bundle of rays, allows implementing uncertainty in different ways and thus easy modifications.

Moreover, an evaluation process of a process-oriented performance measure in HMS was developed in this thesis for the first time. The developed process allows the evaluation of executed and missing actions based on their impact on the achievable performance. Additionally, evaluation criteria were defined for the example application. According to the developed process, measured interaction sequences executed by human operators are combined with interaction sequences generated by the developed model and evaluated using the defined criteria. In the next step of the developed process, the evaluations of sequences are compared to transfer the result measure of combined sequences into a process measure for the measured interaction sequences.

Finally, the novel model-based process-oriented performance measure was validated in a study as part of this thesis. The sensibility of the developed measure regarding the difficulty of the task is proven in this study. This demonstrates that the developed measure can at least be used as any result-oriented measure. However, the main accomplishment of the study is to demonstrate the advantages the developed process-oriented performance measure offers compared to conventional performance measures. These are the ability to look into the details by analyzing specific situations and to identify the impact of specific kinds of actions and types of errors.

This further information gathered by the developed performance measure can be useful in discussions with the human operators. Possible explanations for their deviation from the model are erroneous actions or differences between the human operators' objectives and those implemented in the model. Whatever reason is true, a discussion can give valuable feedback for further improvements of the system. Also the aggregation of the impact of different kind of actions and types of errors on the reached performance gives useful hints for the development of further assistance. The developed performance measure allows the possible impact of assistance to be assessed before this assistance is implemented or developed.

The developed approach of model-based process-oriented performance measurement also has some drawbacks. Some are inevitably connected with the choose approach and are likely to appear if the developed measure is transferred to other applications. Other drawbacks result from the implementation and will probably not affect other kinds of implementation. A principle drawback of this approach is that it is model-based and thus limited by the capability of the model. The model is designed for a specific task and is only able to handle this task. If the developed approach to measure performance is transferred to another task, the task environment as well as the expected operators' behavior must be modeled. The more complex the task, the more extensive the development of this model will be. However, an approach task was used as example application in this thesis. As this is one of the more complicate tasks in ATC, the developed approach should be applicable to other ATC tasks without extensive effort.

The implementation of the performance measure with CPN Tools has the drawback that the calculation of interaction sequence takes a lot of time. This is not only a result of the complexity of the task but also a result of the limitations of the applied software. However, fast calculations were not required for the validation of the developed method in this thesis but might be required if this approach is transferred to other applications. Also the dependency on CPNs can be considered as a drawback. However, even if this approach was developed based on a CPN model, other modeling approaches can be used as well. Every model generating goal-oriented interaction sequences is applicable in the concept of process-oriented performance measure as developed in this thesis.

In summary, the model-based process-oriented human operator performance measure developed during this thesis extends the possibility of existing result oriented measures in HMS and gives valuable information useful for the development and further improvement of assistance.

7.2. Future Work

This thesis initiated the development of process-oriented performance measure in HMS. The validation of the model-based process-oriented human operator performance measure developed in this thesis showed that this measure is ready to use. The concept of process-oriented performance measure as well as the implementation can be extended and improved in differed ways. In the following, directions for possible future research are depicted briefly.

- As part of this thesis, mental prediction uncertainty was integrated into the human operator model. The application of this uncertainty to explain decreases of the reachable performance and to separate performance reductions due to uncertainty and performance reductions caused by an improper set of rules applied by the human operator was demonstrated. However, mental prediction uncertainty was not considered during the analysis of the conducted study. To be able to apply the uncertain human operator model for the evaluation of a study, it is first necessary to quantify the individual uncertainty parameter of a participant. This uncertainty may not only differ between participants but may also change due to learning during an interaction. Thus, a constant uncertainty parameter can only be assumed for experienced participants. Consequently, it is a task for the future to conduct a study with experienced and well-trained participants, to determine the specific uncertainty parameters, and to evaluate the potential of integrating uncertainty for the insights generated by the developed process-oriented performance measure.
- During the result evaluation of combined interaction sequences, possible candidates for erroneous actions are identified out of the given and missing actions. This identification is based on a pre-defined and fixed assignment of actions to objectives, which they can possibly affect. An improvement would be to identify erroneous actions based on their real impact on an objective. This would require to simulate given and missing actions and to evaluate their consequences. For example, an action given by the operator but not planned by the model is detected. Up to now, this action is detected as erroneous action if it fulfills the predefined conditions and a performance reduction is detected. Instead, the real impact could be calculated by adding the additionally given action to the already measured actions and simulating this extended sequence. This would allow the exact impact of this additionally given action on the objectives to be determined. The procedure would be similar in case of missing action. The measured sequence would be extended by an empty interval in this case. The integration of this procedure would be an improvement of the developed method as it would allow determining the impact of actions even in complicate settings in which a pre-defined assignment as applied in this thesis is not possible.
- The developed measure can also be improved by implementing an extended error classification. Instead of only differentiating between omissions and commissions, an extended error classification could split omissions into early, late, and incorrect

actions. The process for this extended classification is detailed in section 5.2.5. The differentiation between actions executed at the wrong time, and incorrect action gives additional information valuable for the improvement of the analyzed HMS. In the case of too late or too early actions, the operator selects the correct actions and only needs help for its execution. In case of incorrect actions, computer-based assistance for the selection of actions would be beneficial.

- The implemented error identification is based on the detection of performance reductions. This requires the interaction sequences generated by the integrated human operator model to represent the maximum reachable performance. However, if the model would include prediction uncertainty as developed in chapter 4, the interaction sequences would no represent the best reachable performance. Consequently, actions of the human operator can not only lead to a performance reduction but also to a performance increase. Consequently, not only errors (performance reduction) but also actions resulting into an outstanding performance (performance increase) could occur. Besides the existing classification of errors, this would require to implement a classification of outstanding performance. While using the optimal performance as criterion indicates which improvements are possible compared to a fully automated system, using a more realistic prediction of human behavior and consequently a not optimal performance as criterion would indicate which improvements are possible with human operators as part of the system.

