EILERS, M. & MÖBUS, C., *Learning of a Bayesian Autonomous Driver Mixture-of-Behaviors (BAD-MoB) Model*, (paper accepted), 1st Intern.Conf. On Applied Digital Human Modeling, 17-20 July, 2010, Intercontinental, Miami Florida, USA

CHAPTER 1 [TIMES NEW ROMAN SIZE 16]

Learning of a Bayesian Autonomous Driver Mixture-of-Behaviors (BAD MoB) Model

Mark Eilers[1], *Claus Moebus*[2]
C.v.O. University / OFFIS, Oldenburg
D-26111 Oldenburg, Germany
http://www.lks.uni-oldenburg.de/
{FirstName.LastName}@uni-oldenburg.de

ABSTRACT

The Human or Cognitive Centered Design (HCD) of intelligent transport systems requires computational *Models of Human Behavior and Cognition* (MHBC). They are developed and used as *driver models* in traffic scenario simulations and risk-based design.

The *conventional approach* is first to develop *handcrafted* control-theoretic or artificial intelligence based prototypes and then to evaluate *ex post* their learnability, usability, and human likeness. We propose a machine-learning alternative: The *Bayesian estimation* of *MHBC*s from behavior traces. The learnt *Bayesian Autonomous Driver (BAD)* models are empirical valid by construction. An ex post evaluation of BAD models is not necessary.

BAD models can be built so that they decompose or compose skills into or from basic skills: BAD Mixture-of-Behaviors (BAD MoB) models. We present an

[1]project ISi-PADAS funded by the European Commission in the 7th Framework Program, Theme 7 Transport FP7-218552
[2]project Integrated Modeling for Safe Transportation (IMOST) sponsored by the Government of Lower Saxony, Germany under contracts ZN2245, ZN2253, ZN2366

efficient implementation which is able to control a simulated vehicle in real-time. It is able to generate complex behaviors of several layers of expertise by mixing and sequencing simpler behavior models.

Keywords: Bayesian Autonomous Driver Models, Mixture of Behavior, Mixture of Experts, Bayesian Real-Time-Control, Levels of Expertise

INTRODUCTION

The skills and the skill acquisition process of human (traffic) agents can be described by a three-stage model consisting of a *cognitive*, an *associative*, and an *autonomous* stage or layer (Fitts, 1967; Anderson, 2002). For each stage, various modeling approaches have emerged: production-system models for the *cognitive and associative* stage, control-theoretic, or probabilistic models for the *autonomous* stage.

Due to the variability of human cognition and behavior, the *irreducible lack of knowledge* about underlying cognitive mechanisms and *irreducible incompleteness* of knowledge about the environment (Bessière, 2008) we conceptualize, estimate and implement probabilistic human traffic agent models. We described first steps to model lateral and longitudinal control behavior of single and groups of drivers with simple *reactive* Bayesian sensory-motor models (Möbus and Eilers, 2008). Then we included the time domain and reported work in progress with dynamic Bayesian sensory-motor models (Möbus and Eilers, 2009a; 2009b). In this paper we propose a dynamic BAD MoB model which is able to decompose complex *maneuvers* into basic *behaviors* and vice versa. The model facilitates the management of sensory-motor schemas (= *behaviors*) in a library. Context dependent driver behavior can then be generated by mixing pure basic *behaviors*.

BASIC CONCEPTS OF BAYESIAN PROGRAMS

BAD MoB models are developed in the tradition of Bayesian expert systems (Pearl, 2009) and Bayesian (Robot) Programming (Bessière et al., 2003, 2008). A Bayesian Program (BP) (Bessiere et al., 2003, 2008, Lebeltel et al., 2004) is defined as a mean of specifying a family of probability distributions. By using such a specification it is possible to construct a driver model, which can effectively control a (virtual) vehicle. The components of a BP are presented in Fig. 1.

An *application* consists of a (behavior model) description and a question. A *description* is constructed from preliminary knowledge π and a data set δ. *Preliminary knowledge* is constructed from a set of pertinent variables, a decomposition of their joint probability distribution (JPD) and a set of forms. *Forms* are either parametric forms or questions in other BPs.

The purpose of a *description* is to specify an effective method to compute a JPD on a set of variables given a set of (experimental) *data* and preliminary knowledge.

