PAPER

Identification of Time-Varying Parameters of Hybrid Dynamical System Models and Its Application to Driving Behavior

Thomas WILHELEM[†a)], **Hiroyuki OKUDA**[†], ***Nonmembers**, **and*** **Tatsuya SUZUKI**[†], ***Member***

SUMMARY This paper presents a novel identification method for hybrid dynamical system models, where parameters have stochastic and time-varying characteristics. The proposed parameter identification scheme is based on a modified implementation of particle filtering, together with a time-smoothing technique. Parameters of the identified model are considered as time-varying random variables. Parameters are identified independently at each time step, using the Bayesian inference implemented as an iterative particle filtering method. Parameters time dynamics are smoothed using a distribution based moving average technique. Modes of the hybrid system model are handled independently, allowing any type of nonlinear piecewise model to be identified. The proposed identification scheme has low computation burden, and it can be implemented for online use. Effectiveness of the scheme is verified by numerical experiments, and an application of the method is proposed: analysis of driving behavior through identified time-varying parameters.

key words: *hybrid dynamical system models, time-varying parameters identification, particle filter, driving behavior identification*

1. Introduction

Thanks to the recent development of computer technology, data-centric system design is attracting great attention [1]–[6]. In the field of system identification, although numerous dedicated mathematical models have been proposed to represent target systems [10]–[13], discrete-continuous hybrid system modeling has great potential to represent complex dynamical behavior including switching mechanisms [9].

As a result of its high describability and understandability, hybrid system modeling has been applied to various domains, such as communication systems, autopilot systems, automotive engine control, traffic control, and chemical processes [2]–[9]. A promising application domain of hybrid system modeling is the human behavior analysis and reproduction, due to the possibility to represent both the decision making and the motion control aspects of human behavior.

From the viewpoint of data-centric modeling, the PieceWise AutoRegressive eXogenous (PWARX) model is extensively used, and the identification of PWARX model has been widely studied. Various methods have been developed [15]. The clustering approach is based on dynamics clustering and on identification of each clustered data set [16]. The bounded-error approach allows to define the maximal identification error [17]. The mixed-integer programming approach guarantees to converge to a global optimum [18]. The algebraic approach defines an analogy to the identification and decomposition of an algebraic variety [19]. Finally the Bayesian approach uses the Bayesian inference to identify both parameters and mode partitionning of hybrid systems from noisy data [20].

From this perspective, the authors have proposed several hybrid system modeling methods, especially focussing on several applications of human driving behavior. A Stochastic Switching ARX (SSARX) model has been developed by extending a conventinal Hidden Markov Model [2]. Hierarchical PWARX (Hi-PWARX) has been developed with the idea to create a hierarchical structure of the data based on unsupervised clustering technique [3]. Finally the Probability-Weighted ARX (PrARX) model has been developed as a new hybrid system model wherein the mode switching is represented by softmax function, which represents the probability of mode occurrence [4].

These works are focused on the identification of time-invariant hybrid system models. When analysis of the human driving behavior is considered, it has been shown that stochastic and time-varying characteristics should be included in addition to decision taking mechanisms [30]. Each driver shows a different response to a given stimuli, leading a driver's individual statistical dispersion in the reproduction of a given action, and under long-time driving situations, drivers' behavior can vary drastically. The understanding of time-varying characteristics of driving behavior through hybrid system model parameters can inform on driving consistency, expressed as short-term variance, and long-term driving characteristics, expressed as global model parameters variations. This can also be used as a new source of information for the design of better driving assistance and health monitoring systems.

These considerations highly motivated us to develop a new identification technique for time-varying parameters of hybrid system models. The parameter identification process should identify time-varying parameters while complying to the parameters dynamics.

In the case of fitting a driving model on real-world measured data, optimality of the identified parameter is not the final goal. The model can never perfectly represent the real situation, and noise in the measurement and time-variability characteristics of the driving behavior are against the concept of optimal solutions. The major concern in this type of identification process is the ability to avoid local minima, and the ability to get parameters estimations within a known error margin. These perspectives were lacking in

Manuscript received March 15, 2017.
Manuscript revised June 14, 2017.

[†]The authors are with the University of Nagoya, Nagoya-shi, 464-8601 Japan.
a) E-mail: wilhelem@nuem.nagoya-u.ac.jp

DOI: 10.1587/transfun.E100.A.2095

the previous works [15]–[24].

To realize stochastic and time-varying hybrid system models parameters identification, parameters are regarded as random variables and identified with a Bayesian approach [20]–[22]. The parameters' time dynamics are also bounded using a time-smoothing technique.

The proposed identification method differs from the conventinal Bayesian approachs [20]–[22] in the sense that as far as we know, former studies did not consider time-varying parameter explicitely. In the main reference article on this topic [20], the prior knowledge of Bayes inference is based on the prior time step. Thus the estimated parameter $\theta(k)$ at time step k depends on the prior time step identified parameter $\theta(k-1)$, on the observation (model output) $z(k)$ and on the probability density function (pdf) p. With this formulation, the identification process directly imposes the parameter time-dynamics through the pdf p. The time-varying dynamics of the parameter are considered implicitely. On the other hand, the method introduced in this article has been developed to consider explicitely the time-varying dynamics of the parameters in the identification process. The prior knowledge of Bayes inference is based on the prior identification-step for each time step, such that $\theta^i(k)$ depends on the same time step k but previous identification iteration $\theta^{i-1}(k)$, where i the identification iteration-step, on the observation (model output) $z(k)$ and on the pdf p. A filtering process based on moving average with a pdf g is implemented to explicitely control the identified parameters time-dynamics during the parameter identification. This method enables us to separate the identification process from the time-smoothing process.

In Sect. 2 the hybrid dynamical system modeling definitions are introduced. In Sect. 3 the existing and novel parameter identification methods are detailed. Then in Sect. 4 the selected application models are presented, and in Sect. 5 examples of parameters identifications are shown. In Sect. 6 driving behavior analysis and modeling are discussed, and the conclusion is given in Sect. 7.

2. Modeling Definitions

The identified models are piecewise hybrid dynamical systems models with time-varying parameters. The usual hybrid dynamical system modeling framework [1] is described as

$$\begin{cases} y(k) = f^1\left(u^1(k)\right) + e^1(k) & \text{if} \quad \mu(k) = 1 \\ & \vdots \\ y(k) = f^M\left(u^M(k)\right) + e^M(k) & \text{if} \quad \mu(k) = M \end{cases} \tag{1}$$

where u is the model input, a time-series vector composed of an external input time-series vector s and of the observed output time-series vector y, such that $u(k) = [s(k), s(k-1), ..., s(k-n_a), y(k-1), y(k-2), ..., y(k-n_b)]$ where $n_a \in \mathbb{N}$ represents the exogenous input order, and $n_b \in \mathbb{N}^*$ represents the autoregressive input order. f^m is a set of functions with $m \in \{1, 2, ..., M\}$ the mode index number, $k \in \{1, 2, ..., K\}$ is the discrete time step, e^m is the modeling error, and $\mu \in \{1, 2, ..., M\}^K$ is the mode index vector.