- As part of this thesis, the developed performance measure was validated with an example application and the feasibility of the developed concept for process-oriented performance measurement in HMS was demonstrated. Consequently, a transfer of this measure to a high-fidelity simulation is possible as the next step. Transferring the approach to a new task requires also adapting the human operator model to this task. However, instead of a CPN-based human operator model existing planning systems can be used for the generation of goal-directed interaction sequences. For example, an AMAN could be used to generate goal-directed interaction sequences in an air traffic control approach task, or an **Surface Manager / Surface management system (SMAN)** could be used in a ground control task. However, it is important to consider if these systems generate interaction sequences representing an optimal solutions from a technical perspective or representing a feasibility solution considering the cognitive limitations of the human operator. In the first case, the necessary modifications to the approach as developed in this thesis are limited to the replacement of the model. In the second case, human operators' action can cause increases of the performance. Consequently, an additional classification of outstanding performance as discussed above is necessary.

- A possible extension of the developed performance measure is to replace to result evaluation criteria by a measure of the proximity to critical situations. The proximity to critical states could in particular be beneficial as a measure for a safety objective as considering only the actually occurred critical situation is of limited value due to the rareness of critical situations. In this thesis, the criteria based on the duration of conflicts were only meaningful as a higher frequency of critical situations was expected due to the student sample. As the approach can be applied to task with continuous and discrete characteristics, the distance to critical situation can either be a continuous distance or a discrete distance. An example for a continuous distance is the time to be elapsed before a critical situation is reached [HOS09]. An example for a discrete weighted distance is the sequence of operators leading to a critical situation [EGVS10]. As critical situation can arise by combination of both continuous and discrete changes, a combined hybrid distance measure would be required.

The developed model-based process-oriented human operator cognitive performance measure gives additionally indicators valuable for improvements of the analyzed HMS. Its main advantage is its ability to evaluate single actions executed or omitted by the human operator. Therefore, the measure is able to identify situations in which errors are conducted and the actions causing problems are executed. Consequently, an analysis or discussion of an interaction can concentrate on situation in which the problems were caused instead on focusing on situations in which the problems became obvious. By accumulating the consequences of different kind of actions and types of errors, the developed human operator performance measure allows estimating the effect of assistance systems even before they are developed.

References

[ABB+04] Anderson, J. R. ; Bothell, D. ; Byrne, M. D. ; Douglass, S. ; Lebiere, C. ; Qin, Y. An integrated theory of the mind. *Psychological Review*, 111(4):1036–1060, 2004.

[Ack92] Ackerman, P. L. Predicting individual differences in complex skill acquisition: dynamics of ability determinants. *The Journal of applied psychology*, 77(5):598–614, 1992.

[ADG+96] Adam, V. ; Danneberg, E. ; Gerling, W. ; Kohrs, R. ; Schenk, H.-D. ; Schubert, M. Systemanforderungen Projekt 4d-Planer. DLR - internal Report 112-96/19, DLR, 1996.

[Bai83] Bainbridge, L. Ironies of automation. *Automatica*, 19(6):775–779, 1983.

[Bai97] Bainbridge, L. The change in concepts needed to account for human behavior in complex dynamic tasks. *IEEE Transactions on Systems, Man, and Cybernetics - Part A: Systems and Humans*, 27(3):351–359, 1997.

[Bau96] Baumgarten, B. *Petri-Netze - Grundlagen und Anwendungen [Petri-Nets - principles and applications]*. Spektrum, Akad. Verl., Heidelberg, 2nd edition, 1996.

[Bau07] Baumgartner, M. The Organisation and operation of european airspace. In Cook, A., (editor), *European air traffic management*, pp. 1–34. Ashgate, Aldershot, England, 2007.

[BD93] Brehmer, B. ; Dörner, D. Experiments with computer-simulated microworlds: escaping both the narrow straits of the laboratory and the deep blue sea of the field study. *Computers in Human Behavior*, 9(2-3):171–184, 1993.

[BDHK83] Buckley, E. P. ; DeBaryshe, B. D. ; Hitcher, N. ; Kohn, P. Methods and measurements in real-time air traffic control system simulation. Report DOT/FAA/CT-83/26, FAA, 1983. [Available from: `www.dtic.mil/dtic/tr/fulltext/u2/a193533.pdf`].

[BKG+00] Bisantz, A. ; Kirlik, A. ; Gay, P. ; Phipps, D. ; Walker, N. ; Fisk, A. Modeling and analysis of a dynamic judgment task using a lens model approach. *IEEE Transactions on Systems, Man, and Cybernetics - Part A: Systems and Humans*, 30(6):605–616, 2000.

[Bre92] Brehmer, B. Dynamic decision making: human control of complex systems. *Acta Psychologica*, 81(3):211–241, 1992.

[Cac98] Cacciabue, P. C. *Modelling and simulation of human behaviour in system control*. Springer, London, England, 1998.

[GPo13] Cognitive and Industrial Psychology, Heinrich Heine Universität Düsseldorf. G*Power version 3. `www.psycho.uni-duesseldorf.de/abteilungen/aap/gpower3`, 2013. [Online; accessed 23-March-2014].

[Coh88] Cohen, J. *Statistical power analysis for the behavioral sciences*. Lawrence Erlbaum Associates, Hillsdale, NJ, USA, 2nd edition, 1988.

[Coo96] Cooksey, R. W. The methodology of social judgement theory. *Thinking & Reasoning*, 2(2-3):141–174, 1996.

[CPS97] Callantine, T. ; Palmer, E. ; Smith, N. Model-based crew activity tracking for precision descent procedure refinement. *IEEE Int. Conf. Systems, Man, and Cybernetics.*, pp. 3126–3131, 1997.

[Dek06] Dekker, S. *The field guide to understanding human error*. Ashgate, Aldershot, England, 2nd edition, 2006.

[DKv00] Dittmann, A. ; Kallus, K. W. ; van Damme, D. Integrated task and job analysis of air traffic controllers - phase 3: baseline reference of air traffic controller tasks and cognitive processes in the ECAC area. Report EATMP HUM.ET1.ST01.1000-REP-05, Eurocontrol, 2000. [Available from: `www.eurocontrol.int/sites/default/files/content/documents/nm/safety/safety-integrated-task-and-job-analysis-of-air-traffic-controllers-phase-3-baseline-reference-of-air-traffic-controller-tasks-and-cognitive--rocesses-in-the-ecac-area.pdf`].

[Dör80] Dörner, D. On the difficulties people have in dealing with complexity. *Simulation & Gaming*, 11(1):87–106, 1980.

[Dör89] Dörner, D. *Die Logik des Mißlingens - Strategisches Denken in komplexen Situationen [The logic of failure - strategic thinking in complex situations]*. Rowohlt, Reinbek bei Hamburg, 1989.

