Identification of time-varying parameters in Gipps model for driving behavior analysis

Thomas Wilhelem, Hiroyuki Okuda, Akihiko Kawashima, Tatsuya Suzuki, *Member*, *IEEE*
Nagoya University, Dept. of Mechanical Science and Engineering
wilhelem@nuem.nagoya-u.ac.jp

***Abstract*—This paper proposes a new method to analyze driver behavior. Analysis of the behavior is done through the observation of the time-evolution of parameters of simple driver models. The behavior analysis is decomposed in two steps. First the driver model have to be selected or designed to represent the average behavior of a large sample of drivers. Then personal driver's behavior evolution can be analyzed over the time. To be able to identify time-varying non-linear hybrid model parameters, an iterative metaheuristic method based on particle optimization and moving average filtering has been created. This method enables to identify parameters of any model type while filtering the parameter time-variation based on the possible parameter dynamics. This methods also enables to interpolate parameters values while model output values are occluded. Demonstration of the identification algorithm efficiency with Gipps car-following driver model is done based on theoretical examples, and time-evolution of parameter are identified from real-world measured data.**

Keywords—driving behavior; parameter analysis; parameter identification; particle optimization; Gipps model

I. Introduction

Modeling and analysis of driving behavior are fundamental issues to realize safe and reliable driving assistance systems, traffic flow models, and ITS systems. Driving behavior can be analyzed from large datasets statistical point of view [1], or by model-based parameters interpretation [2]-[3]. Nevertheless, usual methods do not allow to understand time-varying parameters evolution. This last point is investigated in this paper.

Driver models have been approached from various fields and methodologies. The most traditional approach to the driver model is in cognitive science field. This approach is based on the observation of human cognitive actions with a psychophysical understanding of human behavior [4]. The other traditional model-based approach is by data fitting on basic mathematical models [1]. Recently, more elaborate and complex machine learning methods have been used to accurately model and analyze human behavior [5]-[8]. However, these sophisticated models lead to difficulties to understand driver's behavior. To be able to understand driver behavior, we believe that the driver model should be kept simple enough. Then while a simple driver model can express the driving behavior in a situation, it can provides insufficient accuracy to express the adaptive behavior of the human driver. Thus the novel approach of this paper is to use simple driver models to analyze the driving behavior through the identified time-varying parameters.

Most car-following models are non-linear hybrid systems [9] with heterogeneous types of modes equations. This is due to the fact that drivers are generally not modelled as a linear controller [4]. To tackle the parameter identification problem, a metaheuristic algorithm based on the particle filter method and on differential evolution has been developed [10]-[11]. Due to heterogeneity of the models equations, each mode equation's parameters set is identified independently. This approach leads to regressor vector partial occlusion when the identified mode is not providing the model output. A filtering method is integrated to the identification to lower the amount of identified noise and to be able to extrapolate parameter identification in data occlusion cases.

Among the variety of available driver models [12], the microscopic traffic flow Gipps car-following model has been selected [13]-[14]. This model is widely used to model vehicles behaviors is traffic flow field. Gipps model has been designed so that its parameters are representative of drivers characteristics. Thus time-varying Gipps model parameters can be investigated to analyze drivers behavior evolution. Moreover, understanding the parameters evolution can enable improvement of the models behavior with few added complexity.

In the second section, driver model selection and functioning is explained. In the third section, the parameter identification method is detailed. In the fourth section, demonstration of the identification process efficiency is shown, and in the fifth section application to measurement data is done. Finally conclusion is drawn in the last section.

II. Driver modeling

This section details the driver model's choice for the analysis of drivers' behavior. Then functioning of the selected model is explained.

The goal of this research is to understand user variability and behavior modifications according to models parameter evolution. Thus the candidate model for this study should have been created with the intention to have physically understandable parameters values. Moreover, a widely used model with possibility to run traffic flow simulations would be beneficial to rely on years of solid bibliography and to be able to expend the field on the output of the research.

