Vodovozov, Valery; Aksjonov, Andrei; Petlenkov, Eduard; Raud, Zoja

Neural network-based model reference control of braking electric vehicles

Published in:
Energies

DOI:
10.3390/en14092373

Published: 22/04/2021

Document Version
Publisher's PDF, also known as Version of record

Published under the following license:
CC BY

Please cite the original version:
Neural Network-Based Model Reference Control of Braking Electric Vehicles

Valery Vodovozov 1,*, Andrei Aksjonov 2, Eduard Petlenkov 1 and Zoja Raud 1

1 Department of Electrical Engineering, Tallinn University of Technology, 19086 Tallinn, Estonia; eduard.petlenkov@ttu.ee (E.P.); learnelectronics@narod.ru (Z.R.)
2 Electrical Engineering and Automation, Aalto University, FI-00076 Aalto, Finland; andrei.aksjonov@aalto.fi
* Correspondence: valery.vodovozov@gmail.com

Abstract: The problem of energy recovery in braking of an electric vehicle is solved here, which ensures high quality blended deceleration using electrical and friction brakes. A model reference controller is offered, capable to meet the conflicting requirements of intensive and gradual braking scenarios at changing road surfaces. In this study, the neural network controller provides torque gradient control without a tire model, resulting in the return of maximal energy to the hybrid energy storage during braking. The torque allocation algorithm determines how to share the driver’s request between the friction and electrical brakes in such a way as to enable regeneration for all braking modes, except when the battery state of charge and voltage levels are saturated, and a solo friction brake has to be used. The simulation demonstrates the effectiveness of the proposed coupled two-layer neural network capable of capturing various dynamic behaviors that could not be included in the simplified physics-based model. A comparison of the simulation and experimental results demonstrates that the velocity, slip, and torque responses confirm the proper car performance, while the system successfully copes with the strong nonlinearity and instability of the vehicle dynamics.

Keywords: electric vehicle; model reference controller; neural network; energy recovery; braking

1. Introduction

A common paradox of contemporary energy converters is that while their specific performance improves with each passing year, the overall conversion performance is getting worse. Transportation is one of the areas affected by this regression. The average efficiency of vehicle engines increases annually. However, rising car ownership, heavier vehicles, much more frequent trips, and longer distances travelled per capita have driven the sector’s efficiency down from 26% to 21% over the past 70 years [1].

Since 20 to 70% of the vehicle energy is lost during deceleration [2,3], the problem of braking energy recovery is currently of great importance for improving the situation. A growth of competence in the management of braking road vehicles remarkably affects their energy efficiency [4]. Research in the field of both gradual braking modes and intensive antilock braking systems (ABS) have significantly been boosted in recent decades, filling up multiple books, reports, and research papers [5]. The first field relates to downhill driving, deceleration before traffic lights, parking, and similar regimes, where the driver reduces the velocity of the car in a stable and smooth manner. In this area, many interesting and useful electric vehicle (EV) energy saving solutions have been found [6,7]. In the second case, the driver tries to stop the vehicle abruptly, often without taking into account specific environmental conditions such as slippery surface, road slope, wind strength, cornering, etc. Very little research on energy saving under heavy braking can be found [8]. Even less attention is paid to energy recycling solutions that are so versatile that they could be equally successful in both braking scenarios [9].

The challenge lies in the contradiction between the specific energy needs that arise during an intensive stop and a stop with recuperation. Since the deceleration is commonly
great in the former case, the friction (hydraulic) brakes (FB) are used here, which convert
the friction force into heat. In the latter case, the energy is directed to regenerative electrical
braking (EB), restricted in terms of the state of charge (SOC), voltage, and current. To
resolve this contradiction, blended braking systems are designed that combine traditional
FB and EB, being associated with hybrid energy storage (HES) equipment composed of
batteries on one side and ultracapacitors or/and flywheels on the other [10]. They help
reclaim energy loss in braking, reduce car maintenance costs and tire particle emission, and
extend driving range and time. These benefits are based on the main advantages of EVs in
the sense of braking management, such as fast and accurate torque generation by electrical
motors, easy torque measurement by current sensing, precise wheel speed encoding, and
the ability to adjust each wheel independently due to the small motor dimensions [11].
However, in most existing braking control strategies, regenerative braking force is only
a small fraction of the total braking effort, which contributes to safe braking but leads to
poor braking economics [12].

With both the EB and FB, the braking strength is to be additionally limited to avoid
skidding due to tire-road friction weakening. At present, there are no affordable sensors to
identify nonlinear and nonstationary tire-road friction and make these data available to a
braking controller [13]. Despite numerous attempts, an accurate and general mathematical
model of tire behavior has not been obtained. Therefore, some indirect estimates are needed
that look problematic in general and complicated by the fact that tire properties depend on
variable road conditions and many other features.

In contrast to friction, wheel slip can be calculated easily as follows:

$$
\lambda = \frac{v - \omega r}{v}
$$  \hspace{1cm} (1)

where \(v\)—longitudinal vehicle velocity, \(\omega\)—angular wheel speed, and \(r\)—effective radius of
the wheel. EVs usually utilize wheel-based sensors to measure the angular speed and/or
angular acceleration of the wheel. In [14], the slip is derived based on velocity sensor
signals and vehicle geometry. In [15], a perturbed sliding mode observer is used. A number
of different techniques have been proposed as well to estimate EV velocity [16].

