Speed Control of Electric Vehicle Propulsion with Autotuning at Changeable Driving Modes and Road Conditions

Andrei Aksjonov and Pavel Nedoma
ŠKODA AUTO a.s.
Mladá Boleslav, Czech Republic
andrei.aksjonov@skoda-auto.cz

Valery Vodovozov, Zoja Raud and Eduard Petlenkov
Tallinn University of Technology
Tallinn, Estonia
valery.vodovozov@taltech.ee

***Abstract*— Considering the high demands continually imposed on equipment indispensable for intelligent transportation, this paper focuses on the cruise control in propulsion systems of road electric vehicles. The research lays emphasis on how to arrange optimal dynamics in terms of the speed overshoot and the speed rise time at changeable driving modes and road conditions. The paper addresses two aspects, namely, most accurate following the demanded transition process and most rapid achieving the setpoint. The former issue is typical for industrial vehicles operated in rather stable obstacles (loaders, forklift trucks, carriers, etc.), whereas the latter one concerns traditional road electric cars. Online autotuning of the controller is provided directly in the driving process by applying periodically estimated slope and peak input signals for analysing the speed responses and correcting controller settings. Tuning procedures based on binary logic and fuzzy logic approaches are compared.**

Keywords*— road electric vehicle, propulsion, PID controller, binary logic controller, fuzzy logic controller, autotuning.

I. INTRODUCTION

New approaches are manifested now in the design of road electric vehicles (EV). Propulsion systems of EVs are among the complex nonlinear applications running in different modes, such as highway cruising with acceleration and slowing down, heavy braking with the aid of an antilock braking system (ABS), and parking. Each of modes calls its specific requirements with regard to dynamic performance, positioning accuracy, speed range, torque stability, and overload capacity in the face of changing road and weather circumstances. To avoid unpredictable behaviour of EV, its propulsion system places high demands on intelligent controllers with online autotuning capable to make decisions without human intervention, if need be.

Oftentimes, a conventional proportional-integral-derivative controller (PIDC) is employed, which works properly under a specific set of known system parameters and load conditions. Setting search process of the PIDC is quite elaborated for both the offline and the online stages. Offline tuning is based on the mathematical model of the controlled plant and disturbances, their gains and time constants, to name a few. Because of a wide range of uncertainties, not all data can be exactly known offline; therefore, the natural next step is to consider autotuning, which corrects PIDC settings online to meet the real-time needs. Online tuning is applied in practice by analysing typical EV reactions in critical points and characteristics of the plant, including step response, frequency response, close-loop relay feedback, etc. Some well-known tuning methods, such as Ziegler-Nichols, Cohen-Coon, Chien-Hrones-Reswick, or more sophisticated schemes [1], are successfully adopted to online autotuning aiming to maintain different types of loop performance. Above reasoning results in enough accurate PIDC settings for stable modes of EV running, where the setpoint changes larger and faster than the control variables, while disturbances appear as slow departures of control variables from the setpoint. However, some circumstances hamper PIDC use for electric motion control:

- in unstable conditions, the settings that produce a desired response in one operating point usually do not call a satisfactory response in another one [2];
- PIDC perfectly operates at small deviations of the controlled variables [3], whereas considerable speed or torque changes may call PIDC saturation with possible feedbacks disconnection;
- PIDC is mainly designed to work in the systems with single input and single output (SISO), but the circuits with multiple inputs and multiple outputs (MIMO) and multiple inputs and single output (MISO) often fail in providing acceptable performance [4].

To resolve these conflicting issues, the gain scheduling method is somewhere applied [5], [6], which helps in choosing the optimal PIDC settings best satisfying the current range of plant variables. Nevertheless, gain scheduling cannot consider all possible system states to ensure overall robustness.

To reach more robustness, designers are referred to the fuzzy logic controller (FLC) that, in contrast to classical binary logic, deals with fuzzy sets of linguistic variables (LV) capable to partial membership between 0 (absence of membership) and 1 (full membership) rather than to crisp membership (0 or 1). Thanks to its membership function (MF), the FLC is able to govern nonlinear and complex plants that are difficult to characterise mathematically but are described qualitatively. However, despite the numerous FLC benefits [7], [8], it is worthy of note FLC weakness in terms of vehicle motion:

- in contrast to PIDC, the FLC has problems with dynamic aspects because of its stepwise MF patterns unsuitable for smooth cruising, acceleration, or slowing down;
- attempting to develop a FLC equally suitable for speed control, positioning, and braking is usually daunting as it is difficult to establish fuzzy relations between significant number of variables [9];
- though new sliding and adaptive approaches alleviate some difficulties in constructing fuzzy rule bases [10], they, just like the traditional trial-and-error methods, remain quite sensitive to practitioners' cognitive biases that hamper control reliability.