Based on these considerations, the Gipps microscopic traffic-flow model has been selected [13]. This model is a

978-1-5090-1897-0/16/$31.00

discrete-in-time continuous-in-space collision avoidance type traffic flow model [12]. It has been developed for highway average congestion levels situations. It 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 can be expressed as follow:

$$v_k(t+t_{reac}) = min\left\{\begin{matrix} v_k(t) + 2.5a_n t_{reac}\left(1-\frac{v_k(t)}{v_{k0}}\right)\sqrt{0.025+\frac{v_k(t)}{v_{k0}}} \\ b_k t_{reac} + \sqrt{b_k^2 t_{reac}^2 + b_k\left[2(-\Delta x(t)+s_{k-1}) + v_k(t)t_{reac} + \frac{v_{k-1}(t)^2}{\widehat{b_{k-1}}}\right]} \end{matrix}\right\} \quad (1)$$

with:

- v_k the ego-vehicle velocity, v_{k-1} the leading vehicle velocity
- t_{reac} the apparent driver reaction time
- v_{k0} the desired free flow velocity of the vehicle n
- s_{k0} the length + stopping distance of the ego-vehicle
- a_k the maximum acceleration of the ego-vehicle
- b_k the maximum desired braking acceleration of the ego-vehicle
- $\widehat{b_{k-1}}$ the estimation of the most severe leading vehicle braking.

III. Parameter identification method

This section explains step-by-step the design of the developed time-varying parameter identification method.

The identification process has been designed to be applied to any type of driver model, including nonlinear non-differentiable and time varying problems. Thus an iterative differential evolution method has been selected. Due to the large number of parameters over time, and the desired to be able to handle filtering over time, the differential evolution method has been implemented in a stochastic framework using a SIR (Sample, Importance weighting, Resampling) particle filtering method. To do so, the identified parameter is considered as the state variable. This approach enables fast optimization speed and easy parametrization of the identification algorithm, but does not insure an optimal result. To avoid identification of the measurement noise, a moving average (MA) filtering step is included in the identification process. Multiple parameters can be identified simultaneously as the particle filtering method allows it. Nevertheless, quality of the algorithm convergence would have to be carefully monitored and identification computation burden can become heavy [15].

In the case of heterogeneous equations hybrid models, parameters are only represented in certain modes. Thus when the mode is not used to calculate the model output, the regressor vector cannot be created to identify parameters. This situation is called output data occlusion. The filtering step added to the SIR particle optimization enables to continue parameter identification when the output data is occluded. In the implemented case no prior knowledge is assumed on the parameter dynamics, thus simple linear extrapolation is used.

In the parameter identification process, u_n represents the input data and v_n represents the output data, and $n \in [1, N], N \in \mathbb{Z}$ the time step. The number of resampled particles at each time step is $S \in \mathbb{Z}$ and the multiplicative sampling factor is $Sf \in \mathbb{Z}$.

The required prior knowledge for the identification process are identified parameters upper and lower boundaries, the identification speed $\sigma_{speed} \in \mathbb{R}^{+*}$, the filtering parameters $\sigma_{filterT} \in \mathbb{R}^{+*}$ representing the moving average time width, and $\sigma_{filterN} \in \mathbb{R}^{+*}$ is the standard deviation of the identified parameter time-evolution. The current method does not allow to estimate automatically $\sigma_{filterN}$. θ is the identified parameter. The identification process is the following:

Initial particles are randomly spread over the parameter interval at each time step.

Then, as long as the total residual derivative is too high:

1- **Sampling**: sampling of new parameters particles based on the resampled particles

$$sample\ \theta_n^i(s_2) \sim p\left(\widehat{\theta_n} \middle| \theta_n^{i-1}(s_1), \sigma_{speed}\right)$$

$\widehat{\theta_n}$ represents the correct value of θ_n

i: iteration step

$s_1 \in [1, S]$: resampled particle index

$s_2 \in [1, S * Sf]$: sampled particle index

2- **Importance weighting 1**: least error weighting for each particle.