To specify *preliminary knowledge* the modeler must *define the set of relevant variables* on which the JPD is defined, *decompose the JPD* into factors of (conditional) probability distributions (CPDs) according to conditional independency hypothesis (CIHs), and *define their forms*. Each CPD in the decomposition is a form. Either this is a *parametric form* whose parameters are estimated from batch data (behavior traces) or a *question* to another *application*. Parameter estimation from batch data is the conventional way of estimating the parameters in a BAD model. The Bayesian estimation procedure uses only a small fraction of the data (cases) for updating the model parameters.

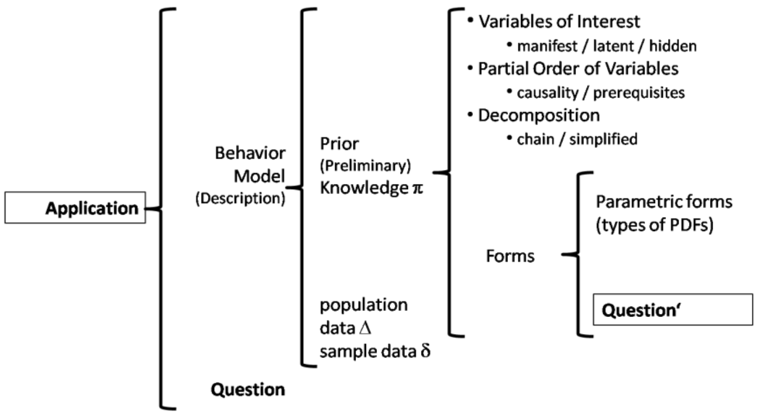

FIGURE 1. Structure of a Bayesian Program (adapted from Bessiere et al., (2003, 2008), Lebeltel et al., (2004)).

Given a description a *question* is obtained by partitioning the variables into *searched*, *known*, and *unknown* variables. A question is defined as the CPD $P(Searched|known, \pi, \delta)$. Various *policies* (Draw, Best, and Expectation) are possible whether the concrete *action* is *drawn* at random, chosen as the *best* action with highest probability, or as the *expected* action.

BAYESIAN-AUTONOMOUS-DRIVER MIXTURE-OF-BEHAVIOR MODELS

We presented a probabilistic model architecture for embedding layered models of human driver expertise which allow sharing of *behaviors* in different driving *maneuvers* (Möbus and Eilers, 2010). These models implement the sensory-motor system of human drivers in a psychological motivated *mixture-of-behaviors (MoB)* architecture with *autonomous and goal-based attention allocation* processes. A Bayesian MoB model is able to decompose complex skills into basic skills and to compose the expertise to drive complex *maneuvers* from basic *behaviors*.

We gave a proof of concept with plausible but artificial data and first modeling results with real data. We demonstrated that the Dynamic Bayesian Network (DBN)-based BAD MoB model has the ability to *predict* agent's behavior, to *abduct* hazardous situations (what could have been the initial situation), to *generate* anticipatory plans and control, and to *plan* counteractive measures by *simulating* counterfactual behaviors or actions *preventing* hazardous situations.

With an increasing number of observable action- or percept-variables and especially latent state- or behavior-variables, inferences in a BAD MoB model can soon become too complex to be computable for real-time-control. Therefore we propose an effective implementation of BAD MoB models, based on the concept of *behavior-combination* (Bessière et al., 2003), that allows to realize DBN-based BAD MoB model by several simpler BPs.

BASIC CONCEPTS OF IMPLEMENTATION

A BAD MoB model as proposed in Möbus and Eilers (2010) intends to model n *behaviors*. It contains a set of *action*-variables A, a set of *percept*-variables $P = P^1, \ldots, P^m$ and a single *behavior*-variable $B = \{1, \ldots, n\}$ with n values for the n behaviors[3]. This BAD MoB model can efficiently be implemented by BPs with three different purposes which we will call: *Action*-, *behavior-classification*- and *gating-models*.

Each *behavior* b_i $i \in \{1, \ldots, n\}$ has to be defined by an *action*-model, with preliminary knowledge π_i and sample data δ_i, consisting of the set of *action*-variables A and an own set of *percept*-variables $P_i \subseteq P$. An *action*-model defines the JPD $P(A, P_i|\pi_i \delta_i)$ that will be used to answer the question $P(A|P_i, \pi_i, \delta_i)$.

Identification of proper behaviors for a given situation is achieved using a *behavior-classification*-model. It consists of the behavior-variable B and a set of *percept*-variables $P_B \subseteq P$. They define the JPD $P(B, P_B|\pi_B \delta_B)$ and will be used to answer the question $P(B|P_B, \pi_B, \delta_B)$.