As a result, most of often-cited FLCs are valid only within the specific bands of parameters and variables. This is a severe restriction on general implementation of FLCs since they require extensive retuning [11]. In particular, in [12] –[15], FLCs are applied for braking only. In [16], the sole mode of FLC usage is parking. MIMO FLCs in [7], [17], [18] meet many challenges with tuning.

Starting with [8], [19], it has been found that the fuzzy-PIDC has often better handling capabilities than both the PIDC and the FLC separately. Nonetheless, such negative issue of the fuzzy-PIDC as MIMO arrangement difficulty complicates its construction and autotuning. There are rather few publications about MIMO fuzzy PIDCs that convert two inputs, usually the speed error and its rate, directly into three PIDC settings: [9], [20] – [22]. A more common approach presented, in particular, in [6], [23] – [25] involves sharing of fuzzy operation among three independent controllers, namely, fuzzy-P, fuzzy-I, and fuzzy-D.

Aside from direct problem solving, the essential idea of many fuzzy-PIDC algorithms lies in the MIMO system alignment with the MISO one by converting the error signal (speed) and its time derivative (acceleration) into some aggregated output with a linear-like control surface [26], which actually combines fuzzy-PI and fuzzy-PD controllers. Particularly, in [19] a Ziegler-Nichols formula is parameterized by a single variable. According to [27], fuzzy-PIDC fails in use of Mamdani's fuzzy reasoning, although two-term fuzzy controllers, PI and PD [10], PD and ID [11] or PD and I [22] may be successfully realised. However, the problem of optimal settings for all PIDC parts remains open in these systems as their control actions are strongly coupled. The contribution of scaling gains to the output action remains unclear that makes tuning methodology rather unreliable.

To address the problems of nonlinearity and time-variability, new control terms were proposed in [4], [21] and [28], but these techniques were also limited by the fixed range of control parameters, resulting in frequent detuning to accommodate worst-case scenarios, for instance, traction upon icy conditions with old tires.

Unlike the listed studies, the present research lays emphasis on how to maintain maybe not the best on its own, but some optimal (say, sample) dynamics in terms of the speed overshoot and the first-matching time at changeable driving modes and road conditions. To that end, the paper addresses two online autotuning aspects, namely, the most accurate following the sample response trace on the one hand, and the most rapid achieving the setpoint on the other. The first issue is typical for such industrial EVs operated in more or less stable conditions as loaders, forklift trucks, carriers, etc. The second one concerns traditional electric cars, buses, etc. The study focuses on the fuzzy-PIDC autotuning based on two periodically computed signals, namely, the slope error and the peak error, with the help of MIMO binary logic and fuzzy logic approaches. National Instruments® LabVIEW™ is used in this research as both the simulation tool and the user interface for data acquisition and analysis of PIDC, FLC, and fuzzy-PIDC.

The paper is organised as follows. First, the principles are justified for a versatile controller composition equally suitable for highway cruising, acceleration, and slowing down of EV under different conditions. Next, controller tuning and binary logic autotuning peculiarities are explained with integration in the propulsion control system. Then, fuzzy-PIDC operation is demonstrated and conclusions are drawn.

II. Control System of EV Propulsion

Controller design issue was intensively investigated by researchers in the past. To build any controller, control variables y, setpoints y^*, and control demands x have to be specified as well as the crisp range of all their possible values called a universe of discourse (UOD) (Fig. 1).

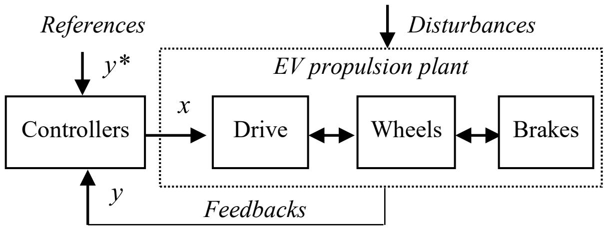

Fig. 1. EV propulsion system.

Every mode of EV running has its specific control variables: speed of cruising, position of parking, and torque of braking. In this study, to manage the EV propulsion plant, all the operations are distributed among free controllers working in separate control loops as shown in Fig. 2.