[Doy99] Doyle, J. Rational decision making. In Wilson, R. A. ; Keil, F. C., (editors), *The MIT encyclopedia of the cognitive sciences*, pp. 701–703. MIT Press, Cambridge, MA, USA, 1999.

[DSK99] Degani, A. ; Shafto, M. ; Kirlik, A. Modes in Human-Machine Systems: constructs, representation, and classification. *The Int. Journal of Aviation Psychology*, 9(2):125–138, 1999.

[EGVS10] Ertle, P. ; Gamrad, D. ; Voos, H. ; Söffker, D. Action planning for autonomous systems with respect to safety aspects. *IEEE Int. Conf. Systems, Man, and Cybernetics*, pp. 2465–2472, Istanbul, Turkey, 2010.

[EK95] Endsley, M. R. ; Kiris, E. O. The Out-of-the-Loop performance problem and level of control in automation. *Human Factors: The Journal of the Human Factors and Ergonomics Society*, 37(2):381–394, 1995.

[End87] Endsley, M. R. The application of human factors to the development of expert systems for advanced cockpits. *Proc. of the Human Factors and Ergonomics Society 31st Annual Meeting*, pp. 1388–1392, 1987.

[End88] Endsley, M. Situation awareness global assessment technique (SAGAT). *Proc. IEEE Nat. Aerospace and Electronics Conf.*, pp. 789–795, 1988.

[End95a] Endsley, M. R. Measurement of situation awareness in dynamic systems. *Human Factors: The Journal of the Human Factors and Ergonomics Society*, 37(1):65–84, 1995.

[End95b] Endsley, M. R. Toward a theory of situation awareness in dynamic systems. *Human Factors: The Journal of the Human Factors and Ergonomics Society*, 37(1):32–64, 1995.

[End99] Endsley, M. R. Level of automation effects on performance, situation awareness and workload in a dynamic control task. *Ergonomics*, 42(3):462–492, 1999.

[Erk99] Erkelens, L. J. Development of noise abatement procedures in the netherlands. Report, Nationaal Lucht- en Ruimtevaartlaboratorium (NLR), 1999.

[Eur12] EUROCONTROL. Two-year forecast december 2012 / flight movements 2012-2014, 2012. [Available from: `www.eurocontrol.int/sites/default/files/press_release/content/documents/official-documents/forecasts/two-year-flight-forecast-2012-2014.pdf`].

[Eur13] EUROCONTROL. Challenges of growth 2013, task 4: European air traffic in 2035, 2013. [Available from: `https://www.eurocontrol.int/sites/default/files/article//content/documents/official-documents/reports/201306-challenges-of-growth-2013-task-4.pdf`].

[Eur14U] EUROCONTROL. Fir/uir lowwer chart, 2014. [Available from: `https://www.eurocontrol.int/sites/default/files/field_tabs/content/documents/nm/cartography/09012014-firuir-upper-airspace-ectl.pdf`].

[Eur14] EUROCONTROL. Fir/uir upper chart, 2014. [Available from: `https://www.eurocontrol.int/sites/default/files/field_tabs/`

content/documents/nm/cartography/09012014-firuir-lower-airspace-ectl.pdf].

[FELB07] Faul, F. ; Erdfelder, E. ; Lang, A.-G. ; Buchner, A. G*power 3: A flexible statistical power analysis program for the social, behavioral, and biomedical sciences. *Behavior Research Methods*, 39(2):175–191, 2007.

[Fit51] Fitts, P. Human engineering for an effective air navigation and traffic control system. Report, National Research Council, Washington, DC, USA, 1951.

[FLN09] Fothergill, S. ; Loft, S. ; Neal, A. ATC-labAdvanced: an air traffic control simulator with realism and control. *Behavior Research Methods*, 41(1):118–127, 2009.

[FN08] Fothergill, S. ; Neal, A. Atc-lab advanced training package. www2.psy.uq.edu.au/~andrew/training_manual.pdf, 2008. [Online; accessed 24-March-2014].

[Fun91] Funke, J. Solving complex problems: exploration and control of complex systems. In Sternberg, R. ; Frensch, P., (editors), *Complex problem solving: Principles and mechanisms*, pp. 185–222. Lawrence Erlbaum, Hillsdale, NJ, USA, 1991.

[Fun98] Funke, J. Computer-based testing and training with scenarios from complex problem-solving research: advantages and disadvantages. *International Journal of Selection and Assessment*, 6(2):90–96, 1998.

[Fun01] Funke, J. Dynamic systems as tools for analysing human judgement. *Thinking & Reasoning*, 7(1):69–89, 2001.

[Gam11] Gamrad, D. *Modeling, simulation, and realization of cognitive technical systems*. PhD thesis, Chair of Dynamics and Control (SRS), Department of Mechanical Engineering, Faculty of Engineering Sciences, Universität Duisburg-Essen, Duisburg, 2011.

[Gaw00] Gawron, V. J. *Human performance measures handbook*. Lawrence Erlbaum Associates, Mahwah, NJ, USA, 2000.

[Gig08] Gigerenzer, G. *Rationality for mortals risk and rules of thumb*. Oxford University Press, New York, NY, USA, 2008.

[GOS07] Gamrad, D. ; Oberheid, H. ; Söffker, D. Supervision of open systems using a situation-operator-modeling approach and higher petri net formalisms. *ISIC. IEEE Int. Conf. Systems, Man, and Cybernetics*, pp. 925–930, Montreal, Que., Canada, 2007.

[GOS09] Gamrad, D. ; Oberheid, H. ; Söffker, D. Automated detection of human errors based on multiple partial state spaces. *6th Vienna Conf. Mathematical Modeling on Dynamical Systems MATHMOD*, pp. 651–659, Vienna, Austria, 2009.

[Gra02] Gray, W. D. Simulated task environments: the role of high-fidelity simulations, scaled worlds, synthetic environments, and laboratory tasks in basic and applied cognitive research. *Cognitive Science Quarterly*, 2(2):205–227, 2002.

[GS89] Green, D. M. ; Swets, J. A. *Signal detection theory and psychophysics.* Peninsula Publishing, Newport Beach, CA, USA, 1989.

[GS09a] Gamrad, D. ; Söffker, D. Reduction of complexity for the analysis of human-machine-interaction. *IEEE Int. Conf. Systems, Man, and Cybernetics*, pp. 1263–1268, San Antonio, TX, USA, 2009.