If $occlusion_{data} = 0$, $w_n^i(s) = \frac{1}{\|f(\theta_n^i(s), u_n) - v_n\|_1 + 1}$

Otherwise, $w_n^i(s) = \frac{1}{(S*Sf)}$

$w_n^i(s)$ represents the weight of particle s at step time n at iteration i

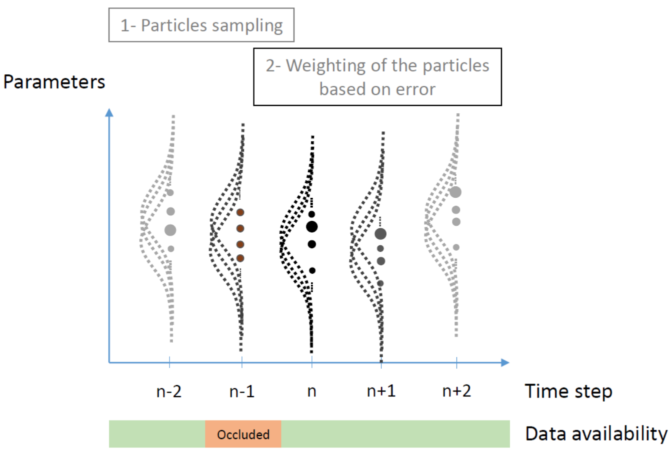

Figure 1: Parameter identification process steps 1 and 2

3- **Importance weighting 2**: filtering process.

a. Parameter estimate at each time step based on 2-

If $occlusion_{data} = 0$, $filt\theta_n^i = \theta_n^i\left(\underset{s}{argmax}\left(w_n^i(s)\right)\right)$

Otherwise, $filt\theta_n^i = extrapolation(filt\theta_{n-}^i, filt\theta_{n+}^i)$

$n-$ and $n+$ represent the previous and following position of parameter estimate with non-occluded regressor vector.

In this example, linear extrapolation is used to estimate missing occluded data cases. If the model time-varying dynamics have a model it can be used to estimate it more accurately.

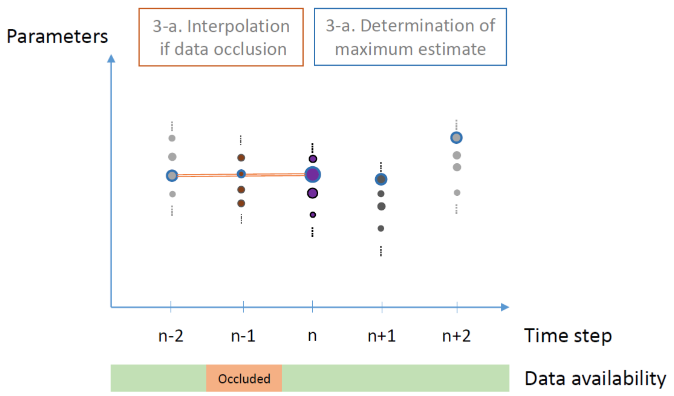

Figure 2: Parameter identification process step 3-a.

b. Creation of moving average filtering weights

$$wT_n^i(n) = N(n, \sigma_{filterT})$$

wT_n^i represents the set of weights over the time steps centered on the time step n at iteration i.

Here Gaussian distribution is used, but the support width is limited. Laplace distribution would actually be recommended.

c. Creation of the filter on the particles

Moving average on parameter estimate parameter for each time step n:

$$avgFilt\theta_n^i = \frac{\sum_j filt\theta_j^i * wT_n^i(j)}{\sum_j wT_n^i(j)}$$

Creation of the dynamics probability distribution:

$$wN_n^i(s) = N(avgFilt\theta_n^i, \sigma_{filterN})$$

d. Adjust the weights from 2-

$$w2_n^i(s) = w_n^i(s) * wN_n^i(s)$$

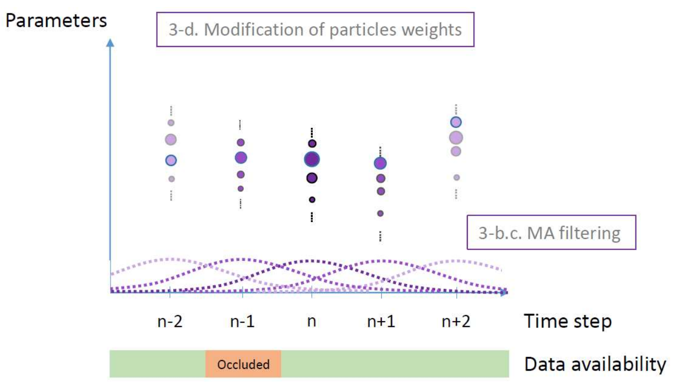

Figure 3: Parameter identification process step 3-b.c.d.