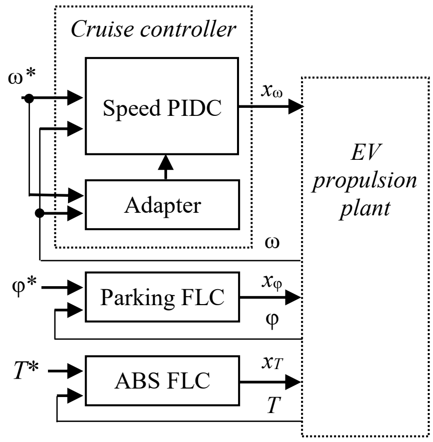

Fig. 2. Control system composition for EV propulsion.

The system includes:

- cruise controller composed of the speed PIDC and an adapter for PIDC autotuning, which setpoint speed ω* and the control variable – real plant speed ω – serve as inputs whereas x_ω is the speed control demand;
- positioning FLC for parking, which setpoint φ* and sensed position φ are applied as inputs whereas the output x_φ governs EV positioning;
- torque FLC for ABS, which setpoint T* and sensed torque T signals are applied as inputs whereas the output x_T establishes EV braking torque.

Parking FLCs are described in [5], [16] and severe other sources. Novel fuzzy ABS organization was presented recently in [29], [30]. The offered cruising speed control is explained below.

A linear model shown in Fig. 3 is used as a first approximation of the speed control system. Here, s is the Laplace operator, δ = ω*– ω – speed error, E – motor electromotive force representing the control demand x_ω, T_p – plant torque representing the main disturbance, $W_c(s)$ – transfer function of the designed controller, and $W_p(s)$ – transfer function of the object, EV propulsion plant.

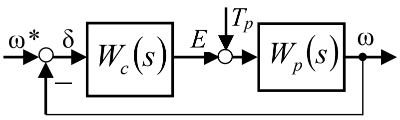

Fig. 3. Approximated linear model of the speed control system.

The close loop system shown in Fig. 3 may be assumed by means of the transfer function

$$W(s) = \frac{\omega(s)}{\omega^*(s)} = \frac{W_c(s)W_p(s)}{1+W_c(s)W_p(s)} \cdot \quad (1)$$

To ensure the demanded response ω*, the transfer function of the controller has to be as follows:

$$W_c(s) = \frac{1}{W_p(s)} \cdot \frac{W(s)}{1-W(s)} \cdot \quad (2)$$

This image includes the transfer function of the plant and a part, which depends on the desired system model $W(s)$. Usually, transfer functions $W(s)$ differ depending on their characteristic polynomials and the process dead time τ_μ. For EV speed cruising, the desired process may be shaped for providing equally balanced performance for both disturbance rejection and reference signal tracking. Particularly, the transfer function described by the second-order polynomial

$$a_1\tau_\mu^2 s^2 + a_1\tau_\mu s + 1 \quad (3)$$

best fulfils this requirement when $2 \le a_1 < 4$ [31], at which the EV has the non-periodic step response with tiny overshoot. Let us call it a sample response towards the given dead time.

Then proceed to the plant. At this stage, the following pair of the Laplace's equations describes the cruising speed responses on the step reference and disturbance:

$$\omega(s) = E(s)\frac{k}{\tau\tau_\mu s^2 + \tau s + 1}, \quad (4)$$

$$\omega(s) = -T_p(s)\frac{\tau(\tau_\mu s + 1)}{J(\tau\tau_\mu s^2 + \tau s + 1)} \cdot \quad (5)$$

Here, $k = \frac{I}{T}$ is the motor gain represented by the ratio between its passport current I and torque T; T_p(s) – the plant torque Laplace image showing the main disturbance; $\tau = \frac{J\omega_{max}}{T_{max}}$ – plant time constant; J – its moment of inertia; ω_{max}, T_{max} – maximal speed and torque of the motor.

Once the desired system and the plant models are defined, the controller turn is coming. Oftentimes, a PIDC is used here, which transfer function in the operator domain is as follows:

$$W_c(s) = \frac{E(s)}{\delta(s)} = k_c\left(1 + \frac{1}{\tau_{int}s} + \tau_{dif}s\right) \cdot \quad (6)$$

Here, k_c is the controller gain, τ_{int} – integral time constant, and τ_{dif} – derivative time constant. The controller gain defines the regulation accuracy, the integral part forces the steady-state error to zero albeit it has an adverse impact to system dynamics, and the derivative part accelerates dynamics whenever necessary. The output of the PIDC fitted to its UOD $\{x_{min}; x_{max}\}$ is applied as a control demand E to the propulsion drive aimed to shift the control variable ω to the setpoint ω* for minimising the speed error δ.