In signal processing field, filtering is usually applied to state-space models. For using particle filtering in parameter identification, dynamical model parameters are usually considered as model states [21]-[22]. Expressed as a state-space simulation model, mode equations of the hybrid dynamical system model (1) become

$$\begin{cases} x^m(k) = h_p^m\left(x^m(k-1), u^m(k)\right) \\ \hat{y}(k) = h_o^m\left(x^m(k), u^m(k)\right) \end{cases} \tag{2}$$

where x is the state vector (here observable), u the simulation model input, $\hat{y}$ the simulation model output, h_p is a process function, h_o is an output fonction and $m = \mu(k)$ the mode index number.

The difference between the dynamical systems expressed by the Eq. (1) and the Eq. (2) is the formulation of the system state. In Eq. (1), the dynamical system is expressed as a transfer function, where the state of the system is implicitely expressed in the model output y, whereas in Eq. (2), the dynamical system is expressed as a state-space system, where the states are explicitely expressed by the state variable x. Thus state-space formulation enables the formulation of a larger set of models, including hiden states models. The formulation (1) is the most common for simple dynamical models such as the applications models of this article, but this formulation does not allow to clearly express the model parameters. Thus, the state-space formulation is introduced to explicitely represent how the identified parameters are included in the filtering problem. Moreover, this formulation allows notations compliance with the reference articles [21]-[22]. In that way, Eq. (2) can be considered as a generalization of Eq. (1).

The parameter vector θ is then assimilated to a state of the state-space model (2) to be identified by the particle filter as

$$\begin{cases} x^m(k) = h_p^m\left(x^m(k-1), \theta^m(k), u^m(k)\right) \\ \hat{y}(k) = h_o^m\left(x^m(k), \theta^m(k), u^m(k)\right) \end{cases} \tag{3}$$

where θ is the parameter vector extended as a model state.

Finally, expressed as a transfer function hybrid dynamical system model, the state-space simulation model (3) becomes

$$\begin{cases} y(k) = f^1\left(u^1(k), \theta_\mu(k)\right) + e^1(k) & \text{if} \quad \mu(k) = 1 \\ & \vdots \\ y(k) = f^M\left(u^M(k), \theta_\mu(k)\right) + e^M(k) & \text{if} \quad \mu(k) = M \end{cases} \tag{4}$$

where $\theta^m(k) \in \Theta^m$ is the parameter vector of the mode m at the time step k and Θ^m the parameter space of the mode m.

The concept of mode output occlusion is also introduced in this section. As described in Eq. (4), the model output y is a composition of the modes m outputs over the time steps k depending on the value of $\mu(k)$. Thus at each time step k, $M-1$ mode outputs are not observable. These non-observable mode outputs are called occluded mode outputs.

During mode output occlusion, time-varying parameters of the mode cannot be identified.

3. Parameter Identification Process

In this section, the method and the implementation of the identification process for time-varying parameters are detailed.

In numerous application cases, modeling is a rough approximation of the real measured data, and the reproduced situations are non-deterministic. Moreover the raw data used for model fitting is often known within an error margin. In those cases, as long as the parameter identification process error is lower than the modeling error or than the raw data error margin, an optimal parameters identification scheme will not bring any advantage over an metaheuristic scheme. Bayesians methods explicitely enable to select the density probability of the estimated parameters, and thus to know the error margin of the estimated parameter in case of identification convergence.

To be able to identify a wide range of hybrid system models, including non-differentiable nonlinear heterogeneous hybrid system models, a suboptimal nonparametric Bayesian method is selected. Parameters estimate (posterior) are calculated based on prior parameters estimates, on the parameters estimation pdf and on an observation. Parameters are estimated from the marginal distribution in (5) [25]. Nonparametric methods do not rely on a fixed functional form of the posterior, but instead create an approximate of the posterior state by a finite number of particles. Thus nonparametric methods converge uniformly to the correct posterior as the number of particles goes to infinity.

Bayes rule used for parameter identification is expressed as follows:

$$p\left(\theta(k)|z(1:k)\right) = \frac{p\left(z(k)|\theta(k)\right)p(\theta(k)|z(1:k-1))}{p\left(z(k)|z(1:k-1)\right)} \tag{5}$$

where $\theta(k) \in \mathbb{R}^{n_\theta}$ is the estimated time-varying parameter vector and $z(k) \in \mathbb{R}^{n_z}$ is the observation. n_θ and n_z are the parameters and observation dimensions in $\mathbb{N}^*$. $k \in \{1, 2, \ldots, K\}$ represents the current time and identification step.

To be able to identify several parameters per mode and to filter each individual parameter time-variation to its time dynamic, a novel implementation scheme of the Bayesian approach based on particle filtering combined with parameter time-smoothing algorithm is proposed in this article.

3.1 Particle Filtering for Parameter Identification

The particle filter (PF), also called Sequential Monte-Carlo (SMC), is a nonparametric Bayesian approach, creating a recursive Bayesian filtering by Monte-Carlo type sampling [22], [25]–[29]. The key idea in PF is to represent the posterior density by a set of random samples drawn from this posterior, to calculate associated weights considering an observation, and then to compute the new estimates based on these samples and weights. Markov assumption on the parameters is used in system identification to make the calculation tractable. Particle filters have been used to determine real-time nonlinear model parameters [21], hybrid system model (Piecewise ARX) constant parameters and modes estimation [20], and nonlinear non-hybrid (NARX) time-varying parameters from a predefined finite parameter set [23]. This litterature uses a conventional implementaion of the PF, and as far as we know, does not consider the parameter time-variation explicitely.

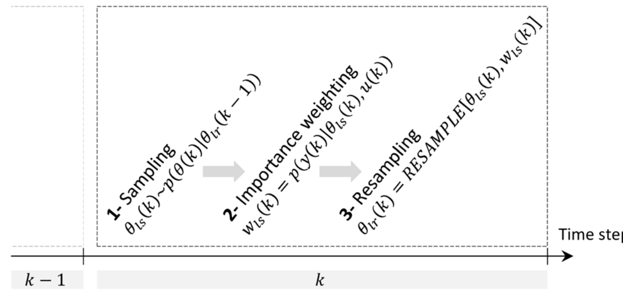

Fig. 1 Standard SIR particle filter identification process.

In this paper the Sample Importance Resample (SIR) scheme is selected over the Sequential Importance Sampling (SIS) scheme, to avoid the degeneracy problem [22], [25]. Moreover no assumption is taken on the ergodicity of the parameter time-variation, thus usage of SIS is not possible. The importance density and the resampling algorithm must be carefully selected to avoid respectively a large variance in the particles weights and sample impoverishment. The main steps of the SIR algorithm are shown in Figure 1.

