Neurocomputing 559 (2023) 126784

Contents lists available at ScienceDirect

Neurocomputing

journal homepage: www.elsevier.com/locate/neucom

On-line learning applied to spiking neural network for antilock braking systems

Javier Pérez [a], Manuel Alcázar [a], Ignacio Sánchez [a], Juan A. Cabrera [a,*], Mikael Nybacka [b,c], Juan J. Castillo [a]

[a] Department of Mechanical Engineering, University of Malaga, Malaga 29071, Spain
[b] Department of Engineering Mechanics, KTH Royal Institute of Technology, Stockholm SE-10044, Sweden
[c] Integrated Transport Research Lab, KTH Royal Institute of Technology, Stockholm SE-10044, Sweden

ARTICLE INFO

Keywords:
Antilock brake system
Spiking neural network
On-line learning
Supervised learning
Vehicle dynamics
Vehicle safety

ABSTRACT

Computationally replicating the behaviour of the cerebral cortex to perform the control tasks of daily life in a human being is a challenge today. First, it is necessary to know the structure and connections between the elements of the neural network that perform movement control. Next, a mathematical neural model that adequately resembles biological neurons has to be developed. Finally, a suitable learning model that allows adapting neural network response to changing conditions in the environment is also required. Spiking Neural Networks (SNN) are currently the closest approximation to biological neural networks. SNNs make use of temporal spike trains to deal with inputs and outputs, thus allowing a faster and more complex computation. In this paper, a controller based on an SNN is proposed to perform the control of an anti-lock braking system (ABS) in vehicles. To this end, two neural networks are used to regulate the braking force. The first one is devoted to estimating the optimal slip while the second one is in charge of setting the optimal braking pressure. The latter resembles biological reflex arcs to ensure stability during operation. This neural structure is used to control the fast regulation cycles that occur during ABS operation. Furthermore, an algorithm has been developed to train the network while driving. On-line learning is proposed to update the response of the controller. Hence, to cope with real conditions, a control algorithm based on neural networks that learn by making use of neural plasticity, similar to what occurs in biological systems, has been implemented. Neural connections are modulated using Spike-Timing-Dependent Plasticity (STDP) by means of a supervised learning structure using the slip error as input. Road-type detection has been included in the same neural structure. To validate and to evaluate the performance of the proposed algorithm, simulations as well as experiments in a real vehicle were carried out. The algorithm proved to be able to adapt to changes in adhesion conditions rapidly. This way, the capability of spiking neural networks to perform the full control logic of the ABS has been verified.

1. Introduction

System control by biologically-based neural networks that mimic real neurons is an issue that is just beginning to be explored. Biological neural systems are capable of performing motion control effectively and robustly. Consequently, biologically-based control structures have also been proposed to control real systems. The versatility and adaptability of control systems in animals or humans represent an opportunity for the development of new advanced control algorithms. However, the optimization of their performance is still a challenge. In the automobile field, one of the control systems that has received the most research and development has been the Anti-lock Brake System (ABS). ABS has proven its effectiveness in emergency situations by reducing the braking distance and maintaining the vehicle's maneuverability [1]. Therefore, algorithms that manage braking pressure during deceleration are a continuous source of study.

ABS control strategies are mainly divided into two different approaches. In the first case, there are systems that perform braking control by taking into account the wheel angular speed and the deceleration. Both input parameters are obtained from wheel speed sensors. In the second case, braking pressure is regulated as a function of the wheel slip [2]. The latter have better behavior than controls based on

* Corresponding author.
E-mail address: jcabrera@uma.es (J.A. Cabrera).

https://doi.org/10.1016/j.neucom.2023.126784
Received 6 March 2023; Received in revised form 29 August 2023; Accepted 11 September 2023
Available online 18 September 2023

wheel deceleration, but, in addition to wheel speed, they also require the estimation of vehicle speed. Furthermore, for optimal performance, the knowledge of the adhesion characteristics of the road is also necessary to define a reference slip. There are also combined strategies to control braking pressure from wheel deceleration, slip and road type. Therefore, one of the main problems to be tackled is the optimization of the control algorithm for all possible adhesion conditions between the tire and the contact surface: asphalt, wet asphalt, gravel, snow, or ice. Furthermore, controller performance is also influenced by the nonlinear generation of forces between the tire and the road [3]. The controller design is consequently complex as it must be able to adapt to all the uncertainties that may be encountered. In addition, unlike other vehicle control algorithms, time is even more critical in ABS systems since the locking of a tire occurs in 200–300 ms.

Most control algorithms [24] are tuned in a prior phase to their final implementation. Control logics are mainly developed using: industrial proportional-integral-derivative controller (PID), rule-based controller [5], the fuzzy logic controller (FLC) [67] sliding mode controller (SMC) [8] and neural control networks [9]. All of them are tuned by simulation or real tests, usually maintaining their structure fixed during normal operation. Offline-trained and fine-tuned algorithm combined with experimental tests may initially appear to be a preferable option due to its lower computational cost. However, the inherent variability and uncertainty of the tire dynamics can result in an inadequate response in the presence of unknown conditions. In contrast, the proposed algorithm can improve its performance by learning from experience in real-time. Consequently, their optimal performance in all conditions, namely those not taken into account during the training phase, is not warranted. These conditions might include: sudden changes in adhesion conditions, driver ability or a poor braking system condition.

To avoid this, the adaptation of the controller during operation using a model predictive controller (MPC) [10] with reinforcement learning, provides a better response to the uncertainties encountered in a real vehicle. However, it requires a high computational cost as it demands the execution of a real-time model of the vehicle and tires. Other authors [11] propose the use of a neural network-based Q-learning scheme. To this end, two classical neural networks are programmed to optimize the response of the ABS controller. In this approach, it is not necessary to know the system to be controlled, but pairs of training data are needed to perform the learning of the neural networks. Although this work shows the adaptation of the controller to changes in the reference slip, a high computational cost is required. In [12], a data-driven control architecture is used in which two controllers are utilized, an inner one that learns from input/output measurement and an outer one, based on a harmonic model predictive controller (HMPC), that provides the optimal slip. Like the previous ones, this control strategy is computationally costly.

In this paper, a new different scheme is presented. Thus, a bio-inspired neural structure [13] that allows control adaptation with a low computational cost is described. Despite being a simple neural network, the adaptation of controller behavior during normal operation is achieved by resorting to on-line learning and neural plasticity. To make use of the neural plasticity observed in biological systems, a neural model capable of reproducing these phenomena has to be developed. For this reason, Spiking Neural Networks, defined as the third generation [14] after artificial neural networks (ANN), have been employed for modeling the neural plasticity.

SNNs are employed to emulate neural structures and achieve bio-inspired ABS control. The simplicity of biological online learning enables the algorithm to be embedded in commercial vehicle control hardware, demonstrating its performance compared to other proposals that require custom hardware or extensive offline learning periods. SNNs [1516] reproduce neuronal activity based on electrical spikes by means of differential equations that model the generation of these spikes. To this end, the encoding and decoding of the variables of the system using sensory and motor neurons is required [1718]. The mathematical model of the SNN is more complex compared to those used for ANN [19]. Consequently, the number of neurons of the network must be reduced to meet real time requirements imposed by braking dynamics. Since the number of neurons is small, to minimize computational costs, a high firing frequency (>10 Hz) is needed in order to properly preserve information during encoding and decoding processes [20]. Alternatively, some authors suggest running the neural model on neuro-morphic hardware [21], but this makes its implementation in a passenger vehicle challenging.

The modeling of neuronal plasticity reproduces the change of strength in the connections between neurons, called synapses [22]. Hebbian learning [23] is used to define the plasticity between the pre- and post-synaptic activity. To modulate this learning, Spike Timing Dependent Plasticity (STDP) [24] regulates the relationship between spikes in the pre- and post-synaptic neurons as a function of time. Depending on the difference in firing time between neurons, a long-term depression (LTD) or long-term potentiation (LTP) is produced, which decreases or increases neuronal activity respectively. There are three possible STDP rules in literature [25]: Hebbian, Anti-Hebbian and All-LTD/LTP. Each one resembles different LTD/LTP behavior depending on timing. To regulate the learning rate, external modulation of the STDP [26], using dopamine according to the performance of the global neural network, is proposed. Furthermore, according to how dopamine is delivered, three types of learning are defined [2728]: supervised [29], unsupervised and reinforced.

This paper proposes using supervised learning [30] to modulate the training of the neural control algorithm of the ABS system of a four-wheel vehicle. This process is carried out by means of a bioinspired baseline control structure and by resorting to active learning [3132]. Whereas passive learning uses pretraining structures, which remain fixed during normal operation of the system, active learning modifies the neural connections during operation, thus enhancing controller performance in all conditions.

As previously mentioned, brake system performance greatly depends on surface detection during braking processes to determine the adhesion conditions. To cope with this fact, the proposed neural structure is responsible for two challenging tasks: classification and control. To this end, the neural structure is composed of two different networks. First, a classification network provides the optimal slip value from vehicle variables obtained through an Extended Kalman Filter (EKF). Next, a control network based on a bio-inspired structure that resorts to equilibrium point (EP) control determines the optimal brake pressure [33]. Consequently, a robust neural controller capable of learning during its operation is proposed.

Experimentation tests using a sensorized vehicle were carried out to evaluate algorithm implementation and performance in a real embedded control system.

The main contributions of this work are summarized as follows:

- Development and implementation of an SNN-based controller that performs road classification and brake pressure regulation for anti-lock braking systems. The controller is able to estimate road conditions to continuously provide the most appropriate braking pressure.
- A new bio-inspired control structure based on the equilibrium point hypothesis. This structure mimics the connections established in the vestibular-ocular reflex (VOR).
- An on-line learning approach using supervised learning via STDP modulation. The weights of the neural network connections are modified in real time to optimally adapt to braking conditions. This feature is a novelty in ABS controls and makes them more dynamic and flexible to environmental changes.
- A robust control logic that imitates biological systems validated through real experimentation in a vehicle.

The paper is organized as follows. In Section 2, the vehicle model and parameter estimation are introduced. The Spiking Neural Network and

learning method are presented in Section 3. Classification and control networks are defined in Sections 4 and 5 respectively. Simulation results and a discussion of the proposed ABS algorithm performance are included in Section 6, which is followed by the experimentation in Section 7. Finally, conclusions are drawn in Section 8.

2. Vehicle model and parameter estimation

The classification and control performance of the proposed structure greatly depends on adequate modelling and estimation of the required input parameters. In this work, state-of-the-art robust simulation and estimation techniques have been adapted and implemented to be used with the developed neural network controller.

A three-degree-of-freedom (3 DOF) model is used to simulate the longitudinal behavior of a four-wheeled vehicle while braking. The lateral and yaw DOF are added to replicate asymmetric braking of the wheels of each side. The dynamics of the vehicle body include both longitudinal and lateral forces.

This work focuses on the whole braking process. Therefore, the braking system and the generation of forces in the tire are modeled separately. The control scheme (Fig. 1) is composed of estimation and neural detection and control algorithms that will be described in the following sections. The nomenclature subsequently used is defined in Table 1.