[GS09b] Gamrad, D. ; Söffker, D. Simulation of learning and planning by a novel architecture for cognitive technical systems. *IEEE Int. Conf. Systems, Man, and Cybernetics*, pp. 2233–2238, San Antonio, TX, USA, 2009.

[GS10] Gamrad, D. ; Söffker, D. Learning from conflicts in real world environments for the realization of cognitive technical systems. *IEEE Int. Conf. Systems, Man, and Cybernetics*, pp. 1995–2002, Istanbul, Turkey, 2010.

[GVM05] Gonzalez, C. ; Vanyukov, P. ; Martin, M. The use of microworlds to study dynamic decision making. *Computers in Human Behavior*, 21(2):273–286, 2005.

[Ham90] Hammond, K. Case-based planning: a framework for planning from experience. *Cognitive Science*, 14(3):385–443, 1990.

[Ham96] Hammond, K. R. *Human judgment and social policy: irreducible uncertainty, inevitable error, unavoidable injustice.* Oxford University Press, New York, NY, USA, 1996.

[Har13] HarperCollins Publishers. Collins English Dictionary. `www.collinsdictionary.com/dictionary/english`, 2013. [Online; accessed 31-May-2013].

[HC10a] Hasevoets, N. ; Conroy, P. AMAN status review 2009. Report, Eurocontrol, 2010. [Available from: `www.eurocontrol.int/sites/default/files/content/documents/nm/fasti-aman-status-review-2009.pdf`].

[HC10b] Hasevoets, N. ; Conroy, P. AMAN status review 2010. Report, Eurocontrol, 2010. [Available from: `www.eurocontrol.int/sites/default/files/content/documents/nm/fasti-aman-status-review-2010.pdf`].

[Hen99] Henrion, M. Uncertainty. In Wilson, R. A. ; Keil, F. C., (editors), *The MIT encyclopedia of the cognitive sciences*, pp. 848–850. MIT Press, Cambridge, MA, USA, 1999.

[HGS99] Hadley, G. A. ; Guttman, J. A. ; Stringer, P. G. Air traffic control specialist performance measurement database. Report DOT/FAA/CT-TN99/17, FAA, 1999. [Available from: `hf.tc.faa.gov/technotes/dot-faa-ct-tn99-17.pdf`].

[HHUR+09] Helmke, H. ; Hann, R. ; Uebbing-Rumke, M. ; Müller, D. ; Wirrkowski, D. Time-based arrival management for dual threshold operation and continous descent approaches. *8th USA/Euope Air Traffic Management Research and Development Seminar*, Napa, CA, USA, 2009.

[Hoc06] Hoc, J.-M. Planning in dynamic situations: some findings in complex supervisory control. In van Wezel, W. ; Jorna, R. ; Meystel, A. M., (editors), *Planning in intelligent systems: aspects, motivations, and methods*, pp. 61–97. John Wiley & Sons, Inc, Hoboken, NJ, USA, 2006.

[Hol98] Hollnagel, E. *Cognitive reliability and error analysis method CREAM*. Elsevier, Oxford, 1998.

[Hol00] Hollnagel, E. Looking for errors of omission and commission or the hunting of the snark revisited. *Reliability Engineering & System Safety*, 68(2):135–145, 2000.

[HOS09] Hasselberg, A. ; Oberheid, H. ; Söffker, D. State-space-based analysis of human decision making in air traffic control. *7th Workshop on Advanced Control and Diagnosis*, Zielona Gora, Poland, 2009. [Available from: `www.issi.uz.zgora.pl/ACD_2009/program/Papers/20_ACD_2009.pdf`].

[HR84] Henneman, R. L. ; Rouse, W. B. Measures of human problem solving performance in fault diagnosis tasks. *IEEE Transactions on Systems, Man, and Cybernetics*, 14(1):99–112, 1984.

[HRHR79] Hayes-Roth, B. ; Hayes-Roth, F. A cognitive model of planning. *Cognitive Science*, 3(4):275–310, 1979.

[HS09] Ha, J. S. ; Seong, P. H. HUPESS: human performance evaluation support system. In Seong, P. H., (editor), *Reliability and Risk Issues in Large Scale Safety-critical Digital Control Systems*, pp. 197–229. Springer, London, England, 2009.

[HS13a] Hasselberg, A. ; Söffker, D. Bewertung von Bedienentscheidungen mit Hilfe von Zustandsraummethoden [evaluation of human operators' decisions with state space methods]. In Söffker, D. ; Kluge, A., (editors), *Online Journal Kognitive Systeme*, volume 1. Universität Duisburg-Essen,

2013. [Available from: `duepublico.uni-duisburg-essen.de/servlets/DocumentServlet?id=31361`].

[HS13b] Hasselberg, A. ; Söffker, D. A human cognitive performance measure based on available options for adaptive aiding. *12th IFAC/IFIP/IFORS/IEA Symposium on Analysis, Design, and Evaluation of Human-Machine Systems*, Las Vegas, NV, USA, 2013. [Available from: `www.ifac-papersonline.net/Detailed/61479.html`].

[HS14] Hasselberg, A. ; Söffker, D. Integration of prediction uncertainty into a human operator planning model realized with Coloured Petri Nets. *3. Interdisziplinärer Workshop Kognitive Systeme: Mensch, Teams, Systeme und Automaten*, Magdeburg, 2014.

[HV98] Howie, D. E. ; Vicente, K. J. Measures of operator performance in complex, dynamic microworlds: advancing the state of the art. *Ergonomics*, 41(4):485–500, 1998.

[HW05] Hollnagel, E. ; Woods, D. D. *Joint cognitive systems.* CRC Press, Boca Raton, FL, USA, 2005.

[ICA13] ICAO. Annex 11 to the convention on international civil aviation: air traffic services, 2013.

[JBPD11] Jha, P. D. ; Bisantz, A. M. ; Parasuraman, R. ; Drury, C. G. Air traffic controllers' performance in advanced air traffic management system: part I - performance results. *The International Journal of Aviation Psychology*, 21(3):283–305, 2011.

[JCH13] Jipp, M. ; Carstengerdes, N. ; Hasselberg, A. Intelligence and working memory capacity impact on performance and situation awareness in case of increasing stages of automation [submitted]. 2013.

