# We are IntechOpen,
# the world's leading publisher of
# Open Access books
# Built by scientists, for scientists

## 6,300
Open access books available

## 171,000
International authors and editors

## 190M
Downloads

## 154
Countries delivered to

Our authors are among the

## TOP 1%
most cited scientists

## 12.2%
Contributors from top 500 universities

**BOOK CITATION INDEX**
CLARIVATE ANALYTICS
INDEXED

**WEB OF SCIENCE™**

Selection of our books indexed in the Book Citation Index
in Web of Science™ Core Collection (BKCI)

## Interested in publishing with us?
## Contact book.department@intechopen.com

**Chapter**

# PID-like Fuzzy Controller Design for Anti-Slip System in Quarter-Car Robot

*José R. García-Martínez, Edson E. Cruz-Miguel,*

*Juvenal Rodríguez-Reséndiz, Luis D. Ramírez-González*
*and Miguel A. Rojas-Hernández*

## Abstract

The design strategy of an adaptive Proportional-Integral-Derivative (PID)-like fuzzy controller for an anti-slip Quarter-Car robotic system is proposed. The proposed control system is constructed by two loops, an external one for lineal speed control and an internal loop for current control. A motion profile is used to follow a trajectory. The slip is computed, such as the difference between the linear velocity given by an S-curve velocity profile and the longitudinal speed calculated according to the rotational speed of the Quarter-Car tire. This difference is the input of the external control loop. Whether the slip is significant, the slave controller must do that both velocities go at the same speed controlling the current of the direct current (DC) motor. On the other hand, the mathematical model of a tire coupled to the DC-motor model is presented to simulate the system and controller response. To test the robustness of the system, different scenarios are presented where the slip coefficient varies depending on the work surface. Three surfaces were selected to test the performance of the controller, dry, wet, and icy surfaces, while the system had a trajectory.

**Keywords:** fuzzy control, slip-control, traction controller, trajectory planning

## 1. Introduction

The human being shows the difficulty in making decisions when there is imprecise information. Fuzzy logic, developed by Lofti Asker Zadeh in 1965, allows emulating of human reasoning and making correct decisions despite the information [1, 2]. It is considered a flexible tool that is based on linguistic rules dictated by experts, composed of a set of mathematical principles based on degrees of membership, whose function is to model information [3]. This modeling is done based on linguistic rules that approximate a function through the relationship between the inputs and outputs of the system. This logic presents membership ranges within an interval between zero and one, unlike conventional logic, in which the content is limited to two values: zero and one [4].

Control systems, on the other hand, are an arrangement of physical components linked or related in such a way that they command, direct or regulate the same system or another [5]; control systems are classified as open-loop control systems and [6] closed-loop control systems. Systems in which the output has no effect on the control action are called open-loop control systems. Systems that maintain a given relationship between the output and the reference input, comparing them and using the difference as a means of control, are called closed-loop control systems or feedback control systems [7, 8].

Fuzzy control is created from the combination of fuzzy logic and control systems techniques, which can be considered an expert closed-loop system in real time, implemented from the experience of an operator or process engineer unfamiliar with the process. It lends itself to being easily expressed in situation-action rules instead of differential equations [9].

Controllers based on fuzzy logic or fuzzy systems are represented by propositional rules *if-then* that can provide an understandable and easy-to-use knowledge representation [10]. This can be seen as a high-level programming language, where the program consists of conditional rules and the compiler or interpreter results in a nonlinear control algorithm, so programming through qualitative statements, represented by of *if-then* statements, to obtain a program that works in quantitative domains, provided by signals from sensors and actuators is the basis of fuzzy control [11]. Intuitively, this implies a loss of information, because there is no single translation from a qualitative entity to a quantitative representation, except in some special cases.

Traction control systems (TCSs) have come to revolutionize the behavior of automobiles as we know them today; however, thanks to the performance shown in human-crewed vehicles, it is possible to open a branch of the study of the traction controllers for mobile robots. The implementation of traction controllers in mobile robots seeks to improve the performance of the robots either due to energy consumption or the accuracy of arrival at the desired point [12]. The energy consumption in mobile robots depends significantly on the adhesion between the tires and the ground; therefore, the greater the bonding, the better stability in autonomous driving produced. This can be translated into more excellent stability in the system. Traction, on the other hand, is considered a vehicular propulsive force produced by the friction present between the tire and the surface. The inherent friction characteristics are nonlinear and uncertain, making TCSs have a high degree of [13] design.