The *action*-models and *behavior-classification*-model are combined by the *gating*-model, which consists of the *action*-variables A, the *percept*-variables P and the *behavior*-variable B. Whereas the JPDs of *action*- and *behavior-classification*-models may be decomposed into simpler terms according to CIHs, the JPD of a *gating*-model is decomposed as follows:

$$P(A, P, B|\pi, \delta)$$
$$= P(P|\pi, \delta) \cdot P(B|P, \pi, \delta) \cdot P(A|P, B, \pi, \delta).$$

The decomposition of a *gating-model* consists of three terms: $P(P|\pi, \delta)$ is the prior distribution of all *percept*-variables and can be derived from (experimental) data or assumed to be uniform. The term $P(B|P, \pi, \delta)$ denotes the probability of

[3] The implementation of BAD MoB models we propose is not restricted to static components, which may be implemented as DBNs. Here we use a static example for reasons of clarity.

each *behavior* for the given percepts and will be defined as a question to the *behavior-classification*-model:

$$P(B|P,\pi,\delta) \equiv P(B|P_B,\pi_B,\delta_B).$$

For each possible behavior $B = b_i, 1 \leq i \leq n$ the term $P(A|P,B = b_i,\pi,\delta)$ is defined as a question to the corresponding i-th *action*-model:

$$P(A|P,B = b_i,\pi,\delta) \equiv P(A|P_i,\pi_i,\delta_i).$$

The question to be answered by a BAD-MoB model is $P(A|P,\pi,\delta)$. By asking this question to the *gating-model* we obtain the weighted sum over all behaviors:

$$\begin{aligned} &P(A|P,\pi,\delta) \\ &= \sum_{i=1}^{n}[P(B = b_i|P,\pi,\delta) \cdot P(A|P,B = b_i,\pi,\delta)] \\ &= \sum_{i=1}^{n}[P(B = b_i|P_B,\pi_B,\delta_B) \cdot P(A|P_i,\pi_i,\delta_i)]. \end{aligned}$$

This structure of a BAD MoB model can be seen as a template. A BAD MoB model can be extended to *hierarchical* BAD MoB model by replacing some of its *action*-models with further BAD MoB models. An example is shown in Fig. 2

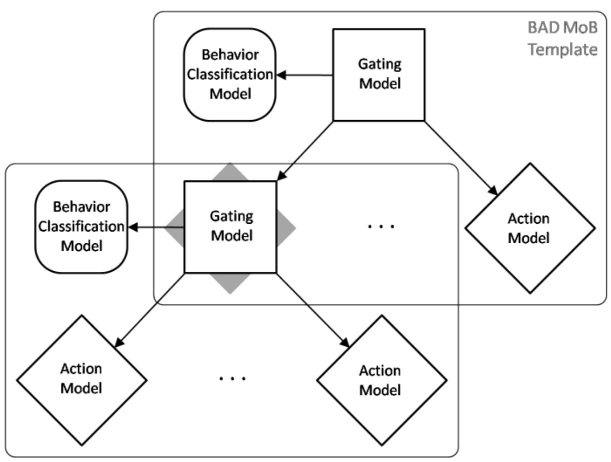

FIGURE 2. Graphical representation of a hierarchical BAD MoB model constructed by BAD MoB templates, where an *action*-model was replaced by a further BAD MoB model. Rectangle nodes represent *gating*-, rounded rectangles represent *behavior-classification*-, and diamond nodes represent *action*-models (notation adapted from Bishop and Svensen (2003)). Directed connections represent that CPDs of the

parent-model are defined to be questions of the child-model.

IMPLEMENTATION

Using the racing simulation TORCS[4] we implemented a BAD MoB model intended to master a complex *driving scenario*. The scenario covers the ability to drive on a racing track together with two other slow vehicles. When approaching a slower car, they should be followed until given the possibility for overtaking.