[JK09] Jensen, K. ; Kristensen, L. M. *Coloured Petri Nets: modelling and validation of concurrent systems.* Springer, Berlin and Heidelberg, 2009.

[KE04] Kaber, D. B. ; Endsley, M. R. The effects of level of automation and adaptive automation on human performance, situation awareness and workload in a dynamic control task. *Theoretical Issues in Ergonomics Science*, 5(2):113–153, 2004.

[Kie06] Kiencke, U. *Ereignisdiskrete Systeme - Modellierung und Steuerung verteilter Systeme [Event-discrete systems - modelling and control of distributed systems].* Oldenbourg, München, 2nd edition, 2006.

[KK91] Klein, G. ; Klinger, D. Naturalistic decision making. *Human Systems IAC*, 2(1):16–19, 1991.

[Kle93] Klein, G. A recognition-primed decision (RPD) model of rapid decision making. In Klein, G. A., (editor), *Decision making in action*. Ablex Pub., Norwood, NJ, USA, 1993.

[Kle98] Klein, G. *Sources of power: how people make decisions*. MIT Press, Cambridge, MA, USA, 1998.

[Kle01] Klein, G. The fiction of opimization. In Gigerenzer, G. ; Selten, R., (editors), *Bounded rationality*. MIT Press, Cambridge, MA, USA, 2001.

[Klu08] Kluge, A. Performance assessments with microworlds and their difficulty. *Applied Psychological Measurement*, 32(2):156–180, 2008.

[KM12] Klußmann, N. ; Malik, A. *Lexikon der Luftfahrt [encyclopedia of aviation]*. Springer, Heidelberg, 3rd Edition, 2012.

[KOE00] Kaber, D. B. ; Onal, E. ; Endsley, M. R. Design of automation for telerobots and the effect on performance, operator situation awareness, and subjective workload. *Human Factors and Ergonomics in Manufacturing*, 10(4):409–430, 2000.

[KPS+06] Kaber, D. B. ; Perry, C. M. ; Segall, N. ; McClernon, C. K. ; Prinzel, L. J. Situation awareness implications of adaptive automation for information processing in an air traffic control-related task. *International Journal of Industrial Ergonomics*, 36(5):447–462, 2006.

[KR99] Kaber, D. B. ; Riley, J. M. Adaptive automation of a dynamic control task based on secondary task workload measurement. *International Journal of Cognitive Ergonomics*, 3(3):169–187, 1999.

[KT79] Kahneman, D. ; Tversky, A. Prospect theory: An analysis of decision under risk. *Econometrica*, 47(2):263–292, 1979.

[KvD99] Kallus, K. W. ; van Damme, D. ; Dittmann, A. Integrated task and job analysis of air traffic controllers - phase 2: task analysis of en-route controllers. Report EATMP HUM.ET1.ST01.1000-REP-04, Eurocontrol, 1999. [Available from: `www.eurocontrol.int/sites/default/files/content/documents/nm/safety/safety-integrated-task-and-job-analysis-of-air-traffic-controllers-phase-2-task-analysis-of-en-route-controllers.pdf`].

[LS97] Lipshitz, R. ; Strauss, O. Coping with uncertainty: a naturalistic decision-making analysis. *Organizational Behavior and Human Decision Processes*, 69(2):149–163, 1997.

[MC05] Macmillan, N. A. ; Creelman, C. D. *Detection theory a user's guide*. Lawrence Erlbaum Associates, Mahwah, NJ, USA, 2nd edition, 2005.

[MMFP00] Manning, C. A. ; Mills, S. H. ; Fox, C. M. ; Pfleiderer, E. Investigating the validity of performance and objective workload evaluation research (power). *3rd USA/Euope Air Traffic Management Research and Development Seminar*, Napoli, Italy, 2000.

[Möh11] Möhlenbrink, C. *Modellierung und Analyse von menschlichen Entscheidungsheuristiken mit farbigen Petrinetzen [Modeling and analysis of human decision heuristics with Coloured Petri Nets]*. PhD thesis, Dept. Mech. Eng., TU Braunschweig, Braunschweig, 2011.

[Mos09] Mosier, K. L. Searching for coherence in a correspondence world. *Judgment and Decision Making*, 4(2):154–163, 2009.

[MOW08] Möhlenbrink, C. ; Oberheid, H. ; Werther, B. A model based approach to cognitive work analysis and work process design in air traffic control. *Human factors for assistance and automation*, pp. 401–414, 2008.

[MP01] Metzger, U. ; Parasuraman, R. The role of the air traffic controller in future air traffic management: an empirical study of active control versus passive monitoring. *Human Factors: The Journal of the Human Factors and Ergonomics Society*, 43(4):519–528, 2001.

[MP05] Metzger, U. ; Parasuraman, R. Automation in future air traffic management: effects of decision aid reliability on controller performance and mental workload. *Human Factors: The Journal of the Human Factors and Ergonomics Society*, 47(1):35–49, 2005.

[MRO12] Manzey, D. ; Reichenbach, J. ; Onnasch, L. Human performance consequences of automated decision aids: the impact of degree of automation and system experience. *Journal of Cognitive Engineering and Decision Making*, 6(1):57–87, 2012.

[MS05] Manning, C. A. ; Stein, E. Measuring air traffic controller performance in the 21st century. In Kirwan, B. ; Rodgers, M. ; Schäfer, D., (editors), *Human factors impacts in air traffic management*, pp. 283–316. Ashgate, Aldershot, England, 2005.

[MS10] Möhlenbrink, C. ; Schnieder, E. Fast and frugal heuristics conquer the airport. *2nd Int. Conf. Applied Human Factors and Ergonomics*, Miami, USA, 2010.

[MSD01] Mumford, M. D. ; Schultz, R. A. ; van Doorn, J. R. Performance in planning: processes, requirements, and errors. *Review of General Psychology*, 5(3):213–240, 2001.

[Obe06] Oberheid, H. A Coloured Petri Net model of cooperative arrival planning in air traffic control. *CPN'06 - Seventh Workshop and Tutorial on Practical*

Use of Coloured Petri Nets and CPN Tools, pp. 177–196, Arhus, Denmark, 2006.

[OGS08] Oberheid, H. ; Gamrad, D. ; Söffker, D. Closed loop state space analysis and simulation for cognitive systems. *8th International Conference Application of Concurrency to System Design*, Xi'An, China, 2008.