4- **Resampling**: keep best fitted based on weight

$draw\ S\ particles\ \theta_n^i(s_1)\ with\ probability\ \alpha\ w2_n^i$

$s_1 \in [1, S]$: resampled particle index

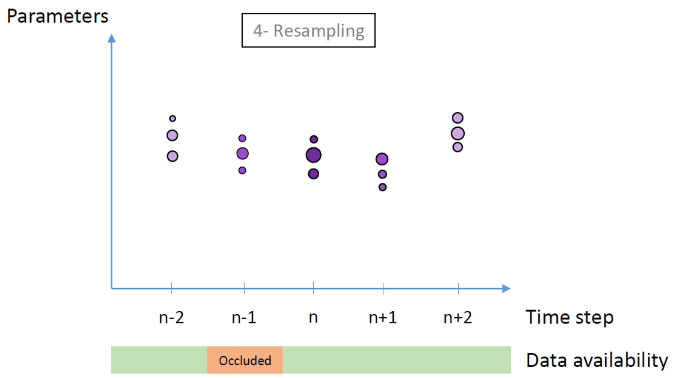

Figure 4: Parameter identification process step 4

IV. Verification by Test Data

In this section, efficiency of the developed parameters identification method is investigated on a single parameter identification. To do so, Gipps model first runs with known (ideal) b time-varying parameters to generate a known output. The lead vehicle used for this demonstration has been recorded from a real vehicle dynamics (see Figure 10). Other Gipps model parameters are constant (see Table 1) and have been identified a-priori. The time-varying parameter is identified using the scheme described in the previous section. Ideal parameter variation is done at different frequencies. Examples where noise is added to the model output signal are also demonstrated. One step of time is 0.6 second. The "mode" represents the output data occlusion. If "mode=1", then the output data can be used to identify the time-varying parameter. The optimization error is based on the norm one difference between the model output velocity and the previously generated velocity. Following examples optimization duration is 32s on an Intel i7 870 personal computer.

Table 1: Gipps model fixed parameter values.

Variable	v_0	s_0	t_{reac}	a	b	$\hat{b}$
Value	20	6.5	0.3	1.7	θ_n	-3.2

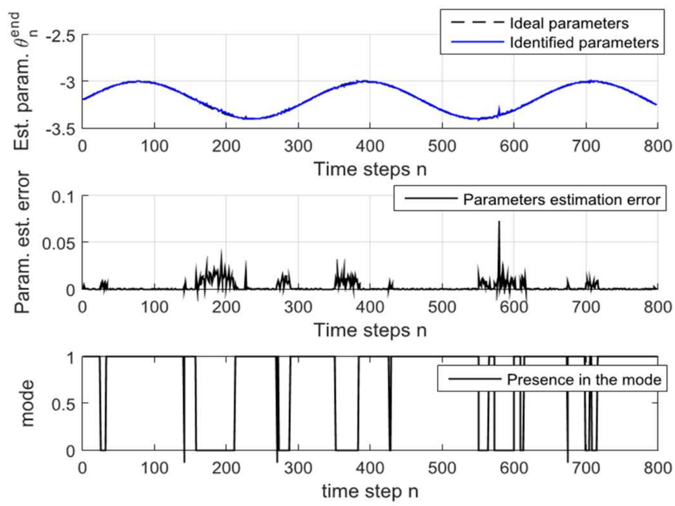

Figure 5: Low frequency parameter evolution case. No noise added on the

model output.