Several methods are available in the literature to transfer from the slip to friction; they
include Pacejka’s “Magic Formula”, Burckhardt model, Rill model, and others [11,17,18].
Nevertheless, there exist diverse views on how to apply the obtained friction-slip estimates
to ensure both the requested stopping rate and maximal energy recovery, taking into
account that friction peak optimality for a given road surface is variable. In particular,
in [9], the highest friction is developed at the slip level of about 12% on the dry road and at
5% on the icy road. On the contrary, in [16], the biggest friction appeared at 20% slip for
dry road and 30% for icy road. Usually, the location of these peaks depends on the initial
car velocity and alternates for forward-rear and right-left wheels. Hence, to avoid skidding,
the designers commonly offer to choose some understated slip level, say 10% [16]. As
a result, the vehicle does not decelerate fast enough and does not save as much energy
as desired.

To exclude the above vagueness, in [9], the torque gradient control method is designed,
where the derivative of the application torque \(T\) with respect to slip \(\lambda\), \(\frac{dT}{d\lambda}\), is used as a
control feedback instead of the slip applied in conventional intelligent braking ABS, such
as [16]. The main benefit of this method is that it does not require any tire model for
implementation. Therefore, this approach is further elaborated in this paper.

Aside from the difficulty in estimating such a critical parameter as the friction factor,
another challenge is to develop a model of the entire vehicle, whose equations of motion
are highly nonlinear and many factors are often unknown. Commonly, models with 2, 4,
and even more degrees of freedom (DOF) are used. The simplest of these is the quarter-
car model [19], which has two DOFs. In the 4-DOF half-car model, the vehicle body is
described by a displacement in the vertical direction, one rotation, and one DOF for each
wheel vertical displacement. The 14-DOF model involves the longitudinal, lateral, and
vertical motions, pitch, roll, and yaw rotations, vertical motions of unsprung mass, and rotations of wheels [20]. The 38-DOF model [21] consists of 6 DOFs of vehicle and 8 DOFs for each wheel. In [22], the vertical and longitudinal vehicle dynamics are taken into account. In all models, the designer has to choose the appropriate level of fidelity, deciding whether or not to include many other effects, such as weight transfer across tires due to acceleration, lag in tire force generation with rapid steering movement, etc. [16,23].

After the development of the vehicle model, the controller is mastered based on data on the driving conditions and vehicle braking capabilities [17].

The main contender to control the gradual braking is a proportional-integral-differential controller (PIDC [24,25]. Classical PIDCs are easy to implement, and sufficient tuning rules are available to achieve a compromise in approaching rapid response and small overshoot [26], based on the famous Ziegler-Nichols and Refined Ziegler-Nichols formulas. Regrettably, when the vehicle performs differently in different operating modes, its parameters cannot be fixed and need to be modified systematically. It is not only a waste of time, but also difficult to implement [27]. The PIDC cannot yield a fine performance in highly order, nonlinear, multi-input, and multi-output environment [28]. As well, they are ineffective and often fail in ABS and braking processes with relatively long dead time. For this reason, for instance, the PIDC, designed in [25], cannot optimize energy recycling due to its orientation to a fixed driving medium. The same can be said about solutions offered in [29–32] that relate not only to the PIDC, but also to the sliding mode [33,34] and the feedback linearization controllers [35].

As distinct from the PIDC, intelligent control equipment has proven to be an emerging and effective solution. This applies to fuzzy logic controllers (FLC), model predictive, genetic algorithm, and neural network (NN) based controllers (NNC) capable to deal successfully with nonlinear, uncertain, and varying braking dynamics. Because of this, the scope of application of intelligent vehicles with highly nonlinear dynamics, driven upon the changeable road pavements, is growing annually [10,16].

The FLCs refer to rule-based controllers. The rule-based approach is considered as a conventional way of energy management; therefore, it is widely used by many vehicle manufacturers, such as Honda® and Toyota® [13,36]. Control here is defined based on a set of “if-then” rules that depend on human expertise, operation boundaries, and safety considerations. The advantages of these systems are their ease of computation and the simplicity in applying updates to commercially produced vehicles. A major drawback of the FLCs is that their quality is completely dependent on the professional level of a specialist who sets them up.

The NNCs can outperform human experts used in FLC, and may be realized more accurately and quickly [27]. The target for using the NNCs is to capture the accuracy of experimental data while saving computational time, so that NN systems are usually designed within reasonable timeframes. This is why different NN methodologies are applied in braking systems of automobiles [35] and aircraft [37].