LEVELS OF EXPERTISE

In reference to (Möbus and Eilers, 2010), this intended *driving scenario* was split up into *driving maneuvers,* namely *Lane-Following*, *Car-Following* and *Overtaking*. *Lane-Following*, a complex maneuver by itself (Möbus and Eilers, 2009a), was supposed to be created by mixing and sequencing the *lane-following.behaviors* for driving through a left curve (*Left*), along a straight road (*Straight*) and through a right curve (*Right*). Accordingly, the maneuver *Car-Following* consists of *car-following.behaviors* for following a car through a left curve (*FollowLeft*), on a straight road (*FollowStraight*) and through a right curve (*FollowRight*). The third maneuver *Overtaking* is composed by the three *overtaking.behaviors* of veering to the left lane (*PassOut*), passing the car (*Pass Car*) and go back to the lane *(PassIn*). Each *action*-model will infer concrete actions for steering wheel angle and a combined acceleration-braking-pedal, which refers to the *driving action* level of expertise. The referring BAD MoB model therefore consists of four *gating*-, four *behavior-classification*- and nine *action*-models on three hierarchical layers, covering four levels of expertise. The structure of the model is shown in Fig. 3.

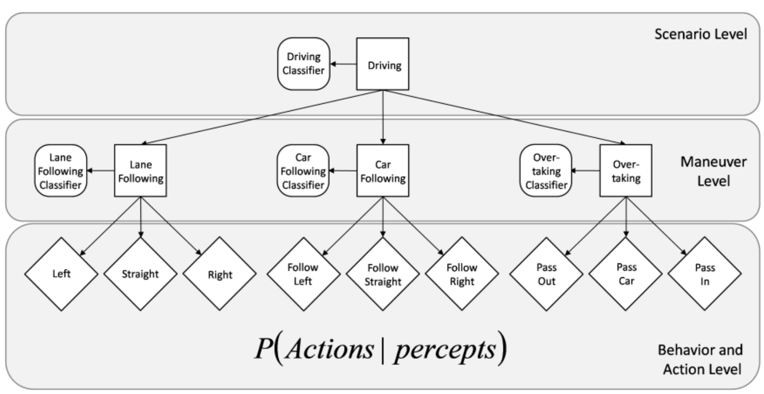

FIGURE 3. Hierarchical structure of BAD MoB model constructed by four *gating*-, four *behavior-classification-,* and nine *action*-models, covering four levels of

[4] http://torcs.sourceforge.net/ (last retrieved 2010-02-26)

expertise.

MODELING PURE BEHAVIORS BY ACTION-MODELS

Each of the nine *action*-models was implemented as a DBN. The *action*-models *Left*, *Straight*, and *Right* share the same preliminary knowledge, specify the same variables and define the same decompositions. They only differ in the experimental data set used for parameter estimation. The same applies for the *FollowLeft*, *FollowStraight*, and *FollowRight action*-models, and for the *action*-models *Pass-Out*, *Pass-Car*, and *PassIn*. Their structure is shown in Fig. 4.

For each time slice, variable $Steer^t$ represents the current steering wheel angle, Acc^t represents the position of a combined acceleration-braking-pedal. $Speed^t$ denotes the longitudinal velocity. A variable $Mid\angle_i^t$ represents the angle between heading vector of the car and the vector to the middle of the right lane in a distance of i meters. In contrast to this, a variable $Cou\angle_i^t$ represents the angle between heading of the car and the course of the road in a distance of i meters. The variables Dis^t and $Car\angle^t$ represent distance and angle to the nearest other vehicle. All pertinent variables were chosen as a tradeoff between computation speed and model performance, guided by statistical methods (i.e. *likelihood maximization*).

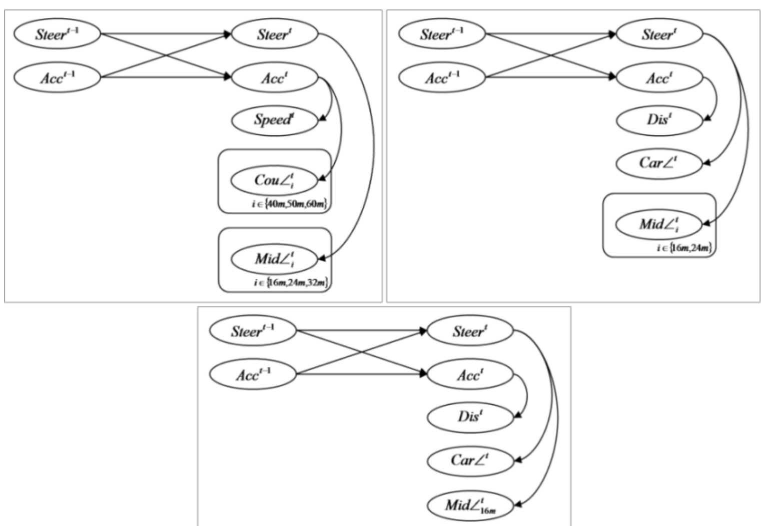

FIGURE 4. ***Upper Left***: DBN of *Left*, *Straight*, and *Right action*-models. The boxes, called plates, denote copies of the nodes shown inside the box. ***Upper Right***: DBN of *FollowLeft*, *FollowStraight*, and *FollowRight action*-models. ***Lower Middle***: DBN of *PassOut*, *PassCar*, and *PassIn action*-models.