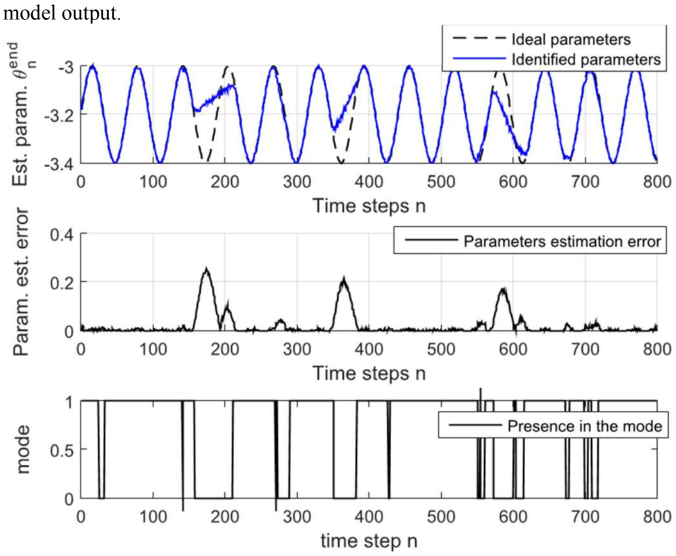

Figure 6: High frequency parameter evolution case. No noise added to the model output.

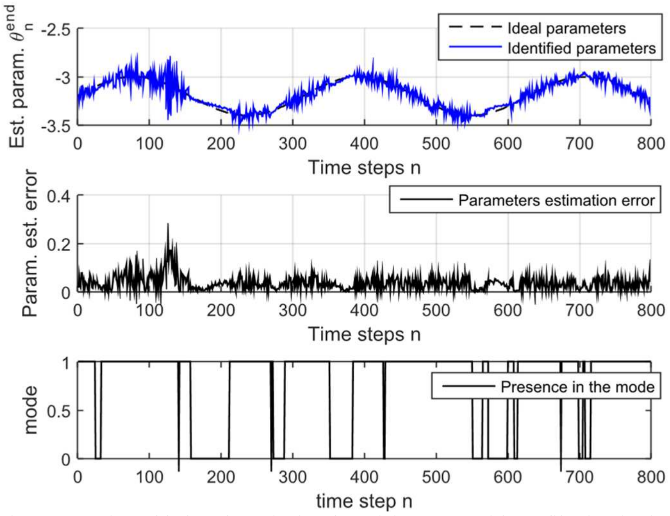

Figure 7: Noise added to the calculated model output. Without filtering in the identification process.

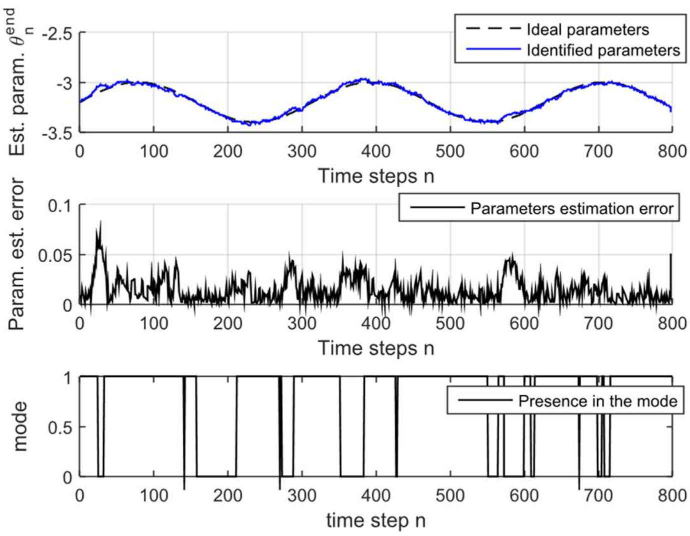

Figure 8: Noise added to the calculated model output. With filtering in the identification process.

As we can observe in Figure 5 and Figure 6, the time-varying parameter is correctly identified and output data occlusion is correctly handled by the identification process. Figure 7 and Figure 8 show that the filtering process enables to avoid most of the noise on the identified parameter estimate. Thus the proposed parameter identification method can be used to interpret model parameters time-varying behavior.