The study [38] is devoted to the use of various NNs in changing driving scenarios, demonstrating the effectiveness of the NNC in extensive and smooth testing modes. In [39], a robust NNC guarantees stability against unknown uncertainties. Emergency braking on a variable-surface road is considered here by tracking the reference wheel slip in different maneuvers. Another paper [16] contributes to the development of an effective engineering solution aimed at improving control of ABS by estimating the friction coefficient with the help of video data. An emerging deep learning method is applied in [22] for the classification of six road surface types and condition combinations: wet gravel, dry gravel, wet cobblestone, dry cobblestone, wet asphalt, and dry asphalt. An experimental investigation of the designed convolutional NN was carried out here to parametrize the vehicle tire model. In [40], a recurrent NN has been proposed as a drive cycle recognizer and NNC coefficient tuner. This network tracks the last 200 s of computed speed data (average, maximum, minimum speed, and acceleration), which are characteristic values for a drive
cycle. The NN has 6 input neurons, 10 neurons with a sigmoid function in the hidden layer, and a single-neural output layer.

The paper [41] represents an interesting example of an intelligent blended braking system. The NN control problem is formulated there as the search for optimal policy in the Markov decision-making process, where the braking space is defined as a set of actions that include no braking, weak braking, average braking, and strong braking. The policy used to manage braking is implemented through computer simulation using deep reinforcement learning. Another intelligent braking system proposed in [12] considers two kinds of braking operations, namely, releasing of the accelerator pedal, and pressing the brake pedal. The radial basis function NN model is trained here through the optimal braking force, while the multiple correlation coefficient method is applied to analyze model errors against a sample database. Several standard test cycles, such as US06, UDDS, LA92, and ECE were used to test the system.

The NNCs are applied successfully in HES management to optimize energy recovery. For instance, a convolutional NN has been used in [42] for estimation of energy/power consumption of EV. In [43], a regenerative EB scheme is offered aiming to transfers braking energy to the HES devices. To this end, the multilayer feedforward NN provides satisfactory capability, comprising EV speed and SOC of the supercapacitor and battery banks in a number of braking situations. As well, [44] uses real data collected over two days of power consumption, trip time, and SOC as training inputs to the NN. The NN output specifies the EV recommended operation mode as a function of time, whereas typical peak loads and off-peak load times, human behavior, seasonal and weather conditions contribute to the model to generate a realistic pattern. Another example is the deep NN-based approach of EV energy demand estimation proposed in [23]. It is based on the time series of the driving cycle properly preprocessed and transformed into 1D or 2D maps to serve as a static input to the NN. Several deep feedforward NN architectures are considered for this application along with various input formats. In [45], the HES-based braking system implements automatic control of the EV regeneration torque of the motor to improve driver’s comfort and energy efficiency. To apply this system, an accurate prediction of the vehicle deceleration states was produced. The original EV deceleration model proposed in this paper is based on the deep NN consisting of a sequential recurrent NN with long-short-term memory and a two-layer conventional NN model.

One of the directions in implementation of the NNC into the on-board EV HES is focusing on determining the optimal coefficients of the classical PIDCs and their influence on the power distribution [46,47]. In [48], the NNC provides twice less transition time compared to the classical PIDC, along with a reduction of energy loss and motor overshoot. The proposed control method made it possible to automatically tune the PIDC, thus reducing the inrush currents and torque surges, and thereby prolonging the service life of vehicle mechanical components. In [49], an NNC replaces the PIDC to control the angular position of the wheel. This double-input NNC provides supervised learning, at which the dataset for training the NN is run before the simulation to get the output results. In [50], the PIDC with an NN uses three special types of neurons: P-neuron, I-neuron, and D-neuron that realize the three fundamental controlling actions. In [51], an adaptive NN-based PIDC is offered for the multi-input multi-output nonlinear vehicle system. Different combinations of NNC and FLC are proposed in [52] to control the EVs in blended braking. A neuro-fuzzy controller applied in [53] solves the torque distribution problem for regenerative braking of a hybrid bus. By merging the FLC and the backpropagation NNC, an algorithm for processing the bus velocity, wheel speed, and the brake pedal travel is implemented. A successful combination of the NNC, sliding-mode controller, and PIDC is proposed in [54]. To reflect the nonlinear time-varying characteristics of the longitudinal dynamics, the NNC is designed based on a sliding mode controller and a single-neuron PIDC, which provides deceleration in emergency braking conditions. Co-simulation utilizing CarSim® and Simulink® was carried out on an intelligent vehicle model. Here, the sum of wind and rolling resistances was taken into account.
An objective of the current research is to offer a new NN-based control system, which ensures high quality blended braking of EVs in both intensive (ABS-fed) and gradual braking scenarios with energy recovery, taking into account the changing road pavement. The main contribution of the paper is in the design of the NNC capable to meet the conflicting requirements of different braking modes and road surfaces. This is the first study where the NN provides the torque gradient control without a tire model resulting in the return of maximal energy to the vehicle HES during the braking process.

Using the author’s torque allocation algorithm, the system determines how to distribute a single driver’s torque request into separate requests between the FB and EB. Multiple simulations demonstrate effectiveness of the proposed NN-based braking system. The model states and inputs are employed here as a guide for developing a coupled two-layer NN. Furthermore, a simulation study demonstrates that the designed system can cover various dynamic behaviors that could not be included in a simplified physics-based model. An experimental part of the research proves the validity of the model and simulations.