2.1. Body model

Three differential equations are used to model the three degrees of freedom describing the motion of the vehicle. These three degrees of freedom are sufficient since only longitudinal and lateral vehicle movement during straight braking is considered. Yaw is also included to replicate situations where the adhesion conditions of the wheels of each side of the vehicle are asymmetric. In addition, it is also used to evaluate the correct behavior of the control algorithm as a deficient performance can cause the vehicle to rotate during longitudinal braking. Therefore, the forces imposed by the tire and the drag define the longitudinal (1), lateral (2) and yaw (3) motion.

$$\ddot{x} = \dot{y}\dot{\theta} + \frac{F_{xfl} + F_{xfr} + F_{xrl} + F_{xrr} + F_{dx}}{m} \tag{1}$$

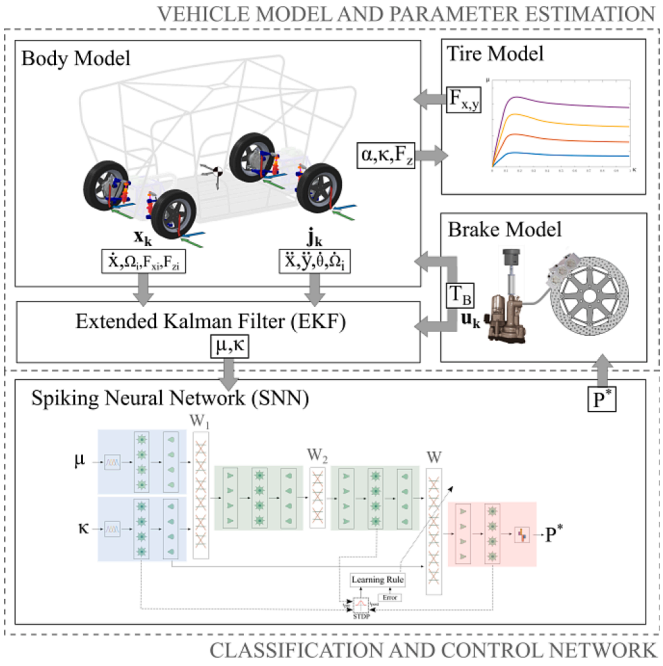

Fig. 1. Vehicle control scheme.

Table 1
Nomenclature.

Symbol	Description	Symbol	Description
x	Longitudinal displacement (CoG)	F_x	Longitudinal force
y	Lateral displacement (CoG)	F_y	Lateral force
θ	Yaw angle	F_z	Vertical force
fl	Front-Left	F_{zn}	Nominal vertical force
fr	Front-Right	F_d	Drag force
rl	Rear-Left	M_d	Drag moment
rr	Rear-Right	T_b	Brake torque
m	Vehicle mass	Ω	Tire angular speed
r	Tire radius	σ	Relaxation length
h	CoG height	C_y	Cornering stiffness
a	CoG distance of front wheels	N_p	Number of pads
b	CoG distance of rear wheels	μ_{pad}	Disc-Pad friction coefficient
w	Vehicle width	D_b	Disc brake diameter
I_w	Wheel inertia	R_m	Mean brake radius

$$\ddot{y} = -\dot{x}\dot{\theta} + \frac{F_{yfl} + F_{yfr} + F_{yrl} + F_{yrr} + F_{dy}}{m} \tag{2}$$

$$\dot{\theta} = \frac{a\left(F_{yfl} + F_{yfr}\right)}{I_{zz}} - \frac{-b\left(F_{yrl} + F_{yrr}\right)}{I_{zz}} + \frac{0.5w\left(F_{xfl} - F_{xfr} + F_{xrl} - F_{xrr}\right)}{I_{zz}} + \frac{M_{dz}}{I_{zz}} \tag{3}$$

For simplicity, the influence of the suspension system of the vehicle is not considered in the equations. Therefore, vertical loads on the front axle (4) and rear axle (5) acting on each wheel individually are obtained from the equilibrium of forces (6–9).

$$F_{zf} = \frac{bmg - \left(\ddot{x} - \dot{y}\dot{\theta}\right)mh + hF_{dx} - M_{dy}}{a+b} \tag{4}$$

$$F_{zr} = \frac{amg - \left(\ddot{x} - \dot{y}\dot{\theta}\right)mh + hF_{dx} + M_{dy}}{a+b} \tag{5}$$

$$F_{zfl} = F_{zf} + \left(mh\left(\ddot{y} + \dot{x}\dot{\theta}\right) + hF_{dy} - M_{dx}\right)\frac{2}{w} \tag{6}$$

$$F_{zfr} = F_{zf} - F_{zfl} \tag{7}$$

$$F_{zrl} = F_{zr} + \left(mh\left(\ddot{y} + \dot{x}\dot{\theta}\right) + hF_{dy} - M_{dx}\right)\frac{2}{w} \tag{8}$$

$$F_{zrr} = F_{zr} - F_{zrl} \tag{9}$$

Additionally, the angular acceleration ($\dot{\Omega}$) of each wheel is described using equation (10), with i={fl,fr,rl,rr}.

$$\dot{\Omega}_i = \frac{T_{bi} - F_{xi}r}{I_w} \tag{10}$$

2.2. Tire model

Tire longitudinal forces are obtained using the slip ratio (κ_i) and the vertical forces on each wheel (F_{zi}). For proper modelling, it should be considered that the generation of forces is not instantaneous [2] but it has a delayed response, so its behavior is modeled using Eq. (11).

$$\sigma\frac{d\kappa_i}{dt} + |\dot{x}|\kappa_i = r\Omega_i - \dot{x} \tag{11}$$

Where (σ), is the longitudinal relaxation length of the tire. Four parameters {B_x, C_x, D_x, E_x} have been defined to determine tire behavior. These, in combination with the vertical load and slip ratio, provide the longitudinal load (F_{xi}) experienced by the tire according to Eq. (12).

$$F_{xi} = F_{zi}D_x sin(C_x arctan\{B_x\kappa_i - E_x[B_x\kappa_i - arctan(B_x\kappa_i)]\}) \tag{12}$$

The ratio between the longitudinal and the vertical loads defines the longitudinal friction coefficient $\mu_{xi} = F_{xi}/F_{zi}$. This friction coefficient has a maximum ($\mu_{xi} = D_x$) at a given slip value, called optimum or reference slip value (κ_i^*). Therefore, the braking algorithm aims to keep the slip ratio as close to its optimal value as possible for improved performance.

Lateral forces on the tire (F_{yi}) are determined using a linear model as a function of the slip angles (α_i), the lateral friction coefficient ($\mu_{yi} = F_{yi}/F_{zi}$) and the vertical load, according to eq. (13). At this stage, a non-linear lateral tire model is not required since system performance will be evaluated in longitudinal tests, in which the slip angle is close to zero with minimum influence on the generation of lateral forces.

$$F_{yi} = -C_y \alpha_i \mu_{yi} \frac{F_{zi}}{2} \tag{13}$$

2.3. Brake model

To reduce the speed of the vehicle, a braking torque is applied on the wheels via the brake discs. A hydraulic system provides the required pressure (P) on the brake pads, pushing them against the disc and thus generating the frictional forces that create the braking torque on the wheels (T_b), according to Eq. (14).

$$T_b = \frac{\mu_{pad} P \pi D_b^2 R_m N_p}{4} \tag{14}$$

The relationship between the pressure demanded by the control system and the actual pressure and the pressure on the brake pads is modeled using a transfer function (15). This includes the dynamics of the control acting on the pump as well as the fluid inside the brake line. This function has been experimentally identified by comparing the target pressure required from the actuator and the actual pressure.

$$H(s) = \frac{-47s + 2354}{s^2 + 29s + 2385} \tag{15}$$

2.4. Extended Kalman filter (EKF)

All input variables to the SNN network cannot be directly measured in a real vehicle. To cope with this fact, the use of an EKF estimator is proposed. The slip ratio ($\kappa_i = 1 - (\Omega_i \bullet r/\dot{x})$ as well as the friction coefficient ($\mu_{xi} = F_{xi}/F_{zi}$) are required to control the braking of the tire. Therefore, these two variables are estimated by the EKF algorithm using the speed ($\dot{x}$) and forces (F_{xi}, F_{zi}), which are included in the state vector ($\boldsymbol{x_k}$) (16), and the measured wheel angular speed Ω. Measured ($\boldsymbol{j_k}$) (17) and system control variables ($\boldsymbol{u_k}$) (18) are defined to obtain the actual value (k) of the state vector.

$$\boldsymbol{x_k} = [\dot{x}, T_{bi}, F_{xi}, F_{zi}]^T \tag{16}$$

$$\boldsymbol{j_k} = \left[\ddot{x}, \ddot{y}, \dot{\theta}, \dot{\Omega}_i\right]^T \tag{17}$$

$$\boldsymbol{u_k} = [T_{bi}]^T \tag{18}$$

Measured variables are obtained from an inertial measurement unit (IMU) located at the center of gravity of the vehicle and the speed sensors installed on each wheel.

The time evolution of the state vector and measured variables is obtained from (19) and (20). A predicted state is calculated for each time step using function (ϕ) according to Eq. (21). Each state evolution is characterized as a random walk ($\dot{x}_{k-1}, F_{xi_{k-1}}$), a control variable ($T_{bik}$) and by using the vertical model (F_{zik}) Eqs. (6) to (9). Similarly, function (h) is calculated (22) to predict measurements. However, in the case of measurement ($\ddot{x}_k, \ddot{y}_k, \dot{\theta}_k, \dot{\Omega}_{i_k}$), the evolution is determined as a function of the body (1–3) and wheel (10) equations. Process noise ($\boldsymbol{\omega_k}$) and observations ($\boldsymbol{v_k}$) are added to achieve a better estimation.

$$\boldsymbol{x_k} = \phi(\boldsymbol{x_{k-1}}, \boldsymbol{u_k}) + \boldsymbol{\omega_k} \tag{19}$$

$$\boldsymbol{j_k} = h(\boldsymbol{x_k}) + \boldsymbol{v_k} \tag{20}$$

$$\phi(\boldsymbol{x_{k-1}}, \boldsymbol{u_k}) = \left[\dot{x}_{k-1}, T_{bik}, F_{xi_{k-1}}, F_{zik}\right] \tag{21}$$

$$h(\boldsymbol{x_k}) = \left[\ddot{x}_k, \ddot{y}_k, \dot{\theta}_k, \dot{\Omega}_{i_k}\right] \tag{22}$$

Hence, covariance matrices Q_k and R_k associated to noises $\boldsymbol{\omega_k}$ and $\boldsymbol{v_k}$ respectively have to be appropriately tuned. The prediction and update method used by the Kalman filter [34] also require the evaluation of the Jacobians for function (ϕ) and (h). An optimization process based on genetic algorithms has been adopted to set the covariance values. In this process, data from braking conditions like those of the experiment have been simulated.

To properly control longitudinal tire braking once the slip and adhesion levels are known, it is necessary to have the optimal level for each surface. This work proposes using a SNN to determine the type of road at any given moment. For this purpose, the same type of neural network, as defined in Section 3, will be used.

3. Spiking neural network

The main novelty of this work is the development of an algorithm based on Spiking Neural Networks capable of performing classification and control tasks. Furthermore, its performance can be improved during normal operation with a proposed on-line learning approach. Consequently, first, neuron and synapse models are outlined in detail in this section along with the STDP learning algorithm. Next, classification and control networks will be described in sections 4 and 5 respectively.

Spiking Neural Networks reproduce the behavior and learning of biological systems more accurately than Artificial Neural Network neurons. These networks resort to differential equations in the neural model to transfer the information, which is temporarily encoded. Therefore, encoding and decoding processes of the variables are required to interact with the neural network, thus resulting in different types of neurons. Therefore, three types of neurons are defined: sensory neurons, interneuron and motor neurons.