V. Application to Measurement Data

In order to observe realistic driver parameter evolution, real world experiment was lead. Two different examinees followed a lead vehicle on a highway. Each driver repeated the experiment twice, once with a soft driving manner, once with an aggressive driving manner. The notions of aggressiveness was let free to the understanding of the driver. The leading vehicle was equipped with a GPS based reference velocity profile display system which showed a predefined velocity pattern on a smartphone (see Figure 9). The ego-vehicle was equipped with a CAN bus acquisition tool, and a millimeter-wave radar. The CAN bus acquisition tool was used to record the GPS position, the velocity, and acceleration at the wheel of the vehicle. The millimeter-wave radar was used to get precise information on the distance to the leading vehicle, and to calculate the relative velocity.

Figure 9: View from the driver of the leading vehicle. The velocity profile display system is squared and zoomed on the right. Current velocity, future velocity and velocity plot were displayed.

A velocity profile reference shown in Figure 10 was created. This velocity profile included a wide range of accelerations and decelerations in order to cover most of the possible driving situations. Low velocity under 5 m/s was excluded from the identification data since the Gipps driver model, used to identify parameter variation is not suited for this kind of situation.

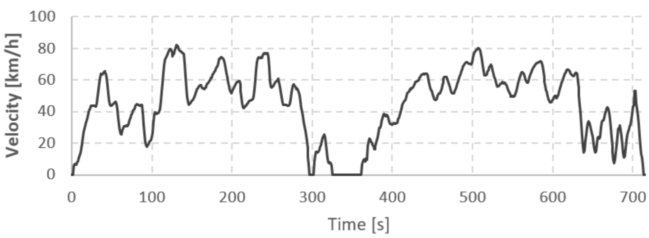

Figure 10: Reference velocity profile of the leading vehicle used for real world experiment.

In this case, where the parameter is identified from measurement data, direct verification of the accuracy of the identified parameters can be done. Thus the identification algorithm convergence is assumed based on the residual derivative value. In this section, time-varying identified parameter values are compared with classic constant parameter identification values. To confirm correct parameter regression, global identification residual with constant and time-varying parameters are compared (see Table 2 and Figure 11). The residual error is calculated as follow:

$$\mathrm{res}_{\mathrm{error}} = \left\| f(\theta_n^{end}, u_n) - v_n \right\|_1 \qquad (2)$$

with u the input data and v the output data, θ is the identified parameter, and n the time step.

Table 2: Residual error with constant parameter simulation and time-varying parameter simulation.

	Constant param.	Variable param.
A soft	519	384
A aggressive	598	511
B soft	339	239
B aggressive	337	262

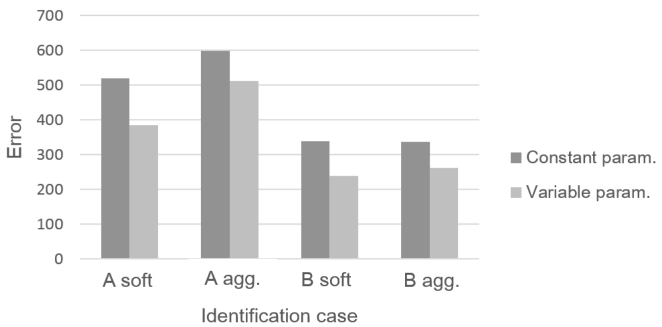

Figure 11: Residual error comparison histogram for two parameter identification methods and four driving patterns.

