Individual Driver Modeling via Optimal Selection of Steering Primitives

M. Flad * C. Trautmann * G. Diehm * S. Hohmann *

* *Karlsruhe Institute of Technology, 76131 Karlsruhe, Germany
(Tel: +49-721-608-42467; e-mail: flad@kit.edu,
trautmann.clemens@web.de, soeren.hohmann@kit.edu)*

Abstract: A shared lane keeping assistance system supports the driver in the steering task. In contrast to an autonomous system, the shared system works in parallel with the driver by applying an additional torque on the steering wheel. This means the system and the driver perform the steering task in cooperation. As the driver is still part of the control loop a driver model is required to predict the steering behavior of the actual driver. In this paper we introduce a new lateral steering model which is suited to characterize individual drivers. This model describes, in contrast to other models, the neuromuscular system, limbs and its control for a specific driver by using a set of dynamic primitives (so called movemes). These movemes build a gray-box model for the neuromuscular system. The steering wheel angle is the predicted output of the moveme model. In order to generate a steering angle trajectory suited for the desired maneuver, the steering model switches between these movemes. Therefore, the central component of the driver model is a framework which determines the optimal switching sequence of the movemes. For this task an optimal control strategy is introduced. The approach is validated using a simulation of an ISO-double lane change with movemes which were identified from a set of real driver trajectories. The results show that the steering trajectories of the driver model highly correspond with the recorded driver trajectories.

Keywords: driver model; driver behavior; shared control; human factors; human-centered design.

1. INTRODUCTION

Unintended departure of the driving lane is one of the most common traffic accident scenarios. In 2011 it was the main cause for major traffic accidents with a percentage of 26.1% of all accidents in Germany Unger [2012]. Advanced driver assistance systems (ADAS) for lane keeping support can improve the lane keeping performance of drivers. See e.g. Mulder et al. [2012]. This leads to the implication that a wide introduction of these systems can significantly reduce the amount of accidents. In the context of aircraft automation (e.g. see Wiener and Curry [1980]), it is long known that full automated systems, which remove the driver from the control loop and force him into a supervisory task, bear several problems. The driver is likely to lose his situation awareness. Furthermore a degeneration of the drivers manual control skills is expected. This can lead to situations where the driver cannot resume manual control adequately after an automation failure occurs.

Therefore, a shared control structure as introduced in Flemisch et al. [2008], Mulder et al. [2012], Itoh et al. [2012], Mars et al. [2014] is desirable for the design of a lane keeping ADAS. Within certain limitations the driver and the ADAS control the vehicle in cooperation and hence, this class of systems has a shared control structure. For the steering task such a structure can be achieved if both - driver and ADAS - can apply a torque on the steering wheel as depicted in Fig. 1.

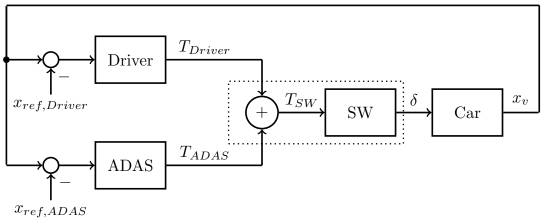

Fig. 1. Shared control structure between driver and ADAS for the lateral steering task. (With steering wheel SW, steering wheel angle δ, torque applied on the steering wheel by the driver T_{Driver}, torque applied by the ADAS T_{ADAS} and the resulting torque T_{SW}, vehicle state x_v, reference of the driver $x_{ref,Driver}$ and reference of the ADAS $x_{ref,ADAS}$)

For a successful cooperation between ADAS and the driver in the control loop depicted in Fig. 1 it is essential that the system can predict the drivers steering behavior Flemisch et al. [2008]. An individual driver model can be used as a basis for a driver specific parameterized assistance system. Some authors suggest that an ADAS should behave like a copy of the driver Abbink et al. [2012] wherefore a specific driver model is required. In long-term, if the future steering behavior of the driver can be predicted, errant interventions by the ADAS can be reduced.

Typically, only low fidelity general models are used in state of the art systems. But for an individual prediction a driver specific model is needed. Thereby it is also necessary to include the driver's neuromuscular behavior and the limb dynamics. To the best of our knowledge such a driver specific model structure, capable of predicting the real steering behavior of the specific driver, has not been presented yet. Furthermore, the driver specific parameters of our new model are easy to obtain compared to the parameter of other driver models (e.g. Pick and Cole [2008]). Our model focuses on the control level of the steering task. We assume there are higher level inputs given which give information about the future maneuver respectively the desired path.

State of the art online capable driver models used for shared control such as Pick and Cole [2008] or Addink and Mulder [2010] assume the neuromuscular dynamic to be modeled by a dynamic model followed by the kinematic model of the limbs and the attached human-machine-interface. The former include well understood mechanisms of the locomotion system as active muscle stiffness and the reflex feedback loop (see Pick and Cole [2008] and Droogendijk [2010]). In those driver models the driving task is mainly performed by a leading controller that calculates the control inputs for the car and commands them to the neuromuscular dynamic. This results in the structure of a driver model depicted in Fig. 2. The leading controller is mainly assumed as a classical linear controller like in Hess and Modjtahedzadeh [1990] or bases on optimal control Keen and Cole [2006], MacAdam [1981].