In robotics, these types of systems are not relatively new but have not been fully addressed due to the scarcity of processing tools; current advances in embedded systems have allowed techniques from different areas to be implemented in [14] robotics. TCSs applied to mobile robots are based on modern controllers and nonlinear sliding mode controllers based on state observers [13–15].

Being wholly linked to friction, traction exhibits nonlinear behavior over time, so modern controllers must append a matrix representing the approximate nonlinear coefficients of friction to act on the control signal. In combination with traction controllers, mobile robots allow the robot to move freely on spaces with smooth, wet, and slippery surfaces. Traction control lets the robot wheels turn at similar speeds on surfaces with near-zero coefficients of friction, and when an imbalance exists, the traction controller must be activated to prevent unnecessary plant slippage, which translates to not reaching the working path [16].

Some authors have worked on different strategies related to slip control and traction control for mobile robots. For instance, [17] introduces an adaptive control strategy for a tracked mobile robot that compensates for the longitudinal slip to reach

a trajectory. Another controller based on the dynamic and kinematic model with slip is presented in [18]. [19] proposes an algorithm for optimal slip control of wheeled robots with the trade-off between traction and energy consumption based on observing a change in a robot's velocity on different soil surfaces. An optimal slip ratio control using a current sensing method is presented in [20]. The controller consisted of a fuzzy PID structure. In [21] shows a slip control for a nonholonomic wheeled robot where the kinematic model of a differential-driven wheeled mobile robot is used to solve the trajectory tracking problem using fuzzy and optimal fuzzy logic. The trajectory tracking problem of a wheeled mobile robot which is actuated by two independent electrical motors is attacked in [22]. A simple non-mode-based fuzzy logic controller is used to reduce the tracking error provoked by the slippage.

As the reader can see, there are many works with different perspectives to attack the slip control in other structures. Hence, this work proposes a new methodology to control the slippery of a Quarter-Car robot using an internal loop based on fuzzy logic inference to compute the gains of a Proportional-Integral (PI) structure. The slip is calculated, such as the difference between the linear velocity given by an S-curve velocity profile and the longitudinal speed calculated according to the rotational speed of the Quarter-Car tire. This difference is the input of the external control loop. Whether the slip is significant, the slave controller must do that both velocities go at the same speed controlling the current of the DC motor. The external loop controls the angular velocity so that the linear velocity has the same magnitude. The outer loop uses an adaptive PID-like fuzzy controller structure.

The chapter is divided into the following sections. Section 2 describes the dynamic model of the test bench. Section 3 presents the methodology for self-tuning PID-like fuzzy controller. In Section 4, the simulations and results of the experimentation are presented, and, finally, in Section 5, the conclusions are given.

## 2. Dynamic model of one-wheel robot

This section presents the fundamental parts that compound the Quarter-Car robot structure. **Figure 1** shows the platform used for the project simulations. The platform consists of a wheel–motor coupling that will emulate the behavior of a mobile robot
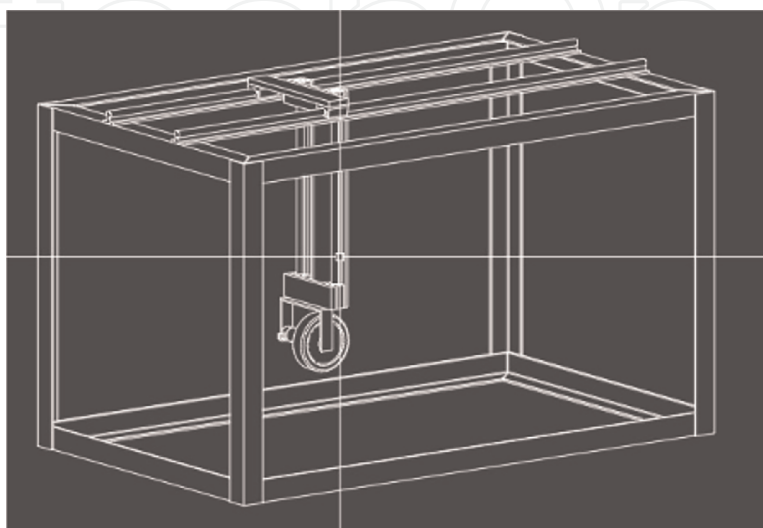


**Figure 1.**
*One-wheel robot for control tests.*

axis when different surfaces interact. The measurements will be carried out: angular velocity, longitudinal velocity, current, and slip. Slippage will be controlled so that the tire does not slip and can correctly follow a trajectory.

