2018 26th Mediterranean Conference on Control and Automation (MED)
Zadar, Croatia, June 19-22, 2018

A New Approach of Neurofuzzy-based Control Design and Analysis applied on an EV Electromechanical System

Jemma J. Makrygiorgou[1] and Antonio T. Alexandridis[2], *Member, IEEE*

***Abstract*— Electric vehicles (EVs) constitute an attractive solution for transportation caused by both environmental and economic reasons. In view of their complexity as electromechanical structures, EVs are multi-input nonlinear systems which cannot be easily controlled and analysed. Hence, in order to operate an EV in a reliable and desired manner, neurofuzzy controllers (NFCs) are proposed, implemented and examined. The latter can deal with uncertain environments, due to their capabilities in learning, adaptation and fault tolerance by incorporating a properly adapted expert knowledge. However, in the paper, the cumbersome task of proving stability of the NFC driven electromechanical system is provided by slightly modifying the adaptation dynamics needed for the learning procedure. This is the main theoretical contribution presented by this work, since it makes possible to ensure input-to-state stability for the entire system and convergence of the system states to the desired equilibrium. The conducted simulations verify the satisfactory system performance and confirm the attained theoretical results.**

I. INTRODUCTION

Electric vehicles (EVs) constitute a new trend in the world of automotive industry. In the near future, many companies would base their entire model of cars around electricity, a reasonable outcome if one takes into account all benefits that EVs offer. The choice of EVs is the most straightforward way for decreasing our personal impact on the environment, since they are running without fuel and consequently there are no emissions associated [1], especially when electricity is produced by renewable energy sources. Additionally, EVs restrict noise pollution, as the electric motors they use provide smooth and noiseless drives. Besides the environmental advantages EVs offer economic benefits to their owners, too. Specifically, low maintenance of electric motors and cheap, with concerning to petrol and fuel prices, charging tarrifs comprise them in cost effective solutions for transportation [2].

In their nature, EVs are complex electromechanical systems described by a nonlinear mathematical model. At the physical level, power should be handled with the most efficient way between the batteries and the motor. Practically, the driver sets the car velocity, in accordance to the road and driving conditions, by using the car pedal, while all the other regulations for a fast and smooth response are automatically regulated by internally controlling the power electronic devices [3, 4]. The design of suitable controllers based on the system model suffers from the aforementioned nonlinearities. To overcome this obstacle, several EV control strategies have been reported [5-8], with the fuzzy logic approach being good enough, since it provides an efficient way to cope with system nonlinearities [9].

Nevertheless, due to lack of appropriate systematic methodology for their design and their static development, FLCs fail to meet the experts expectations and to deal with input disturbances and system uncertainties. Attempting to dissolve those problems and automate the FLCs design, neural network (NN) techniques are embraced, due to their promising capabilities in learning, adaptation and fault tolerance [10]. The integration of fuzzy logic with NNs gives rise to a unified framework of neurofuzzy systems that incorporates expert knowledge and contains enough intelligence to perform accuracy tasks in uncertain environments. Though the unified framework of NNs and fuzzy logic have helped in accessing and exploiting better their respective advantages, there are concerns about the stability and performance analysis of neurofuzzy structures.

In the present work, keeping in mind the entire nonlinear system model, the implementation of FLCs on a NN architecture is performed. Three neurofuzzy controllers (NFCs) are proposed and developed; two on the synchronously rotating reference frame at the ac/dc interface, namely the $d-$ and $q-$ stator input voltage components that are related to the voltage source converter duty-ratio, and, one on the dc voltage component that is related to the dc/dc boost converter duty-ratio. The aforementioned controllers are designed in order to achieve optimum motor flux extraction, track the motor angular velocity based on the driver decisions and automatically determine the battery operation and charging performance, respectively. The conducted simulations confirm the controllers sufficient function and the satisfactory system response.

It is worth noting that both the design approach and the analysis are deployed by introducing a slightly modified procedure, capable to overcome the main drawback of the NFC schemes, i.e. the weakness of proving stability. Specifically, an innovative learning technique is proposed that uses an adaptation mechanism based on the ac and dc current-state variables, instead of the original external loop variables such as the rotor speed and the dc-link voltage. This enables, the entire EV system combined with the proposed NFCs to be analysed for its stability via the application of advanced Lyapunov techniques. To achieve this task, the nonlinear system model is considered along with the NFC

[1]Jemma J. Makrygiorgou is with the Department of Electrical and Computer Engineering, University of Patras, Rion, 26504, Greece dmak@upatras.gr

[2]Antonio T. Alexandridis is with the Department of Electrical and Computer Engineering, University of Patras, Rion, 26504, Greece a.t.alexandridis@ece.upatras.gr

978-1-5386-7891-6/18/$31.00 ©2018 IEEE

adaptation dynamics. The overall analysis is concentrated to initially prove input-to-state stability, thus ensuring a kind of robustness, and subsequently convergence to the unique equilibrium in steady state. The results are very satisfactory while simultaneously the current loop based adaptation mechanism ensures overload protection, in a manner similar to that of the familiar to engineers conventional inner-loop current controllers [11].