III. PIDC Tuning and Binary Logic Autotuning

At first, the PIDC is tuned offline to obtain the required steady-state accuracy and sample dynamics shown in Fig. 4. Decision-making rules used for tuning the proportional, integral, and derivative parts of the PIDC are given, particularly, in [31].

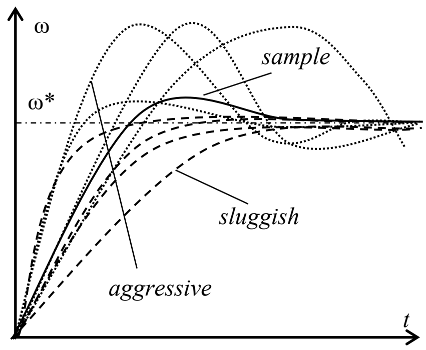

Fig. 4. Step responses and distortion combinations.

As offline tuning is based on approximated plant and disturbance models, the issues associated with system nonlinearity and load instability usually lead to distorted loop behaviour, and the system appears either too sluggish (dashed lines in Fig. 4) or too aggressive (dotted lines). Against this backdrop, an online autotuning is initialised.

To that end, the controller periodically evaluates the trends of the plant step responses aiming to detect their deviation from a prescribed sample course. If a deviation is found, a control action is generated to correct the PIDC settings with the following essence.

Every abnormal response is seen as a combination of two types of dynamic errors: the slope and the peak.

The slope characterises the speed rise time, i.e. the first-matching time during which the speed response approaches the setpoint. Call the slope error the ratio between the rise time of the sample speed response $t_{sample(\omega=\omega^*)}$ and the rise time of the distorted process $t_{proc(\omega=\omega^*)}$:

$$slope = \frac{t_{sample(\omega=\omega^*)}}{t_{proc(\omega=\omega^*)}} . \quad (7)$$

The peak characterises the speed overshoot. Call the peak error the ratio between the overshoot of the distorted process $\delta_{proc(\max)}$ and the overshoot of the sample one $\delta_{sample(\max)}$:

$$peak = \frac{\delta_{proc(\max)}}{\delta_{sample(\max)}} . \quad (8)$$

Based on the above definitions, the following stepwise binary logic PIDC autotuning algorithm is implemented:

- to reduce (or enlarge) a peak, time constant τ_{int} is increased (or, appropriately, decreased) in increments, and, if possible, time constant τ_{dif} is decreased (or increased) together, until the sample peak reaching;
- often, any change of τ_{int} calls the reverse change of the slope, therefore, the slope correction is further needed;
- to reduce (or enlarge) a slope, gain k_c is decreased (or, appropriately, increased) in increments, until the sample slope approaching.

The simple-iteration method with alternating step size was employed for the gain and time constants reduction and enlargement. The step size is defined by the size of error whereas the number of steps depends on the slope and peak tolerances.

It is noteworthy that the derivative term is seldom employed in EV control, mainly due to the fact, that it increases sensibility to noise. Although most of the PIDCs incorporate this action, it is quite usual for the plant operators to inhibit it. In the automotive applications, the derivative mode is sometimes used running and braking, whereas the integral mode is applied universally. Thereby, the derivative term is accompanied by "if possible" remark.

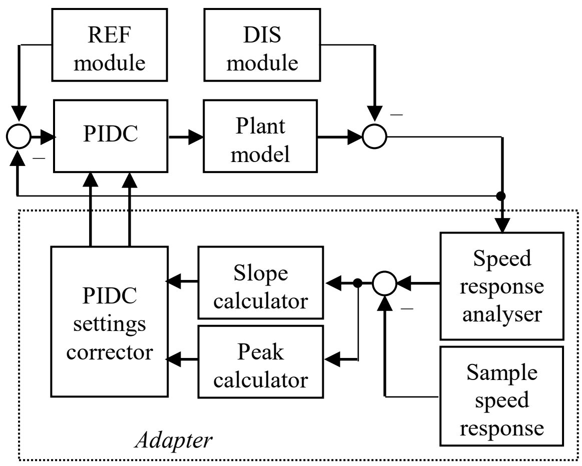

Fig. 5. Model of the speed control system with autotuning.

In Fig. 5, the speed control system with autotuning is shown. Here, the reference (REF) and disturbance (DIS) modules generate the PIDC and plant speed responses directed to the Adapter, which compares the plant response with the sample one aiming to normalize the slope and peak errors.