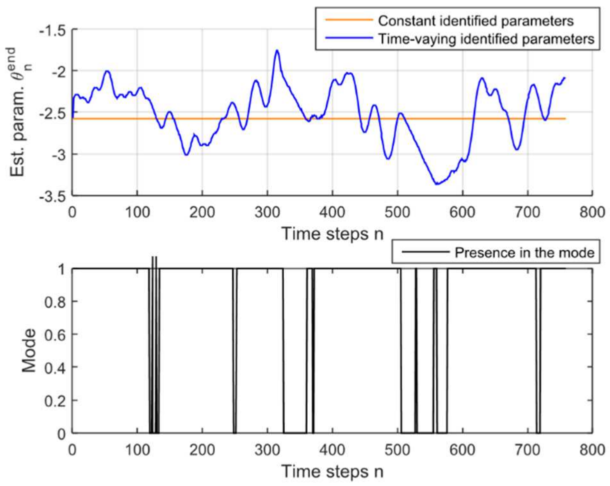

Figure 12: Driver A, soft driving case

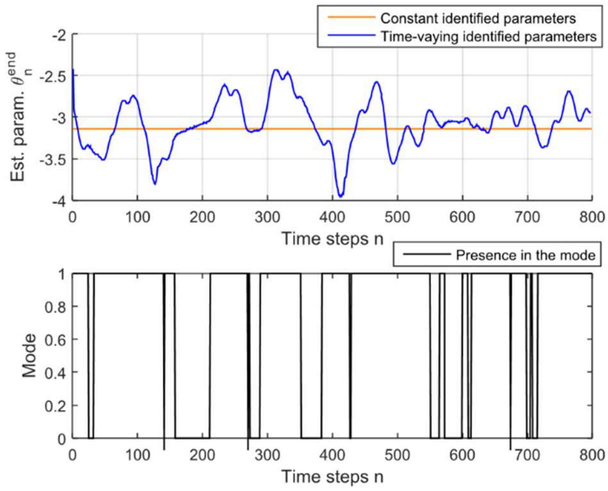

Figure 13: Driver A, aggressive driving case

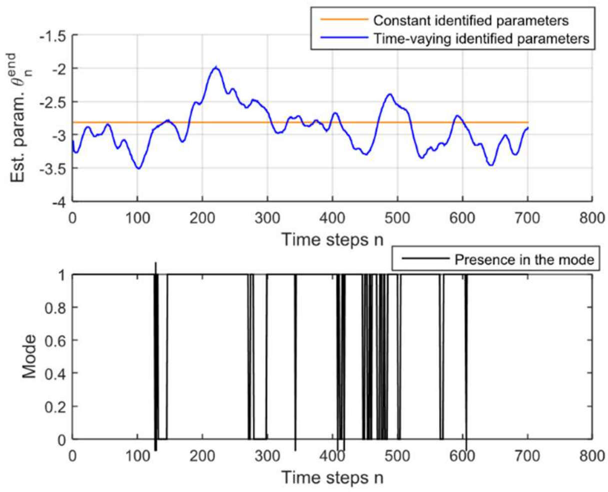

Figure 14: Driver B, soft driving case

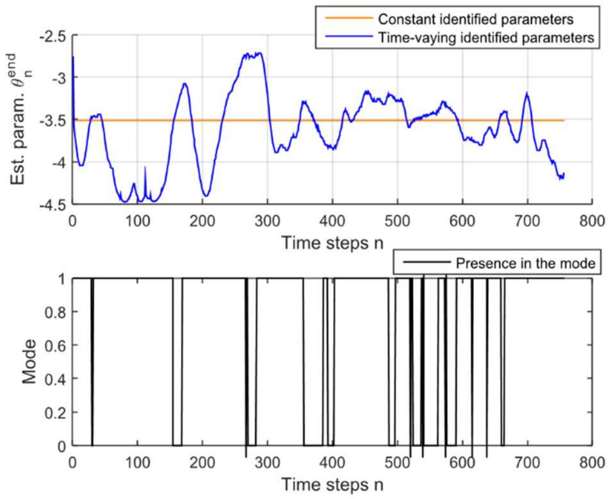

Figure 15: Driver B, aggressive driving case

In Figure 12 to Figure 15, it can be observed that the tendency of the identified parameter b, representing the desired deceleration of the driver, is logically correlated with the driving instructions. Fig. 11, 12 and 13 show clean identification values and data occlusion could be handle properly. In Figure 15, instant values of the parameter are sometimes too low. This could mean that other parameter of the model, like the apparent reaction time t_{reac} should also be optimized, or that the Gipps model is not correctly suited for this type of driving situation. To be able to identify parameters independently, a prior study on the correlation between the model parameters should be thoroughly done. Otherwise, simultaneous identification of several model parameters should be performed.