II. Neurofuzzy Control - Basics

As it is widely known, NFCs get rise from the combination of fuzzy logic and NN frameworks. Knowledge of some basic theoretical notions is essential, in order to proceed with the design of such a controller. To this end, a brief presentation of both frameworks is provided.

A. Fuzzy Logic Control

Fuzzy logic controllers make use of human knowledge and experience by implementing linguistic control laws, in the form of $if-then$ rules [12]. The aforementioned conditional statements can model the qualitive aspects of human knowledge and reasoning processes, without employing precise quantitive analysis. Specifically, an $if-then$ statement defines the set of facts that must be true, before a set of actions is executed.

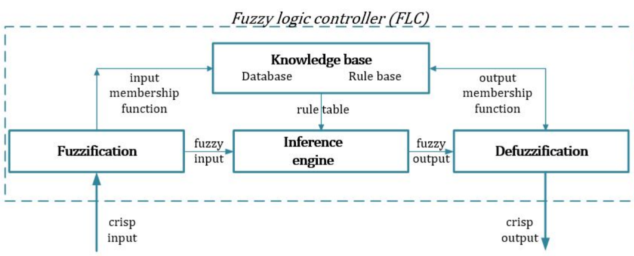

Fig. 1. The fuzzy logic controller structure

A general FLC scheme, as shown in Fig. 1, consists of the fuzzifier, the rule base, the database, the inference engine and the defuzzifier [13]. The inference engine is considered as the heart of an FLC, and is responsible for performing all the steps of fuzzy reasoning, which is the procedure of making decisions based on the available inputs and producing the corresponding outputs. In particular, except from the *fuzzification* on the premise part and *defuzzification* on the consequent part, the main steps of fuzzy reasoning include:

- the *firing strength calculation*, which derives from the combination (through a T-norm operator) of membership values on the premise part of the FLC and,
- the generation of *qualified consequent* of each rule depending on the signal strength.

The overall output of the FLC is the weighted average of each rule crisp output, induced by the rules firing strength and output membership functions.

B. Neural Networks

On the other hand, a neural network is considered as a processor that has the ability of storing experiential knowledge and making it available for use [14]. NNs mimic the organization and the function of human brain, as knowledge is acquired through a learning process and stored into inter-neuron weights. It is apparent that NNs do not require detailed information of the system model, and create the relationship between the input and the system variables as a consequence of the recorded training data.

A general NN structure consists of nodes (neurons) and directional links with adaptable weights, through which the node are connected [15]. Each node performs a particular function on incoming signals, while a set of parameter pertaining to this node. The parameter set of an adaptive NN is the union of the parameter sets of each adaptive node.

In order to achieve a desired input-output mapping, those parameters are updated according to the provided training data and a learning procedure [15]. Actually, the network uses the input data to produce an output, which afterwards is compared to the training pattern. In case there is a difference, the connection weights are altered, in such a direction that the error is compensated. Application of a learning method to a system necessitates either knowledge of a system model, or measurements of system outputs signals. In most cases, knowledge of the system model is a cumbersome task, therefore, the most well-known learning procedures use signal measurements.

C. Neurofuzzy Design

In many real world systems, the available expert knowledge and its formulation into strict $if-then$ statements used in FLC schemes, is substantially affected by different system disturbances and parameter uncertainties. The ability of NNs to effectively approximate uncertain nonlinear procedures, provides a unique tool which can be incorporated into the inference engine of an FLC with aim to execute the steps of fuzzification, defuzzification and fuzzy reasoning in a self-corrected manner. Exploiting, therefore, the capabilities and advantages of both the FLCs and NNs the implementation of neurofuzzy controllers results.

III. Neurofuzzy Controller Design for the EV System

In the present work, our main goal is to create suitable controller schemes that ensure the electromechanical system of an EV to converge to the desired operating point. Therefore, nonlinearities in the model, as well as uncertainties on the system parameters point out the usefulness of applying neurofuzzy controllers.

A. EV System Model

An EV consists of different types of power devices, such as the electric storage system of batteries, the controlled power electronic devices and the motor [16]. In the present case, lithium-ion batteries are considered, while the torque on the car wheels is provided by a permanent magnet synchronous machine. A dc/dc bidirectional boost converter feeds or injects power from the battery and a three phase voltage source converter is used to drive the motor. The mathematical representation of the EV system in the $d-q$

synchronously rotating reference frame is described by the following nonlinear model (1)-(7) [16].