In Fig. 6, the designed LabVIEW user interface is presented. Here, two selectors are intended for the simulation mode choice: the Reference/Disturbance switch and the Manual/Auto switch. REF settings, DIS settings, Manual PID settings, and PID autotuning settings occupy the separate areas alongside the response chart display. A line of placards reflects process information.

The main benefit of this binary logic autotuning procedure is its accuracy whereas the drawbacks relate to its slowness. Simulation shows that it costs a long time to settle for a step response. Adapter requires up to 50 online PIDC cycles for autotuning in large inertia delay systems.

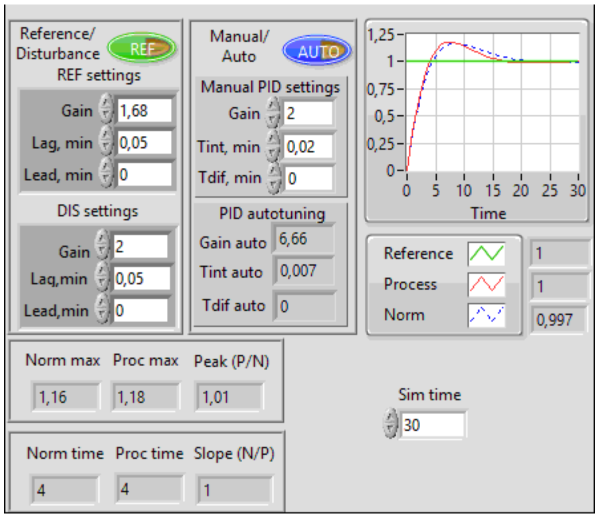

Fig. 6. LabVIEW user interface of the speed control system.

IV. Cruise PIDC with Fuzzy Logic Tuning

As it was shown in [32], non-linearly constrained and badly modelled problems can often be solved in lesser number of iterations using fuzzy approach, in which a rather "clever" adapter is capable to make well-informed decisions regarding directions of search and step size. Therefore, as the next step to speed up the process, the binary logic adapter was replaced with the FLC.

The speed FLC has a MIMO topology with the Mamdani's-type inference mechanism. In the developed fuzzy-PI controller, the slope and the peak errors were assigned as input LVs. From the slope side, the responses are classified using the following fuzzy sets: *VB* – very bluff (>>1), *B* – bluff (>1), *N* – normal (≈1), *S* – sloping (<1), and *VS* – very sloping (<<1). Depending on the peak errors, the responses are classified as *VB* – very big (>>1), *B* – big (>1), *N* – normal (≈1), *S* – small (<1), and *VS* – very small (<<1). The PIDC gain correction and the PIDC integral time constant correction are the FLC outputs. Here, *dK* and *dT* were chosen as output LVs and their MFs were assigned as *BDn* – big down, *Dn* – down, *Z* – zero change, *Up* – up, and *BUp* – big up. By applying "If–Then" modus ponens, an appropriate rule base has been developed (Table I).

TABLE I. RULE BASE OF SPEED FLC

slope	peak				
	VS	***S***	***N***	***B***	***VB***
VS	dK=*BUp*	dK=*BUp*	dK=*BUp*	dK=*BUp*	dK=*BUp*
	dT=*BDn*	dT=*Dn*	dT=*Z*	dT=*Up*	dT=*BUp*
S	dK=*Up*	dK=*Up*	dK=*Up*	dK=*Up*	dK=*Up*
	dT=*BDn*	dT=*Dn*	dT=*Z*	dT=*Up*	dT=*BUp*
N	dK=*Z*	dK=*Z*	dK=*Z*	dK=*Z*	dK=*Z*
	dT=*BDn*	dT=*Dn*	dT=*Z*	dT=*Up*	dT=*BUp*
B	dK=*Dn*	dK=*Dn*	dK=*Dn*	dK=*Dn*	dK=*Dn*
	dT=*BDn*	dT=*Dn*	dT=*Z*	dT=*Up*	dT=*BUp*
VB	dK=*BDn*	dK=*BDn*	dK=*BDn*	dK=*BDn*	dK=*BDn*
	dT=*BDn*	dT=*Dn*	dT=*Z*	dT=*Up*	dT=*BUp*

Using the centre of gravity as a defuzzification method, two outputs are further converted to the enhanced crisp settings: k_c and τ_{int}. Fig. 7 represents the fuzzy triangle sets for the input and output MFs that have closed frontiers of UODs.