The biological components of a neuron are the dendrite, nucleus, and axon. Information reaches the neuron through the dendrite in the form of a current. This travels to the nucleus where, if a threshold level is exceeded, the neuron generates a spike. These spikes are transmitted through the axon to its terminals, which are connected to the following neuron. This neuronal connection between a dendrite and the axon terminals is called synapse. To generate the spikes, a coding process is carried out in the sensory neurons, where the information obtained is codified into a train of spikes. In contrast, to transform the information into an action, a decoding process is required in the motor neurons. Next, the mathematical modelling of these processes is presented. All in all, these neurons are composed of a nucleus connected to dendrites, axons, coding and decoding, depending on their function (Fig. 2).

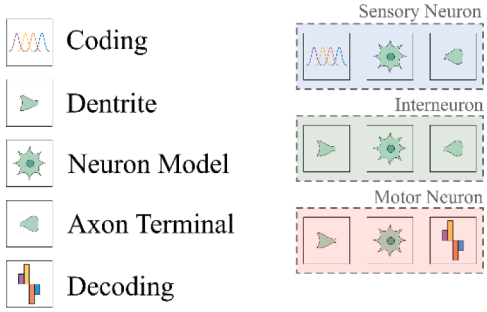

Fig. 2. Neuron configuration.

3.1. Coding

Encoding is responsible for converting an input variable (in) into an intensity value (I) for m neurons. To this end, a Gaussian bell (25) associated with each input neuron (i) is used. This regulates the amount of input current to each neuron, being spread over the entire input space linearly according to (23) and (24), where I_{min} and I_{max} are the minimum and maximum input current respectively (see Table 2). Gaussian bells are uniformly distributed according to partitioning factor (β), mean value (μ_i) and width (σ) over the entire input space [35].

$$\sigma = \beta \frac{I_{max} - I_{min}}{m - 2} \tag{23}$$

$$\mu_i = I_{min} + (i - 1)\frac{\sigma}{\beta} \tag{24}$$

$$I = I_{max} e^{\frac{-(in-\mu_i)^2}{2\sigma^2}} \tag{25}$$

3.2. Neuron model

The neuron model proposed by Izhikevich [15] is used in this work due to its low computational cost and good biological representation. Thanks to its greater number of advanced neuro-computational features, the Izhikevich's model [36] facilitates the exploration of other more complex biological structures in future research related to this work. While other simpler models, such as Leaky Integrate-and-Fire (LIF), may offer a lower computational cost, Izhikevich's neuron model makes it possible for future research to incorporate advanced features such as spike latency and accommodation into the brake control system.

It consists of two differential Eq. (29) that model the membrane potential (v) and recovery (u) of the neuron.

$$\begin{cases} \dfrac{dv}{dt} = 0.94v^2 + 5v + 140 - u + I \\ \dfrac{du}{dt} = a(bv - u) \end{cases} \tag{29}$$

$$If v \geq 30mV then \begin{cases} v \leftarrow c \\ u \leftarrow u + d \end{cases} \tag{30}$$

Thus, when the potential exceeds a threshold level, a spike is produced. Additionally, the neuron must be reset according to equation (30). Hence, the input current value is converted into electrical pulses. Parameters a, b, c and d define the firing response of the neuron. To minimize distortion and increase firing frequency, values {0.1, 0.2, −65, 2} have been used, which gives a fast spiking (FS) response.

3.3. Synapse model

The synapse model reproduces the connection between two neurons, from the axon to the dendrites (Fig. 2, interneuron). The release of neurotransmitters as well as their reception are modeled. The release of neurotransmitters (ε) from the axon is modeled using a first-order function (31) with a time constant (τ_s). This response emulates the neurotransmitter concentration in the synapse after the occurrence of a firing, making use of the time that has elapsed since the last spike ($\Delta t = t - t_i$).

On the other hand, the neurotransmitter reception after the response (ε) of all the connected neurons generates an input current (I) which depends on the strength of each synapse (w_{ij}). The current generated is the sum of all connections with the presynaptic neuron, according to equation (32). Where i and j represent the presynaptic and postsynaptic neurons respectively.

$$\frac{d\varepsilon(\Delta t)}{dt} = -\frac{\varepsilon(\Delta t)}{\tau_s} + \delta(\Delta t) \tag{31}$$

$$I = \sum w_{ij}\varepsilon_i \tag{32}$$

3.4. Decoding

The neuron output response (ε) must be transformed into an output variable (y) (Fig. 2, motor neuron). As in coding, the same partitioning factor and width are used for linear decoding (28), according to Eqs. (26) and (27). A gain is thus associated with each neuron of the output layer, giving the sum of all the neurons as a result, like Heinemanńs size principle [37].

$$\sigma = \beta \frac{I_{max} - I_{min}}{m - 1} \tag{26}$$

$$\mu_i = I_{min} + (i - 1)\frac{\sigma}{\beta} \tag{27}$$

$$y = \sum_{i=1}^{m} \varepsilon_i \mu(i) \tag{28}$$

3.5. Learning algorithm

The output of the SNN control algorithm is directly linked with the strength of the synapses between the neurons. Consequently, the response of the algorithm can be optimized for specific braking conditions if the synapses are adequately modified. The proposed learning algorithm is responsible for this adaptation strategy. The learning structure used is based on on-line learning as it allows online training when a large error is made. Time-dependent plasticity (STDP) is used for this purpose in combination with the error made by the neural network at a given time (*error(t)*).

STDP models neuronal plasticity [25] as a function of the difference in firing time (τ) between the presynaptic (pre) and postsynaptic (post) neurons (33). Thus, learning can be Hebbian, Cells that fire together, wire together, Anti-Hebbian, All-LTP or All-LTD. It depends on how long-term potentiation (LTP) and long-term depression (LTD) are defined according to the temporal difference.

$$\tau = t_{post} - t_{pre} \tag{33}$$

Table 2
Neural Network Parameters.

Symbol	Description	Value
m_{SN}	Number of sensory neurons	$64 = 32(\mu) + 32(\kappa)$
m_{IN}	Number of interneurons	$64 = 32(W_1 \rightarrow W_2) + 32(W_2 \rightarrow W)$
m_{MN}	Number of motor neurons	$32(P^*)$
$[a, b, c, d]$	Neuron model parameters	$[0.02, -0.1, -55, 6]$
τ_S	Synapse time constant	1 ms
β	Partitioning factor	2
$[Imax_{min}$	Input current range	$[0, 388]$
τ_C	Eligibility time constant	10 *ms*
τ_D	Dopamine time constant	20 *ms*
Δt	Time interval	1 ms

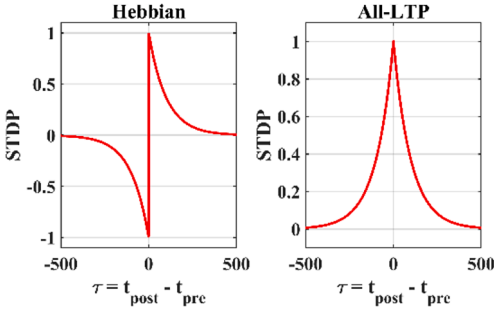

Fig. 3. STDP rules.

Hebbian learning (Fig. 3) (34) takes place when a low firing frequency is involved [38], which is related to the memorization process. Accordingly, it will be used in the learning performed by the classification neural network. On the contrary, ALL-LTP learning (35) is naturally found in brain areas with a high firing frequency, which is associated with motor control [39]. Consequently, this learning strategy is preferred during the on-line learning phase of the control network.

$$STDP(\tau) = sign(\tau)e^{-\frac{|\tau|}{\tau_{STDP}}} \tag{34}$$

$$STDP(\tau) = e^{-\frac{|\tau|}{\tau_{STDP}}} \tag{35}$$

The learning process involves adapting the weights of the neural connections progressively to reduce the error (equation (39)). This learning depends on eligibility (C), dopamine release (D) and a learning factor (μ_l). The latter has been added to control the learning rate to adjust it for each particular case. Eligibility (C) tries to keep the post-synaptic and presynaptic pulses synchronized, as it can be seen in differential equation (36), which includes a delay as a function of the time constant (τ_C). If the difference between t_{post} and t_{pre} is close to zero, the STDP function increases or decreases the weights up to their maximum possible values (see Fig. 3). In contrast, dopamine (D) depends on a reinforcement signal that triggers its release with a time delay determined by the time constant (τ_D) Eq. (37). The combination of these two gives rise to the synaptic strength (S) Eq. (38) which defines the learning rule used to modify the synapse, following the scheme of Fig. 4. In this case, this reinforcement signal is error (t), whose value is related to the response of the neural control. Furthermore, as our ABS control scheme is composed of two neural networks, one for road type classification and one for braking pressure control, error (t) depends on the performance of both networks, as it will be described in the following sections.

$$\frac{dC}{dt} = -\frac{C}{\tau_C} + STDP(\tau)\delta(t - t_{pre/post}) \tag{36}$$

$$\frac{dD}{dt} = -\frac{D}{\tau_D} + error(t) \tag{37}$$

$$\frac{dS}{dt} = C\hat{A}\cdot D \tag{38}$$

$$w_{ij}(t) = w_{ij}(t - \Delta t) + \mu_l dS \tag{39}$$

4. Classification network

As mentioned above, the ABS system proposed in this work is composed of two neural networks. The first one, the classification neural network, is responsible for providing the optimal slip value of the road where the vehicle is circulating. To this end, the neural network resorts to data obtained from the EKF estimator. Thus, this network has the estimated wheel slip (κ) and friction coefficient (F_x/F_{z0}) as inputs and the optimal slip as output. Next, once the input parameters are known, the network recognizes which type of road it belongs to (Fig. 5) and

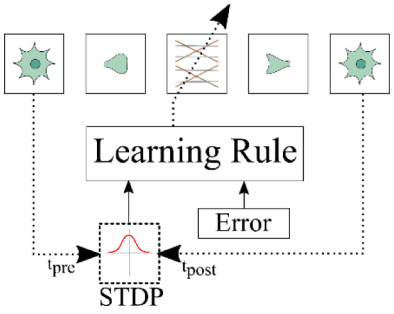

Fig. 4. Learning scheme.

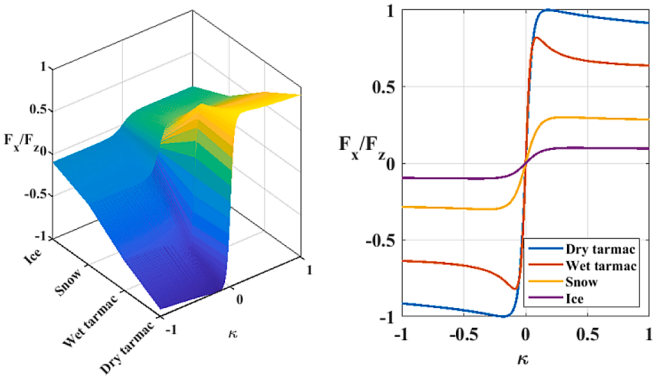

Fig. 5. Tire data for different road conditions.

determines the optimal slip.