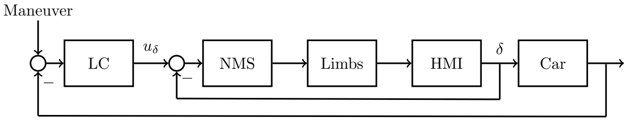

Fig. 2. Structure of common biological inspired driver models with leading controller LC, neuromuscular system NMS, limb dynamics and human-machine-interface HMI (Steering wheel angle δ, set-point from leading controller u_δ)

In the state of the art of driver models, less attention is paid to how a human realizes its individual motion. There exists strong evidence in neurobiology that humans realize their subliminal motion by combining a finite set of individual elementary building blocks called motor primitives. See Mussa-Ivaldi and Solla [2004] and Hart and Giszter [2010]. In this paper we therefore assume that humans perform the lateral vehicle control task using a finite set of motor primitives. This set of driver specific steering primitives can be determined by analyzing recorded steering wheel trajectories. This leads to the new driver model structure introduced in the next section.

2. MODEL STRUCTURE

According to previous work Flad et al. [2013] we name the motor primitives "movemes". Each moveme is defined by an affine state space model. It is assumed that all movemes have the same structure and only differ in their parameterization. The differential equation is given by

$$\dot{\mathbf{x}}_p(t) = \mathbf{A}_i \mathbf{x}_p(t) + \mathbf{b}_i, \tag{1}$$

where $\mathbf{A}_i \in \mathbb{R}^{n_p \times n_p}$, $\mathbf{b}_i \in \mathbb{R}^{n_p}$, $\mathbf{x}_p = (x_{p,1}, ..., x_{p,n_p})^T$ is the state vector of the movemes with the length $n_p \in \mathbb{N}^+$. Thereby, the steering wheel angle is represented by $x_{p,1} = \delta$. The other elements of $\mathbf{x}_p$ depends on the used identification method. It is indeed also possible to use the driver's steering torque as basis for the movemes. The index i refers to the specific parameter set of the moveme i. The set of all movemes is given by $\mathbb{I} = \{1, 2, ..., m\}$ where m is the number of available movemes. We assume that the driver specific parameter set $\mathbf{A}_i$ and $\mathbf{b}_i$, $\forall i \in \mathbb{I}$ of the movemes are identified from real steering wheel trajectories and are known. These identification tasks can be done using Diehm et al. [2013b]. In this paper we are then confronted with the following question: For a given maneuver, which moveme of the individual set has to be chosen such that the car performs the maneuver.

To this end we introduce a switching vector $\boldsymbol{\omega}(t) = (\omega_1(t), ..., \omega_m(t)) \in \mathbb{W}$ with $\mathbb{W} = \{0, 1\}^m$, that specifies which moveme is active at time t. The generated trajectory is given by

$$\dot{\mathbf{x}}_p(t) = \sum_{k=1}^{m} \omega_k(t) \left(\mathbf{A}_k \mathbf{x}_p(t) + \mathbf{b}_k\right) \tag{2}$$

with

$$x_{p,1}(t) = \delta(t)$$

and with the steering wheel angle $\delta(t)$. At each time, only one of the movemes can be active, which results in

$$\sum_{k=1}^{m} \omega_k(t) = 1. \tag{3}$$

This leads to the new driver model structure which is depicted in Fig. 3. By switching between different individual movemes, the steering wheel is controlled and thus the vehicle lateral dynamics. Considering the human driver, the driving task is performed by higher human cognition levels which control the neuromuscular system by switching between these movemes.

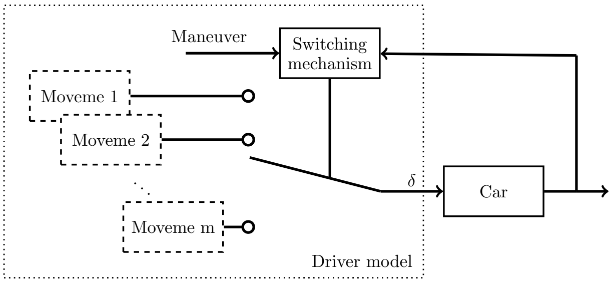

Fig. 3. Proposed driver model structure

In Flad et al. [2013] first experimental results of the driver model are proposed which shows the validity of the proposed structure. In this paper we focus on the control aspects to model the switching mechanism and its implementation.

3. SWITCHING MECHANISM

3.1 Basic Idea

In addition to the concept of primitives it is known in neuroscience that humans apply optimality principles

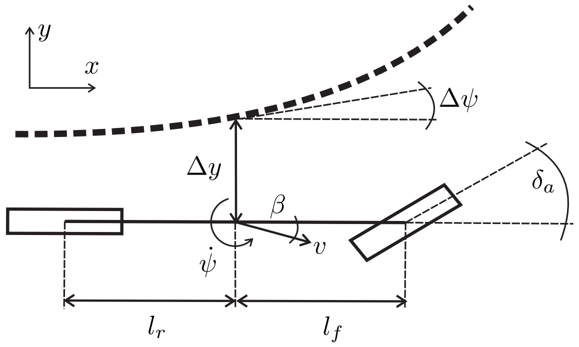

Fig. 4. Quantities of the single-track vehicle model

in their motion control Todorov [2004], Bhushan and Shadmer [1999]. Thus motion control can be modeled as an anticipatory open-loop optimal control problem. Anticipatory behavior is also a basic principle for modeling the steering task in traditional driver models Peng [2002].