$$L_{ds}\dot{I}_{ds} = -r_s I_{ds} + p\omega_r I_{qs} L_{qs} + v_{ds} \quad (1)$$

$$L_{qs}\dot{I}_{qs} = -r_s I_{qs} - p\omega_r I_{ds} L_{ds} - p\omega_r \psi_m + v_{qs} \quad (2)$$

$$J\dot{\omega}_r = -b\omega_r + \frac{3}{2}p(L_{ds} - L_{qs})I_{ds}I_{qs} + \frac{3}{2}p\psi_m I_{qs} - T_m \quad (3)$$

$$C\dot{V}_{dc}V_{dc} = -\frac{V_{dc}^2}{R_{dc}} - \frac{3}{2}(v_{ds}I_{ds} + v_{qs}I_{qs}) + v_{bat}I_{bat} \quad (4)$$

$$L_{bat}\dot{I}_{bat} = -R_{ser}I_{bat} - V_S - V_L - v_{bat} + V_0 \quad (5)$$

$$C_S\dot{V}_S = -\frac{V_S}{R_S} + I_{bat} \quad (6)$$

$$C_L\dot{V}_L = -\frac{V_L}{R_L} + I_{bat} \quad (7)$$

where I_{ds}, I_{qs} are the $d-$ and $q-$ axis stator current components, ω_r is the rotor mechanical angular frequency, and T_m represents the external mechanical torque at the machine rotor. Additionally, V_{dc} represents the dc-link voltage, I_{bat} is the output of the total battery array, and V_S and V_L are the battery voltage drops. The control input of the battery-side boost converter is denoted by v_{bat}, while v_{ds} and v_{qs} represent the control inputs of the motor, that actually stand for the $d-$ and $q-$ motor voltage components. All other system parameters are summarized in Table 1.

TABLE I

SYSTEM PARAMETERS

r_s	Stator resistance	0.005582 Ω
L_{ds}	d-axis stator inductance	17.3 μH
L_{qs}	q-axis stator inductance	17.3 μH
J	Moment of inertia	1.8 kgm^2
b	Viscous friction coefficient	0.003852 $\frac{Nmsec}{rad}$
ψ_m	Permanent magnet flux	0.0175 Wb
C	Capacitance of the dc-link	640 μF
L_{bat}	Inductance of the battery boost converter	200 mH
R_{ser}	Series output battery resistance	0.0745 Ω
R_S	Short term battery resistance component	0.0467 Ω
C_S	Short term battery capacitance component	703.6 H
R_L	Long term battery resistance component	0.0498 Ω
C_L	Long term battery capacitance component	4475 H

B. Design of Neurofuzzy Controllers

Three neurofuzzy controllers are implemented, one for each control input v_{ds}, v_{qs}, v_{bat}. Each controller has two inputs defined as the error e_i and the change in error de_i and one output v_i, where i stands for $ds-$, $qs-$, and $bat-$, corresponding to the three control inputs. Specifically, the first NFC provides the ac/dc VSC $d-$axis voltage component, v_{ds}, using as input the $d-$axis current difference from its reference value $I_{ds,ref}$. The latter is considered zero in order to achieve maximum torque (perpendicular rotor and stator fluxes) [17]. The second one provides the ac/dc VSC $q-$axis voltage component, v_{qs}, using as input the rotor angular velocity difference from its reference value $\omega_{r,ref}$, i.e. a signal that is manually provided by the car driver. The last NFC provides the bidirectional dc/dc boost converter dc voltage component, v_{bat}, using as input the dc-link voltage difference from its reference value $V_{dc,ref}$, which is assumed as predefined constant.

The main architecture of the implemented neurofuzzy controllers is depicted in Fig. 2. As it is easily seen, one can divide the nodes [18, 19] based on their function into the layers [20].

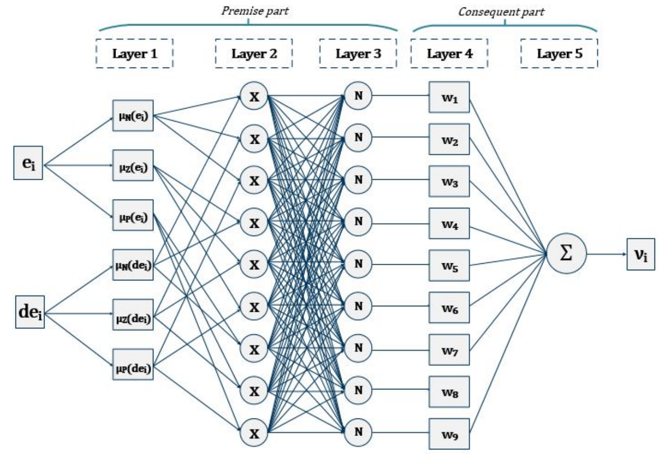

Fig. 2. The neuro-fuzzy controller architecture

- *Layer 1* performs the fuzzification of the neural network inputs. Actually, every node in this layer has a node function of the form $func_1 = \mu_j(x)$, where x is the input of the node and j corresponds to the different linguistic labels (negative (N), zero (Z), positive (P)). Therefore, $func_1$ is the membership function of j that specifies the degree to which the given x satisfies the quantifier j. In most cases, $\mu_j(x)$ is a bell shaped function, expressed as $\mu_j(x) = exp\{-(\frac{x-b_{x,j}}{a_{x,j}})^2\}$, where $\{a_{x,j}, b_{x,j}\}$ are premise parameters.
- In *layer 2* firing strengths of the ensemble of the implemented *if-then* rules are calculated. Every node multiplies the incoming signals (application of a T-norm operator) and the outputs u_k represent the firing strength of the corresponding rules, where $k = 1, ..., m$ stands for the number of implemented rules.
- Every node in *layer 3* calculates the normalized firing strengths; i.e. the ratio of each rule firing strength to the sum of all rules firing strength, $\bar{u}_k = \frac{u_k}{\sum_{k=1}^{m} u_k}$.
- Every node in *layer 4* has a function of the form $func_4 = \bar{u}_k w_k$, where w_k are consequent parameters. The values of the aforementioned parameters w_k are derived from the learning procedure provided in the following.
- Finally, the node in *layer 5* computes the overall output as the summation of all incoming signals, $func_5 = \sum_i \bar{u}_k w_k = \bar{\mathbf{u}}^T \mathbf{w}$, where $\bar{\mathbf{u}}^T \mathbf{w}$ is the vector representation of the aforementioned equation.