[OHS11] Oberheid, H. ; Hasselberg, A. ; Söffker, D. Know your options - analysing human decision making in dynamic task environments with state space methods. In de Waard, D. ; Gérard, N. ; Onnasch, L. ; Wiczorek, R. ; Manzey, D., (editors), *Human Centred Automation*, pp. 285–300. Shaker Publishing, Maastricht, the Netherlands, 2011.

[Osm10] Osman, M. Controlling uncertainty: a review of human behavior in complex dynamic environments. *Psychological Bulletin*, 136(1):65–86, 2010.

[OTK+08] Oberheid, H. ; Temme, M.-M. ; Kuenz, A. ; Mollwitz, V. ; Helmke, H. Fuel efficient and noise-reduced approach procedures using late merging of arrival routes. *German Aerospace Congress 2008*, Darmstadt, Germany, 2008.

[OWM10] Oberheid, H. ; Weber, B. ; Möhlenbrink, C. Eye movement analysis to evaluate ghosting and targeting aids for controller assistance. In de Waard, D. ; Axelsson, A. ; Berglund, M. ; Peters, B. ; Weikert, C., (editors), *Human Factors: A system view of human, technology and organisation*, pp. 291–311. Shaker Publishing, Maastricht, the Netherlands, 2010.

[OWR09] Oberheid, H. ; Weber, B. ; Rudolph, M. Visual controller aids to support late merging operations for fuel efficient and noise reduced approach procedures. *Proc. of the Human Factors and Ergonomics Society 53rd Annual Meeting*, San Antonio, TX, USA, 2009.

[PBD+92] Parasuraman, R. ; Bahri, T. ; Deaton, J. ; Morrison, G. ; Barnes, M. Theory and design of adaptive automation in aviation systems. Report NAWCADWAR-92 033-60, Naval Air Warfare Center, Warminster, PA, USA, 1992. [Available from: `www.dtic.mil/dtic/tr/fulltext/u2/a254595.pdf`].

[PCV09] Parasuraman, R. ; Cosenzo, K. A. ; Visser, E. d. Adaptive automation for human supervision of multiple uninhabited vehicles: effects on change detection, situation awareness, and mental workload. *Military Psychology*, 21(2):270–297, 2009.

[Pet62] Petri, C. A. *Kommunikation mit Automaten [Communication with machines]*. PhD thesis, Dept. Math. and Physics, Technische Hochschule, Darmstadt, 1962.

[PMH00] Parasuraman, R. ; Masalonis, A. J. ; Hancock, P. A. Fuzzy signal detection theory: basic postulates and formulas for analyzing human and machine performance. *Human Factors: The Journal of the Human Factors and Ergonomics Society*, 42(4):636–659, 2000.

[PR97] Parasuraman, R. ; Riley, V. Humans and automation: use, misuse, disuse, abuse. *Human Factors: The Journal of the Human Factors and Ergonomics Society*, 39(2):230–253, 1997.

[PSW00] Parasuraman, R. ; Sheridan, T. ; Wickens, C. A model for types and levels of human interaction with automation. *IEEE Transactions on Systems, Man, and Cybernetics - Part A: Systems and Humans*, 30(3):286–297, 2000.

[PW12] Papenfuss, A. ; Weber, B. Just talk - an experimental task for investigation of team situation awareness. In de Waard, D. ; Brookhuis, K. ; Dehais, F. ; Weikert, C. ; Röttger, S. ; Manzey, D. ; Biede, S. ; Reuzeau, F. ; Terrier, P., (editors), *Human Factors: a view from an integrative perspective. Proceedings HFES Europe Chapter Conference Toulouse*, pp. 255–268, 2012. [Available from: `hfes-europe.org`].

[RA03] Rantanen, E. M. ; Ashley, N. Taxonomies of measures in air traffic control research. *12th International Symposium on Aviation Psychology*, Dayton, OH, USA, 2003.

[Ran04] Rantanen, E. R. Development and validation of objective performance and workload measures in air traffic control. Report, Institute of Aviation, University of Illinois at Urban-Champaign, 2004. [Available from: `www.aviation.illinois.edu/avimain/papers/research/pub_pdfs/techreports/04-19.pdf`].

[Ras79] Rasmussen, J. *On the structure of knowledge a morphology of mental models in a man-machine system context.* Risø National Laboratory, Roskilde, Denmark, 1979.

[Ras83] Rasmussen, J. Skills, rules, and knowledge; signals, signs, and symbols, and other distinctions in human performance models. *IEEE Transactions on Systems, Man, and Cybernetics*, 13(3):257–266, 1983.

[Ras86] Rasmussen, J. *Information processing and human machine interaction An approach to cognitive engineering.* North-Holland, New York, 1986.

[Rea90] Reason, J. *Human error.* Cambridge University Press, Cambridge, England, 1990.

[RJR+00] Remington, R. W. ; Johnston, J. C. ; Ruthruff, E. ; Gold, M. ; Romera, M. Visual search in complex displays: factors affecting conflict detection by air traffic controllers. *Human Factors: The Journal of the Human Factors and Ergonomics Society*, 42(3):349–366, 2000.

[RN11] Rothrock, L. ; Narayanan, S., (editors). *Human-in-the-Loop simulations.* Springer, London, England, 2011.

[Rot01] Rothrock, L. Using time windows to evaluate operator performance. *International Journal of Cognitive Ergonomics*, 5(1):1–21, 2001.

[Rot11] Rothrock, L. Performance measurement and evaluation in Human-in-the-Loop simulations. In Rothrock, L. ; Narayanan, S., (editors), *Human-in-the-Loop simulations*, pp. 15–53. Springer, London, England, 2011.

[RSA+98] Reichmuth, J. ; Schick, F. ; Adam, V. ; Hobein, A. ; Link, A. ; Teegen, U. ; Tenoort, S. PD/2 Final Report. PHARE/EHQ/MAN/FR PHARE/DLR/PD/2-10.2/SSR, Eurocontrol, 1998. [Available from: `www.eurocontrol.int/phare/gallery/content/public/documents/97-70-13pd2_final.pdf`].

[RV07] Rantanen, E. M. ; Vlach, P. M. Development of air traffic control measures database. *Proc. of the Human Factors and Ergonomics Society 51th Annual Meeting*, pp. 1124–1128, Baltimore, MD, USA, 2007.