For the neural network to be able to identify the features associated with each surface, the use of interneurons conforming the so-called hidden layers is required. Depending on the complexity of the surface to be recognized, or if the input variable encodes a lot of information, for example from a camera image [42], a different number of hidden layers can be used. In this approach, due to the previous estimation of the EKF [34], the problem is simplified since each pair of data (μ_κ) is associated to a particular road type with the exception of some of them that overlap for low slip levels. Although this is reflected in the results obtained by the network, they are not considered disadvantageous since braking control requires slip generation during its normal operation. Taking all this into account, it was decided to use a structure with a single hidden layer (Fig. 6) consisting of 20 sensory neurons (10 per input), 15 interneurons and 10 motor neurons. Each motor neuron is associated with a slip level so that the one with the highest activation is the one that establishes the optimal slip level.

The learning has been performed off-line, i.e., using data obtained from the tire model presented in sub-section 2.2 with different parameters {B_x, C_x, D_x, E_x}. Therefore, weight matrices W1 and W2 that define the strength of the connections are modified with a training data set ($\boldsymbol{\mu_{xi}}$, $\boldsymbol{\kappa_i}$,$\boldsymbol{\kappa_i^*}$) by applying the STDP learning algorithm with Hebbian rule, both detailed in the previous section. To obtain the reinforcement signal that triggers learning in the neural network, $\boldsymbol{error(t)} = \boldsymbol{\kappa_i^*} - \boldsymbol{\kappa_{i_{SNN}}^*}$ is proposed. This error is defined as the difference between the optimal slip ($\boldsymbol{\kappa_i^*}$), which is obtained from tire models on different roads and with different slip levels ($\boldsymbol{\kappa_i}$) and friction coefficient ($\boldsymbol{\mu_{xi}}$) and the optimal slip provided by the neural network ($\boldsymbol{\kappa_{iSNN}^*}$). Thus, when the error is high, dopamine release will be activated (equation 37) and the synaptic connections that are most synchronized at that instant will be updated (equation 36). These synchronized connections are the ones that affect output of the neural network ($\boldsymbol{\kappa_{iSNN}^*}$) to a greater extent. Therefore, it is necessary to adapt their weights to reduce the error made.

Fig. 7 shows an example of the learning process of the classification network. A.tir file is utilized to define the training and validation dataset. This file is an industry-standard providing the parameters required to describe the behavior of the tire according to Equation 12. To obtain the dataset the road condition ($\boldsymbol{\mu_{xi}}$) is modified each 200 ms randomly from low to high adherence. During this period, for each grip condition ($\boldsymbol{\mu_{xi}}$), the slip ($\boldsymbol{\kappa_i}$) is linearly varied from 0 (pure rolling) to 1 (tire lock). Then, for each combination of ($\boldsymbol{\mu_{xi}}$, $\boldsymbol{\kappa_i}$), a corresponding optimal slip ($\boldsymbol{\kappa_i^*}$) is calculated using the parameters from the.tir file and Equation 12.

At first, it can be observed how the errors made by the classification neural network are high. As there is no prior knowledge about the connections of W1 and W2, the weights of the connections are randomly initialized with a normal distribution. Consequently, the performance of the network is poor. Next, as the learning process evolves, the neural

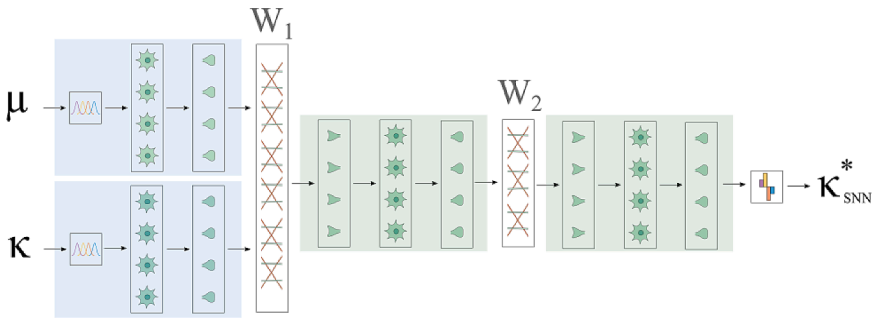

Fig. 6. Classification network structure.

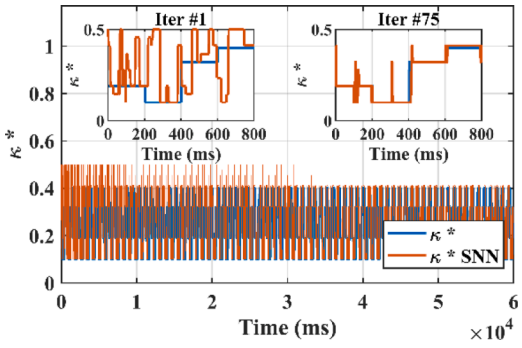

Fig. 7. STDP learning over time.

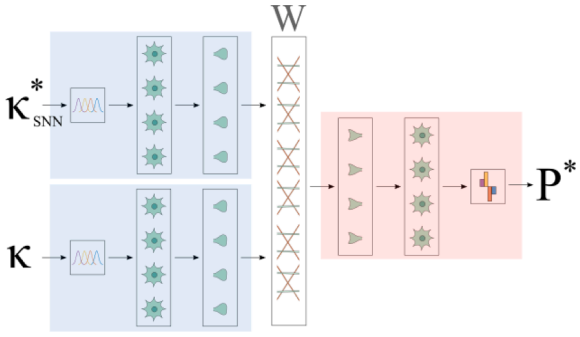

Fig. 8. Controller structure.

network yields more appropriate values of the optimal slip progressively. After 75 iterations, the network is already capable of providing a reliable value for the optimal slip. In this off-line learning process, we proceeded to teach the neural network with different road types, which have different reference slips (κ^*), using the slip (κ) and the friction coefficient (μ_x) as inputs. When the slip is very low or close to zero, the neural network cannot provide an accurate value since it does not have enough information due to the fact that the adhesion curves are overlapped (see Fig. 5). For this reason, peaks can be observed in the output of the network, as seen in Iter #75. This fact does not imply an instability of the ABS control system since the brake pressure will always be increased when the slip value is close to zero. This will be made possible by the configuration of the control neural network, as it will be described in the next section.

5. Control network

The neural network controller is responsible for determining a pressure setpoint (P^*) in order to minimize braking distance without losing control of the vehicle. For this purpose, a wheel slip control (WSC) is used [43]. This WSC needs to be aware of both the slip level (κ) and the optimal value (κ^*_{SNN}). These variables are provided by the EKF and the classification neural network respectively. The proposed structure (Fig. 8) shows the internal structure of the control neural network.

A structure similar to those found in the neural pathways of reflexes, called reflex arcs [44], is used to establish the initial neural connections. These reflex arcs define the relationships between sensory neurons and motor neurons, either directly or through an interneuron [45]. According to previous work [40], it is proposed to use a control structure extracted from the vestibular-ocular reflex (VOR) arcs [41]. This structure generates the control action by means of two motor neurons that react antagonistically. Together they maintain the equilibrium at the required operating point as proposed by the equilibrium point hypothesis (EPH).

To replicate this antagonistic behavior by means of excitatory and inhibitory connections, neuronal pathways are proposed according to Fig. 9. On one hand, if the slip is higher than the optimal slip level ($\kappa^*_{SNN} - \kappa < 0$), the pressure in the brake system is reduced. On the other hand, if the slip is lower than the optimal one ($\kappa^*_{SNN} - \kappa > 0$), the pressure is increased.

As shown in Fig. 9, for a surface with an optimum slip (κ^*_{SNN}), the output of the controller, i.e. the pressure (P^*) in the brake system, depends on the current slip value (κ). For low slip values ($\kappa = +$), i.e. below the optimum value, the network connections will boost ($P^* = + +$) and inhibit ($P^* = +$), thus providing a high value of braking pressure (a-low slip). When both slips have a similar value neither of the paths is boosted so the applied pressure will be intermediate (b-optimal slip). On the other hand, when the slip value is high ($\kappa = + +$), the neurons applying a reduced value of pressure ($P^* = +$) are excited and ($P^* = + +$) inhibited, thus reducing braking pressure (c-high slip). This ensures the stability of the controller as long as the learning algorithm does not change the inhibitory and excitatory relationships, thus only modifying its synaptic strength.

Since the output of the classifier is one of the inputs of the controller, the overall structure is simplified according to Fig. 10. Network parameters are defined in Table 2. Since the classification network was trained off-line, the algorithm running in real time performs the on-line learning exclusively for the neural connections of the controller. For this purpose, STDP learning is used with an All-LTP type rule making use of the error defined by $error(t) = (\kappa^*_{SNN} - \kappa)$. Therefore, weight matrices W1 and W2 are not modified in real time. They have already been optimized off-line as explained in the previous section. On the other hand, weight matrix W is updated in real time and by means of the error signal. This error represents the difference between the optimal slip (κ^*_{SNN}) and the real slip (κ), as shown in Fig. 10. Consequently, the proposed strategy allows the ABS control system to learn online, thus modifying its connections in real time to improve its performance.

6. Simulations

To demonstrate the effectiveness of the controller, simulations have

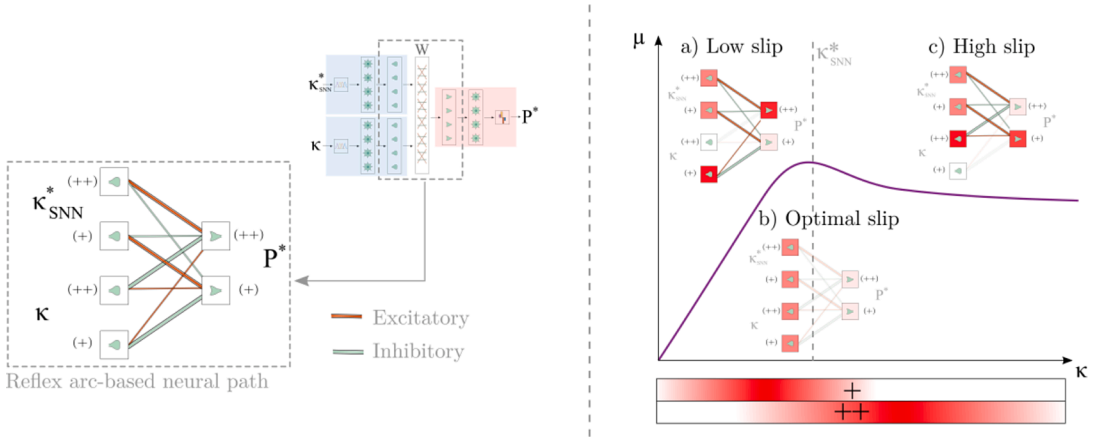

Fig. 9. Reflex arc-based synapse connections (left) and pressure states for different slip ratios (right).

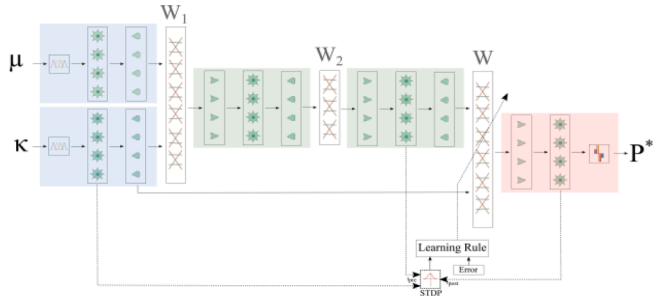

Fig. 10. Classification and control network.

been carried out to reproduce emergency braking, including variations in the level of adhesion. Initially, the structure of the network is defined with a prefix value of synaptic strength (W) based on bio-inspired neural connections. The learning algorithm is then in charge of increasing or decreasing the strength to minimize the slip error made. This reduces the braking distance and ensures maneuverability, avoiding wheel locking, which allows for maintaining steering capacity as the lateral friction remains available.