Standard particle filtering method for parameter identification is expressed as follow:

Algorithm 1 SIR Particle Filter

$\{\theta_{lr}(k)\}_{lr=1}^{Lr} = SIR\left[\{\theta_{lr}(k-1), u(k), y(k)\}_{lr=1}^{Lr}\right]$
for $lr = 1 : Lr$ **do**
  Step 1: Sampling
  - Draw $\theta_{ls}(k) \sim p\left(\theta(k)|\theta_{lr}(k-1)\right)$
end for
for $ls = 1 : Ls$ **do**
  Step 2: Importance weighting
  - Calculate $w_{ls}(k) = p\left(y(k)|\theta_{ls}(k), u(k)\right)$
end for
- Calculate total weight: $w^{tot}(k) = \sum_{ls=1}^{Ls} w_l(k)$
for $ls = 1 : Ls$ **do**
  - Normalize: $w_{ls}(k) = (w^{tot}(tot))^{-1} * w_{ls}(k)$
end for
Step 3: Resampling, using Algorithm 2 in [25]
$\{\theta_{lr}(k)\}_{lr=1}^{Lr}$ =RESAMPLE$\left[\{\theta_{ls}(k), w_{ls}(k)\}_{ls=1}^{Ls}\right]$

In Algorithm 1, $lr \in \{1, 2, \ldots, Lr\}$ is the resampled particle index, $ls \in \{1, 2, \ldots, Ls\}$ is the sampled particle index, k is the discrete time step, $\theta \in \Theta$ the estimated parameter, w is the associated weigth, u the model input, and y is the model output.

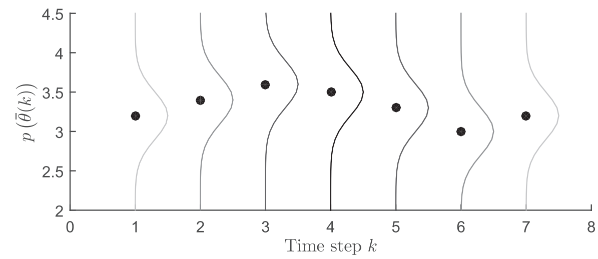

Fig. 2 Example of Normal weighting distributions $p(\bar{\theta}(k))$ over the time steps k, used to define smoothening particles weights. Black points represent the smoothened parameter estimate $\bar{\theta}(k)$, curves represent the probability distribution $p(\bar{\theta}(k))$ over the possible particles values.

3.2 Smoothing Algorithm

An algorithm of smoothing over time is implemented in the parameters identification process to avoid the effect of noise on the parameter identification, and to be able to identify parameters time-variation within specified dynamics. Parameters dynamics filtering can be done independently for each parameter, enabling dynamics decoupling of the parameters during the identification process. Smoothing can be done directly by solving the problem $p(\theta(k)|y(1:K))_{K\in\mathbb{N}^*}, k \in \{1, 2, \ldots, K\}$, but time-smoothed particle filtering methods as presented in [28] tend to be complex to implement. Thus a simple moving average method has been adapted in this work.

Parameter estimates profile is generated based on the maximum likelihood estimate (see (8)) at each time steps. This profile is time-smoothened by using a moving average method (MA) weighted by the pdf g. The smoothened parameter estimate profile is then used to attribute smoothing weights to all the particles of the identification process based on the pdf p (see Fig. 2).

3.3 Algorithm Initialization

In this section, two methods are proposed to initialize the parameter identification algorithm.

Case A: Data classification is known without a-priori knowledge on parameters.

The minimum a-priori knowledge required to initialize the parameter identification method is the modes partitioning of the hybrid system model. Numerous methods can be used to segment the data [24]. Once the mode separation is obtained, initial particles can be spread uniformly over the candidate parameters space.

Case B: Prior knowledge from Bayesian parameter identification approach.

Initial parameters and modes partition can be determined using a conventinal method developed for the identification of hybrid systems parameters. The most suitable solution for nonlinear hybrid system models is the Bayesian online identification method [20]. This method provides good results for single parameter identification, and can also be used to provide a-priori knowledge for multiple parameter cases. Nevertheless, the problem proposed in [20] is not well posed for multiple parameters cases. In this algorithm, the particles weights would have to be determined based on a number of time steps at least equal to the number of identified parameters per mode. This initialization method is only suitable for cases with clear mode separation in the data. Regarding the applications of this paper, initialization of the identification process based on real driving data is done using Case A.

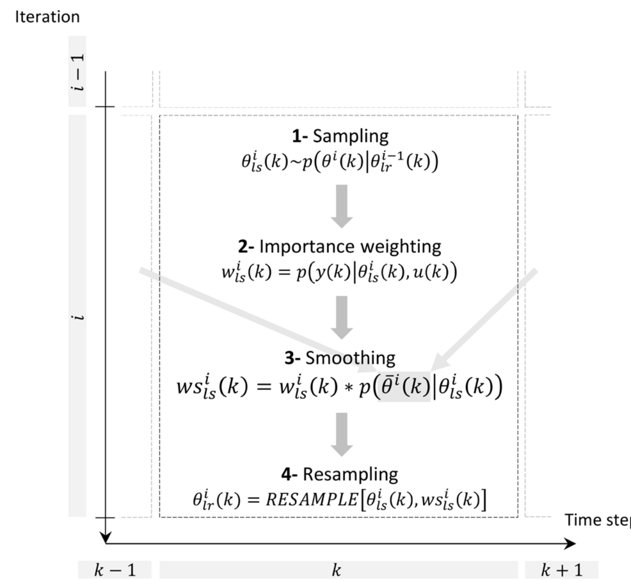

Fig. 3 Novel parameters identification scheme, composed of an iterative SIR particle filter and a time-smoothing algorithm. $\bar{\theta}^i(k)$ represents the smooth parameter estimate at iteration i and discrete time step k. Definition of $\bar{\theta}^i(k)$ is given in Eq. (9).

3.4 Novel Parameter Identification Scheme

In this section, the proposed parameter identification algorithm is described. In order to identify time-varying parameters of hybrid dynamical system models with parameter smoothing over time, the particle filter is implemented as an iterative process instead of a time dependent process (see Fig. 3). Variables and indexes notations are detailed in Tables 1 and 2. The pseudocode of the proposed parameter identification method is detailed in Algorithm 2, and the method is described as follows:

- **Initialization:** Particles representative of each parameter are initialized.

- **Step 1:** Sampling of the particles is done at each time step k using a standard particle sampling scheme. A particle set $\Xi = \{\theta_{ls}, w_{ls}\}_{ls \in \{1,2,\ldots,Ls\}}$ is generated for each parameter of each mode at each time step, based on:

$$\theta^i_{ls}(k) \sim p\left(\theta^i(k)|\theta^{i-1}_{lr}(k)\right). \tag{6}$$

- **Step 2:** If the mode output is not occluded, the particles weights are calculated using

$$w^i_{ls}(k) = p\left(y(k)|\theta^i_{ls}(k), u(k)\right), \tag{7}$$

otherwise all particles representing a parameter have equal weights.