The proposed identification method does allow parameter analysis. To understand how to model the parameter behavior, longer driving recording and more information on the driving situation and on the surrounding environment would be required. It would then be possible to implement a simple parameter evolution model based on the driver behavior modification and correlated to the driving environment.

VI. Conclusion

This paper proposes a new point of view to analyze drivers' behaviors. Analysis of the behavior is done through the observation of the time-evolution of parameters of simple driver models. The behavior analysis is decomposed in two steps. First the driver model is selected or designed to represent the average behavior of a large sample of drivers. Then personal driver's behavior evolution can be analyzed over the time. To be able to identify time-varying non-linear hybrid model parameters, an iterative metaheuristic identification method based on SIR particle filtering and moving average filtering has been created. This method enables to identify parameters of any model type while filtering the parameter time-variation based on the possible parameter dynamics. This methods also enables to interpolate parameters values while model output values are occluded. Demonstration of the identification algorithm efficiency on a single parameter could be done based on theoretical example, and time-evolution of parameter could be identified from real-world measured data.

Our future research will extend the identification process demonstration to simultaneous multiple identification case, and use this parameter identification method to model parameter time-variations, and analyze the influence of parameter time-variation traffic flow simulation.

References

[1] Daiheng Ni, Traffic Flow Theory: Characteristics, Experimental Methods, and Numerical Techincs, Butterworth-Heinemann, 2016, pp.175-249

[2] H. Okuda, N. Ikami, T. Suzuki, Y. Tazaki, K. Takeda, "Modeling and Analysis of Driving Behavior Based on a Probability-Weighted ARX Model", *IEEE Trans. on Intel. Trans. Systems*, Vol. 14, No. 1, March 2013

[3] T. Akita, T. Suzuki, S. Hayakawa, S. Inagaki, "Analysis and Synthesis of Driving Behavior based on Mode Segmentation", *Intern. Conf. on Control Auto. And Sys.*, 2008

[4] Charles C.MacADAM, "Understanding and Modeling the Human Driver", *Swets & Zeitlinger, Vehicle System Dyn.*, Vol. 40, No. 1-3, pp. 101-134, 2003

[5] S. Panwai, H. Dai, "Comparative Evaluation of Microscopic Car-Following Behavior", *IEEE Trans. on Intel. Trans. Systems*, Vol. 6, No. 3, Sept. 2005

[6] J. Ackermann, J. Guldner, W. Sienel, R. Steinhauser, "Linear and nonlinear controller design for robust automatic steering", *IEEE Control Syst. Society*, March 1995

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

[8] J. Sjöberg, Q. Zhang, L. Ljung, A. Benveniste, B. Delyon, P.Y. Glorennec, H. Hjalmarsson, A. Juditsky, "Nonlinear black-box modeling in system identification: A unified overview", *Automatica*, Vol. 31, no. 12, pp. 1691–1724, Dec. 1995.

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

[10] S. Thrun, W. Burgard, D. Fox, Probabilistic Robotics, The MIT Press, 2005, pp. 96-112

[11] G. Chowdhary, R. Jategaonkar, "Aerodynamic parameter estimation from flight data applying extended and unscented Kalman filter", *Aerospace Science and Technology*, Vol. 14, March 2010, pp.106-117

[12] M. Brackstone, M. McDonald, "Car-following: a historical review", *Transportation research part F*, pp.181-196

[13] P.G. Gipps, "A behavioural Car-Following Model For Computer Simulation", *Trans. Res. Part B*, Vol. 15, No. 2, 1981, pp 105-111

[14] B. Ciuffo, V. Punzo, M. Montanino, "Thirty Years of Gipps's Car-Following Model", *Trans. Res. Record: Journal of the Trans. Res. Board*, Vol. 2315, 2012

[15] M. S. Arulampalam, S. Maskell, N. Gordon, T. Clapp, "A tutorial on Particle Filters for Online Nonlinear/Non-Gaussian Bayesian Tracking", *IEEE Trans. On Signal Processing*, Vol. 50, No. 2, Feb. 2002