In the next sections, an NN-based braking system and its parts are introduced. Then, two design phases of the NNC are explained, namely, identifying the NN model, which predicts the EV behavior, and training the NNC using the identified NN model. Once the control system is trained and validated, its responses are estimated against the various driver’s requests and environment conditions. To verify the effect of the proposed control strategy in the hardware environment, a set of experimentations is described. Finally, the results are discussed and conclusions are drawn.

2. Materials and Methods

2.1. Braking System

The braking system discussed in this paper (Figure 1) involves an NN-based Model Reference Controller (MRC), a torque allocator (TA) coupled to the physics-based model of an electric vehicle with blended braking (PBEV) supplied from a hybrid energy storage (HES), a reference system (REF), and computational modules that estimate wheel slip and torque gradient.
The REF block simulates a driver pressing or releasing the brake pedal with different forces. The MRC unit produces the actuating torque $T^*$, further divided by the TA ratio into two fractions: electrical $TE^*$ and friction $TF^*$, both entering the PBEV block. During braking, the HES block consumes the electrical fraction of the energy $kJ$ generated by the PBEV, while its value does not exceed the permissible SOC, voltage, and current restrictions. The application torque $T$ of the PBEV decelerates the EV at such an intensity as to meet the driver's reference $T^*$ on the one hand, and avoids wheel skidding caused by the wheel slip, on the other. The electrical current recharges the HES block from the EB, while the friction current recharges the HES block from the EB, while the pressure signal adjusts the FB. Braking will complete when the driver releases the pedal or the vehicle stops.

2.2. Physics-Based Vehicle Model

The physics-based vehicle model (PBEV) used in this study was initially established in [9] based on experimental data describing the interacted rotational properties and the physical features of the EV. In Figure 2, its Simulink composition is shown. It is made up of an adjustable electrical drive (eDrive) implementing a regenerative EB, a friction drive (fDrive) integrated with the FB, a special drive (Drive-U) responsible for direct torque control, energy recovery, and a Wheel unit, which simulates the vehicle inertia.

2.2. Physics-Based Vehicle Model

The physics-based Simulink model of the vehicle.

To ensure direct torque control with consideration of the system non-linearity and the space vector modulation, the AC6 Interior PM Synchronous Motor Drive block from the Specialized Power Systems/Motor Drive block from the Simulink/Specialized Power Systems MATLAB/Simulink library was used for the EB core. It is composed of the following four main parts: the electrical motor of 288 V, 100 kW, a three-phase voltage source inverter powered by a direct voltage from the HES, able to regenerate and supply, and energy recovery. Together with the Drive-I sub-block, it arranges a space vector modulation loop with a PI current regulator. The Drive-U sub-block is responsible for direct torque control, electrical power supply, and energy recovery. Together with the Drive-I sub-block, it arranges a torque stabilization loop with a PI current regulator. The Drive-U sub-block and the Wheel block belong to the speed loop activated in gradual deceleration and shorted in emergency braking. In the Wheel block (Figure 4), the static fraction of the application torque (Static $T$) is taken into account when the velocity drops to a low vehicle settling velocity level ($v_{home}$) and the EB becomes ineffective.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>$K_T$</td>
<td>1.5</td>
</tr>
<tr>
<td>$K_I$</td>
<td>1</td>
</tr>
<tr>
<td>$R$</td>
<td>10</td>
</tr>
<tr>
<td>$K_U$</td>
<td>500</td>
</tr>
<tr>
<td>$K_W$</td>
<td>1.5</td>
</tr>
</tbody>
</table>

Figure 3. The replacement circuit of the electrical drive used for NN learning.

Here, the Drive-U sub-block is responsible for direct torque control, electrical power supply, and energy recovery. Together with the Drive-I sub-block, it arranges a torque stabilization loop with a PI current regulator. The Drive-U sub-block and the Wheel block belong to the speed loop activated in gradual deceleration and shorted in emergency braking. In the Wheel block (Figure 4), the static fraction of the application torque (Static $T$) is taken into account when the velocity drops to a low vehicle settling velocity level ($v_{home}$) and the EB becomes ineffective.
Here, the Drive-U sub-block is responsible for direct torque control, electrical power supply, and energy recovery. Together with the Drive-I sub-block, it arranges a stabilization loop with a PI current regulator. The Drive-T sub-block and the Wheel block belong to the speed loop activated in gradual deceleration and shorted in emergency braking. In the Wheel block (Figure 4), the static fraction of the application torque (Static T) is taken into account when the velocity drops to a low vehicle settling velocity level (vhome) and the EB becomes ineffective.

Figure 3. The replacement circuit of the electrical drive used for NN learning.

Figure 4. The Wheel block of the PBEV.

The data used in the simulation are presented in Table 1.

Table 1. The data used in the simulation.