## 2.1 DC motor

Direct current (DC) motors are the most common actuators within control systems. It directly provides a rotational movement, and together with the wheels, rails, and cables, it can perform a translational action. The equivalent electrical circuit of the armature and the free body diagram of the rotor are shown in **Figure 2**. It is necessary to obtain a dynamic model that allows a correct analysis. The dynamic model of this servo system depends on the electrical and mechanical characteristics, such as the resistance $R_a$, the inductance $L_a$, the inertia $J$ of the armature, the back electromotive force $v_b$, and the friction $D$.

From **Figure 2**, it is possible to derive the following equations based on Newton's second law for rotational motion and Kirchhoff's second law, Eqs. (1) and (2), respectively.

$$\frac{d\omega_m}{dt} = \frac{K}{J}i_a - \frac{D}{J}\omega_m \tag{1}$$

$$\frac{di_a}{dt} = \frac{e_a}{L_a} - \frac{K}{L_a}\omega_m - \frac{R_a}{L}i_a \tag{2}$$

$$\frac{d\theta_m}{dt} = \omega_m \tag{3}$$

## 2.2 Tire

Tire friction models are also indispensable for accurately reproducing friction forces for simulation purposes. A common assumption in most tire friction models is that the normalized tire friction, Eq. (4), is a nonlinear function of the normalized
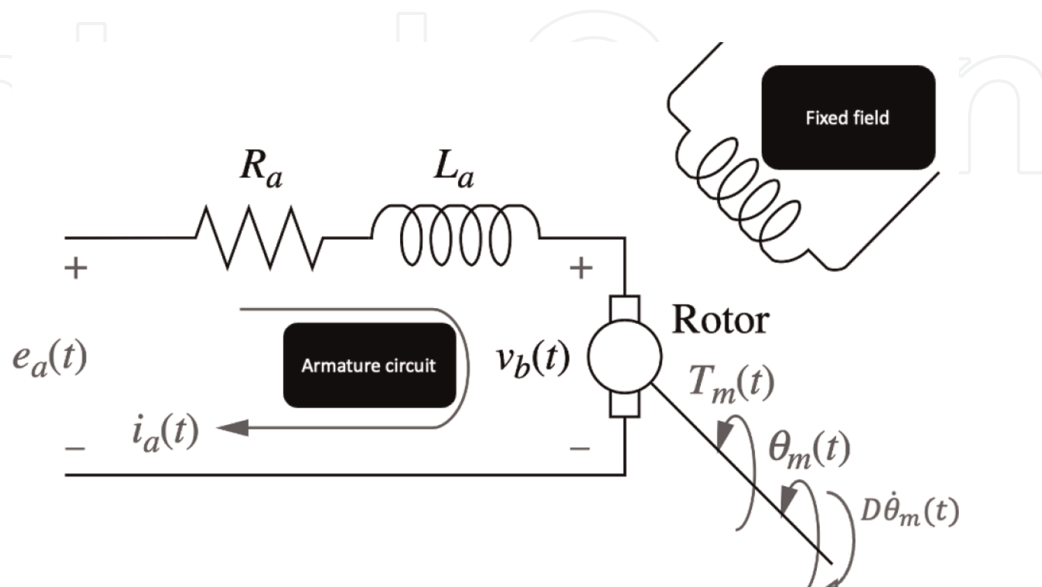


**Figure 2.**
*Dynamic model of a DC motor.*

relative velocity between the surface and the tire (slip coefficient $s$) with a different maximum.

$$\mu = \frac{F}{F_n} = \frac{\text{Friction force}}{\text{Normal forcel}} \tag{4}$$