Algorithm 2 Time-varying parameters identification

Θ^i = Ident_Params$\left[\Theta^{i-1}, u, y\right]$
Step 1: Sampling
for $k = 1 : K$ (time) **do**
  for $m = 1 : M$ (mode) **do**
    for $n = 1 : N$ (parameters) **do**
      - Draw $\left\{\theta^{m,i}_{n,ls}(k)\right\}^{Ls}_{ls=1} \sim \left\{p\left(\theta^{m,i}_n(k)|\theta^{m,i-1}_{n,lr}(k)\right)\right\}^{Lr}_{lr=1}$
    end for
  end for
end for
Step 2: Importance weighting
for k, m, ls (time, mode, sampled particles) **do**
  if $\mu(k) == m$ **then**
    - Calculate $w^{m,i}_{ls}(k) = p\left(y(k)|\theta^{m,i}_{ls}(k), u(k)\right)$
  else
    - $w^{m,i}_{ls}(k) = Ls^{-1}$
  end if
  - Normalize $w^{m,i}_{ls}(k)$ over ls
end for
Step 3: Smoothing
for k, m (time, mode) **do**
  if $\mu(k) == m$ **then**
    - Point estimate index: $\hat{l}^m(k) = \underset{ls}{\text{argmax}}\left(w^{m,i}_{ls}(k)\right)$
  else
    - Extrapolate: $\hat{l}^m(k)\alpha\left\{\hat{l}^m(\mu_{k^-} == m), \hat{l}^m(\mu_{k^+} == m)\right\}$
  end if
  - Estimate: $\hat{\theta}^{m,i}(k) = \theta^{m,i}_{\hat{l}^m(k)}(k)$
end for
for k, m, ls (time, mode, sampled particles) **do**
  - Smooth estimate: $\bar{\theta}^{m,i}_n(k) = \sum^K_{j=1}\left[\hat{\theta}^{m,i}_n(j) * g_n(j|k)\right]$
  for $ls = 1 : Ls$ (sampled particles) **do**
    - Smoothing weights: $s^{m,i}_{ls}(k) = p\left(\bar{\theta}^{m,i}(k)|\theta^{m,i}_{ls}(k)\right)$
  end for
  - Normalize $s^{m,i}_{ls}(k)$ over l
end for
for k, m (time, mode) **do**
  for $ls = 1 : Ls$ (sampled particle) **do**
    - Final weight: $ws^{m,i}_{ls}(k) = w^{m,i}_{ls}(k) * s^{m,i}_{ls}(k)$
  end for
  - Normalize $ws^{m,i}_{ls}(k)$ over ls
end for
Step 4: Resampling, using Algorithm 2 in [25]
for k, m (time, mode) **do**
  - $\left\{\theta^{m,i}_{lr}(k)\right\}^{Lr}_{lr=1}$ =RESAMPLE$\left[\left\{\theta^{m,i}_{ls}(k), ws^{m,i}_{ls}(k)\right\}^{Ls}_{ls=1}\right]$
end for

- **Step 3:** If the mode output is not occluded, the point estimate of $\theta^i_{ls}(k)$ is calculated for each parameter:

$$\hat{\theta}^i(k) = \underset{\theta^i_{ls}(k)\in\Xi}{\text{argmax}}\left(w^i_{ls}(k)\right). \tag{8}$$

Otherwise $\hat{\theta}^i(k)$ is extrapolated from adjacent point estimates at non-occluded time steps.
Then smooth parameters estimates are calculated from the point estimates and the moving average weight function g_n as follows:

$$\bar{\theta}^i(k) = \sum^K_{j=1}\left[\hat{\theta}^i(j) \cdot g_n(j|k)\right]. \tag{9}$$

Smoothing weights are associated to each particles, and calculated based on the pdf p, the smooth point estimates and the particles values:

$$s^i_{ls}(k) = p\left(\bar{\theta}^i(k)|\theta^i_{ls}(k)\right). \tag{10}$$

Finally, particles weights are recalculated based on the particles weights (7) and the smoothing weights (10):

$$ws^i_{ls}(k) = w^i_{ls}(k) * s^i_{ls}(k). \tag{11}$$

- **Step 4:** Particles are resampled using a standard particle resampling scheme, based on the recalculated weights (11):

$$\theta^i_{lr}(k) = \text{RESAMPLE}\left[\theta^i_{ls}(k), w^i_{ls}(k)\right], \tag{12}$$

and i is iterated:

$$i = i + 1. \tag{13}$$

The algorithm goes to Step 1 if it did not reach the end criteria (stability of the modeling error e, Eq. (4)).

Table 1 Identification scheme parameters and variables.

θ	One identified parameter
θ_l	One identified particle
ν	All parameters stacked in a vector
y	Observation
μ	Mode index vector
w	Particle weight
s	Smoothing weight
ws	Total weight

Table 2 Identification scheme indexes.

k	Discrete time step
i	Identification iteration step
m	Mode index number
n	Parameter number index
lr	Resampled particle number index
ls	Sampled particle number index

With this method, the modes of the hybrid dynamical system model are identified separately. Thus the parameters of each mode are different spaces and can take similar values.

In case of mode output occlusion, values of the parameters are extrapolated thanks to the integrated time-smoothing process. Extrapolation of the parameters could be improved by replacing the selected MA time-smoothing method with a more sophisticated method. The proposed parameter identification scheme has a few limitations. The first limitation is the number of modes of the hybrid model. A high number of modes induces frequent output occlusion, and thus results in low parameter identification accuracy. The second limitation is the number of identified parameters. A high number of identified implies strong time-smoothing as explained in Sect. 3.6, and thus it reduces the added value of this method over a constant parameter identification method. Finally, very large problems are to avoid due to the high number of required particles, involving long calculation durations. A typical problem solved with this method would be composed of up to 15 parameters distributed in 3 modes.

3.5 Tuning Parameters Details

The created model has two main tuning parameters: the particle filtering pdf p and the smoothing pdf g. If the pdf are normal distributions, the tuning parameters can be expressed as standard deviations. Be aware that normal distributions are not the best distributions for weighting due to the quick drop to zeros of the pdf envelope. It can be advised to use a weighting pdf p_{Step2} different from the sampling and resampling pdf p_{Step1} and p_{Step4}. For example, the weights w_{Step2} can be a negative power of the modeling error.

The particle filtering distribution p is used to sample, weight and resample. For filtering purpose, this pdf is usually associated to the signal noise. For parameter identification, this parameter is usually associated to the input signal noise, to the algorithm convergence speed and to the identifed parameters precision. In the case of the developed parameter identification scheme, the particle filtering pdf p is not used to provide any knowledge from time-prior conditional probability. Thus the pdf p is only related to the algorithm convergence speed and to the identifed parameters precision.

The smoothing distribution g proposed in the the developed parameter identification scheme is used to generate a filtered point estimate for each parameter at each discrete time step k. Thus g is the tuning parameter related to the input noise and to the parameters dynamics. Due to the creation process of the final weight ws, g is also related to the algorithm convergence speed. Strong smoothening will have a negative effect on the algorithm convergence speed.

3.6 Unicity of the Solution

In this section, unicity of the identification equation system is verified.

For a defined mode m, the identification problem can be formulated as

$$\begin{cases} y^m(1) &= h(u^m(1), \theta^m(1)) + e_{h(1)} \\ \quad\vdots & \\ y^m(k) &= h(u^m(k), \theta^m(k)) + e_{h(k)} \\ \quad\vdots & \\ y^m(K) &= h(u^m(K), \theta^m(K)) + e_{h(K)} \end{cases} \tag{14}$$

where $k \in \{1, 2, ..., K\}$ is the time step, y^m is the mode output vector, h is a model, u^m is the mode input vector, θ^m is the mode parameters vector and e is the modeling error.