<table>
<thead>
<tr>
<th>Symbol</th>
<th>Parameter</th>
<th>Unit</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>J</td>
<td>Moment of inertia</td>
<td>kgm²</td>
<td>5.0</td>
</tr>
<tr>
<td>KE</td>
<td>Electromotive force factor</td>
<td>Vs/rad</td>
<td>10</td>
</tr>
<tr>
<td>KF</td>
<td>Gain of friction drive</td>
<td>A/Nm</td>
<td>1.0</td>
</tr>
<tr>
<td>KT</td>
<td>Magnetomotive force factor</td>
<td>V/A</td>
<td>500</td>
</tr>
<tr>
<td>KU</td>
<td>Current loop gain</td>
<td>Vs/Nm</td>
<td>500</td>
</tr>
<tr>
<td>KW</td>
<td>Gear ratio</td>
<td>rad/m</td>
<td>1.5</td>
</tr>
<tr>
<td>R</td>
<td>Resistance of motor winding</td>
<td>Ohm</td>
<td>10</td>
</tr>
<tr>
<td>T0</td>
<td>Static torque</td>
<td>Nm</td>
<td>1.8</td>
</tr>
<tr>
<td>TB *</td>
<td>Actuating torque</td>
<td>Nm</td>
<td>1000</td>
</tr>
<tr>
<td>TF</td>
<td>Time constant of friction drive</td>
<td>s</td>
<td>0.1</td>
</tr>
<tr>
<td>TI</td>
<td>Time constant of current sensor</td>
<td>s</td>
<td>0.01</td>
</tr>
<tr>
<td>TU</td>
<td>Time constant of HES</td>
<td>s</td>
<td>7.0</td>
</tr>
<tr>
<td>Ts</td>
<td>Sampling time</td>
<td>s</td>
<td>2 x 10⁻⁶</td>
</tr>
<tr>
<td>vhome</td>
<td>Vehicle settling velocity</td>
<td>m/s</td>
<td>10</td>
</tr>
<tr>
<td>v0</td>
<td>Initial vehicle velocity</td>
<td>m/s</td>
<td>100</td>
</tr>
</tbody>
</table>

2.3. Torque Allocation

The torque allocation module TA algorithmically distributes the actuating braking torque T*, generated at the MRC output, between the front and rear wheels in a fixed ratio [55] and allocates it between the FB and EB based on the real-time SOC, voltage, and permissible EB current.

To transfer the maximal fraction of the actuating torque T* to the EB, the ability of the electrical drive to develop sufficient power, voltage, and current to charge all energy storage devices of the HES is used. Meanwhile, in order to keep the battery and the ultracapacitor within their proper SOC borders, the torque allocation algorithm verifies whether the electrical current and motor torque meet the real-time HES restrictions.

In [9], an appropriate torque allocation algorithm was offered. There, when the control system recognizes the actuating torque request T*, the EB is activated, and either the ultracapacitor or the battery HES section runs. The FB torque does not appear until either
any of the SOC levels exceeds the allowed overcharging barriers or the electrical motor produces maximal power. As soon as the motor torque becomes insufficient, the system runs FB and EB together:

$$TP^* = T^* - TE^*$$

(2)

Only when both SOC levels exceed their boundaries, the solo FB is used due to the inability to regenerate. Therefore, a common trait of this strategy is to include regeneration in all braking scenarios, even during heavy braking with ABS, leaving the solo FB only if the HES is saturated.

2.4. Model Reference Controller

Using the data generated by the designed PBEV model, the NN-based MRC solves the problem of effective EV braking with maximal recovery of braking energy.

To design the NN, the Deep Learning Toolbox™ blockset from MATLAB/Simulink® was used in this study. An interactive development environment for the NN controllers was applied, namely the MRC block. In response to the reference in the form of the driver’s torque request $TB^*$, the MRC issues an actuating braking torque command $T^*$ to stop the PBEV without wheels locking under a variety of road conditions and EV velocities. The MRC includes two NNs, which are an NN controller (NNC) generating the actuating braking torque command and an NN model of the PBEV (NNEV) predicting its behavior.

Two stages of the MRC design were conducted (Figure 5), namely: (a) identification of the NNEV and (b) training the NNC using the identified NNEV. The aim of the NNEV identification was to obtain the NNEV parameters capable of representing the behavior of the unknown PBEV. The mean squared error (MSE) between the PBEV and NNEV outputs was used as the NNEV training signal (\(\varnothing\)).

![Figure 5. Two phases of the MRC design: (a) identification of the NNEV and (b) training the NNC.](image)

In the NNEV identification phase, the batch mode was enforced, in which 400 training samples were prepared that cover the range of expected MRC inputs. To prepare the training data, the reference training function trainlm was utilized consisted of a series of steps of random heights and random intervals. The main training sample characteristics are presented in Table 3.
range of expected MRC inputs. To prepare the training data, the reference training function \textit{trainlm} was utilized consisted of a series of steps of random heights and random intervals. The main training sample characteristics are presented in Table 3.

Table 3. MRC training sample characteristics.