Therefore, for each of the three control inputs we have constructed an adaptive network which is functionally equivalent to a FLC with nine rules as shown in Fig. 3. Three membership functions are associated with each input, so the input space is partitioned into nine fuzzy subspaces, each governed by a fuzzy $if-then$ rule. The premise part of a rule defines a fuzzy subspace, while the consequent part specifies the output within this fuzzy subspace [20].

The premise parameters of the proposed NFC schemes $a_{x,j}$, $b_{x,j}$ are assumed fixed and are selected based on the *a priori* knowledge of the system. Conversely, the consequent parameter matrices $\mathbf{w}_i$ are updated via a gradient learning algorithm that takes into account the given training data set.

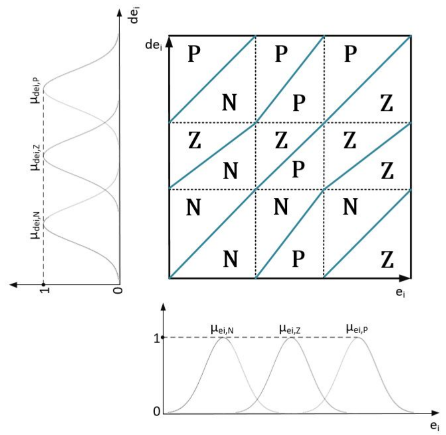

Fig. 3. Corresponding fuzzy subspaces of the implemented neurofuzzy controller

Specifically, the proposed learning procedure is a variant of the widely known back propagation algorithm [21, 22], that tries to improve the performance of the neural network by reducing the total error via altering the weights along its gradient. The solution to the aforementioned parameter estimation problem consists of finding the extremum of a cost function V, that quantifies the error that it is desired to be minimized.

In our case, the proposed cost function is selected as

$$V_i = \frac{1}{2}d_i^2 + \frac{1}{2}s_i^2 \tag{8}$$

In (8) $s_i = v_i - v_{i,ref}$ corresponds to the error of each neural network output with the provided from the given training data set reference value. Also, $d_i = x_i - x_{i,ref}$ stands for the error of each directly controlled input current I_{ds}, I_{qs} and I_{bat} with its reference value, instead of the real error inputs of the implemented NFCs. This substitution is feasible since a) in accordance to the dynamic equation (1), v_{ds} regulates directly the state I_{ds}, b) the dynamic equation (2) indicates that v_{qs} actually regulates the error $\omega_r - \omega_{r,ref}$ via the state I_{qs}, and, c) in a similar manner, (5) indicates that v_{bat} regulates the error $V_{dc} - V_{dc,ref}$ via the state I_{bat}. It is apparent that except from I_{ds}, which takes as reference command the zero signal, also I_{qs} take its reference value as determined by $I_{qs,ref} = \frac{2J}{3p^2\psi_m}(\frac{\beta}{J}\omega_{r,ref} - T_m)$, and consequently, I_{bat} by $I_{bat,ref} = \frac{V_{dc,ref}}{R_{dc}v_{bat}} + \frac{3}{2v_{bat}}v_{qs}I_{qs,ref}$, which correspond to the desired in steady state $\omega_{r,ref}$ and $V_{dc,ref}$, respectively.

A necessary condition that ensures the minimization of the cost function V_i is to force its derivative with respect to all different consequent parameter matrices $\mathbf{w}_i$ to be zero.

$$\frac{\partial V_i}{\partial \mathbf{w}_i} = d_i\frac{\partial d_i}{\partial \mathbf{w}_i} + s_i\frac{\partial s_i}{\partial \mathbf{w}_i} = 0$$
$$\frac{\partial V_i}{\partial \mathbf{w}_i} = (x_i - x_{i,ref})\frac{\partial x_i}{\partial \mathbf{w}_i} + (v_i - v_{i,ref})\frac{\partial v_i}{\partial \mathbf{w}_i} = 0$$

After lengthy manipulations, the following learning rules are attained

$$\dot{\mathbf{w}}_{ds} = -\gamma_{ds}[(I_{ds} - I_{ds,ref}) + (v_{ds} - v_{ds,ref})]\bar{\mathbf{u}}_{ds} \tag{9}$$
$$\dot{\mathbf{w}}_{qs} = -\gamma_{qs}[(I_{qs} - I_{qs,ref}) + (v_{qs} - v_{qs,ref})]\bar{\mathbf{u}}_{qs} \tag{10}$$
$$\dot{\mathbf{w}}_{bat} = -\gamma_{bat}[(I_{bat} - I_{bat,ref}) + (v_{bat} - v_{bat,ref})]\bar{\mathbf{u}}_{bat} \tag{11}$$

where γ_{ds}, γ_{qs}, and γ_{bat} are predefined positive constants that affect the algorithms speed of convergence.