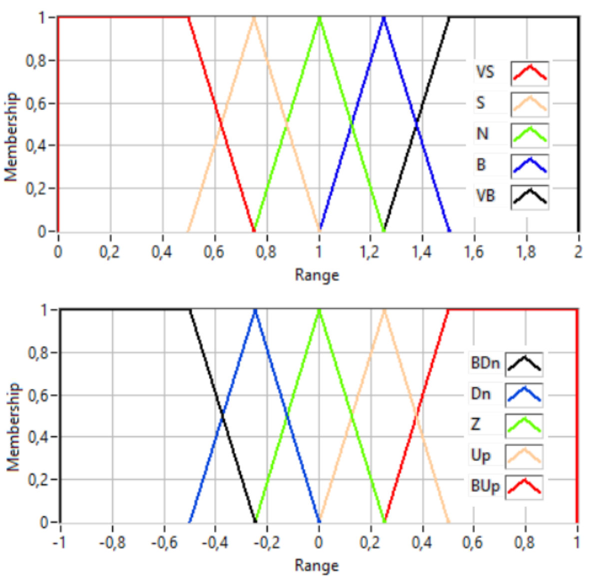

Fig. 7. Fuzzy sets for the input and output LVs.

V. EXPERIMENTAL VERIFICATION

To validate the designed tuning methodology, the laboratory setup FESTO® TP801 was applied equipped with a brushless dc motor MTR-AC55, a servo converter SEC-AC305, and an axis slider replicating a moving vehicle part. The plant has the following data: nominal motor voltage 325 V, current 2.15 A, torque 0.66 Nm, and speed 6800 rpm; maximal current 6.4 A at torque 0.98 Nm; moment of inertia of the plant 1 kgcm^2 and dead time 15 µs.

Before experimentation, the PIDC settings were assigned that provide the sample response of the setup. Next, moment of inertia and friction of the slider were changed resulting in the response distortion. This detuned response is shown in Fig. 8 (a). After that, autotuning was conducted on the model shown in Fig. 5 where the setup plant module and the distorted parameters were replicated quite accurately. In tuning, both the binary logic and the fuzzy logic algorithms were used. The settings obtained were uploaded to the setup PIDC.

Experimental speed and current traces of a configured with the binary logic controller settings (b) and configured with the FLC settings (c) are displayed in Fig. 8 also. Experimentation demonstrates that the PIDC settings obtained from the binary logic algorithm provide the precise slider run up with sample speed response whereas the FLC-driven process looks rather sluggish but close enough to the sample as well.

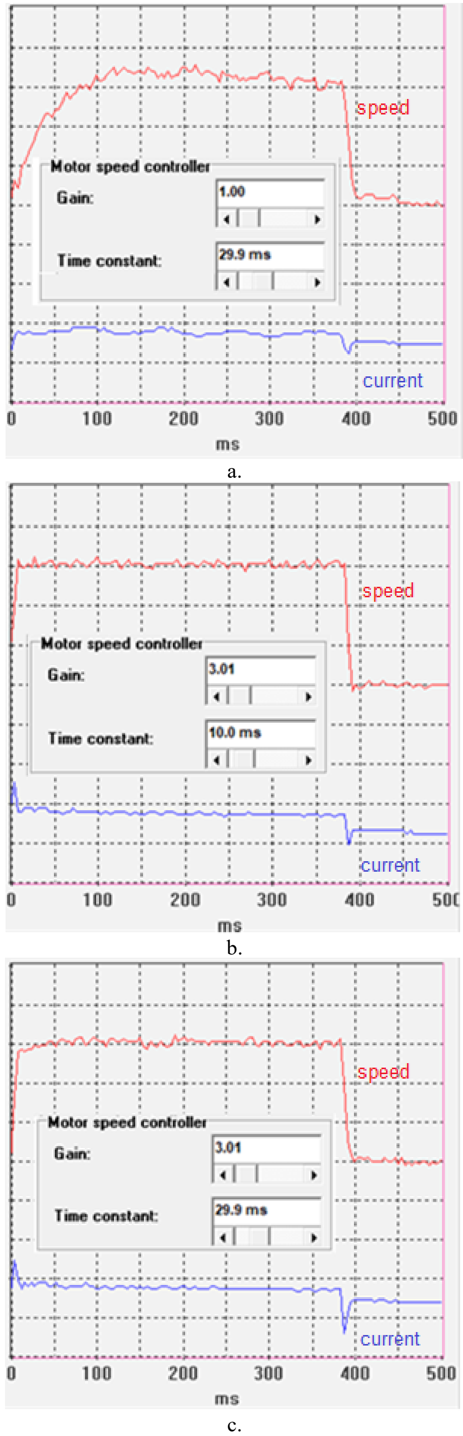

Fig. 8. Experimental traces recorded from the laboratory setup.