<table>
<thead>
<tr>
<th>Sample Parameter</th>
<th>Unit</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Sampling interval</td>
<td>s</td>
<td>0.05</td>
</tr>
<tr>
<td>Number of samples</td>
<td></td>
<td>400</td>
</tr>
<tr>
<td>Maximum sample interval</td>
<td>s</td>
<td>2</td>
</tr>
<tr>
<td>Minimum sample interval</td>
<td>s</td>
<td>0.1</td>
</tr>
<tr>
<td>Maximum reference value</td>
<td></td>
<td>1</td>
</tr>
<tr>
<td>Minimum reference value</td>
<td></td>
<td>0</td>
</tr>
</tbody>
</table>

Since the MRC output is limited to 0 and 1, the sample datasets have been exposed to preprocessing. Both NNs were trained for function approximation (nonlinear regression) so that the MRC output follows the reference as closely as possible.

3. Results and Discussion

3.1. NNEV Identification

In the NNEV identification phase, the batch mode was enforced, in which 400 training samples given in Figure 6a (sections Network Architecture and Training Data) were applied to the NNEV before the weight updating. It was performed offline based on the measurements of the NNEV inputs and outputs of the system. The NNEV output (Figure 6b, Plant Input) was compared to the PBEV output (Figure 6b, Plant Output) aiming to minimize the MSE between the responses. To perform identification, the training data were shared among the training, validation, and testing phases as shown in Figure 6a (section Training Parameters). The number of training epochs was specified as 10 in this section. As follows from Figure 6b, the PBEV accurately follows the NNEV.

After performing identification, a final NNEV architecture was chosen, as shown in Figure 6c. It has 2 neurons in the input layer, 10 neurons in the only hidden layer with the sigmoid activation functions, and an output layer with a single neural linear activation function. The sigmoid \textit{tansig} activation functions allow the NNEV to learn nonlinear relationships between inputs and output. The linear \textit{purelin} activation function is suitable for solving the nonlinear regression problem related to the braking torque estimation. It was found that adding more hidden layers or hidden neurons does not improve the performance while increasing computation complexity, which is not desirable in a time critical context. Another obstacle to increasing the number of hidden layers might be the opacity of the system, which prevents effective security checks.

The NNEV identification results are shown in Figure 6c–e. The convergence curve of the model shows that MSE has almost converged at the 10th epoch where its value is of $7.6574 \times 10^{-6}$. Therefore, the model can be qualified as a well-trained one, and 10 epochs chosen for training are sufficient to meet the demands of the model application. The final regression of 0.99997 after training, validation, and testing also qualifies the model as a good one.
minimize the MSE between the responses. To perform identification, the training data were shared among the training, validation, and testing phases as shown in Figure 6a (section Training Parameters). The number of training epochs was specified as 10 in this section. As follows from Figure 6b, the PBEV accurately follows the NNEV.

Figure 6. Cont.
Figure 6. Cont.
After performing identification, a final NNEV architecture was chosen, as shown in Figure 6c. It has 2 neurons in the input layer, 10 neurons in the only hidden layer with the sigmoid activation functions, and an output layer with a single neural linear activation function. The sigmoid `tansig` activation functions allow the NNEV to learn nonlinear relationships between inputs and output. The linear `purelin` activation function is suitable for solving the nonlinear regression problem related to the braking torque estimation. It was found that adding more hidden layers or hidden neurons does not improve the performance while increasing computation complexity, which is not desirable in a time-critical context. Another obstacle to increasing the number of hidden layers might be the opacity of the system, which prevents effective security checks.

The NNEV identification results are shown in Figure 6c–e. The convergence curve of the model shows that MSE has almost converged at the 10th epoch where its value is of $7.6574 \times 10^{-6}$. Therefore, the model can be qualified as a well-trained one, and 10 epochs chosen for training are sufficient to meet the demands of the model application. The final regression of 0.99997 after training, validation, and testing also qualifies the model as a good one.

3.2. NNC Training and Validation

Once the NNEV is identified, it was used to train the NNC. Among the algorithms suitable for the NNC performance optimizing, the fastest of the gradient types was applied, namely, the backpropagation algorithm able to perform computations backward through the NN. The NNC was trained using the PBEV model, having one input and one output to generate data for the MRC training algorithm. The strategy of training supports both the offline and online tuning the NNC weights in a way that the MSE between the reference and the NNC output converges to zero.

The reference data displayed in Figure 7a,b are the same as for the NNEV identification. An NNC architecture was chosen, as shown in Figure 6c. It has one neuron in the input layer, four neurons in the hidden layer with the sigmoid activation functions, and...
an output layer with a single neural linear activation function. To perform training, the reference data were broken into two segments. The number of training epochs for each segment was specified as 10.

Figure 7. Cont.
Figure 7. Cont.
The NNC training results are shown in Figure 7c–e. As follows from these results, the convergence speed of the NNC is very fast, therefore, the optimization is obtained quickly. According to the convergence curve, the MSE has almost converged at the 10th epoch, where its value is $108.35 \times 10^{-6}$. The regression of 0.99933 qualifies the NNC as well trained.

After the training, the MRC was validated using the double-step reference command ($TB^*$) imitating the driver's brake pedal pressing/releasing. According to Figure 8, the response of the PBEV model to the MRC command ($T^*$) follows the driver's reference quite accurately: the maximal dynamics error is 100 Nm (10%) and the steady-state error is 9 Nm (1%).
3.3. Experimental Data Used for the MRC Testing

After the MRC training and validation, the PBEV model responses were estimated against various driver requests and MRC commands. To test the effectiveness of the designed control strategy in the hardware environment, a set of preliminary conducted experiments was used.