IV. Stability analysis

As mentioned above, the main drawback of NFCs are the weekness of proving stability. However, it is adequate for the closed-loop system analysis both the system states and the weight parameters $\mathbf{w}_i$, seen as states, to be stable. To this end, taking into account both the system (1)-(7) and the controller weight components (9)-(11) in a common form, a system of the form $\dot{x} = f(x, v_c, v_d)$ is obtained, where $v_c = [v_{ds}\ v_{qs}\ v_{bat}]^T$ represent the controlled inputs, while $v_d = [0\ 0\ -T_m\ V_0\ 0\ 0\ \gamma(I_{ds,ref} + v_{ds,ref}\bar{\mathbf{u}}_{ds})\ \gamma(I_{qs,ref} + v_{qs,ref}\bar{\mathbf{u}}_{qs})\ \gamma(I_{bat,ref} + v_{bat,ref}\bar{\mathbf{u}}_{bat})\ 0]^T$ are considered as disturbance.

To proceed with the stability analysis, as a first step we are going to prove the input-to-state stability property [23] for the closed loop system. Based on [Theorem 4.8, [24]], the construction of a suitable Lyapunov function is of great importance. In our case, the following Lyapunov function is selected.

$$\begin{aligned} V = &\frac{3}{4}\gamma_{ds}L_{ds}I_{ds}^2 + \frac{3}{4}\gamma_{qs}L_{qs}I_{qs}^2 + \frac{1}{2}J\omega_r^2 \\ &+ \frac{1}{2}\gamma_{bat}L_{bat}I_{bat}^2 + \frac{1}{2}C_S V_S^2 + \frac{1}{2}C_L V_L^2 \\ &+ \frac{3}{4}\mathbf{w}_{ds}^T\mathbf{w}_{ds} + \frac{3}{4}\mathbf{w}_{qs}^T\mathbf{w}_{qs} + \frac{1}{4}\mathbf{w}_{bat}^T\mathbf{w}_{bat} + \frac{1}{2}\gamma_{ds}CV_{dc}^2 \end{aligned}$$

Taking into account that $\gamma_{ds} = \gamma_{qs} = \gamma_{bat} = \gamma$, the derivative of the aforementioned storage function after lengthy manipulations is obtained as follows

$$\begin{aligned} \dot{V} = &-\frac{3}{2}\gamma r_s I_{ds}^2 - \frac{3}{2}\gamma r_s I_{qs}^2 - \beta\omega_r^2 - \gamma R_{ser}I_{bat}^2 \\ &- \frac{1}{R_S}V_S^2 - \frac{1}{R_L}V_L^2 - \frac{3}{2}\gamma\mathbf{w}_{ds}^T\bar{\mathbf{u}}_{ds}^T\bar{\mathbf{u}}_{ds}\mathbf{w}_{ds} \\ &- \frac{3}{2}\gamma\mathbf{w}_{qs}^T\bar{\mathbf{u}}_{qs}^T\bar{\mathbf{u}}_{qs}\mathbf{w}_{qs} - \frac{1}{2}\gamma\mathbf{w}_{bat}^T\bar{\mathbf{u}}_{bat}^T\bar{\mathbf{u}}_{bat}\mathbf{w}_{bat} \\ &- \gamma\frac{1}{R_{dc}}V_{dc}^2 + \frac{3}{2}\gamma T_m - \gamma V_0 + \frac{3}{2}\gamma I_{ds,ref} \\ &+ \frac{3}{2}\gamma v_{ds,ref}\bar{\mathbf{u}}_{ds} + \frac{3}{2}\gamma v_{qs,ref}\bar{\mathbf{u}}_{qs} + \frac{1}{2}\gamma v_{bat,ref}\bar{\mathbf{u}}_{bat} \end{aligned}$$

The latter can be written as

$$\dot{V} = -x^T R x + y^T v_d \tag{12}$$

where R stands for the positive definite diagonal matrix, given by (13)

$$\begin{aligned} R = diag\{&\frac{3}{2}\gamma r_s, \frac{3}{2}\gamma r_s, \beta, \gamma R_{ser}, \frac{1}{R_S}, \frac{1}{R_L}, \\ &\frac{3}{2}\gamma\bar{\mathbf{u}}_{ds}^T\bar{\mathbf{u}}_{ds}, \frac{3}{2}\gamma\bar{\mathbf{u}}_{qs}^T\bar{\mathbf{u}}_{qs}, \frac{3}{2}\gamma\bar{\mathbf{u}}_{bat}^T\bar{\mathbf{u}}_{bat}, \frac{1}{R_{dc}}\} \end{aligned} \tag{13}$$