The learning capability of the network is evaluated by means of simulations in which the transition between two roads with different levels of adhesion ($D_{x1} = 1$, $D_{x2} = 0.4$) is reproduced. Simulations start with a braking process on a road with a high friction coefficient ($\mu_x = D_x = 1$). Next, when the speed reaches 60 km/h, the surface is changed to a road with a low friction coefficient ($\mu_x = D_x = 0.4$). Table 3 shows the vehicle parameters used in these simulations.

Table 3
Vehicle Simulation Parameters.

Parameter	Value	Unit
Weight	863	kg
Wheelbase	2.6	m
Track width	1.6	m
Wheel radius	0.344	m
Wheel + Hub Motor Inertia	2.33	Kg m^2
Center of gravity (CoG) (Front axis)	[0.0, −1.4,0.5]	m
Frontal Area	3	m^2
Drag coefficient	0.4	–
Tire corner stiffness	11e3	N/rad
Lateral relaxation length (σ_y)	0.2	m
Longitudinal stiffness factor (B_x)	9	–
Longitudinal shape factor (C_x)	2	–
Longitudinal peak factor (D_x)	1–0.4	–
Longitudinal curvature factor (E_x)	0.8	–
Longitudinal relaxation length (σ_x)	0.025	m
Disc-pad friction coefficient	0.5	–
Piston bore	25e-3	m^2
Mean brake disc radius	0.133	m
Number of pads	6	–

The classification network has been previously trained to detect both surfaces properly. Therefore, these simulations aim to demonstrate the stability of the learning process by integrating the EKF estimator and the SNN classifier in conjunction with the SNN controller. It takes a total of 20 iterations for the algorithm to converge to a state with a minimal dopamine level, achieving a reduced braking distance, as shown in Fig. 11. As shown in Fig. 11, thanks to the learning process, the neural ABS controller reduces the braking distance around 30 m after only 20 iterations.

Both Fig. 12 and Fig. 13 show the velocities and brake torque respectively in three iterations of the learning process. In iteration 1 it can be seen that the response of the network is not sufficient to generate the necessary slip, so braking is not optimal. Nevertheless, it is observed how, during the transition between high and low adhesion, the algorithm adapts quickly. This is something that occurs throughout the learning process, showing its robustness when a sudden change in adhesion is applied. Iteration 10 shows a response with a higher slip than at the beginning of the learning process. However, the front and the rear wheels behave differently as the algorithm has not managed to adapt to the load transfer yet. Despite the reduction in braking distance, the error is still high, so learning must continue.

Finally, in iteration 20, the error is low, as shown by the reduction in the associated value of dopamine (p). Furthermore, it is observed that the front and rear wheel speeds experience a similar slip level, which results in the finalization of the learning process.

Previous simulations have demonstrated the ability of the proposed ABS to learn on-line. However, the stability, robustness and performance of the controller have to be demonstrated. As indicated in [2], the classical stability theory to validate such complex designs is extremely difficult to apply. Traditional approaches can only prove closed-loop stability of the system model used in the analysis, but not in the real plant. Robustness and performance can be evaluated by performing simulations and real testing in different conditions and comparing obtained results to those provided by a state-of-the-art competitor. To this end, simulations have been conducted to reproduce the main tests required during the homologation process of the braking system of a vehicle according to guidelines given by the United Nations in Regulation 13 (E/ECE/- TRANS/505/Rev.1/Add.12/Rev.8. 3. Regulation No. 13). All these combinations of tests as well as the incorporation of system constraints ensure the stability of the controller and the vehicle during the braking process. Sensor noise was added in all simulations to reproduce real conditions. Performance indexes were obtained and compared to those achieved by a competitor to evaluate the ABS performance. Finally, real tests were carried out to evaluate learning

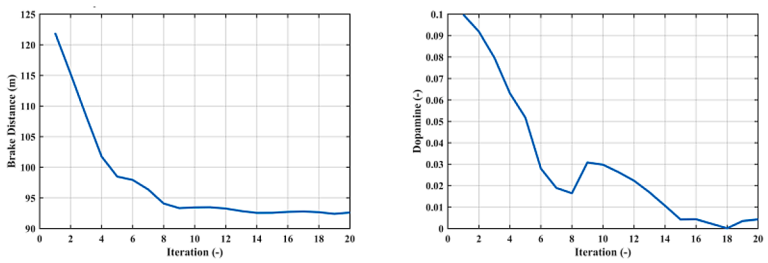

Fig. 11. Distance (left) and Dopamine (right) level under the supervised learning process during simulated emergency braking.

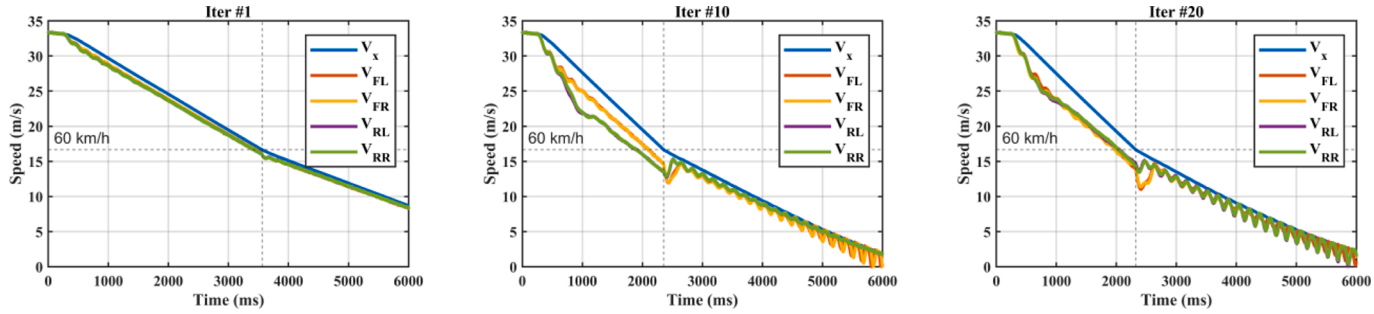

Fig. 12. Vehicle and wheel speeds under the supervised learning process during simulated emergency braking.

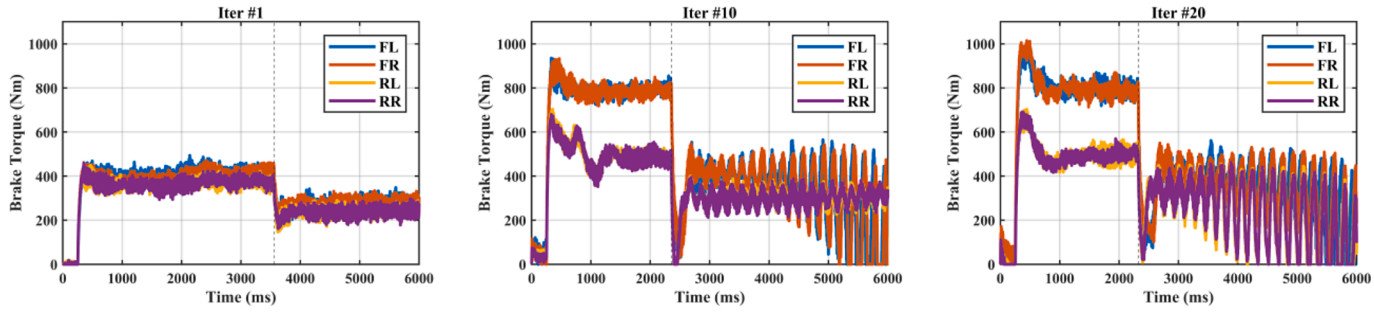

Fig. 13. Braking torques under the supervised learning process during simulated emergency braking.

ability, robustness and performance.

First, a high number of simulations were conducted in steady and transient conditions at different speeds and in various surfaces. For comparison purposes, it is necessary to have an objective way to quantify the performance on the different cases evaluated. To this end, three key performance indicators (KPIs) were calculated for the steady case simulations: *ABS Index of Performance (ABSIP), Integral Pitch Variation (IPV) and Integral Pitch Variation (IPV)*. Similarly, three different KPIs were used for transient cases: Mean Deceleration (MDj), Peak to Peak (PTPj), and Maximum yaw rate (MYRj). In addition to the KPIs, for the sake of comparison, a reference control algorithm was used. Thus, an algorithm based on threshold slip control (THR) was employed [46]. THR controllers are widely used in ABS applications due to their good performance and easy control, being widely used in passenger vehicles. They are created from practical experience and intensive simulation studies. They have good performance but require fine-tuning through extensive testing. The stability and robustness assessment of THR is done through experiments on the target vehicle. Three control actions of brake pressure are usually defined: apply, release, and hold brake pressure.

The response of the THR controller has been experimentally fine-tuned to achieve the shortest braking distance on the tested surfaces. This fine-tuning process involves setting the angular deceleration limits of the wheel and defining the appropriate slip limits. The final setting for THR control was the following: a slip switching threshold of 0.2. Minimum and maximum acceleration limits were 4 m/s^2 and 30 m/s^2 respectively. Finally, a deceleration limit -60 m/s^2 was set.

The main goal of this work is to show that an algorithm with a predefined neural structure can improve its behavior without becoming unstable, providing improved results compared to a well-known braking control algorithm. This makes a clear difference to other advanced control algorithms since this proposal does not require a previous offline training process in several situations.

KPIs [2] are used to evaluate the performance between controllers. There is a large number of indicators depending on each variable to be evaluated and its impact on safety, maneuverability, comfort, durability, and so on. In this case, only the KPIs necessary during an emergency have been used, leaving out those related to comfort or durability of the actuators since, in an emergency, the priority is to maximize grip while maintaining maneuverability. The KPIs are divided into steady-state and transient, both of which will be used to analyze the response of the proposed controller compared to skid and the threshold-based controllers in both situations.

6.1. Steady-State

6.1.1. ABS Index of performance (ABSIP)

It compares the braking distance obtained by the controller with and without ABS (locking the tire) (37). Thus, the overall effectiveness of the controller is obtained.

$$ABSIP = \frac{d_{ABS}}{d_{SKID}} \tag{37}$$

6.1.2. Peak to Peak PTP (PTP)

It measures the agility of the response of the controller. For this purpose, the difference between the maximum (max) and optimum (opt) angular speed during the first cycle is calculated. The value for each wheel is obtained according to (38).

$$PTP = \frac{\Omega_{max} - \Omega_{opt}}{\Omega_{max}} \tag{38}$$

6.1.3. Integral Pitch variation (IPV)

Yaw variation (39) does not affect the braking distance directly. However, it limits the driver's ability to estimate distances since the human brain makes greater errors in estimation in the presence of excessive yaw, which leads to danger.

$$IPV = \int_{t_i}^{t_f} |\theta| dt \tag{39}$$

Table 4 shows the results for the two studied algorithms. It shows both emergency braking on constant surfaces and transitions. This list of conditions has been extracted from Regulation 13 (E/ECE/-TRANS/505/Rev.1/Add.12/Rev.8. 3. Regulation No. 13). Consequently, it is possible to assess the stability, performance and robustness [2].