The hardware-in-loop (HIL) setup was used, which emulated an electric sport utility vehicle with four independent in-wheel motor powertrains in dynamically changing and vaguely defined environmental conditions. The core of the setup was represented by a decoupled testbed created by the ZF TRW Automotive® (Koblenz, Germany), connected to the high-precision vehicle dynamic software IPG CarMaker® (Karlsruhe, Germany). This testbed ensures smooth coordination between FB and RB, fast response time, flexible packaging, and integration with other chassis and powertrain control systems. To reproduce real pressure dynamics, the brake line pressure is measured there in the four brake calipers in a range of 20 MPa with a frequency of 1 kHz. The HIL system under investigation provided separate control of each of the four wheels using two interacted electrical drives. The electrohydraulic brake of the HIL setup was connected to the host computer via the dSPACE® (Paderborn, Germany) modular platform with the hardware components responsible for analogue-to-digital and digital-to-analogue data converting in real-time experiments, the control of the testbed, and the communication with the EV simulator. The mainboard of the HIL platform distributed the tasks between four core processors and communicated with a local host computer featuring 32 channels at 16-bit resolution and conversion time of less than 5 μs. The mass of the sport utility vehicle simulated by IPG CarMaker® was 2117 kg, and wheel radius was 0.2 m. It was assumed that the vehicle moves in a straight-line maneuver at a velocity of 100 km/h, fed by an electrical drive with a maximal permissible torque of 200 Nm, wheel speed of 157 rad/s, and inertia of 2.1 kgm², connected to the wheel imitator through the gear with a ratio of 10.5.

Details of experimentation and the set of experimental braking curves for front and rear wheels were reported in [8]. The most remarkable of them rely on the front wheels, where both the EB and the FB are operated together, whereas for the rear wheels, the FB is not actuated. These plots are selected, combined, and mutually scaled in Figure 9. The curve (w FL, light blue) demonstrates the velocity of the front left wheel, which follows the vehicle longitudinal velocity (v, blue line). Its wheel slip (L%, purple), friction torque (TF, green), and electrical torque (TE, red) curves are displayed. On the dry asphalt, since the EB torque is not sufficient to retain the optimal slip, the control system requests the additional FB torque. At the end of the slowing down, the regeneration is turned off, and the FB completes the deceleration alone. On the icy road, regeneration lasts all the way, while the FB does not participate in braking, except for the end of the process. An evident chattering phenomenon at low velocity is seen in the experimental torque plots.
3.4. Simulation Results of the MRC Testing

Figure 10 introduces the traces obtained from the model shown in Figure 1. Here, in response to a driver’s reference $T_B^* = 420$ Nm, the application torque $T^*$ was allocated between the electrical ($T_E$) and friction ($T_F$) fractions, the first of which was limited by the value of $200$ Nm. Since the EB response is fast and accurate, the application $T_E$ value turned out to be almost the same as the actuating $T_E^*$, unless it exceeds the limitation of the maximal electrical power. The torque oscillations are rather small here, as they are damped by the torque loop of the electrical drive. At low velocity, the friction increases dramatically, and the EB turns off.
3.5. Discussing the Concomitant Results

In order to investigate the effect of the NN control, taking into account the assessment of the road surface, the motion of a vehicle on a changing pavement, from dry to icy, was simulated. Figure 10 displays the traces obtained from the simulation of the volatile driving conditions. Here, the system successfully detects a change of the road surface based on analysis of the torque gradient. At the beginning, the deceleration was around 30 m/s² on a dry surface. At the 1.5 s, the road surface suddenly changes from dry to icy. As the new gradient is recognized, the total application torque needed to ensure an intensive stop drops to 60 Nm. The FB is no longer required, as the electrical torque is sufficient to slow the car down within the optimal wheel slip zone. Therefore, only electrical braking is performed further. At low velocity, the EB turns off, and the FB runs alone. Herewith, an obvious reduction in chattering is observed in the torque plots at low velocity. First, this is because the designed model takes into account an increase of friction due to its static fraction. Second, because the torque loop of the electrical drive remains close...
Figure 11. Simulation of the road changing from dry asphalt to icy surface.

Based on the energy curves (kJ, black) and assuming 50% regenerative efficiency, it is now useful to compare the MRC-driven system to the FLC-driven system published in [9]. This comparison is done with an aim to verify the robustness of the designed MRC in the case of changes in road conditions. A comparison of the simulation and experimental results demonstrates that both the MRC and FLC systems achieve good performance with the MRC, whereas the FLC-driven system shows higher oscillation. First, the NN-driven system handles electrical torque accurately. At gradual braking, the MRC maintains the same 60 Nm as in the experiment and successfully keeps optimal wheel slip in both braking modes. The NN-based system demonstrates the same braking times as the experimental system, both in the ABS (2.6 s) and in gradual (10 s) modes. At the same time, the torque chattering upon the NN control is minimal. There, it stays within a 100 Nm band at worst, whereas the experimentally obtained oscillation exceeds 500 Nm at the end of braking.