BEHAVIOR-IDENTIFICATION BY BEHAVIOR-CLASSIFICATION-MODELS

For behavior identification each *behavior-classification*-model was implemented in form of a DBN. In each time slice, the *behavior-classification*-models define a single *behavior*-variable representing the current *driving maneuver* or *behavior*, namely DM^t for the *Driving-Maneuver-Classification* model, LFB^t for the *Lane-Following-Behavior-Classification* model, CFB^t for the *Car-Following-Behavior-Classification* model, and OB^t for the *Overtaking-Behavior-Classification* model. For all *behavior-classification*-models each time slice is implemented as *naïve Bayesian classifier*. The pertinent variables were chosen as a tradeoff between computation speed and model performance, guided by statistical methods (i.e. *likelihood maximization*). The structure of the *behavior-classification*-models is shown in Fig. 5.

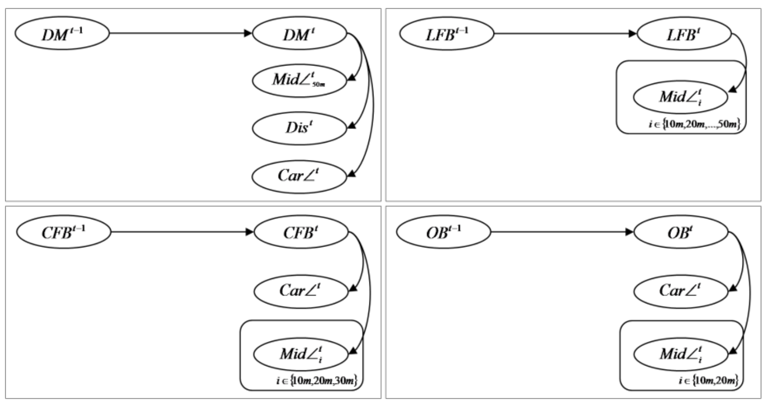

FIGURE 5. ***Upper Left****:* DBN of *Driving-Maneuver-Classification* model. ***Upper Right****:* DBN of *Lane-Following-Behavior-Classification* model. ***Lower Left****:* DBN of *Car-Following-Behavior-Classification* model. ***Lower Right****:* DBN of *Overtaking-Behavior-Classification* model.

BEHAVIOR-COMBINATION BY GATING-MODELS

Following the structure shown in Fig. 3, the *action*-models were combined by the *Lane-Following-Maneuver-*, *Car-Following-Maneuver-*, and *Overtaking-Maneuver-gating* model using their corresponding *behavior-classification*-models for behavior identification. These three *gating-models* were then combined by the *Driving-Scenario-Gating* model using the *DMC* model for maneuver identification. Considering the defined decomposition of *gating*-models, we will relinquish to show their structure.

LEARNING BY DATA COLLECTION AND BEHAVIOR ANNOTATION

For the purpose of data collection four laps were driven by a single driver, two laps at a time on two different TORCS racing tracks, containing several complex chicanes like s-shaped curves and hairpins. Instructions were given to drive sensual, stay on the right side of the road and observe a speed limit of approximately 110 km/h (70 mph). When approaching a slower car, it should be followed in short distance until a longer straight road segment would allow an overtaking-maneuver. Experimental data for parameter estimation was then obtained by recording time series of all current variable values. As values of *behavior*-variables were unknown during recording, the time series were annotated offline, manually setting the appropriated behaviors.

RESULTS

First results are very promising. With the recorded experimental data the BAD MoB model is able to accomplish the racing tracks used for data collection and other tracks of comparable complexity. The model successfully performs *Car-Following* and *Overtaking* maneuvers (an example of model-ability is shown in Fig. 6, videos are available at http://www.lks.uni-oldenburg.de/46350.html). Compared to former BAD models (Möbus and Eilers, 2008, 2009a) the driving performance was considerably improved: the BAD MoB model now stays on the right lane, sticks to the intended high speed and does not collide with roadsides anymore. In addition, the use of the proposed BAD MoB model structure significantly improved performance towards implementation of combined BAD MoB models.