We therefore model the switching mechanism by using the concept of optimal preview control. We assume that the driver has a mental model of his vehicle and the steering task which includes a model of his internal neuromuscular system (the moveme model of section 2). Based on these models the driver determines the optimal open-loop sequence of movemes with respect to an objective function and the given maneuver. Experimental results suggest that drivers tend to use a fixed and finite preview time Ungoren and Peng [2005], Ayres et al. [2001]. We include this in our model by determining the movemes for a fixed preview horizon. Godthelp [1988] states that there is a constant switching between an error-neglecting strategy, where the driver ignores minor path errors, and an error-correction strategy, which is applied when the errors can still be comfortably corrected. This behavior is also included in our driver model.

3.2 Formalization

The mental model of the vehicle is modeled using a linear single-track model (see Fig. 4). The dynamic of the vehicle model is given by

$$\dot{\mathbf{x}}_v(t) = \mathbf{A}_v(t)\mathbf{x}_v(t) + \mathbf{b}_v(t)\delta(t) \tag{4}$$

with the state vector

$$\mathbf{x}_v(t) = \left(\beta(t), \dot{\psi}(t), \psi(t), y(t)\right)^T$$

the system matrix and input vector

$$\mathbf{A}_v = \begin{pmatrix} {}^{(-C_f-C_r)}\!/_{Mv} & {}^{(-Mv^2-C_f l_f+C_r l_r)}\!/_{Mv^2} & 0 & 0 \\ {}^{(C_r l_r - C_f l_f)}\!/_{J_z} & {}^{(-C_f l_f^2 - C_r l_r^2)}\!/_{J_z v} & 0 & 0 \\ 0 & 1 & 0 & 0 \\ v & 0 & v & 0 \end{pmatrix}$$

$$\mathbf{b}_v = \frac{1}{i_s}\begin{pmatrix} {}^{C_f}\!/_{Mv} \\ {}^{(C_f l_f)}\!/_{J_z} \\ 0 \\ 0 \end{pmatrix},$$

where M is the vehicle mass, v is the velocity which is assumed to be constant, J_z is the yaw inertia, i_s the transmission ratio of the steering system and C_f respectively C_r constants of the linearized tire models. ψ is the yaw angle, β is the slip angle and y the lateral position as depicted in Fig. 4. The model is suited for steering maneuvers up to an lateral acceleration of $4m/s^2$ (see Mitschke and Wallentowitz [2004]) and hence well suited for our application.

The combination of the moveme model structure (2) and the vehicle model (4) results in the state space model

$$\dot{\mathbf{x}}(t) = \mathbf{f}(\mathbf{x}(t), \boldsymbol{\omega}(t), t) \tag{5}$$

with $\mathbf{x} = (\mathbf{x}_v, \mathbf{x}_p)^T$ and $\mathbf{f} : \mathbb{R}^{4+n_p} \times \mathbb{W} \to \mathbb{R}^{4+n_p}$. Due to (2) the model is nonlinear and non smooth. The input for the combined systems is now the binary vector $\boldsymbol{\omega}(t)$ with m elements. According to our basic idea, the trajectories of this vector should be determined optimal according to the following objective function. The model focuses on the control level, therefore explicit a-priori reference trajectories of the maneuver are used to define the maneuver. Hence the error

$$e_y(t) = y(t) - y_{ref}(t), \tag{6}$$

between the predicted trajectory of the lateral position $y(t)$ and the given reference of the maneuver $y_{ref}(t)$ and the error of the yaw angle

$$e_\psi(t) = \psi(t) - \psi_{ref}(t) \tag{7}$$

is minimized.

Path planning is assumed to be performed in a higher level of the driver model. But from the technical point of view the path planning problem can also be included in the switching framework and more abstract information like street maps and the navigable space on the road can be used as inputs for the optimization.

To include the aforementioned error-neglecting strategy observed for real drivers Godthelp [1988], a function $N(\boldsymbol{\omega}(t), t_0, T_s)$ is proposed which counts the changes of movemes. Changing the active moveme increases the value of N. We introduce the objective function for the finite preview horizon $[t_0, t_0 + T_s]$ by

$$J(\cdot) = q_p N(\boldsymbol{\omega}(t), t_0, T_s) + \int_{t_0}^{t_0+T_s} \mathbf{e}(t)^T \mathbf{Q}\mathbf{e}(t)\, dt, \tag{8}$$

$$\mathbf{e}(t) = \begin{pmatrix} e_\psi(t) \\ e_y(t) \end{pmatrix}, \mathbf{Q} = \begin{pmatrix} q_\psi & 0 \\ 0 & q_y \end{pmatrix}$$

with some weighting factors $q_p, q_\psi, q_y \in \mathbb{R}^+$. Ideally these parameters are determined for the specific driver by solving the inverse optimal control problem. This is not discussed here.