4. Conclusions

A comparison of the simulation and experimental results demonstrates that the created system demonstrates the desirable control behavior for various test scenarios, including the following:

- Both emergency and gradual braking. The EV velocity, slip, and torque responses confirm the proper system performance. The NN-driven system handles torque accurately. At both emergency and gradual braking, the EV velocity, slip, and torque responses confirm the proper system performance. The NN-driven system handles torque accurately.
- Robustness of the designed MRC: By imitating changes in road conditions, the robustness of the designed MRC was evaluated. A comparison of the simulation and experimental results on two different roads demonstrates the desirable control behavior for various test scenarios, including the proper system performance. The NN-driven system handles torque accurately.
- A comparison of the simulation and experimental results demonstrates that the created NN-based system demonstrates the desirable control behavior for various test scenarios, including the proper system performance. The NN-driven system handles torque accurately.
setup, the experience of the staff, and the time required to implement in the same control architecture. The assumed outcomes of this solution will be improved EV performance, increased mileage, and longer battery life.

Author Contributions: Conceptualization, methodology, software, writing, investigation, V.V.; validation, Z.R. and A.A.; resources, A.A.; project administration, funding acquisition, E.P. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by the Estonian Research Council grant PRG 658.

Data Availability Statement: The data presented in this study are available on request from the corresponding author.

Conflicts of Interest: The authors declare no conflict of interest.

Abbreviations

<table>
<thead>
<tr>
<th>Abbreviation</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ABS</td>
<td>antilock braking system</td>
</tr>
<tr>
<td>DOF</td>
<td>degree of freedom</td>
</tr>
<tr>
<td>EB</td>
<td>electrical brake</td>
</tr>
<tr>
<td>EV</td>
<td>electric vehicle</td>
</tr>
<tr>
<td>FB</td>
<td>friction brake</td>
</tr>
<tr>
<td>FLC</td>
<td>fuzzy logic controller</td>
</tr>
<tr>
<td>HES</td>
<td>hybrid energy storage</td>
</tr>
<tr>
<td>HIL</td>
<td>hardware-in-the-loop</td>
</tr>
<tr>
<td>kJ</td>
<td>recovered energy, kJ</td>
</tr>
<tr>
<td>λ, L%</td>
<td>p.u. and percentage wheel slip, appropriately</td>
</tr>
<tr>
<td>MRC</td>
<td>model reference controller</td>
</tr>
<tr>
<td>MSE</td>
<td>mean squared error, Nm</td>
</tr>
<tr>
<td>NN</td>
<td>neural network</td>
</tr>
<tr>
<td>NNC</td>
<td>neural network controller</td>
</tr>
<tr>
<td>NNEV</td>
<td>neural network model of an electric vehicle</td>
</tr>
<tr>
<td>PBEV</td>
<td>physics-based model of electric vehicle with blended braking</td>
</tr>
<tr>
<td>PIDC</td>
<td>proportional-integral-differential controller</td>
</tr>
<tr>
<td>REF</td>
<td>Reference</td>
</tr>
<tr>
<td>SOC</td>
<td>state of charge</td>
</tr>
<tr>
<td>TA</td>
<td>torque allocator</td>
</tr>
<tr>
<td>$T^*$, $T$</td>
<td>actuating and application braking torque, appropriately, Nm</td>
</tr>
<tr>
<td>$T^*_{\text{ref}}$</td>
<td>driver’s reference, Nm</td>
</tr>
<tr>
<td>$TE^*$, $TE$</td>
<td>electrical fraction of actuating and application braking torque, appropriately, Nm</td>
</tr>
<tr>
<td>$TP^*$, $TF$</td>
<td>friction fraction of actuating and application braking torque, appropriately, Nm</td>
</tr>
<tr>
<td>v</td>
<td>longitudinal velocity of the vehicle, km/h</td>
</tr>
<tr>
<td>vw</td>
<td>longitudinal velocity of the wheel, km/h</td>
</tr>
<tr>
<td>w</td>
<td>angular speed of the wheel, rad/s</td>
</tr>
</tbody>
</table>

References


17. Wang, B.; Guan, H.; Lu, P.; Zhang, A. Road surface condition identification approach based on road characteristic value. J. Terramech. 2014, 56, 103–117. [CrossRef]


23. Topić, J.; Škugor, B.; Deur, J. Neural network-based modeling of electric vehicle energy demand and all electric range. Energies 2019, 12, 1396. [CrossRef]


38. Aamir, M. On replacing PID controller with ANN controller for DC motor position control. *Energies* 2018, 11, 1377. [CrossRef]


42. Modia, S.; Bhattacharyya, J.; Basak, P. Estimation of energy consumption of electric vehicles using deep convolutional neural network to reduce driver’s range anxiety. *ISA Trans.* 2020, 98, 454–470. [CrossRef]


45. Min, K.; Yeon, K.; Jo, Y.; Sim, G.; Sunwoo, M.; Han, M. Vehicle deceleration prediction based on deep neural network at braking conditions. *Int. J. Automot. Technol.* 2020, 21, 91–102. [CrossRef]