The identification problem is formulated as

$$\begin{aligned} \theta &= \underset{\theta}{\operatorname{argmin}} \textstyle\sum_k \|e_{h(k)}\| \\ &= \underset{\theta}{\operatorname{argmin}} \textstyle\sum_k \|\, y(k) - h\,(r(k), \theta)\,\|. \end{aligned} \tag{15}$$

If the model parameters are considered as constant, $\forall k \in \{1, 2, ..., K\}, \theta(k) = \theta$, and theoretically, if $\|e_{h(k)}\| = 0$, as many system equations are the dimention of the parameter vector v are required to get a unique solution. Thus hybrid systems identification processes usually use large sets of data to get a good guess of the model parameters values. In the case of time-varying parameters identification, the assumption of constant parameter is not valid. Thus each equation of the system (14) is independent. The identification problems have $dim(\theta) - 1$ degree of freedom. A solution to avoid this issue is frequency decoupling. It can be considered that the identified parameters have very different natural frequencies, and thus lower frequencies parameters can be considered as constant. From this point of view, the problem is about solving a system with $dim(\theta) = 1$, and a unique solution exist. This frequency decoupling can be applied in Algorithm 2 through the g_n smoothing pdfs.

4. Selected Application Models

This section details the selected application models. The application goal of this research is to understand drivers variability and behavior modifications according to models parameters' evolution. Thus the selected application models must have been used for driving modelling and must have physically understandable parameters values.

Based on these considerations, PieceWise AutoRegressive eXogenous (PWARX) and Gipps microscopic trafficflow [14] models have been selected. The PWARX model is a linear hybrid system model, generalization of the classical ARX models. Various hybrid ARX models have been used over the years to describe driving behavior [2]–[4]. Physical meaning could be attributed to probability weighted ARX model parameters [4]. Gipps model is a discrete-in-time continuous-in-space collision avoidance type traffic flow model [11]–[14]. It has been developed for highway use with the aim to have physically comprehensive parameters. This model is non-homogeneous and nonlinear.

4.1 PWARX Model

The PWARX model can be formulated as

$$y(k)=f(r(k)),$$
$$r(k)=[y(k-1)\dots y(k-n_a)\ u(k)\dots u(k-n_b)]^\top,$$

$$f(r(k))=\begin{cases} \theta^{1\top}\begin{bmatrix} r(k) \\ 1 \end{bmatrix} & \text{if } r(k)\in X^1 \Leftrightarrow \mu(k)=1 \\ \vdots & \\ \theta^{M\top}\begin{bmatrix} r(k) \\ 1 \end{bmatrix} & \text{if } r(k)\in X^M \Leftrightarrow \mu(k)=M \end{cases} \tag{16}$$

where k represents the discrete time step, r is the regression vector, y is the model output, and u is the exogenous input. The model orders n_a and n_b and the mode number M are supposed to be known. $\theta_m \in \mathbb{R}^{n_{pw}+1}$ represents the parameter vector of the mode m, where $n_{pw} = n_a + n_b$. $\mu \in \{1, 2, ..., M\}^K$ is the mode index vector.

The data partitions X^m are assumed to be bounded convex polyhedra, described by

$$X^m = \{r \in \mathbb{R}^{n_{pw}} | H^m r \le h^m\} \tag{17}$$

where H^m and h^m are the real valued matrix and vector describing the mode partitioning. $X = \cup_{m=1}^{M} X^m$ is assumed to be a bounded convex polyhedron, and $\forall(i, j) \in \{1, 2, ..., M\}^2, i \neq j, H^i \cap H^j = \{0\}$.

4.2 Gipps Microscopic Traffic Flow Model

Gipps model is composed of two equations. An equation based on data fitting to reproduce acceleration behavior of the driver, and an equation based on the mathematical derivation of the required braking to maintain a safety distance with a delay characteristic and a delay margin.

Gipps model is expressed by

$$\begin{cases} v_\eta(k+\tau_{reac}) = v(a) \quad \text{if} \quad v(a) \le v(b) \Leftrightarrow \mu(k)=1 \\ v_\eta(k+\tau_{reac}) = v(b) \quad \text{if} \quad v(a) > v(b) \Leftrightarrow \mu(k)=2 \\ \\ v_a = v_\eta(k) + 2.5 a_\eta \tau_{reac}(1-\frac{v_\eta(k)}{v_{\eta 0}})\sqrt{0.025+\frac{v_\eta(k)}{v_{n0}}} \\ v_b = b_\eta \tau_{reac} + \Big(b_\eta^2 \tau_{reac}^2 + b_\eta\Big[2(-\Delta x(k) + s_{\eta-1}) \\ \qquad + v_\eta(k)\tau_{reac} + \dfrac{v_{\eta-1}(k)^2}{bm_{\eta-1}}\Big]\Big)^{1/2} \end{cases} \tag{18}$$

where v_η is the velocity for the vehicle number η (ego-vehicle), $v_{\eta-1}$ is the velocity of the vehicle $\eta-1$ (leading vehicle), k represents the discrete time step, τ_{reac} is the apparent driver reaction time steps, $v_{\eta 0}$ is the desired free flow velocity of the vehicle η, $s_{\eta-1}$ is the length + stopping distance of the vehicle $\eta-1$, a_η is the maximum acceleration of the vehicle η, $\Delta x = x_{\eta-1} - x_\eta$ is the relative distance between the vehicle $\eta-1$ and the vehicle η, b_η is the maximum desired braking acceleration of the vehicle η, $bm_{\eta-1}$ the estimation of the most severe braking of vehicle $\eta-1$.

The inputs of Gipps model are $v_\eta(k)$, $v_{\eta-1}(k)$ and Δx. The output of Gipps model is $v_\eta(k+\tau_{reac})$. The parameters of Gipps model for the vehicle η are a_η, $v_{\eta 0}$, τ_{reac}, b_η, $s_{\eta-1}$ and $bm_{\eta-1}$

5. Parameters Identification Examples

In this section, validation of the developed parameters identification scheme is proposed. To validate the identification procedure, PWARX and Gipps model are used. This allows to cover linear and nonlinear cases, and single and multiple parameters cases. The particle filtering pdf $p = p_{Step1} = p_{Step4}$ is Gaussian with a mean equal to zero and a standard deviation σ_p, weighting of the particles is done based on the inverse of the square root of the modeling error, and the MA smoothing pdf g is Gaussian with a mean equal to zero and a standard deviation σ_g. Parameter identification examples are done with at most two simultaneously identified time-varying parameters per mode, and two modes per model. As expalined in Sect. 3.6, the number of parameters should be kept low in order to be able to identify high frequency time-variations. The number of modes can be higher, but a high number of modes implies frequent mode output occlusion, and thus reduces the parameters identification precision.

5.1 One Identified Parameter Case, PWARX Model

The test input, output and parameters are generated by the following system:

$$f(x) = \begin{cases} \theta^1(k)^\top \begin{bmatrix} x(k) \\ 1 \end{bmatrix} & \text{if } \mu(k)=1 \\ \theta^2(k)^\top \begin{bmatrix} x(k) \\ 1 \end{bmatrix} & \text{if } \mu(k)=2 \end{cases}$$
$$\begin{cases} \theta^1(k)^\top = \Big[0.5 + \frac{sin(k*12)}{4} \quad 0.5\Big] \\ \theta^2(k)^\top = \Big[-1 + \frac{sin(k*6)}{6} \quad 2\Big] \end{cases} \tag{19}$$
$$x(k) = u(k) = [-2.5\ -2.49\ \dots\ 2.5]$$

where $k \in \{1, 2, ..., K\}$ is the discrete time step, and μ is the mode index vector.