VI. CONCLUSIONS

Simulation and experimental study show that the proposed PIDC arrangement has competitive advantages over known control approaches used in automotive applications. It implements online autotuning directly in the driving process by applying periodically computed slope and peak estimations for analysing the speed responses and correcting PIDC settings. Both the binary logic and the fuzzy logic controllers are capable to implement nonlinear control strategies described by mathematical and linguistic variables, appropriately, in the conditions when the process, the plant, and the disturbance models change unpredictably. The methodology offered demonstrates high versatility as it supports different modes of the EV running.

ACKNOWLEDGMENT

This study has received partial funding from the European Union's Horizon 2020 research and innovation program under grant agreement No. 675999.

REFERENCES

1 M. Korkmaz, Ö. Ydoğdu and H. Doğan, "Design and performance comparison of variable parameter nonlinear PID controller and genetic algorithm based PID controller," International Symposium on Innovations in Intelligent Systems and Applications, Trabzon, Turkey, 2012, pp. 1 – 5.

2 H. Zhou, F. Jia, H. Jing, Z. Liu and L. Güvenç, "Coordinated longitudinal and lateral motion control for four wheel independent motor-drive electric vehicle," IEEE Transactions on Vehicular Technology, vol. 67, no. 5, pp. 3782 – 3790, May 2018.

3 J. Zhao, Y. Ma, H. Zhao, Y. Cui and H. Chen "PID slip control based on vertical suspension system for in-wheel-motored electric vehicles," Chinese Control And Decision Conference (CCDC), Shenyang, China, 2018, pp. 1126 – 1131.

4 P. Gil, C. Lucena, A. Cardoso and L. B. Palma, "Gain tuning of fuzzy PID controllers for MIMO systems: A performance-driven approach," IEEE Transactions on Fuzzy Systems, vol. 23, no. 4, pp. 757 – 768, Aug. 2015.

5 LabVIEW PID and Fuzzy Logic Toolkit User Manual, National Instruments, 2009, 126 p.

6 A. M. O. Fini, M. B. Gogani and M. Pourgholi, "Fuzzy gain scheduling of PID controller implemented on real time level control," 4th Iranian Joint Congress on Fuzzy and Intelligent Systems (CFIS), Zahedan, Iran, 2015, pp. 1 – 5.

7 K. S. Jena, A. V. Joseph and P. R. R. Senapati, "Fuzzy logic based approach for controlling of a vehicle in its longitudinal motion, Middle-East Journal of Scientific Research, vol. 24 (S1), pp. 346 – 352, 2016.

8 W. Siler and H. Ying, "Fuzzy control theory: The linear case," Fuzzy Sets Syst., vol. 33, pp. 275 – 290, 1989.

9 P. Xue, H. Wang, J. Hou and W. Li, "Based on the fuzzy PID brushless DC motor control system design," International Conference on Measurement, Information and Control (MIC), Harbin, China, 2012, pp. 703 – 706.

10 B. M. Mohan and A. Sinha, "Analytical structures for fuzzy PID controllers?," IEEE Transactions on Fuzzy Systems, vol. 16, no. 1, pp. 52 – 60, Feb. 2008.

11 Q. Li and D. Shen, "A new incremental fuzzy PD+fuzzy ID fuzzy controller," WASE International Conference on Information Engineering, Taiyuan, China, 2009, pp. 615 – 619.

12 M. Santos, S. Dormido and J. M. de la Cruz, "Fuzzy PID convrollers vs. fuzzy PI controllers," 5th IEEE International Conference on Fuzzy Systems, New Orleans, LA, USA, 1996, pp. 1598 – 1604.

13 Y. Tao, X. Xie, H. Zhao, W. Xu and H. Chen, "A regenerative braking system for electric vehicle with four in-wheel motors based on fuzzy control," 36th Chinese Control Conference, Dalian, China, 2017, pp. 4288 – 4293.

14 P. Dhivya and A. Murugesan, "Intelligent car braking system with collision avoidance and ABS," International Journal of Computer Applications, 2015, pp. 16 – 20.