and y represents the output vector

$$y = \begin{bmatrix} I_{ds} & I_{qs} & \omega_r & I_{bat} & \mathbf{w}_{ds} & \mathbf{w}_{qs} & \mathbf{w}_{bat} \end{bmatrix}^T$$

or in a matrix form

$$y = Cx \tag{14}$$

with $$C = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \end{bmatrix}$$

Note that from (12) and (13) one can recognize the well-known [24] strict passivity condition $\dot{V} \leq y^T u_d$. Additionally, based on the properties of a positive definite matrix, (12) provides the following inequality

$$\dot{V} \leq -\lambda_{min}\{R\}||x||^2 + ||x||||C^T v_d|| \tag{15}$$

where $\lambda_{min}\{R\}$ is the smallest eigenvalue of R.

Then to use the term $-\lambda_{min}\{R\}||x||^2$ to dominate $||x||||C^T v_d||$ we can introduce a positive scalar: $0 < \theta < 1$, such that

$$\dot{V} \leq -(1-\theta)\lambda_{min}\{R\}||x||^2 - \theta\lambda_{min}\{R\}||x||^2 + ||x||||C^T v_d|| \tag{16}$$

and taking into account that $||C^T v_d|| = ||v_d||$, we can finally obtain

$$\dot{V} \leq -(1-\theta)\lambda_{min}\{R\}||x||^2, \quad \forall ||x|| \geq \frac{||v_d||}{\theta\lambda_{min}\{R\}} \tag{17}$$

Therefore, system $\dot{x} = f(x, v_c, v_d)$, driven by the proposed NFCs is input-to-state stability and its states will be ultimately bounded by a class $\mathscr{K}$ function $\rho(v_d) = ||v_d||/\theta\lambda_{min}R$, [24].

Input-to-state stability property ensures that all the system states are bounded for bounded inputs (bounded input bounded state - BIBS). Therefore, under the assumption of the existence of a unique equilibrium and based on the BIBS and passivity properties and the relative theory [25], it is shown that there exists a continuous, differentiable, bounded, non-increasing storage function suitable to prove convergence to that equilibrium.

V. Simulation Results

In order to verify the effectiveness and the robustness of the developed NFCs, several simulations are conducted. In the aforementioned simulations the unknown external torque T_m is assumed piece-wise constant as Fig. 4 depicts, while the internal battery voltage is considered constant at $V_0 = 840$ V. The resultant control signals v_{ds}, v_{qs} and v_{bat} from the three implemented controllers are depicted in Fig. 5, 6 and 7, respectively. The recovering features of the proposed controllers to the imposed load torque variations are presented in Fig. 8, 9, 10, 11 and 12. Specifically, Fig. 8 provides the mechanical angular velocity ω_r, which effectively tracks its reference value $\omega_{r,ref}$, while Fig. 9 presents the dc-link voltage regulation to its constant reference value $V_{dc,ref} = 900$ V, despite torque variations (disturbance). Additionally, Fig. 10 depicts the $d-$axis current, which obviously is regulated to its reference value, $I_{ds,ref} = 0$ A, and therefore motor optimum torque operation is achieved. Based on the aforementioned results, all the proposed controllers achieve their design goals.