The proposed SNN-based algorithm provides better results according to performance indicator ABSIP and PTP. For ABSIP only in extremely low grip ($D_x = 0.3$) situations does the THR algorithm perform better. This is due to the constant level of grip for low levels of adherence. Therefore, the influence of the slip level in the braking distance is low. However, the SNN controller maintains a low slip rate which, despite the higher braking distance, preserves the maneuverability of the vehicle during braking. Regarding PTP, the proposed algorithm obtains less variation, improving the maneuverability of the vehicle. This is a key factor in low-adherence situations, e.g., on ice, where maintaining control of the vehicle is a priority. Therefore, it can be concluded that the SNN algorithm is the one that offers the best compromise between braking distance and maneuverability. The last KPI gives information about yaw (IPV), resulting in favorable behavior for both THR and SNN. Therefore, it is not required for experimentation.

6.2. Transient

The three transitions performed are assessed, also using the KPIs associated with the transient response during the jump.

Table 4
Steady State KPIs.

D_x	Initial Speed	ABSIP THR	ABSIP SNN	PTP THR	PTP SNN	IPV THR	IPV SNN
1	130 km/h	89,5%	**82,5%**	0,20	**0,03**	0,15	**0,12**
0.7	90 km/h	87,5%	**87,4%**	0,33	**0,11**	**0,02**	0,06
0.3	40 km/h	**87,4%**	98,1%	0,50	**0,03**	**0,01**	0,02
1.1 → 0.58 (@100 km/h)	120 km/h	92,6%	**90,4%**	0,21	**0,18**	**0,03**	0,08
0.8 → 0.3 (@40 km/h)	50 km/h	91,3%	**90,5%**	0,59	**0,43**	**0,01**	0,04
0.3 → 0.8 (@55 km/h)	70 km/h	**85,3%**	86,4%	0,27	**0,05**	**0,02**	0,04
0.7 (Rough)	70 km/h	97,7%	**94,0%**	0,48	**0,28**	**0,03**	0,02
0.3 (Rough)	40 km/h	**95,3%**	103,9%	0,70	**0,22**	**0,01**	**0,01**

6.2.1. Mean deceleration (MDj)

The average deceleration obtained during transition is calculated. The deceleration is measured between the start of the jump and one second after it occurs according to Eq. (40).

$$MD_{jump} = \int_{t_{i,j}}^{t_{i,j}+1} a_x dt \tag{40}$$

6.2.2. Peak to peak (PTPj)

The same calculation as in equation (38) is done in this case for the first cycle after the jump.

6.2.3. Maximum yaw rate (MYRj)

The maximum yaw rate is used to quantify the lateral during the jump (41). This is because a sudden change in adhesion can generate the yaw of the vehicle although this situation does not usually occur as both wheels on each side of the vehicle experience the change in grip at the same time.

$$MYR_{jump} = max\left[\dot{\theta}\right]_{i,jump}^{f,jump} \tag{32}$$

The results obtained for the transitions show better performance of the proposed SNN-based controller (Table 5). This is due to its ability to adapt to change while the one based on threshold control cannot adapt to change. Hence, during the transition, it obtains worse deceleration and deviation values. As in steady-state, the value obtained associated with the yaw is reduced, so vehicle stability is maintained without large yaw changes in both cases.

From previous simulations, it can be concluded that the proposed algorithm is safe, robust and stable. Furthermore, performance is satisfactory, providing improved KPI_S compared to its competitor.

7. Experimentation

In order to validate the simulations performed, real tests with an instrumented vehicle were carried out. The performance of the proposed algorithm was validated using the Research Concept Vehicle Model E (RCV-E), an over-actuated by-wire electric vehicle designed by the Integrated Transport Research Lab at the KTH Royal Institute of Technology (Fig. 14). This made it possible to implement the developed algorithm using SIMULINK by flashing it onto the dSPACE hardware that controls all the actuators and handles and all the signals on the RCV-E.

The braking system used consists of a hydraulic pump from the manufacturer Nissin used in the CBR1000RR from 2008 to 2014. This pump is capable of delivering up to 120 bars of pressure through the use of a master cylinder actuated by a DC motor and a lead screw. This allows performing a brake-by-wire system according to the scheme shown in Fig. 14, where each wheel of the vehicle is equipped with an independent system.

The control loop in our implementation has an execution time of 10 ms, with the algorithm runtime being less than 1 ms, as implemented on a 900 MHz dSPACE. This neural controller not only enables real-time

Table 5
Transient KPIs.

D_x	MDj THR	MDj SNN	PTPj THR	PTPj SNN	MYRj THR	MYRj SNN
1.1 → 0.58	0,56	**0,59**	0,21	**0,18**	**0,007**	0,028
0.8 → 0.3	0,24	**0,25**	0,59	**0,43**	0,001	**0,001**
0.3 → 0.8	0,63	**0,68**	0,27	**0,05**	0,000	**0,000**

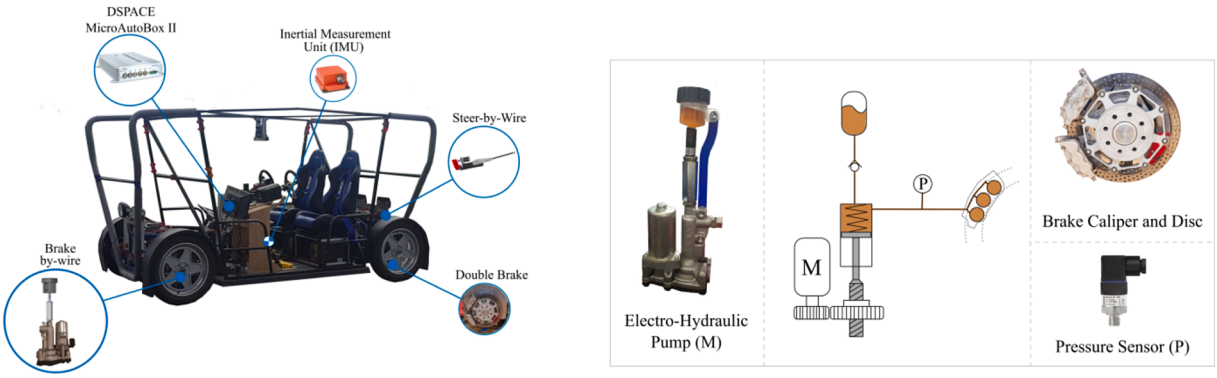

Fig. 14. Research concept vehicle model E (RCV-E) (Left) and electro-hydraulic proportional brake system (right).

execution but also allows for the emulation of firing frequencies between 300 and 400 Hz. By minimizing latency, the controller's response time is significantly improved. A maximum time of 10 ms to meet the real-time requirements imposes a similar control loop period as other approaches [12]. However, these implementations can last hours to perform offline learning. In contrast, our solution not only meets real-time requirements but also uses less computational cost compared to similar controllers. For example, in [11], an MPC requires between 3 and 5 ms on similar hardware. While other controllers like eMPC have lower execution times (0.1 ms), incorporating online learning becomes challenging.

This scheme also includes a pressure sensor to close the control loop, enabling the control algorithm to set a target pressure that the pump manages to maintain by means of a PI controller. The pump is connected to the brake caliper driving the hydraulic fluid into a total of 6 pistons that transmit force to the pads in contact with the brake disc. This allows the pressure generated by the pump to be converted into braking torque by friction.

The validation is focused on the control algorithm, which is the main contribution of this work. Tests were carried out at 25 km/h since the vehicle is limited to that speed by design. Tests were performed on high grip surface, low grip surface, and transition between both. Each test consisted of multiple iterations of consecutive emergency braking. Tests were conducted on the Arlanda Drive Lab test track 1 in Stockholm. This track has a test area with two road surfaces with extreme grip levels (Fig. 15). One of them is high-performance asphalt with high-water absorption and optimal grip (High). The other one is a painted surface with low water evacuation that simulates low adherence conditions (Low).

To characterize these surfaces, a braking operation was performed by locking the wheels to measure the average deceleration obtained using the IMU installed in the vehicle, obtaining an approximate value of adhesion with $D_x = 0.9$ and $D_x = 0.6$ for a high and low adhesion condition respectively. These values would allow obtaining the optimal slip levels using the tire model presented before.

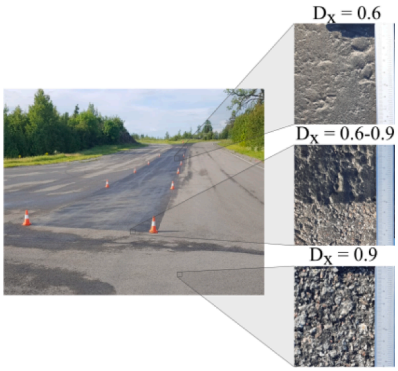

Fig. 15. DriveLab: Arlanda test track 1 (Road Transition).

To evaluate the on-line learning capability of the SNN-based ABS in real conditions, the weights of connections of the neural network were initially selected at the beginning of the test according to EPH to warrant stable behavior.

As expected, in the first test, the algorithm showed a non-optimal performance. Nevertheless, as consecutive tests were performed, the network modified the weights by modulating the applied pressure as observed in simulations carried out in the previous section.

The first tests were carried out on a high-adhesion surface, in which it was observed how deceleration increased considerably as experimentation proceeded (Iteration 1 → Iteration 3 → Iteration 5). Figs. 16 and 17 show the velocity profile during braking in simulation and experimentation phases respectively. It can be seen that a similar braking time for the same number of iterations was obtained in both cases. Although in real tests the speeds of the real vehicle oscillated more than of the simulated vehicle, this effect can be attributed to the tire model used, since only 4 parameters $\{B_x, C_x, D_x, E_x\}$ were used to model the real behavior of the tire. Even so, the controller demonstrates its adaptability by showing a similar response and by correctly controlling the slip level (Fig. 18 and Fig. 19). Similar to the braking torque evolution observed in previous simulations (Fig. 13), the experimental braking pressure (Fig. 20) also increases in each iteration until it stabilizes.

It should be noted that the learning factor (μ_l) used for experimentation is higher than for simulation, so only 5 iterations were needed to obtain a value close to the optimal one. This allows for performing fewer trials although it slightly penalizes the response obtained. The learning algorithm modifies the weights of the network to optimize braking, giving place to a higher deceleration. It has to be remarked that the variation of the weights occurs over a preset structure based on reflex arcs. Consequently, the controller will not become unstable. Thanks to the bio-inspired structure, learning can be performed on-line by means of on-line learning, avoiding wheel locking while searching for the optimal braking level. This highlights the importance of the prefixed neural structure that provides the necessary robustness.

Figs. 21 and 22 show the results of a surface transition from low to high at mid-brake. As it can be observed, the algorithm performs transition without large oscillations, reducing the braking distance as previously presented in the simulations.

Table 6 shows a summary of the results. A comparison has been made between the proposed algorithm based on SNN and a threshold control-based one for high and low surfaces. The results obtained indicate that the proposed algorithm obtains a better deceleration in all situations. The difference is even greater than in simulation since the threshold control algorithm is not able to adjust to real behavior. This is due to the low speed (6 m/s) of the test, making control even harder as adhesion increases.