Only the first parameter of each mode is studied. This parameter identification example uses 10 particles per parameter, with a sampling coefficient of 10, $\sigma_p = 0.05$, and $\sigma_g = 2$. Initialization is done with random particles in the $[-2.5; 2.5]$ range. The results of Fig. 4 are obtained in 5 identification iterations. As shown in Fig. 4, the developed parameter identification method can filter out noise in parameter values while preserve the parameters dynamics. It can be observed that the parameters are correctly identified. The identification time was 62 seconds on an Intel i5@3GHz

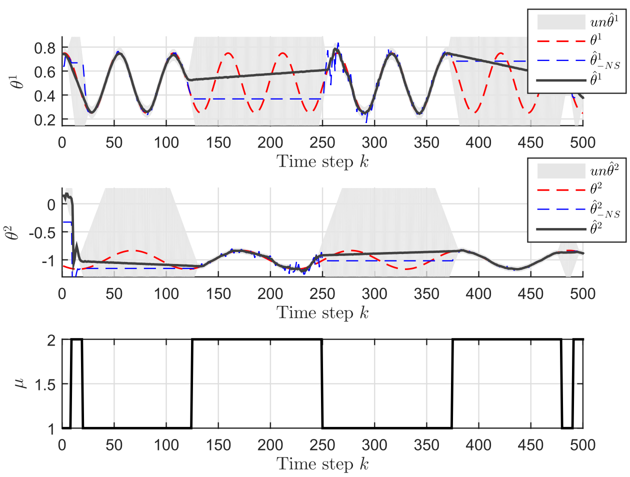

Fig. 4 Parameter identification of a two modes one time-varying parameter per mode PWARX model.
$un\hat{\theta}^m$ represents the uncertainty of the parameter estimation, based on σ_p and on mode output occlusion, θ^m the true parameters values, $\hat{\theta}^m_{-NS}$ the non-smoothened parameters estimate values and $\hat{\theta}^m$ the final parameters estimates.

computer with 8GB or RAM.

5.2 Multiple Identified Parameters Case, PWARX Model

In this section, results from the PWARX model with multiple simultaneous time-varying identification are shown. The problem definition is the same than in (19), with the following time-varying parameters

$$\begin{cases} \theta^1(k)^\top = \left[1 + \frac{sin(k*12)}{4} \quad 0.5 + \frac{sin(k*1.2)}{8}\right] \\ \theta^2(k)^\top = \left[-1 + \frac{sin(k*6)}{6} \quad 2\right]. \end{cases} \tag{20}$$

This parameter identification example uses 10 particles per parameter, with a sampling coefficient of 10, $\sigma_p = 0.05$, $\sigma^1_g = 2$ and $\sigma^2_g = 20$. Initialization is done with random particles as shown in Fig. 5. 10 iterations i are done. As shown in Fig. 6, simultaneous identification of multiple time-varying parameters could be realized. When parameters cannot be identified, values of the parameters are extrapolated. Identification error can be observed, as the number of particles is kept low to enable fast identification. Nevertheless, values of time-varying parameters are closer to ideal than a constant parameter. In Fig. 7, $e^m = \sum_{k[\mu(k)=m]} \|f^m(\theta(k)) - f^m(\hat{\theta}(k))\|_1$ represents the total modeling error per mode m. When identification error stabilizes, the identification process convergence is assumed. The modeling errors do not converge to zero due to the filtering and to the extrapolation processes. The identification time was 118 seconds on an Intel i5@3GHz computer with 8GB or RAM. This computation speed opens the method to online parameters identification.

5.3 One Parameter Case, Nonlinear Model

In this section, the efficiency of the model is assessed with

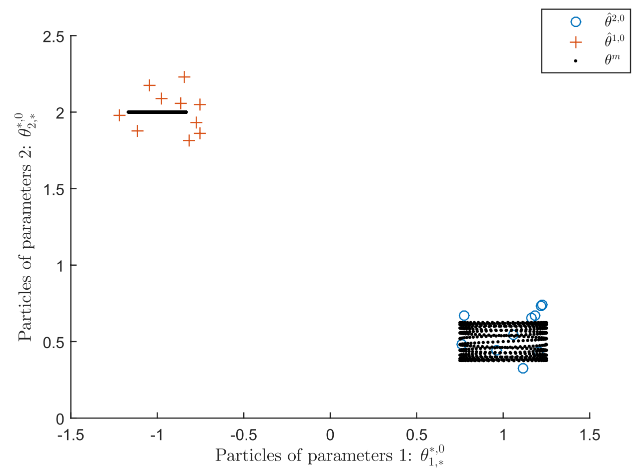

Fig. 5 Initial particles for two modes two parameters per mode PWARX model. In orange crosses the initial particules for $m = 1$, in blue circles the initial particles for $m = 2$ and in black dots the true parameters values.

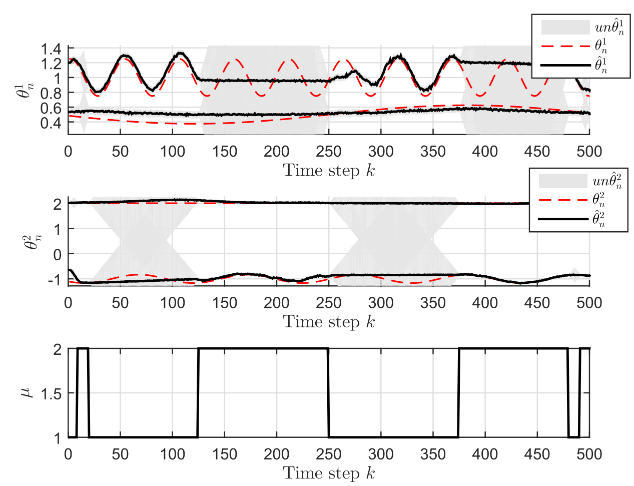

Fig. 6 Parameter identification of a two modes two parameters per mode PWARX model.
$un\hat{\theta}$ represents the uncertainty of the parameter estimation, based on σ_p and on mode output occlusion. θ^m_n are the true parameter values of parameter number n of mode m and $\hat{\theta}^m_n$ are the final parameter estimates.

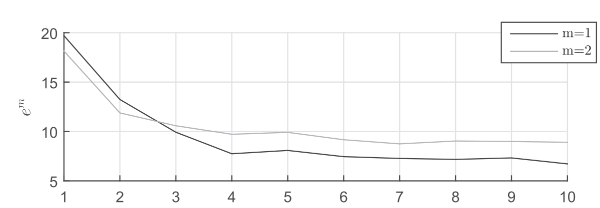

Fig. 7 Parameter identification error of a two modes two identified parameters per mode PWARX model.

the heterogeneous nonlinear car-following Gipps traffic flow model, expressed in Eq. (18).

The identified parameter is b of the braking equation v_b, corresponding to the mode $m = 2$. Gipps model first runs with a known time-varying parameter $b(k)$ to generate a known output. Other Gipps model parameters are known

Table 3 Gipps model parameters values.