Hence $\boldsymbol{\omega}(t)$ is calculated by solving the following dynamic optimization problem:

$$\min_{\boldsymbol{\omega}(t)} \left(q_p N(\boldsymbol{\omega}(t), t_0, T_s) + \int_{t_0}^{t_0+T_s} \mathbf{e}(t)^T \mathbf{Q}\mathbf{e}(t)\, dt \right) \tag{9a}$$

with respect to

$$\dot{\mathbf{x}}(t) = \mathbf{f}(\mathbf{x}(t), \boldsymbol{\omega}(t), t), \tag{9b}$$

$$\mathbf{x}(t_0) = \mathbf{x}_0, \tag{9c}$$

$$\sum_{k=1}^{m} \omega_k(t) = 1, \tag{9d}$$

$$\omega_k(t) \in \{0, 1\} \;\; \forall k \in \mathbb{I}, \tag{9e}$$

$$\mathbf{h}(\mathbf{x}(t)) \leq \mathbf{0}. \tag{9f}$$

The constraint function $\mathbf{h} : \mathbb{R}^{n_p+4} \rightarrow \mathbb{R}^{n_h}$ is used to specify constraints on the state trajectories. For example, the function $\mathbf{h}$ is used to restrict the steering wheel angle δ to its mechanical limitations.
Together with the moveme set and if (9) is solved iteratively at $t_0 + \kappa T_c$ based on the actual states with $\kappa \in \mathbb{N}^+$and T_c the control time $T_c \leq T_s$, the individual closed-loop driver model is completely determined.

4. SUB-OPTIMAL SWITCHING MECHANISM

Problem (9) is a constrained boolean dynamic optimization problem as m input trajectories $\boldsymbol{\omega}(t) \in \{0,1\}^m$ need to be determined. Compared to other boolean problems like the control of power converters as in Holderbaum [2002] m is large. In addition the system equations are nonlinear. To the best of the authors knowledge there exists no method which can determine the exact solution of the continuous problem (9) in finite time. Suboptimal solutions are still very time consumptive as state of the art algorithms scale up exponentially with m or requires a numerical approximation of the gradient. For an overview of optimal control for nonlinear integer problems see Sager [2006], Kirches [2011]. We propose a new suboptimal solution which is online capable.

The trajectories $\boldsymbol{\omega}(t)$ are time discretized with T_p, see Fig. 5. This means the trajectories change only at fixed time steps $t = \alpha T_p$ with $\alpha \in \mathbb{N}^+$ and are constant during the intervals between them. It is obvious that with a sufficiently small T_p and a set of movemes, which allows the adequate reconstruction of the required steering wheel trajectory, a driving maneuver can be performed within certain error bounds, when full enumeration is used to solve the optimization problem (9). Nevertheless, the calculation effort of a full enumeration algorithm scales exponentially with m. It can hence not be used to solve the given problem in a practical framework.

Therefore, we propose a Greedy based procedure instead. This means we determine the optimal solution of $\boldsymbol{\omega}(t)$ for the first time interval $[t_0, t_0 + T_p]$ without considering the following intervals. The optimal solution is thereby simply determined by directly simulating the system for all m corresponding movemes. The solution of the n-th interval $[t_0 + (n-1)T_p, t_0 + (n)T_p]$ is then calculated based the result of the $(n-1)$-th interval. This approach does not regard switching costs in the objective function as represented by the function N. Therefore, the concept is applied to all possible combinations of discrete time intervals for which the trajectories $\boldsymbol{\omega}(t)$ are constant. For illustration all possible combinations for the example in Fig. 5 are depicted in Fig. 6. In the following the algorithm is introduced in more detail.

At first the prediction horizon T_s of problem (9) is discretized into time intervals with the duration of T_p for which the trajectories $\boldsymbol{\omega}(t)$ are assumed to be constant (see Fig. 5 for an example with $T_s = 4T_p$). Based on the time discretization all possible combinations of time intervals in which the trajectories are constant are determined. This results in

$$n_c = 2^{\left\lceil \frac{T_s}{T_p} \right\rceil - 1}$$

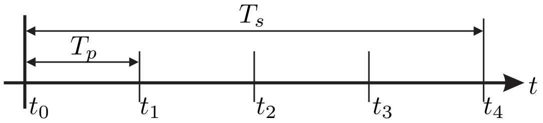

Fig. 5. Time discretization of the prediction horizon for the example $T_s = 4T_p$

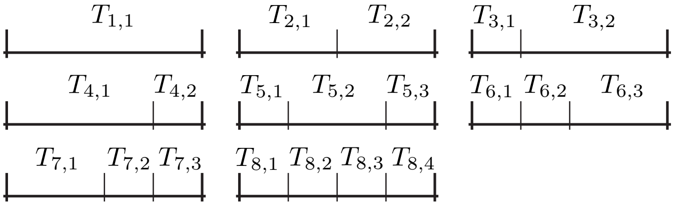

Fig. 6. All 8 possible combinations for the prediction horizon of fig. 5

possible combinations. The ceiling function $\lceil a \rceil$ is defined by $\lceil a \rceil = \min\{n \in \mathbb{Z} | n \geq a\}$. Thereby the combination k consists of $n_k \in \{1, ..., \lceil T_s/T_p \rceil\}$ time intervals $[T_{k,1}, T_{k,n_k}]$. During each of the intervals one of the m trajectories of $\boldsymbol{\omega}(t)$ is constantly 1 all others are equal to 0. For each of the combinations $k \in \{1, ..., n_c\}$ the optimal sequence of movemes is determined by using the Greedy principle. Thereby the algorithm determines the optimal $\boldsymbol{\omega}(t)$ for the first time interval $t \in T_{k,1}$ based on the objective function (8) and the actual state $\mathbf{x}(t_0)$. To determine the optimal solution the system is simulated for all m possibilities. The optimal solution for the time interval $T_{k,n}$ is then determined based on the system state