[SBD+11] Sager, S. ; Barth, C. M. ; Diedam, H. ; Engelhart, M. ; Funke, J. Optimization as an analysis tool for human complex problem solving. *SIAM Journal on Optimization*, 21(3):936, 2011.

[Sch98] Schick, F. V. Methods and measurements for the evaluation of ATM tools in real-time simulations and field tests. *2nd USA/Euope Air Traffic Management Research and Development Seminar*, Orlando, FL, USA, 1998.

[Sch06] Schaub, H. Störungen und Fehler beim Denken und Problemlösen [Dysfunction and error in thinking and problem solving]. In Funke, J., (editor), *Denken und Problemlösen*, pp. 447–482. Hogrefe, Göttingen, 2006.

[Sch11] Schweig, S. Erstellung einer Mikrowelt zur entscheidungs- und konfliktorientierten Analyse von Bediener/innen technischer Prozesse [development of a microworld for the decision- and conflictoriented analysis of operators of technical processes]. Diplomarbeit, Universität Duisburg-Essen, Duisburg, 2011.

[SES2012] Sesar. European ATM master plan - edition 2. Technical report, SESAR, 2012. [Available from: `www.atmmasterplan.eu/download/29`].

[Sha99] Shafir, E. Decision making. In Wilson, R. A. ; Keil, F. C., (editors), *The MIT encyclopedia of the cognitive sciences*, pp. 220–223. MIT Press, Cambridge, MA, USA, 1999.

[Sim55] Simon, H. A. A behavioral model of rational choice. *The Quarterly Journal of Economics*, 69(1):99, 1955.

[Söf03] Söffker, D. *Systemtheoretische Modellbildung der wissengeleiteten Mensch-Maschine-Interaktion [System-theoretic modeling of knowledge-driven human-machine-interaktion]*. Logos Wissenschaftsverlag, Berlin, 2003.

[Söf04a] Söffker, D. Understanding MMI from a system-theoretic view - part I: formalization MMI introducing modified situation-operator concept. *9th IFAC, IFIP. IFORS, IEA Symp. Analysis, Design and Evaluation of Human-Machine Systems*, Atlanta, GA, USA, 2004.

[Söf04b] Söffker, D. Understanding MMI from a system-theoretic view - part II: concepts for supervision of human and machine. *9th IFAC, IFIP. IFORS, IEA Symp. Analysis, Design and Evaluation of Human-Machine Systems*, Atlanta, GA, USA, 2004.

[Söf08] Söffker, D. Interaction of intelligent and autonomous systems - part I: qualitative structuring of interaction. *Mathematical and Computer Modelling of Dynamical Systems*, 14(4):303–318, 2008.

[SV78] Sheridan, T. ; Verplank, W. Human and computer control of undersea teleoperators. Technical report, Man-Machine Systems Laboratory, MIT, 1978.

[TCE+11] Temme, M. ; Chritoffels, L. ; Elmenhorts, E.-M. ; Felux, M. ; Forster, C. ; Gerdes, I. ; Gerz. T ; Heimann, D. ; Isermann, U. ; Isfort, J. ; Jipp, M. ; Kaltenhäuser, S. ; König, R. ; Korn, B. ; Linke, F. ; Mollwitz, V. ; Röder, M. ; Scharnweber, A. ; Schmid, R. ; Stump, R. ; Tafferner, A. flexiGuide - Projektplan [flexiGuide - project plan]. DLR - internal Report 112-2011/24, DLR, 2011.

[TD08] Trochim, W. M. K. ; Donnelly, J. P. *Research methods knowledge base*. Atomic Dog/Cengage Learning, Mason, OH, USA, 3, 2008.

[WJM06] van Wezel, W. ; Jorna, R. ; Meystel, A. M., (editors). *Planning in intelligent systems: aspects, motivations, and methods*. John Wiley & Sons, Inc, Hoboken, NJ, 2006.

[Völ90] Völckers, U. Arrival planning and sequencing with COMPAS-OP at the Frankfurt ATC-center. *American Control Conference, 1990*, pp. 496–501, 1990.

[Wel99] Wellman, M. P. Utility theory. In Wilson, R. A. ; Keil, F. C., (editors), *The MIT encyclopedia of the cognitive sciences*, pp. 859–861. MIT Press, Cambridge, MA, USA, 1999.

[Wer06] Werther, B. Colored Petri Net based modeling of airport control processes. *International Conference Computational Intelligence for Modelling ,Control and Automation*, pp. 108–144, Sydney, NSW, Australia, 2006.

[WFS+10] Weber, B. ; Friedrich, M. ; Schätzle, S. ; Oberheid, H. ; Preusche, C. Vibrotaktiles Feedback zur Aufmerksamkeitslenkung bei komplexen Lotsentätigkeiten [Vibrotactile feedback to guide attention during control tasks]. In Grandt, M. ; Bauch, A., (editors), *Innovative Interaktionstechnologien für Mensch-Maschine Schnittstellen*, pp. 71–85. DGLR, Bonn, 2010.

[WK09] Westergaard, M. ; Kristensen, L. M. The Access/CPN framework: a tool for interacting with the CPN Tools simulator. In Franceschinis, G. ; Wolf, K., (editors), *Lecture Notes in Computer Science*, volume 5606, pp. 313–322. Springer, Berlin, 2009.

[WLS+10] Wickens, C. D. ; Li, H. ; Santamaria, A. ; Sebok, A. ; Sarter, N. B. Stages and levels of automation: an integrated meta-analysis. *Proc. of the Human Factors and Ergonomics Society 54th Annual Meeting*, pp. 389–393, San Francisco, CA, USA, 2010.

[WO08] Weber, B. ; Oberheid, H. Future Air Ground Integration (FAGI) - Ergebnisse der FAGI Voruntersuchung VU1 - Late Merging [Results of FAGI pre-trials VU1]. DLR - internal Report 112-2008/44, DLR, 2008.

[WO10] Weber, B. ; Oberheid, H. FAGI D 6.6.1 Ergebnisbericht. DLR - internal Report 112-2010/05, DLR, 2010.

[YLVC02] Yu, X. ; Lau, E. ; Vicente, K. J. ; Carter, M. W. Toward theory-driven, quantitative performance measurement in ergonomics science: The abstraction hierarchy as a framework for data analysis. *Theoretical Issues in Ergonomics Science*, 3(2):124–142, 2002.