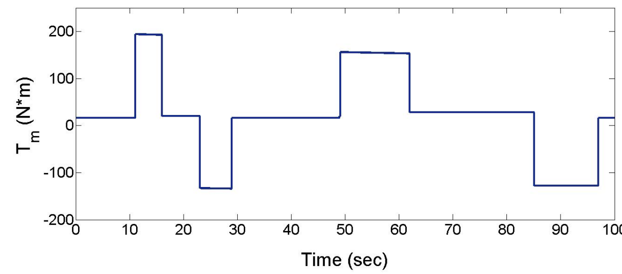

Fig. 4. External mechanical torque T_m

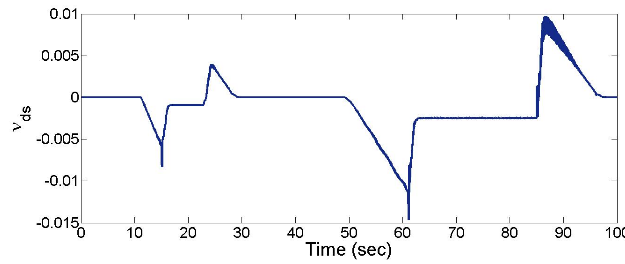

Fig. 5. d-axis stator voltage component v_{ds}

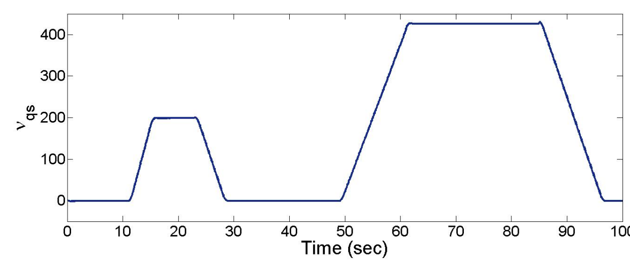

Fig. 6. q-axis stator voltage component v_{qs}

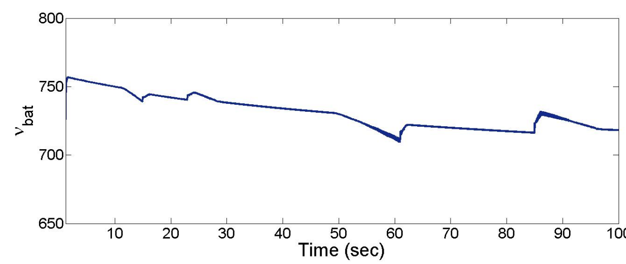

Fig. 7. Battery-side boost converter voltage component v_{bat}

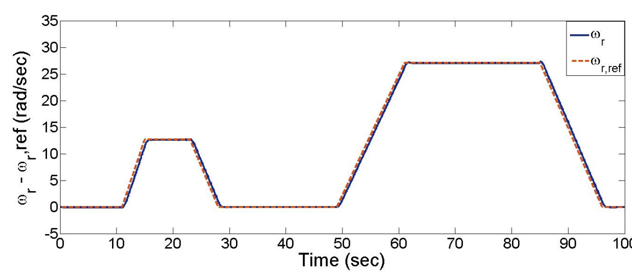

Fig. 8. Mechanical angular frequency vs its reference $\omega_r - \omega_{r,ref}$

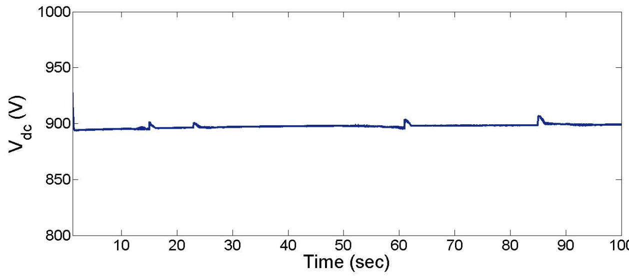

Fig. 9. Dc-link voltage V_{dc}

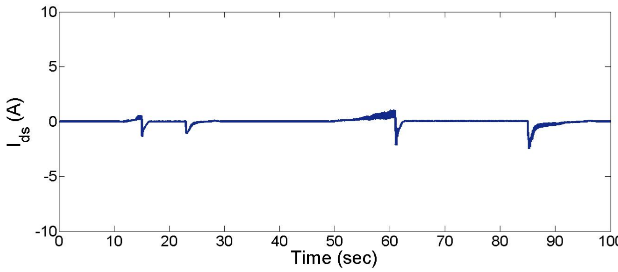

Fig. 10. d-axis stator current I_{ds}

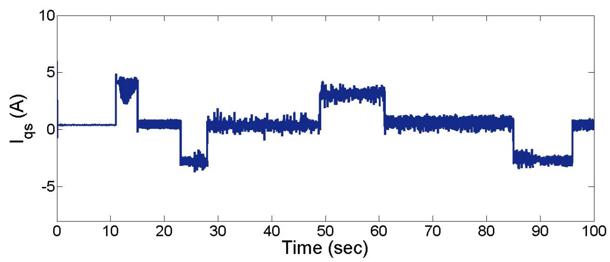

Fig. 11. q-axis stator current I_{qs}

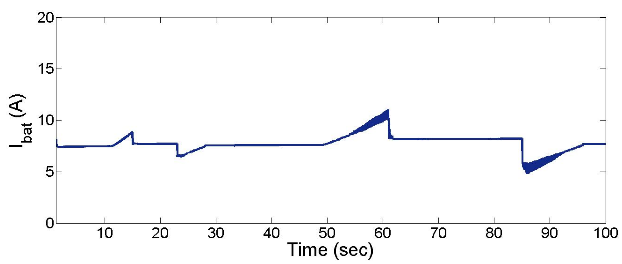

Fig. 12. Battery current I_{bat}

Another important result is depicted in Fig. 11, and 12. In particular, it is shown that the $q-$axis and the battery currents I_{qs}, I_{bat}, effectively track their reference values to a suitable equilibrium that change in order to follow the load variations and the mechanical angular velocity, and to maintain the dc voltage constant, by injecting/extracting the necessary power to/from the battery, respectively. In all cases, the system desired operation is achieved.

VI. Conclusions

NFC schemes are implemented and applied on the complex electromechanical EV structure, in an innovative manner. This permits the resulting closed-loop system to be analysed for its stability via a Lyapunov-based method, that exploits input-to-state stability property to further conclude convergence to the desired equilibrium. Simulation results establish the NFCs effective performance and confirm system stability with smooth transient response.

References

[1] R. Sioshansi, and P. Denholm, ”Emissions Impacts and Benefits of Plug-In Hybrid Electric Vehicles and Vehicle-to-Grid Services,” *Journal on Environmental Science & Technology*, vol. 43, no. 4, pp. 1199-1204, Jan. 2009.