$$\mathbf{x}\left(t_0 + \sum_{i=1}^{n-1} T_{k,i}\right)$$

which is obtained by propagating the state $\boldsymbol{x}(t_0)$ with the system equation (5) and the determined foregoing optimal solution. When reaching the end of the prediction horizon $[t_0, t_0 + T_s]$ the objective function J_k is evaluated. Note that the objective function now implicitly includes the switching costs $N(\boldsymbol{\omega}(t), t_0, T_s)$ because all n_c combinations are compared to select the optimal solution. The flowchart of the algorithm is depicted in Fig. 7.

Lemma 1. The complexity of the proposed algorithm depicted in Fig. 7 scales only linearly with m.

Proof 1. Straight forward, according to the algorithm $n_c m$ simulations for $[t_0, t_0 + T_s]$ of the system (5) and calculations of the objective function have to be performed. As T_s and n_c are independent from m the algorithm scales linearly with m.

The algorithm significantly reduces the required calculation time of the discretized problem and it is possible to solve the problem online by using state of the art hardware. It is obvious that the algorithm does not fulfill Bellman's principle and therefore does not guarantee that the global optimal solution is determined. However the algorithm guarantees, at least locally, an optimal solution. In some circumstances the computing effort can be further reduced. This will be discussed in the next section.

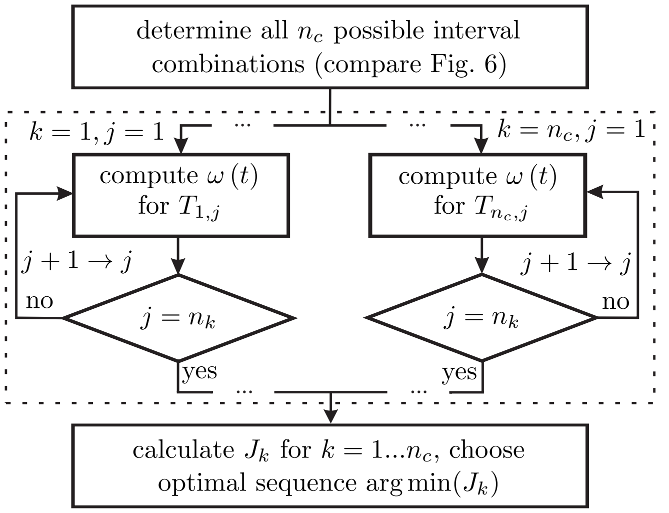
determine all n_c possible interval combinations (compare Fig. 6)
$k = 1, j = 1$
...
...
$k = n_c, j = 1$
compute $\omega(t)$ for $T_{1,j}$
compute $\omega(t)$ for $T_{n_c,j}$
$j + 1 \to j$
$j + 1 \to j$
no
$j = n_k$
$j = n_k$
no
yes
yes
...
...
calculate J_k for $k = 1...n_c$, choose optimal sequence $\arg\min(J_k)$

Fig. 7. Flowchart of the greedy based algorithm

4.1 Special Case

When the driver steering model considers only the error of the vehicle trajectory at discrete time points for the objective functions, the objective function simplifies to:

$$J = \int_{t_0}^{t_0+T_s} w(t)\left(e_y(t)\right)^2 dt. \tag{10}$$

With $w(t)$ a Dirac comb respectively a sampling function defined by

$$w(t) = q_y \sum_{k=-\infty}^{\infty} \zeta(t - kT) \tag{11}$$

and the Dirac delta function ζ. If we also assume that $T = T_p$ and T_p is small enough so that the dynamic of the steering wheel angle in the interval $[t_0, t_0 + T_p]$ can be neglected, the optimization problem can be solved significantly faster.

For the aforementioned assumptions J is a convex function with respect to δ_i in $[t_0, t_0 + T_p]$. With the constant steering wheel angle in the interval δ_i.

The resulting steering wheel angle $\delta(t_0 + T)$ at a specific time point $t_0 + T$ that results by applying the moveme i ($\omega_i(t) = 1$ for $t \in [t_0, t_0 + T]$) can be calculated by D'Azzo and Houpis [1975] with

$$\mathbf{x}_p(t_0 + T) = \mathbf{\Phi}(T)\mathbf{x}_p(t_0) + \mathbf{h}(T) \tag{12}$$

and

$$\mathbf{\Phi}(T) = e^{\mathbf{A}_i T},\ \mathbf{h}(T) = \sum_{\nu=1}^{\infty} \mathbf{A}_i^{\nu-1}\frac{T^\nu}{\nu!}\mathbf{b}_i.$$

Note, the matrix $\mathbf{\Phi}(T)$ and the vector $\mathbf{h}(T)$ can be calculated offline and stored for each of the m movemes.