Im Rahmen von Forschungs- und Projektarbeiten im Lehrstuhl SRS und im Institut FL wurde von Dipl.-Ing Andreas Hasselberg und Univ.-Prof. Dr.-Ing. Dirk Söffker die nachstehende Diplomarbeit inhaltlich betreut, wobei Bestandteile und Ergebnisse aus den Forschungs- und Projektarbeiten sowie der studentischen Qualifikationsarbeit wechselseitig in die jeweiligen Arbeiten und somit auch in diese Promotionsarbeit eingeflossen sind.

[Sch11] Schweig, S. Erstellung einer Mikrowelt zur entscheidungs- und konfliktorientierten Analyse von Bediener/innen technischer Prozesse [development of a microworld for the decision- and conflictoriented analysis of operators of technical processes]. Diplomarbeit, Universität Duisburg-Essen, Duisburg, 2011.

A. G*Power calculations

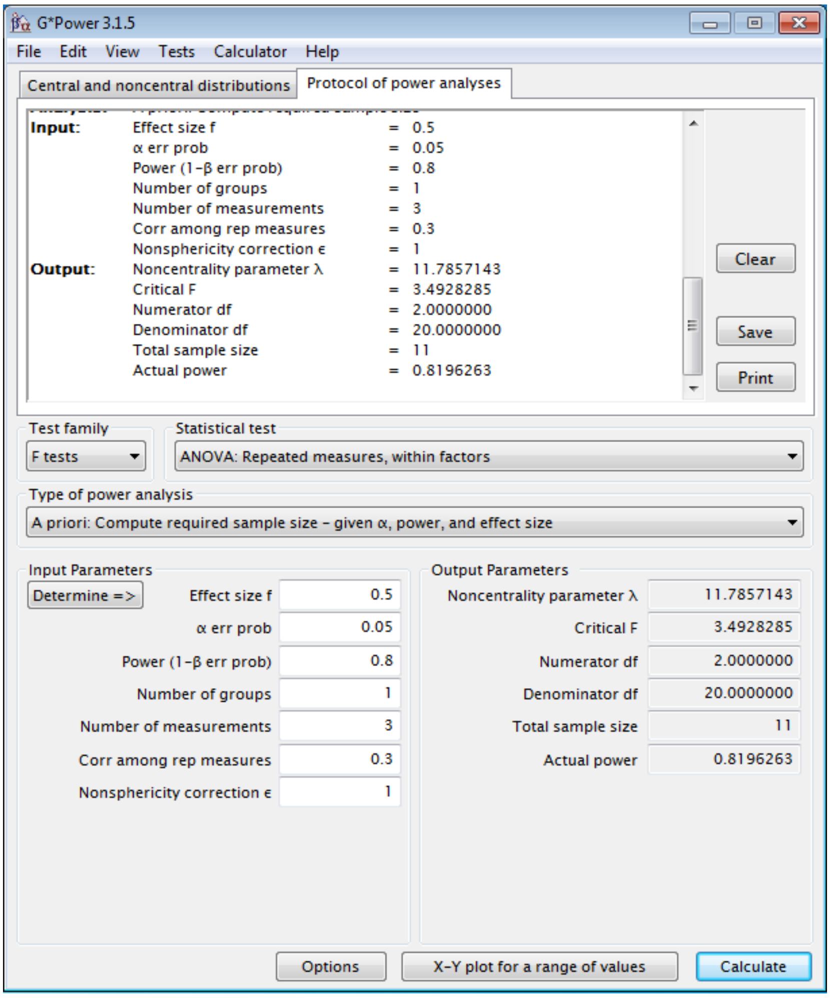


Figure A.1.: Calculation of required participants for factor *Difficulty* with G*Power

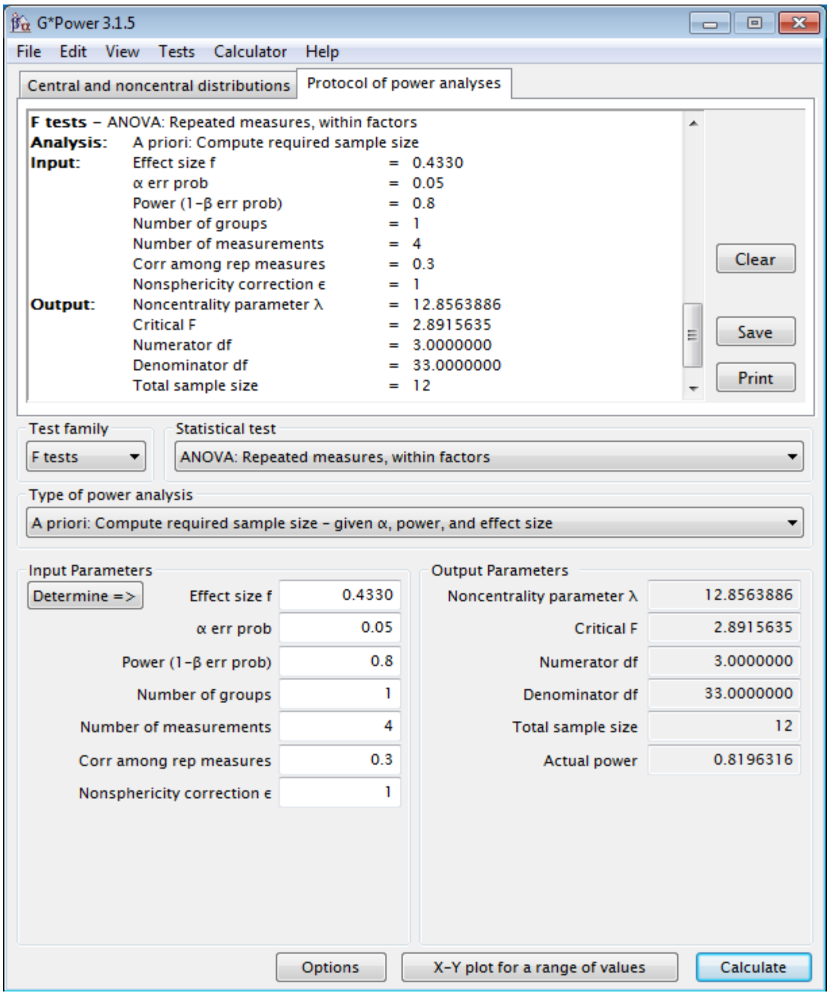


Figure A.2.: Calculation of required participants for factor *Repetition* with G*Power