Furthermore, it is understood that $\mu$ also depends on vehicle speed and road surface conditions, among other factors. This work considers the simplified motion dynamics of a Quarter-Car model. The system is represented by Eqs. (5) and (6).

$$m\dot{v} = F \tag{5}$$

$$J\dot{\omega} = -rF + u \tag{6}$$

Where $m$ is $\frac{1}{4}$ of the mass of the vehicle, and $J$ and $r$ are the inertia and radius of the wheel, respectively. $v$ is the linear speed of the tire, and $\omega$ is the angular speed of the wheel, $u$ is the acceleration or braking torque, and $F$ is the friction force as shown in **Figure 3**.

The most common tire friction models used in the literature are those of algebraic slip/force relationships. They are defined as one-to-one maps (memory-less) between the friction $F$ and the longitudinal slip rate $s$, which is defined in Eq. (7).

$$s = \begin{cases} s_f = \dfrac{r\omega - v}{v} & \text{if } v > r\omega, \ v \neq 0 \text{ for breaking} \\[2mm] s_m = \dfrac{r\omega - v}{r\omega} & \text{if } v < r\omega, \ \omega \neq 0 \text{ for movement} \end{cases} \tag{7}$$

Slippage results from the reduction in the effective circumference of the rim as a result of surface deformation due to the tire rubber's elasticity. This, in turn, implies that the longitudinal velocity $v$ will not be equal to $r\omega$. The absolute value of the slippage is defined in the interval $[0, 1]$. When $s = 0$, there is no slip (pure rotation), while $|s| = 1$ indicates total slip/skid.
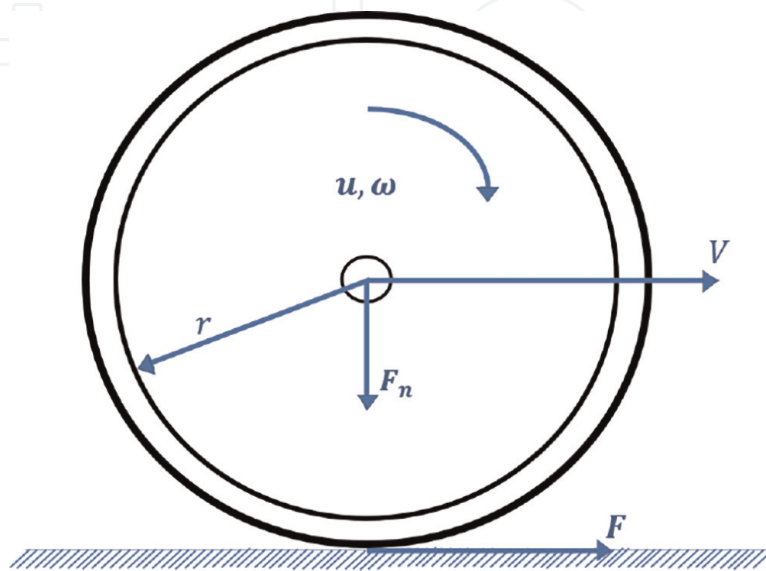


**Figure 3.**
*One-wheel system with concentrated friction.*

### 2.3 Slip-force models

Slip/force models aim to describe the slip movement through its force/surface dependency mapping $F(s) : s \mapsto F$. They can also depend on the vehicle speed $v$, that is, $F(s, v)$, and vary when the characteristics of the road change.

One of the best-known models of this type is the Pacejka model [23], also known as the "Magic Formula". It has been shown that this model agrees adequately with the experimental data obtained under particular conditions of constant linear and angular velocity. Pacejka's model is represented by Eq. (8).

$$F(s) = D \sin(C\arctan(Bs - E(Bs - \arctan(Bs)))) \tag{8}$$

It can be seen that the Eq. (8) contains a set of parameters: $B$, $C$, $D$, and $E$. These parameters depend on the tire's physical properties and the vehicle's dynamic state. In the Eq. (8), $D$ represents the maximum coefficient, $C$ represents the shape coefficient and influences the shape of the curve, $B$ is the stiffness coefficient, and $E$ is the coefficient of curvature [24].