When δ_i for a time interval is calculated by using (12), all m movemes can be sorted with respect to δ_i. Based on this sorted list, an one dimensional search strategy can be applied to determine the optimal $\boldsymbol{\omega}(t)$ for a time interval instead of directly simulate and compare all possibilities. The flowchart of the hence so called fast search algorithm is depicted in Fig. 8.

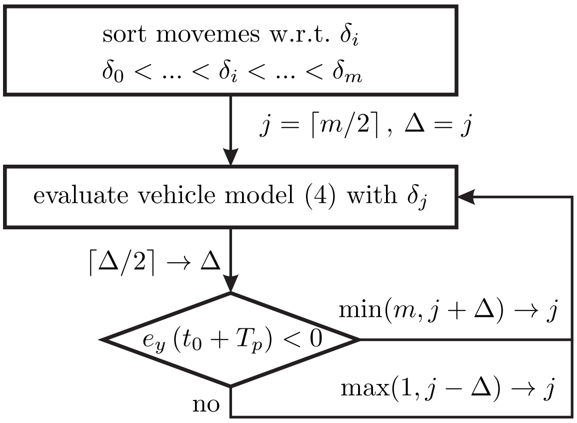
sort movemes w.r.t. δ_i
$\delta_0 < ... < \delta_i < ... < \delta_m$
$j = \lceil m/2 \rceil,\ \Delta = j$
evaluate vehicle model (4) with δ_j
$\lceil \Delta/2 \rceil \to \Delta$
$\min(m, j + \Delta) \to j$
$e_y(t_0 + T_p) < 0$
$\max(1, j - \Delta) \to j$
no

Fig. 8. Flowchart of the fast search algorithm used to determine a moveme for an interval $[t_0, t_0 + T]$

Lemma 2. The fast search strategy reduces the computation complexity of the algorithm with respect to the number of movemes m to $\lceil log_2(m) \rceil$.

Proof 2. With the stored matrices the computation of the steering wheel angle is fast compared to the other calculations and can be neglected. Therefore, the complexity of the algorithm depends linearly on the required simulations of the vehicle model that are necessary to determine the optimal moveme in an interval. We denote the iteration of the search loop in Fig. 8 with an index and name the second last iteration l. The step length Δ_l of the search procedure is always 2. To get the slowest converging scenario we assume Δ_k to be odd in every step. The reduction ratio Δ_k/Δ_{k+1} for an odd number Δ_k is smaller than for an even number. In this scenario starting backwards from l the previous Δ_{l-k} steps can be computed by

$$\Delta_{l-k} = 2 \cdot 2^k - \sum_{n=0}^{n=k-1} 2^n = 2^k \left(2 - \sum_{n=0}^{n=k-1} 2^{n-k}\right).$$

The term in brackets is between 1 and 2, so Δ_{l-k} in step $l - k$ can be estimated by $2^k \leq \Delta_{l-k} \leq 2^k \cdot 2$. For $k = l \to \Delta_0 = \lceil m/2 \rceil$. Inserting m in the previous equation and applying the log function yields $l \leq \lceil \log_2(m) - 1 \rceil \leq l + 1$. With this an upper bound of the number of iterations for the search algorithm can be estimated by $\lceil \log_2(m) \rceil$.

5. RESULTS

We focus on the comparison of the two proposed algorithms in this section. For a deeper evaluation of the driver model itself and other scenarios see Flad et al. [2013]. The program CarMaker (3.5.4) for Simulink by IPG Automotive is used for all simulations. In the simulations a validated nonlinear two-track model of a VW Golf GTD is used. The parameter of the linear vehicle model are given in Tab. 1. The calculations are done on a computer with an Intel(R)Core(TM) i7-2600 3,40 GHz processor and 8 GB RAM.

Table 1. Parameters used for the linear single-track vehicle model

	M	J_z	C_f	C_r	l_f	l_r	i_s
Value	1634	1814	2400	2400	0.91	1.67	16
Unit	kg	kgm^2	$N/°$	$N/°$	m	m	$-$

Table 2. Time horizons and parameters of the objective function used for the switching controller

Parameter	T_s	T_c	T_p	q_ψ	q_y	q_p
Value	$1.5s$	$0.3s$	$0.3s$	1	0.2	0.1

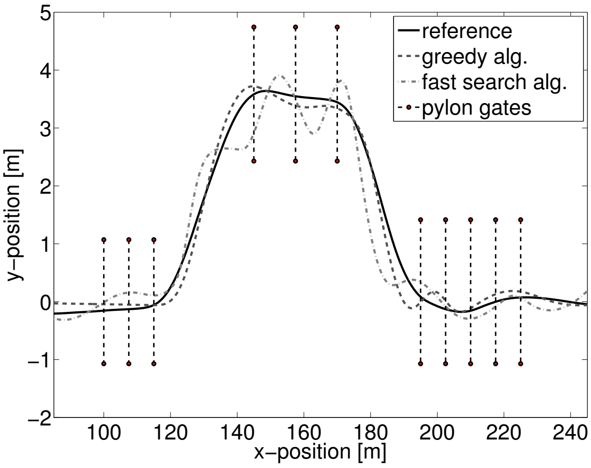

Fig. 9. Lateral-position of ISO double lane-change

Table 3. Summary of the ISO double lane-change results

	Greedy based alg.	Fast search alg.
Computing Time	$9.58s$	$5.53s$
RMSE y-position	$0.15m$	$0.31m$
Max. Error y-position	$0.31m$	$0.55m$
RMSE δ	15.9°	44.9°
Max. Error δ	74.7°	172.1°
Number of switches N	31	35

5.1 Parameters

T_c is set according to an usual average driver's reaction time. The lengths of the time horizons and the parameter of the objective function are given in Tab. 2. For a detailed introduction of the parameters please be referred to Flad et al. [2013].