[2] S.M. Lukic, and A. Emadi, ”Effects of drivetrain hybridization on fuel economy and dynamic performance of parallel hybrid electric vehicles,” *IEEE Trans. on Vehicular Technology*, vol. 53, no. 2, pp. 385 - 389, Mar. 2004.

[3] M. Ehsan , S.E. Gao, and A. Emadi, *Modern Electric, Hybrid Electric, and Fuel Cell Vehicles - Fundamentals, Theory, and Design*, first edn., CRC Press LLC, 2005.

[4] S. Soylu (editor), *Electric Vehicles - Modelling and Simulations*, InTech, Sept. 2011.

[5] V.I. Utkin, ”Sliding mode control design principles and applications to electric drives,” *IEEE Trans. on Industrial Electronics*, vol. 40, no. 1, pp. 23 - 36, Feb. 1993.

[6] S.G. Wirasingha, and A. Emadi, ”Classification and Review of Control Strategies for Plug-In Hybrid Electric Vehicles”, *IEEE Trans. on Vehicular Technology*, vol. 60, no. 1, pp. 111 - 122, Jan. 2011.

[7] H.D. Lee, S.J. Kang, and S.K. Sul, ”Efficiency-optimized direct torque control of synchronous reluctance motor using feedback linearization,” *IEEE Trans. on Industrial Electronics*, vol. 46, no. 1, pp. 192 - 198, Feb. 1999.

[8] L. Feiqiang, W. Jun, and L. Zhaodu, ”Motor torque based vehicle stability control for four-wheel-drive electric vehicle,” *IEEE Vehicle Power and Propulsion Conference (VPPC 09)*, pp. 1596 - 1601, Sept. 2009.

[9] J.J. Makrygiorgou, and A.T. Alexandridis, ”Fuzzy Logic Control of Electric Vehicles: Design and Analysis Concepts,” *IEEE Proc. on Ecological Vehicles and Renewable Energies Conference (EVER)*, pp. 1-6, Apr. 2017.

[10] R.E. King, *Computational Intelligence in Control Engineering*, 1st edn., CRC Press, 1999.

[11] P.C. Sen, ”Electric motor drives and control-past, present, and future,” *IEEE Trans. on Industrial Electronics*, vol. 37, no. 6, pp. 562 - 575, Dec 1990.

[12] R. Reznik, *Fuzzy Controllers*, 1st edn., Newnes, 1997.

[13] K.M. Passino, *Fuzzy Control*, 1st edn., Addison Wesley Longman, Inc., 1998.

[14] S.A. Kalogirou, ”Applications of artificial neural-networks for energy systems,” *Journal on Applied Energy*, vol. 67, no. 12, pp. 17-35, Sept. 2000.

[15] R. Rojas, *Neural Networks: A Systematic Introduction*, 1st edn., Springer-Verlag Berlin Heidelberg, 1996.

[16] J.J. Makrygiorgou, and A.T. Alexandridis, ”Nonlinear Dynamic Modeling and Stability Analysis of Electric Vehicles,” *IEEE Proc. on American Control Conference (ACC)*, pp. 643-648, Boston, Jul. 2016.

[17] A.T Alexandridis, G.C. Konstantopoulos and Q.C. Zhong, Advanced integrated modeling and analysis for adjustable speed drives of induction motors operating with minimum losses, *IEEE Trans. on Energy Conversion*, vol. 30, no. 3, pp. 1237-1246, Sept. 2015.

[18] J. Godjevac, and N. Steele, ”Neuro-fuzzy control of a mobile robot,” *IEEE Trans. on Neurocomputing*, vol. 28, no. 13, pp. 127-143, Oct. 1999.

[19] J.S.R. Jang, ”ANFIS: Adaptive-Network-Based Fuzzy Inference System,” *IEEE Trans. on Systems, Man, and Cybernetics*, vol. 23, no. 3, pp. 665 685, Jun. 1993.

[20] C.T. Lin, and C.S.G. Lee, ”Neural-network-based fuzzy logic control and decision system,” *IEEE Trans. on Computers*, vol. 40, no. 12, pp. 1320 - 1336, Dec. 1991.

[21] Y. Chauvin, D.E. Rumelhart (editors), *Backpropagation: Theory, Architectures, and Applications*, 1st edn., Hillsdale, N.J. : Lawrence Erlbaum Associates, 1995.

[22] S.I. Horikawa, T. Furuhashi, and Y. Uchikawa, ”On fuzzy modeling using fuzzy neural networks with the back-propagation algorithm,” *IEEE Trans. on Neural Networks*, vol. 3, no. 5, pp. 801 - 806, Sep. 1992.

[23] E.D. Sontag, and Y. Wang, On characterizations of the input-to-state stability property, *System & Control Letters*, vol. 24, no. 5, pp. 351-359, April 1995.

[24] H.K. Khalil, *Nonlinear Systems*, 3rd edn., Upper Saddle River, NJ:Prentice-Hall, 2002.

[25] G.C. Konstantopoulos and A.T. Alexandridis, Generalized nonlinear stabilizing controllers for Hamiltonian-passive systems with switching devices, *IEEE Trans. on Control System Technology*, vol. 21, no. 4, pp.1479-1488, July 2013.