15 P. Khatun, C. M. Bingham, N. Schofield and P. H. Mellor, "An experimental laboratory bench setup to study electric vehicle antilock braking / traction systems and their control," IEEE 56th Vehicular Technology Conference, Vancouver, Canada, 2002, pp. 1490 – 1494.

16 M. A. Abu, Z. Kornain, M. H. Rosli and I. M. Iqbal, "Automated car braking system using LabVIEW," IEEE Symposium on Industrial Electronics and Applications (ISIEA), Bandung, Indonesia, 2012, pp. 246 – 250.

17 J. Soriano, A. Olarte and M. Melgarejo, "Fuzzy controller for MIMO systems using defuzzification based on boolean relations (DBR)," 14th IEEE International Conference on Fuzzy Systems, Reno, NV, USA, 2005, pp. 271 – 275.

18 Y. Jian and L. Changliang, "Design of self-tuning PID controller with fuzzy variable parameters based on LabVIEW," IEEE International Conference on Information and Automation, Lijiang, China, 2015, pp. 2586 – 2591

19 S. Z. He, S. H. Tan, F. L. Xu and P. Z. Wang, "PID self-tuning control using a fuzzy adaptive mechanism," 2nd IEEE International Conference on Fuzzy Systems, San Francisco, CA, USA, 1993, pp. 708 – 713.

20 Y. Xiaojin, P. Yan, S. Jinhao, L. Yezi and Q. Jianling, "Self-adaptive tuning of fuzzy PID control of PV grid-connected inverter," 6th International Conference on Fuzzy Systems and Knowledge Discovery, Tianjin, China, pp. 160 – 162.

21 L. P. Sun, Z. H. Li and W. H. Cao, "Design and realization of fuzzy self-tuning PID controller based on TMS320LF2407A DSPs for microwave-vacuum wood drying," 6th International Conference on Fuzzy Systems and Knowledge Discovery, Tianjin, China, pp. 71 – 75.

22 A. F. Amer, E. A. Sallam and W. M. Elawady, "Fuzzy pre-compensated fuzzy self-tuning fuzzy PID controller of 3 DOF planar robot manipulators," IEEE/ASME International Conference on Advanced Intelligent Mechatronics, Montreal, Canada, 2010, pp. 599 – 604.

23 A. Rubaai, M. J. Castro-Sitiriche and A. Ofoli, "DSP-based implementation of fuzzy-PID controller using genetic optimization for high performance motor drives," IEEE Industry Applications Annual Meeting, New Orleans, LA, USA, 2007, pp. 1649 – 1656.

24 S. H. Lian, C. H. Messom and Y. P. Kong, "Fuzzy hybrid PID controller of a steam heated dryer," IEEE International Fuzzy Systems Conference, Seoul, South Korea, 1999, pp. 1707 – 1711.

25 K. Sharma and D. K. Palwalia, "A modified PID control with adaptive fuzzy controller applied to DC motor," International Conference on Information, Communication, Instrumentation and Control (ICICIC), Cairo, Egypt, pp. 1 – 6.

26 B. Hu, G. K. I. Mann and R. G. Gosine "New methodology for analytical and optimal design of fuzzy PID controllers," IEEE Transactions on Fuzzy Systems, vol. 7, no. 5, pp. 521 – 539, Oct. 1999.

27 M. Mizumoto, "Realization of PID controls by fuzzy control methods," IEEE International Conference on Fuzzy Systems, Adelaide, Australia, 1992, pp. 709 – 715.

28 X.-G. Duan, H.-X. Li and H. Deng "A simple tuning method for fuzzy PID control," IEEE International Conference on Fuzzy Systems, Hong Kong, China, 2008, pp. 271 – 275.

29 A. Aksjonov, V. Vodovozov and E. Petlenkov, "Design and experimentation of fuzzy logic control for an anti-lock braking system," 15th Biennial Baltic Electronics Conference (BEC), Tallinn, Estonia, 2016, pp. 207−210.

30 A. Aksjonov, V. Vodovozov, K. Augsburg and E. Petlenkov, "Design of regenerative anti-lock braking system controller for 4 in-wheel-motor drive electric vehicle with road surface estimation," International Journal of Automotive Technology 19 (4), pp. 727−742, 2018.

31 V. Vodovozov, Electrical Drive: Performance, Design and Control, Saarbrücken, Germany: LAP, 2014, 320 p.

32 Y. Huang and S. Yasunobu, "A general practical design method for fuzzy PID control from conventional PID control," 9th IEEE International Conference on Fuzzy Systems, San Antonio, TX, USA, 2000, pp. 969 – 972.