The used moveme database consists of $m = 71$ movemes. n_p was chosen to 2. The parameters are identified using five test-runs of the ISO double lane-change maneuver with a fixed speed of 30 km/h. These are performed by a driver on a fixed frame driving simulator. Thereby the steering wheel trajectory is recorded and the movemes are calculated by the identification algorithm described in Diehm et al. [2013a,b]. The identification algorithm returns 71 different movemes for all five test-runs.

5.2 Simulation of ISO double lane-change

We use one of the five recorded real driver trajectories as a reference to test our switching controller. The used ISO double lane-change reference trajectory is also one of the trajectories used to identify the movemes. Full enumeration requires approximately 200 billion simulations of the system each of $1.5s$ simulation time. Hence full enumeration is not applicable. Fig. 9 shows the lateral position of the car during the lane-change maneuver when the proposed driver model is used as steering controller. The greedy based algorithm which uses the objective function (8) is depicted, as well as the fast search algorithm which uses the objective function of the special case in section 4.1 with included switching costs. Fig. 10 shows the steering wheel angle trajectories corresponding to Fig. 9. Switches of $\boldsymbol{\omega}(t)$ are marked with crosses for the greedy, respectively circles for the fast search algorithm. Additionally the results of the simulation are summarized in Tab. 3. Because of the limitation of fixed moveme switching times αT_p with $\alpha \in \mathbb{N}^+$ it is not possible for the switching controller to reconstruct the human reference trajectories exactly. As the fast search algorithm only solves a simplified optimal control problem, the result shows more deviations from the reference. There is a significant deviation to the reference steering angle trajectory as this algorithm does not consider the yaw error $e_\psi(t)$. For the straight road leading to the lane change the error of the steering wheel angle with respect to the reference is below 1.5° for the greedy based and 10° for the fast search algorithm. Overall the model in combination with the greedy algorithm succeeds in reflecting the human behavior. At some point the model deviate appreciably from the reference steering angle. Naturally the accuracy of the model is higher if it is used to predict the steering angle for a future time interval based on the correct initial steering angle Flad et al. [2013]. Also the parameters of the switching mechanism (horizons, objective function etc.) have been selected based on reference data and are not specific for the present driver. The model precision can be further increased if these specific parameters are identified.

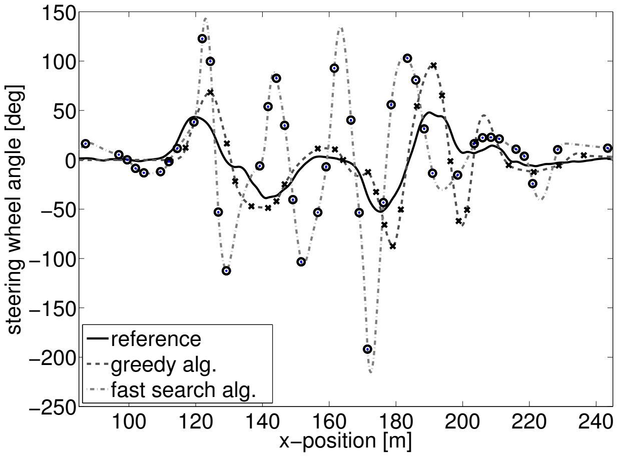

Fig. 10. Steering wheel angle. Time steps where switches of movemes occur are marked with crosses and circles

The computing time in Tab. 3 includes the time overhead the system amongst other minor things needs to calculate the nonlinear vehicle model $T_{OH} \approx 2.5s$. The lane-change maneuver lasts $35s$. The average computation time to solve the optimal control problem once using the fast search algorithm is about $30ms$. Based on Matlab code the fast search algorithm is about three times faster than the greedy algorithm.

While the greedy algorithms yields superior results with respect to the control task and reflects the human behavior better the fast search has a significant better performance regarding the calculation time.

6. CONCLUSION

The driver steering model proposed in this article combines different important neurobiological aspects. First the driver movement is modeled using motor primitives. Second, these elementary movements are combined with the idea that motion control should be modeled as an optimal control problem. Third, the steering model includes an error-neglecting strategy of the driver. These ideas result in a mathematical driver model used to describe an individual driver. From the control theoretical point of view the driver model is described as an optimal dynamic integer control problem. Using state of the art techniques this problem cannot be solved online. We proposed a new numeric algorithm which solves the problem within an acceptable computation effort. The driver model output shows a high similarity with the recorded reference steering behavior of the real driver. The new model is thus capable to predict the steering behavior of an individual driver online.