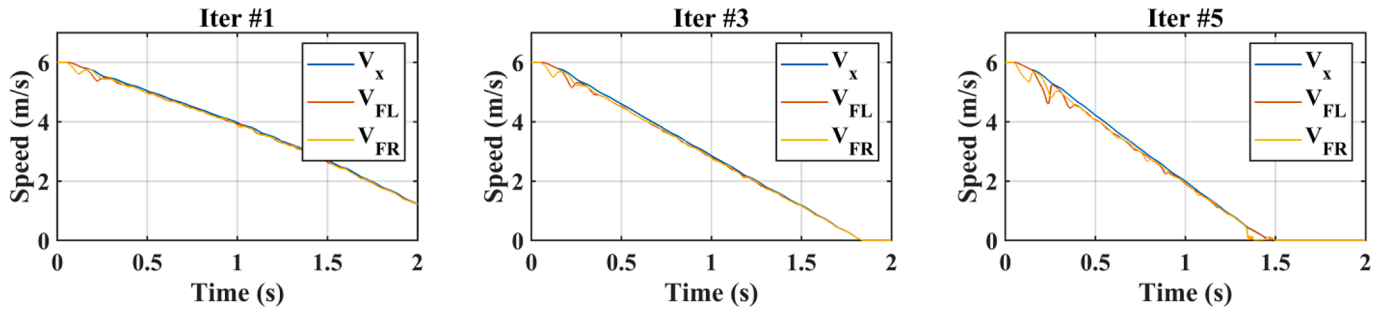

Fig. 16. Vehicle speeds under on-line learning process (Simulation).

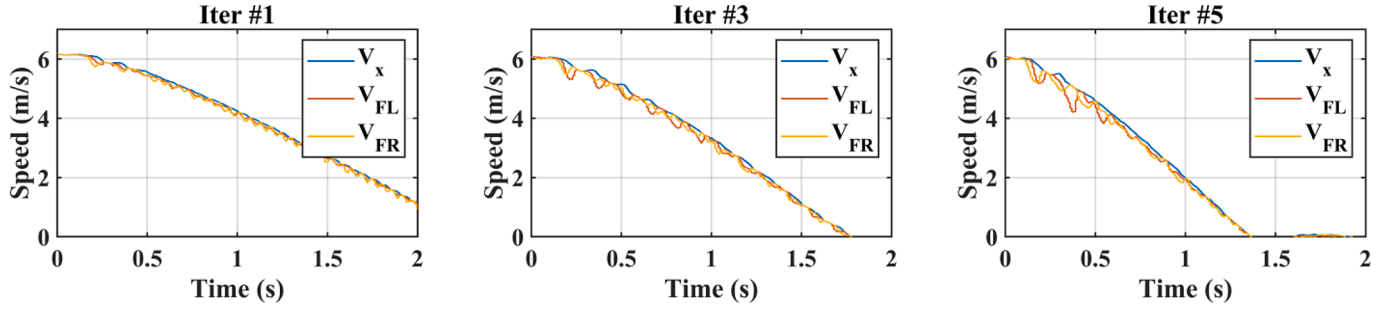

Fig. 17. Vehicle speeds under the on-line learning process (Experimentation).

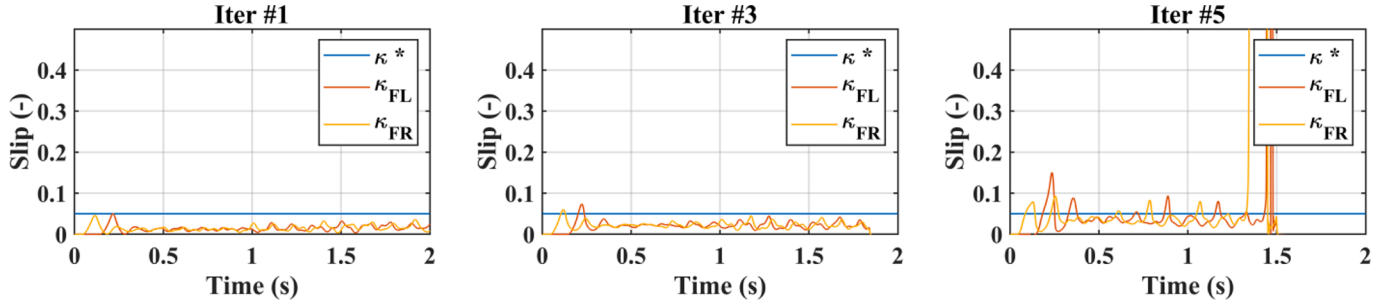

Fig. 18. Longitudinal slip under the on-line learning process (Simulation).

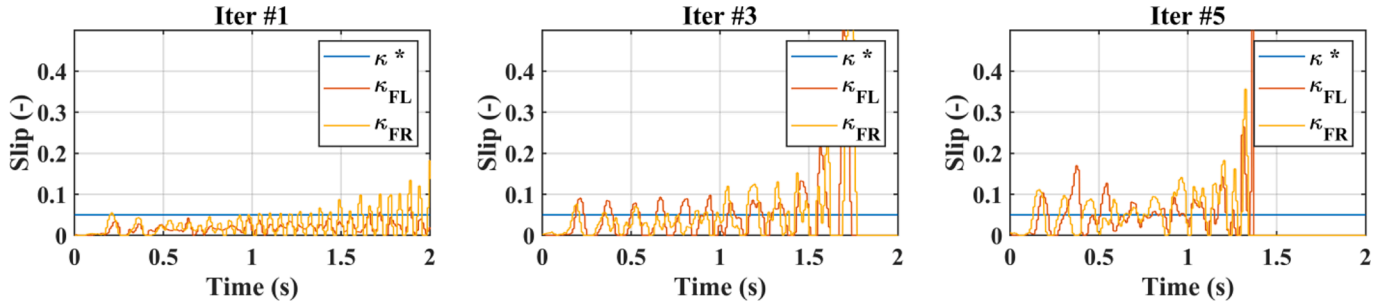

Fig. 19. Longitudinal slip under the on-line learning process (Experimentation).

8. Conclusions

This work has tackled a novel approach to establishing a new strategy of vehicular control inspired by biological systems that increases safety in emergency braking scenarios. The biological-like control has been performed by means of an SNN endowed with on-line learning capability based on STDP modulation. Thus, it has been possible to replicate the natural ability to adapt to external disturbances or changes in system dynamics that biological systems have.

A slip control algorithm based on Spiking Neural Networks has been proposed. This control is able to adapt to varying conditions of tire-road contact dynamics. The combination of classification and control in a single neural structure demonstrates the capability of performing full neural control using the proposed neuron model. Results obtained in simulations and experimentations proved the effectiveness of the proposed algorithm compared to a state-of-the art controller widely used in ABS of passenger vehicles. Furthermore, the ability of the neural structure to learn and adapt to unknown conditions during simulation and

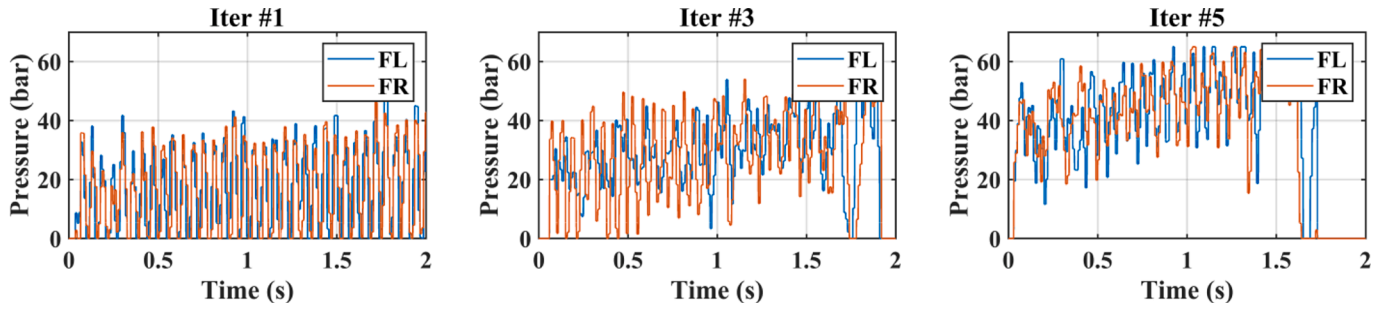

Fig. 20. Braking torque under the on-line learning process (Experimentation).

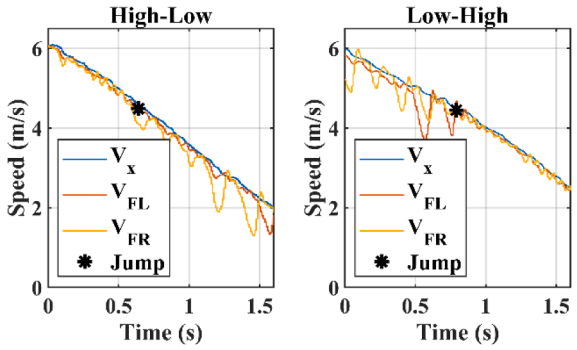

Fig. 21. Vehicle speeds after training for different situations (Experimentation).

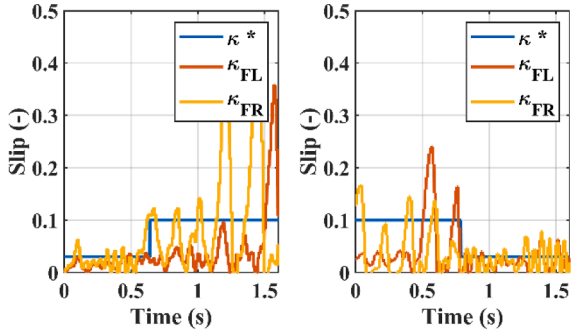

Fig. 22. Longitudinal slip after training for different situations (Experimentation).

Table 6
Steady State KPIs (6 m/s) (Simulation and Experimentation).

	D_x	ABSIP THR	ABSIP SNN	PTP THR	PTP SNN
Simulation	0.9 (High)	96.5%	**91.3%**	0,17	**0,15**
	0.6 (Low)	95.67%	**89.08%**	0.81	**0.27**
Experimentation	0.9 (High)	99.12%	**97.5%**	0,63	**0,10**
	0.6 (Low)	98.56	**94.91%**	0,74	**0,12**

testing is remarkable.

Real tests verified the effective performance of the control algorithm. Future research will be focused on experimental validation of the EKF and combined neural control response. This task requires a large number of road scenarios and types of vehicles. Similarly, the robustness of the controller will be further improved, taking into account variations in system parameters, such as brake system components or tires. On the whole, the integration of the EKF estimation process in the same neural control and classification network would represent an end-to-end neural process. Therefore, this is the next challenge to be developed in vehicular control with SNN.

Control algorithms such as torque vectoring, brake blending, stability control, or steer-by-wire use a similar structure and would benefit from the work presented. Other high-level control algorithms such as advanced driver assistance systems (ADAS) or those needed for a self-driving car are clear targets for an extension of the proposed algorithm.

Declaration of Competing Interest

The authors declare the following financial interests/personal relationships which may be considered as potential competing interests: Juan A. Cabrera reports financial support was provided by Spain Ministry of Science Technology and Innovation.

Data availability

No data was used for the research described in the article.

Acknowledgements

This work was partly supported by the Ministry of Science and Innovation under grant PID2019-105572RB-I00, partly by the Regional Government of Andalusia under grant UMA18-FEDERJA-109, and partly by the University of Malaga as well as the KTH Royal Institute of Technology and its initiative, TRENoP.