Variable	v_0	s_0	τ_{reac}	a	b	bm
Value	20	6.5	0.3	1.7	$\theta(k)$	-3.2

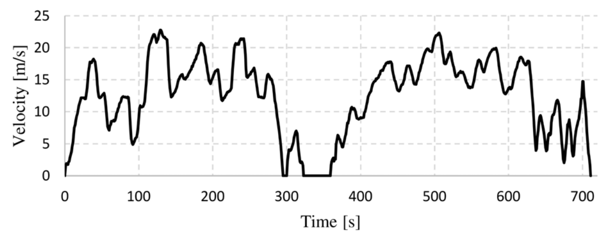

Fig. 8 Leading vehicle velocity data used in the generation of the example output data and for the time-variying parameter identification.

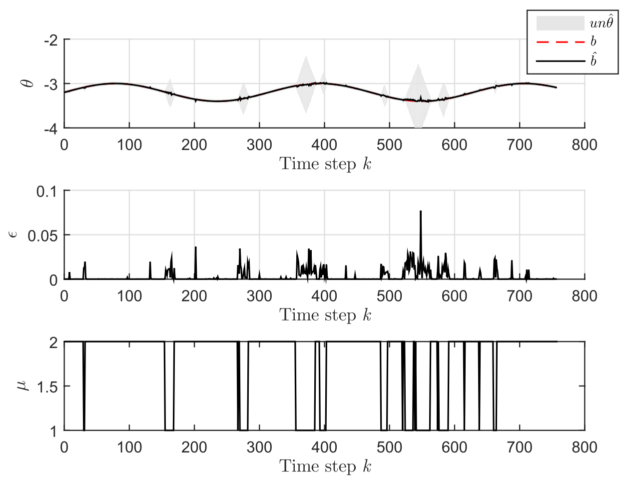

Fig. 9 Low frequency parameter time-variation case.
$un\hat{\theta}$ represents the uncertainty of the parameter estimation, based on σ_p. b is the true parameter and $\hat{b}$ is the final parameter estimate.

and constant (see Table 3). The leading vehicle velocity and relative distance used for this example are extracted from a real-world measurement. The velocity of the leading vehicle is showed in Fig. 8.

Figure 9 and Fig. 10 show the identification of the time-varying parameter $b(k)$, with different time-variation frequencies. As described in Eq. (18), the parameter b is in the braking mode, enabling parameter identification when $\mu(k) = 2$.

As shown in Fig. 9 and Fig. 10, it can be observed that the parameter b is correctly identified. $\epsilon(k) = \|\theta(k) - \hat{\theta}(k)\|_1$ represents the time-varying parameter estimation error. This parameter identification error is low as long as $\mu(k) = 2$. Thus the proposed parameter identification method can be used to interpret driving behavior through model time-varying parameters. The identification time was 26 seconds on an Intel i5@3GHz computer with 8GB or RAM. This speed of execution opens this method to online parameter identification.

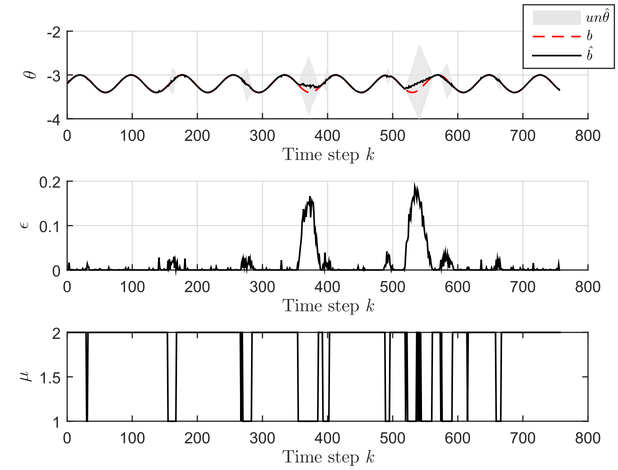

Fig. 10 High frequency parameter time-variation case.
$un\hat{\theta}$ represents the uncertainty of the parameter estimation, based on σ_p. b is the true parameter and $\hat{b}$ is the final parameter estimate.

6. Driving Behavior Applications Discussion

In this section, a discussion about the application for driver behavior analysis and modeling is proposed.

A large variety of driver modeling methods have been researched in the past years. Initial studies focused on the human psychophisical reactions [10], then an important highlight has been done on the creation of microscopic traffic flow models from data analysis [11]–[14], and more recently a focus is done on the usage of machine learning methods [1]–[6]. While most of these models can have parameters attributed a physical meaning, understanding of the driver behavior from the point of view of the parameters is not common practice. Different aspects of the driving behavior could be investigated from the analysis of time-varying parameters: the driver consistency expressed by high frequency parameters variations or a statistical parameter variance, and the driver behavior modifications expressed by low frequency parameters variations or global parameters changes [31]. Online analysis of the parameters could also be considered for indirect sensing of the human state during the driving operation [4].

Once analysis of drivers is done, we believe that modeling of the human stochasticity could be done. This new class of modeling techniques could be implemented in future driver models for more realistic traffic flow simulations, for naturalistic traffic vehicles behavior prediction in autonomous cars, or for advanced driver-personalized driving assistance systems.

7. Conclusion

In this paper, an iterative metaheuristic method based on particle filtering and moving average time-smoothing has been created to identify time-varying nonlinear hybrid system model parameters. The proposed method enables us to identify the parameters of nonlinear non-heterogeneous hybrid dynamical system models, while filtering the parameters time-variation based on the possible parameters dynamics. This method has a low computation burden and can be implemented for online use. The effectiveness of this method is verified by numerical experiments, including linear and nonlinear hybrid dynamical system models, represented by PWARX and Gipps car-following driver model. Finally, the application of time-varying parameters identification to driving behavior analysis and modeling is proposed.

References

[1] L. Ljung, System Identification: Theory for the User, Prentice-Hall, May 2012.

[2] S. Sekizawa, S. Inagaki, T. Suzuki, S. Hayakawa, N. Tsuchida, T. Tsuda, and H. Fujinami, "Modeling and recognition of driving behavior based on stochastic switched ARX model," IEEE Trans. Intell. Transp. Syst., vol.8, no.4, pp.593–606, Dec. 2007.

[3] R. Terada, H. Okuda, T. Suzuki, K. Isaji, and N. Tsuru, "Multi-scale driving behavior modeling using hierarchical PWARX model," IEEE 13th Annual Conf. on Intel. Trans. Systems, pp.1638–1644, Sept. 2010.

[4] H. Okuda, N. Ikami, T. Suzuki, Y. Tazaki, and K. Takeda, "Modeling and analysis of driving behavior based on a probability-weighted ARX model," IEEE Trans. Intell. Transp. Syst., vol.14, no.1, pp.98–112, March 2013.

[5] P. Angkititrakul, C. Miyajima, and K. Takeda, "Modeling and adaptation of stochastic driver-behavior model with application to car following," IEEE Intelligent Vhl. Symposium (IV), pp.814–819, 2011.

[6] G.S. Aoude, V.R. Desaraju, L.H. Stephens, and J.P. How, "Driver behavior classification at intersections and validation on large naturalistic data set," IEEE Trans. Intell. Transp. Syst., vol.13, no.2, pp.724–736, June 2012.

[7] P.J. Antsaklis, "Special issue on hybrid systems: Theory and applications a brief introduction to the theory and applications of hybrid systems," Proc. IEEE, vol.88, no.7, pp.879–887, Aug. 2002

[8] T. Wilhelem, H. Okuda, B. Levedahl, and T. Suzuki, "Energy consumption evaluation based on personalized driver-vehicle model," IEEE Trans. Intell. Transp. Syst., vol.18, no.6, pp.1468–1477, Oct. 2016.