In future research it is planned to apply the model to improve shared control steering assistance systems.

REFERENCES

D.A. Abbink, D. Cleij, M. Mulder, and M.M. van Paassen. The importance of including knowledge of neuromuscular behaviour in haptic shared control. In *IEEE International Conference on Systems, Man, and Cybernetics*, pages 3350 –3355, 2012.

D.A. Addink and M. Mulder. Neuromuscular analysis as a guideline in designing shared control. *Advances in Haptics*, pages 501–514, 2010.

T.J. Ayres, L. Li, D. Schleuning, and D. Young. Preferred time-headway of highway drivers. In *IEEE Proceedings Intelligent Transportation Systems, 2001.*, pages 826–829, 2001.

N. Bhushan and R. Shadmer. Evidence for a forward dynamics model in human adaptive motor control. *Advances in Neural Information Processing Systems*, 11: 3–9, 1999.

J.J. D'Azzo and Constantine H. Houpis. *Linear control system analysis and design: conventional and modern.* McGraw-Hill, New York, 1975.

G. Diehm, S. Maier, M. Flad, and S. Hohmann. An identification method for individual driver steering behaviour modelled by switched affine systems. In *IEEE Conference on Decision and Control*, 2013a.

G. Diehm, S. Maier, M. Flad, and S. Hohmann. Online identification of individual driver steering behaviour and experimental results. In *IEEE Conference on Systems, Man, and Cybernetics*, 2013b.

C.G.I. Droogendijk. A new neuromuscular driver model for steering system development. Master's thesis, TUDelft Mechanical, Maritime and Materials Engineering, 2010.

M. Flad, C. Trautmann, G. Diehm, and S. Hohmann. Experimental validation of a driver steering model based on switching of driver specific primitives. In *IEEE Conference on Systems, Man, and Cybernetics*, 2013.

F. Flemisch, J. Kelsch, C. Löper, A. Schieben, J. Schindler, and M. Heesen. Cooperative control and active interfaces for vehicle assistance and automation. In *FISITA World automotive Congress; Munich*, 2008.

H. Godthelp. The limits of path error-neglecting in straight lane driving. *Ergonomics*, 31(4):609–619, 1988.

C.B. Hart and S.F. Giszter. A neural basis for motor primitives in the spinal cord. *The Journal of Neuroscience*, 27:1322–1336, 2010.

R.A. Hess and A. Modjtahedzadeh. A control theoretic model of driver steering behavior. *Control Systems Magazine, IEEE*, 10(5):3 –8, 1990.

W. Holderbaum. Control strategy for boolean input systems. In *Proceedings of the American Control Conference, 2002*, volume 2, pages 1260 – 1265, 2002.

M. Itoh, T. Inagaki, and H. Tanaka. Haptic steering direction guidance for pedestrian-vehicle collision avoidance. In *IEEE Conference on Systems, Man, and Cybernetics, 2012*, pages 3327–3332, 2012.

S.D. Keen and D.J. Cole. Steering control using model predictive control and multiple internal models. *Proceedings of the 8th International Symposium on Advanced Vehicle Control*, 2006.

C. Kirches. *Fast Numerical Methods for Mixed-Integer Nonlinear Model-Predictive Control.* Vieweg+Teubner Verlag, Wiesbaden, 2011.

C.C. MacAdam. Application of an optimal preview control for simulation of closed-loop automobile driving. *IEEE Transactions on Systems, Man and Cybernetics*, 11(6): 393 –399, June 1981.

F. Mars, D. Mathieu, and Jean-Michel Hoc. Analysis of human-machine cooperation when driving with different degrees of haptic shared control. *IEEE Transactions on Haptics*, (99), 2014.

M. Mitschke and H. Wallentowitz. *Dynamik der Kraftfahrzeuge.* VDI. Springer, Berlin, 4 edition, 2004.

M. Mulder, D.A. Abbink, and E.R. Boer. Sharing control with haptics: Seamless driver support from manual to automatic control. *Human Factors*, 54(5):786–798, 2012.

F.A. Mussa-Ivaldi and S.A. Solla. Neural primitives for motion control. *IEEE Journal of Oceanic Engineering*, 29(3):640 – 650, 2004.

H. Peng. Evaluation of driver assistance systems - a human centered approach. *Proceedings of the 6th International Symposium on Advanced Vehicle Control*, 2002.

A.J. Pick and D.J. Cole. A mathematical model of driver steering control including neuromuscular dynamics. *Journal of Dynamic Systems, Measurement, and Control*, 130(3):031004, 2008.

S. Sager. *Numerical methods for mixed-integer optimal control problems.* PhD thesis, University of Heidelberg, 2006.

E. Todorov. Optimality principles in sensorimotor control. *Nature neuroscience*, 7(9):907–915, 2004.

T. Unger. ADAC Unfallforschung Jahresbericht 2011. Technical report, ADAC, 2012.

A.Y. Ungoren and H Peng. An adaptive lateral preview driver model. *Vehicle System Dynamics*, 43(4):245–259, 2005.

E.L. Wiener and R.E. Curry. Flight-deck automation: promises and problems. *Ergonomics*, 23(10):995–1011, 1980.