References

[1] K. Reif. Brakes, Brake Control and Driver Assistance Systems. Springer Vieweg. 2014.
[2] F. Pretagostini, L. Ferranti, G. Berardo, V. Ivanov, and B. Shyrokau. “Survey on Wheel Slip Control Design Strategies, Evaluation and Application to Antilock Braking Systems.” IEEE Access, vol. 8, no. Section III. 10951–10970. 2020.
[3] H.B. Pacejka, Tire and Vehicle, Dynamics (Pembroke, Ont.) (2012).
[4] V. Ivanov, D. Savitski, B. Shyrokau, A Survey of Traction Control and Antilock Braking Systems of Full Electric Vehicles with Individually Controlled Electric Motors, IEEE Transactions on Vehicular Technology 64 (9) (2015) 3878–3896.
[5] H. Jing, Z. Liu, H. Chen, A switched control strategy for antilock braking system with on/off valves, IEEE Transactions on Vehicular Technology 60 (4) (2011) 1470–1484.
[6] J.P. Fernandez, M.A. Vargas, J.M.V. Garcia, J.A.C. Carrillo, J.J.C. Aguilar, Coevolutionary Optimization of a Fuzzy Logic Controller for Antilock Braking Systems under Changing Road Conditions, IEEE Transactions on Vehicular Technology 70 (2) (2021) 1255–1268.
[7] A. Amirkhani, M. Shirzadeh, M. Molaie, An Indirect Type-2 Fuzzy Neural Network Optimized by the Grasshopper Algorithm for Vehicle ABS Controller, IEEE Access 10 (2022) 58736–58751.
[8] H. Wang, S. Wu, Q. Wang, Global Sliding Mode Control for Nonlinear Vehicle Antilock Braking System, IEEE Access 9 (2021) 40349–40359.
[9] T.T. Lee, C.F. Hsu, S. Lee, Neural-Network Hybrid Control for Antilock Braking Systems, IEEE Transactions on Neural Networks 14 (2) (2003) 351–359.
[10] D. Tavernini, et al., An Explicit Nonlinear Model Predictive ABS Controller for Electro-Hydraulic Braking Systems, IEEE Transactions on Industrial Electronics 67 (5) (2020) 3990–4001.

[11] R. Mircea-Bogdan, P. Radu-Emil, Data-driven model-free slip control of anti-lock braking systems using reinforcement Q-learning, Neurocomputing 275 (2018) 317–329.
[12] A. Sassella, V. Breschi, S. Formentin, S.M. Savaresi, A data-driven switching control approach for braking systems with constraints, Nonlinear Analysis: Hybrid Systems 46 (2022) 101220.
[13] K. Voutsas, J. Adamy, A Biologically Inspired Spiking Neural Network for Sound Source Lateralization, IEEE Transactions on Neural Networks 18 (6) (2007) 1785–1799.
[14] X. Wang, X. Lin, X. Dang, Supervised learning in spiking neural networks: A review of algorithms and evaluations, Neural Networks 125 (2020) 258–280.
[15] E.M. Izhikevich, "Simple Model of Spiking Neurons" 14 (6) (2003) 1569–1572.
[16] J.A.K. Ranjan, T. Sigamani, J. Barnabas, A novel and efficient classifier using spiking neural network, The Journal of Supercomputing 76 (9) (2020) 6545–6560.
[17] A. Arriandiaga, E. Portillo, J.I. Espinosa-Ramos, N.K. Kasabov, Pulsewidth Modulation-Based Algorithm for Spike Phase Encoding and Decoding of Time-Dependent Analog Data, IEEE Transactions on Neural Networks and Learning Systems 31 (10) (2020) 3920–3931.
[18] I.M. Comşa, K. Potempa, L. Versari, T. Fischbacher, A. Gesmundo, J. Alakuijala, Temporal coding in spiking neural networks with alpha synaptic function, IEEE Transactions on Neural Networks and Learning Systems (2019) 1–14.
[19] B. Rueckauer, I.A. Lungu, Y. Hu, M. Pfeiffer, S.C. Liu, Conversion of continuous-valued deep networks to efficient event-driven networks for image classification, Frontiers in Neuroscience 11 (DEC) (2017) 1–12.
[20] J. Su, J. Li, HF-SNN: High-Frequency Spiking Neural Network, IEEE Access 9 (2021) 51950–51957.
[21] T. DeWolf, P. Jaworski, and C. Eliasmith. "Nengo and Low-Power AI Hardware for Robust, Embedded Neurorobotics," Front Neurorobot. vol. 14, no. October. 1–11. 2020.
[22] C. Liu, W. Shen, L. Zhang, Y. Du, Z. Yuan, Spike Neural Network Learning Algorithm Based on an Evolutionary Membrane Algorithm, IEEE Access 9 (2021) 17071–17082.
[23] G. Lagani, F. Falchi, C. Gennaro, G. Amato, Hebbian semi-supervised learning in a sample efficiency setting, Neural Networks 143 (2021) 719–731.
[24] A. Taherkhani, A. Belatreche, Y. Li, L.P. Maguire, A Supervised Learning Algorithm for Learning Precise Timing of Multiple Spikes in Multilayer Spiking Neural Networks, IEEE Transactions on Neural Networks and Learning Systems 29 (11) (2018) 5394–5407.
[25] D.E. Shulz, D.E. Feldman. Spike Timing-Dependent Plasticity. Elsevier Inc. 2013.
[26] E.M. Izhikevich, Solving the distal reward problem through linkage of STDP and dopamine signaling, BMC Neuroscience 8 (S2) (2007).
[27] K. Doya. "What are the computations of the cerebellum, the basal ganglia and the cerebral cortex ?" vol. 12. 961–974. 1999.
[28] X. Guan, L. Mo, Unsupervised Conditional Reflex Learning Based on Convolutional Spiking Neural Network and Reward Modulation, IEEE Access 8 (2020) 17673–17690.
[29] A. Tavanaei, M. Ghodrati, S.R. Kheradpisheh, T. Masquelier, A. Maida, Deep learning in spiking neural networks, Neural Networks 111 (2019) 47–63.
[30] S.R. Nandakumar, I. Boybat, M. Le Gallo, E. Eleftheriou, A. Sebastian, B. Rajendran, Experimental Demonstration of Supervised Learning in Spiking Neural Networks with Phase-Change Memory Synapses, Scientific Reports 10 (1) (2020) 1–11.
[31] Raychaudhuri T, Hamey L, Bell R. "Neural network control using active learning." In Control 95: meeting the challenge of Asia Pacific growth, Australia. 369-373. 1995.
[32] Hasenjäger M., & Ritter H. "Active Learning in Neural Networks," Studies in Fuzziness and Soft Computing, 137–169. 2002.
[33] A. G. Feldman. "Referent control of action and perception". 2015.
[34] P. Taylor, C. Cheng, D. Cebon, Dual extended Kalman filter for vehicle state and parameter estimation, Veh. Syst. Dyn. Int. J. Veh. Mech. Mobil 37–41 (2010).
[35] Y. Oniz, O. Kaynak, Control of a direct drive robot using fuzzy spiking neural networks with variable structure systems-based learning algorithm, Neurocomputing 149 (2015) 690–699.
[36] E.M. Izhikevich, Which Model to Use for Cortical Spiking Neurons? IEEE Transactions on Neural Networks 15 (5) (2004) 1063–1070.
[37] E. Henneman, Relation between Size of Neurons and Their Susceptibility to Discharge, Science 126 (3287) (1957) 1345–1347.
[38] H. Tan, S. Majumdar, Q. Qin, J. Lahtinen, S. van Dijken, Mimicking Neurotransmitter Release and Long-Term Plasticity by Oxygen Vacancy Migration in a Tunnel Junction Memristor, Advanced Intelligent Systems 1 (2) (2019) 1900036–1902019.
[39] K.A. Buchanan, J.R. Mellor, The activity requirements for spike timing-dependent plasticity in the hippocampus, Frontiers in Synaptic Neuroscience 2 (JUN) (2010) 1–5.
[40] J. Pérez Fernández, M. Alcázar Vargas, J.M. Velasco García, J.A. Cabrera Carrillo, J.J. Castillo Aguilar, A biological-like controller using improved spiking neural networks, Neurocomputing 463 (2021) 237–250.
[41] A.G. Feldman, L.Z. Eye, Head movements and vestibulo-ocular reflex in the context of indirect, referent control of motor action, Journal of Neurophysiology (2020).
[42] E. Šabanovi, V. Žuraulis, O. Prentkovskis, and V. Skrickij. "Identification of Road - Surface Type Using Deep Neural Networks for Friction Coefficient Estimation." 2020.
[43] D. Savitski, V. Ivanov, B. Shyrokau, T. Pütz, J. De Smet, J. Theunissen, Experimental investigations on continuous regenerative anti-lock braking system of full electric vehicle, Int.J Automot. Technol. 17 (2) (2016) 327–338.
[44] S.E. Haggerty, W.M. King, The Interaction of Pre-programmed Eye Movements with the Vestibulo-Ocular Reflex, Frontiers in Systems Neuroscience 12 (2018) 1 13.
[45] E.R. Kandel, W, The molecular biology of memory storage: A dialogue between genes and synapses", Science 294 (5544) (2001) 1030–1038.
[46] U. Kiencke, L. Nielsen, Automotive Control Systems for Engine, Driveline, and Vehicle, Springer-Verlag, Berlin Heidelberg, Berlin, Heidelberg, 2005.

Javier Pérez, received a B.S. and a M.S. in Industrial Engineering, from the University of Malaga, Spain. He is Ph.D. in Mechanical Engineering and is a researcher of Mechanical Engineering at the University of Malaga. His research interests include modeling and control of vehicle safety systems, electric vehicles and spiking neural networks.

Manuel Alcázar, received a B.S. in Mechanical Engineering and a M.S. in Industrial Engineering from the University of Jaén, Spain. He is Ph.D. in Mechanical Engineering and is a researcher of Mechanical Engineering at the University of Malaga. His research interests include electric vehicles and tire dynamics.

Ignacio Sánchez, received a B.S. in Mechanical Engineering and a M.S. in Mechatronic Engineering from the University of Malaga, Spain. He is a researcher of Mechanical Engineering at the University of Malaga. His research interests include vehicle dynamics and parameter estimation.

Juan A. Cabrera, received a B.S. in Mechanical Engineering, a M.S. in Computer Science, and a Ph.D. Degree in Mechanical Engineering from the University of Malaga, Spain. He is Full Professor of Mechanical Engineering at the University of Malaga. His research interests include modeling and control of vehicle safety systems and parameter estimation, advanced vehicle systems, genetic algorithms applied to synthesis of mechanisms, tire models and control systems, and spiking neural networks.

Mikael Nybacka, received his Mechanical Engineering degree in 2005 at Lulea University of Technology where he also received his Ph.D. degree in 2009. He currently works at KTH Royal Institute of Technology as an Associate Professor in Vehicle Dynamics engaged in research and education in vehicle dynamics and applied vehicle dynamics control. He is also responsible for the Research Concept Vehicles at Integrated Transport Research Lab at KTH. His research focus is vehicle validation and driver-vehicle interaction but also urban vehicle concepts, autonomous vehicles, fault-tolerant control, and over-actuated vehicles.

Juan J. Castillo received a B.S., M.S., and Ph.D. Degree in Mechanical Engineering from the University of Malaga, Spain. He is Full Professor of Mechanical Engineering at the University of Malaga. His research interests include vehicle dynamics, modeling and control of vehicle safety systems, vehicle testing, optimization algorithms and parameter estimation.