[9] R. Goebel, R.G. Sanfelice, and A.R. Teel, Hybrid Dynamical Systems, Princeton University Press, 2012.

[10] C.C. MacADAM, "Understanding and modeling the human driver," Swets & Zeitlinger, Vehicle System Dyn., vol.40, no.1-3, pp.101–134, 2003.

[11] D. Ni, Traffic Flow Theory, Butterworth-Heinemann, 2016.

[12] M. Brackstone and M. McDonald, "Car-following: A historical review," Transportation Research Part F, vol.2, no.4, pp.181–196, 1999.

[13] S. Panwai and H. Dai, "Comparative evaluation of microscopic car-following behavior," IEEE Trans. Intell. Transp. Syst., vol.6, no.3, pp.314–325, Sept. 2005.

[14] P.G. Gipps, "A behavioural car-following model for computer simulation," Trans. Res. Part B, vol.15, no.2, pp.105–111, 1981.

[15] A.Lj. Juloski, W.P.M.H. Heemels, G. Ferari-Tecate, R. Vidal, S. Paoletti, and J.H.G. Niessen, "Comparison of four procedures for the identification of hybrid systems," Hybrid Systems: Computation and Control, pp.354–369, 2005.

[16] G. Ferrari-Trecate, M. Musellu, D. Liberati, and M. Morari, "A clustering technique for the identification of piecewise affine systems," Automatica, vol.39, no.2, pp.205–217, 2009.

[17] A. Bemporad, A. Garulli, S. Paoletti, and A. Vicino, "A bounded-error approach to piecewise affine system identification," IEEE Trans. Autom. Control, vol.50, no.10, pp.1567–1580, Oct. 2005.

[18] J. Roll, A. Bemporad, and L. Ljung, "Identification of piecewise affine systems via mixed-integer programming," Automatica, vol.40, no.1, pp.37–50, 2004.

[19] Y. Ma and R. Vidal, "Identification of deterministic switched ARX systems via identification of algebraic varieties," Hybrid Systems: Computation and Control, vol.3431, pp.449–465, 2005.

[20] A.Lj. Juloski, S. Weiland, and W.P.M.H. Heemels, "A Bayesian approach to identification of hybrid systems," IEEE Trans. Autom. Control, vol.50, no.10, pp.1520–1533, Oct. 2005.

[21] G. Chowdhary and R. Jategaonkar, "Aerodynamic parameter estimation from flight data applying extended and unscented Kalman filter," Aerosp. Sci. Technol., vol.14, no.2, pp.106–117, 2010.

[22] C. Andrieu, A. Doucet, S.S. Singh, and V.B. Tadic, "Particle method for change detection, system identification, and control," Proc. IEEE, vol.92, no.3, pp.423–438, 2004.

[23] A. Hartmann, S. Vinga, and J.M. Lemos, "Hybrid identification of time-varying parameter with particle filtering and expectation maximization," IEEE 21st Mediterranean Conf. on Control & Automation, pp.884–889, 2013.

[24] S. Paoletti, A.Lj. Juloski, G. Ferrari-Trecate, and R. Vidal, "Identification of hybrid systems: A tutorial," Eur. J. Control, vol.13, no.2-3, pp.242–260, 2007.

[25] M.S. Arulampalam, S. Maskell, N. Gordon, and T. Clapp, "A tutorial on particle filters for online nonlinear/non-Gaussian Bayesian tracking," IEEE Trans. Signal Process., vol.50, no.2, pp.174–188, Feb. 2002.

[26] P.M. Djuric, J.H. Kotecha, J. Zhang, Y. Huang, T. Ghirmai, M.F. Bugallo, and J. Miguez, "Particle filtering: A review of the theory and how it can be used for solving problems in wireless communication," IEEE Signal Process. Mag., vol.20, no.5, pp.19–38, Oct. 2003.

[27] C. Andrieu, N. de Freitas, A. Doucet, and M.I. Jordan, "An introduction to MCMC for machine learning," Mach. Learn., vol.50, no.1, pp.5–43, Jan. 2003

[28] A. Doucet and A.M. Johansen, "A tutorial on particle filtering and smoothing: Fifteen years later," Handbook of Nonlinear Filtering, vol.12, no.3, pp.656–704, Dec. 2008.

[29] S. Thrun, W. Burgard, and D. Fox, Probability Robotics, The MIT Press, 2006.

[30] C.J. Nash, D.J. Cole, and R.S. Bigler, "A review of human sensory dynamics for application to models of driver steering and speed control," Biol. Cybern., vol.110, no.2, pp.91–116, June 2016.

[31] T. Wilhelem, H. Okuda, A. Kawashima, and T. Suzuki, "Identification of time-varying parameters in Gipps model for driving behavior analysis," IEEE Int. Conf. on Systems, Man, and Cybernetics, pp.2984–2989, Oct. 2016.

Thomas Wilhelem was born in Paris, France, in 1990. He received a Master's degree in Solid Mechanics and Automobile and Railway transportation from ENSTA ParisTech, Palaiseau, France. He is currently pursuing the Ph.D degree with the department of Mechanical Science and Engineering of Nagoya University, Nagoya, Japan. His research interests include hybrid dynamical systems, and the modeling and analysis of human driving behavior, focused on the ability to reproduce driver personalized energy consumption values.

Hiroyuki Okuda was born in JAPAN, in 1982. He received the B.E. and M.E. degrees in Advanced Science and Technology from Toyota Technological Institute, JAPAN in 2005 and 2007, respectively, and he received the Ph.D degree in Mechanical Science and Engineering from Nagoya University, JAPAN in 2010. From 2012 to 2016, he was an assistant professor of the Green Mobility Collaborative Research Center in Nagoya University. Currently, he is an assistant professor of the Department of Mechanical Science and Engineering of Nagoya University. His research interests are in the areas of system identification of hybrid dynamical system and its application to modeling of human behavior, design of human centered mechatronics and biological signal processing. Dr. Okuda is a member of the IEEJ, SICE and JSME.

Tatsuya Suzuki was born in Aichi, JAPAN, in 1964. He recieved the B.S., M.S. and Ph.D degrees in Electronic Mechanical Engineering from Nagoya University, JAPAN in 1986, 1988 and 1991, respectively. From 1998 to 1999, he was a visiting researcher of the Mechanical Engineering Department of U.C.Berkeley. Currently, he is a Professor of the Department of Mechanical Science and Engineering of Nagoya University, Vice Director of Green Mobility Collaborative Research Center in Nagoya University and Principal Investigator in JST-CREST. He won the best paper award in International Conference on Advanced Mechatronic Systems 2013 and the outstanding paper award in International Conference on Control Automation and Systems 2008. He also won the journal paper award from IEEJ, SICE and JSAE in 1995, 2009 and 2010, respectively. His current research interests are in the areas of analysis and design of hybrid dynamical systems and discrete event systems particularly focusing on the human-centric mobility systems and energy management systems. Dr. Suzuki is a member of the SICE, ISCIE, IEICE, JSAE, RSJ, JSME, IEEJ and IEEE.