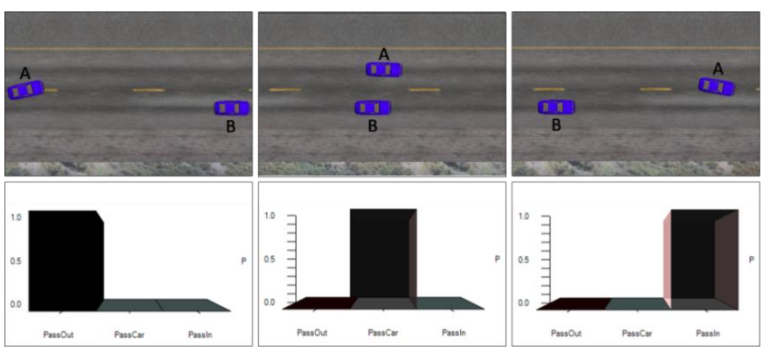

FIGURE 6. Sequencing of behaviors during *Overtaking* maneuver. Upper row shows snapshots of BAD MoB model (A) in TORCS simulation overtaking slower vehicle (B), lower row shows corresponding CPD of *overtaking.behavior* variable OB^t.

CONLUSION AND OUTLOOK

We believe that the proof of concept is convincing: Bayesian Autonomous Driver Models with Mixture-of-Behavior are expressive enough to describe and predict a wide range of phenomena. Next we have to implement further models creating a library of behaviors of various levels of expertise. To that end a careful selected taxonomy of scenarios, maneuvers, behaviors, and control actions without and with alter agents has to be defined and studied.

REFERENCES

Anderson, J.R. (2002): *Learning and Memory*, John Wiley

Bessière, P. and the BIBA INRIA Research Group (2003): Survey: Probabilistic Methodology and Techniques for Artefact Conception and Development, Technical Report RR-4730, INRIA.

Bessière, P., Laugier, Ch. and Siegwart, R. (eds.) (2008) Probabilistic Reasoning and Decision Making in Sensory-Motor Systems, Berlin: Springer, ISBN 978-3-540-79006-8

Bishop, C. M. and Svensén, M. (2003): Bayesian hierarchical mixtures of experts. In: Kjaerulff, U. and C. Meek (Ed.): Proceedings of the Nineteenth Conference on Uncertainty in Artificial Intelligence, pp. 57-64.

Fitts, P.M. and Posner, M.I., Human Performance, Belmont, CA: Brooks/Cole, ISBN 0-13-445247-X, (1967)

Möbus, C. and Eilers, M. (2008): First Steps Towards Driver Modeling According to the Bayesian Programming Approach, p.59, in L. Urbas, et al. (eds) KogWis. Christoph Hille, Dresden, ISBN 978-3-939025-14-6

Möbus, C. and Eilers, M. (2009a): Further Steps Towards Driver Modeling according to the Bayesian Programming Approach, Conference Proceedings, HCI 2009, Digital Human Modeling, San Diego, Springer: Lecture Notes in Computer Science (LNCS) and Lecture Notes in Artificial Intelligence (LNAI)

Möbus, C., Eilers, M., Garbe, H., and Zilinski, M. (2009b): Probabilistic, and Empirical Grounded Modeling of Agents in Partial Cooperative (Traffic) Scenarios, Conference Proceedings, HCI 2009, Digital Human Modeling, San Diego, Springer: LNCS and LNAI)

Möbus, C., Eilers, M., Zilinski, M., and Garbe, H. (2009c): Mixture of Behaviors in a Bayesian Driver Model, 8. Berliner Werkstatt, Mensch-Maschine-Systeme - Der Mensch im Mittelpunkt technischer Systeme, 7.-9. Oktober 2009, Berlin-Brandenburgische Akademie der Wissenschaften, VDO-Verlag

Möbus, C. and Eilers, M. (2010): Mixture-of-Behaviors and Levels Group Möbus, C. & Bessiere, P., Models of Human Behavior and Cognition in the Bayesian Programming Framework, 1st Intern.Conf. On Applied Digital Human Modeling, 17-20 July, 2010, Intercontinental, Miami Florida, USA

Pearl, J. Causality – Models, Reasoning, and Inference, 2nd ed., Cambridge University Press, 2009, ISBN 978-0-521-89560-6