### 2.4 Quarter-car robot model

For the design of the traction controller, the Eqs. (1)–(3) represents the engine dynamics, and the Eqs. (5) and (6) represent the dynamics of the wheel of a differential robot.

**Figure 4** shows the interaction of the dynamic equations of the motor with the wheel of the robot. It is observed that the relationship between the equations is the angular velocity of the motor shaft and the longitudinal movement of the robot, as expressed by the Eq. from (9) to the Eq. (13).

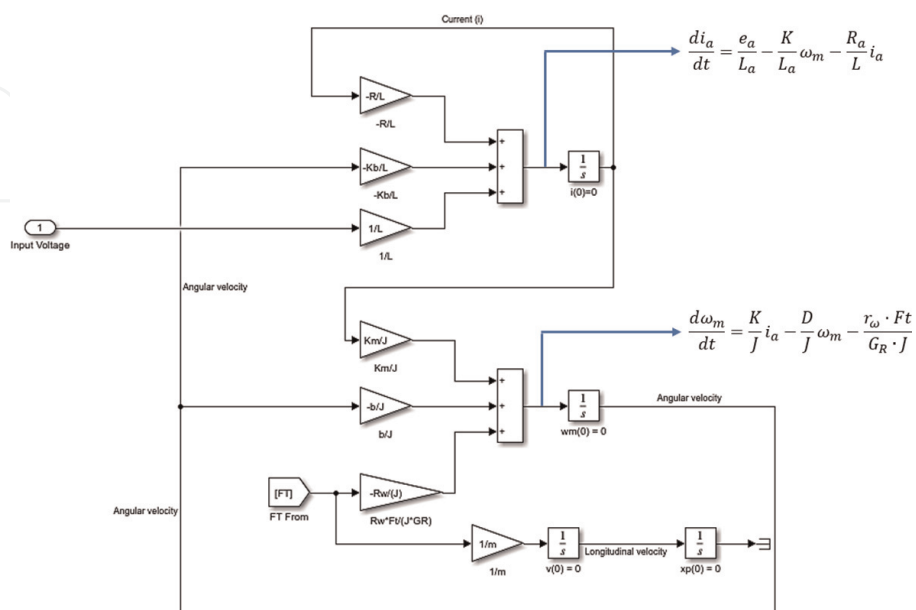$$\frac{di_a}{dt} = \frac{e_a}{L_a} - \frac{K}{L_a}\omega_m - \frac{R_a}{L}i_a \tag{9}$$



**Figure 4.**
*Block diagram of the motor-wheel system from Eqs. (9)–(13).*

$$\frac{d\omega_m}{dt} = \frac{K}{J}i_a - \frac{D}{J}\omega_m - \frac{r_w \cdot Ft}{G_R \cdot J} \tag{10}$$

$$\frac{d\theta_m}{dt} = \omega_m \tag{11}$$

$$\frac{dv}{dt} = \frac{Ft}{m} \tag{12}$$

$$\frac{dx_p}{dt} = v \tag{13}$$

The slip ratio is presented in Eq. (14) and is a function of the angular velocity.

$$s = \frac{\omega_b - \omega_f}{|\omega_f| + tol}, \quad tol \simeq 0 \tag{14}$$

On the other hand, rewriting the Eq. (14) as a function of the radius of the wheel ($r_w$) and the longitudinal velocity ($v$).

$$s = \frac{\omega r_w - v}{|\omega r_w| + tol}, \quad tol \simeq 0 \tag{15}$$

Where $\omega r_w$ and $v$ represent the longitudinal speed of the wheel computed according to the radial speed times the radius of the tire and the longitudinal velocity measured, respectively. To calculate the traction force, the Eq. (16) of Pacejka and Sharp [23] is used.

$$F(s) = D\sin(C\arctan(Bs - E(Bs - \arctan(Bs)))) \tag{16}$$

where $B$, $C$, $D$, and $E$ are constants, $s$ represents the slip value. **Figure 5** shows the implementation of Eqs. (15) and (16) in block diagram form.

On the other hand, if an imbalance is detected between the wheels of the robot, it is necessary to adjust the speed of the wheel that rotates at a higher rate with the one that turns at a lower speed [25]. To carry out this criterion, it is necessary to propose
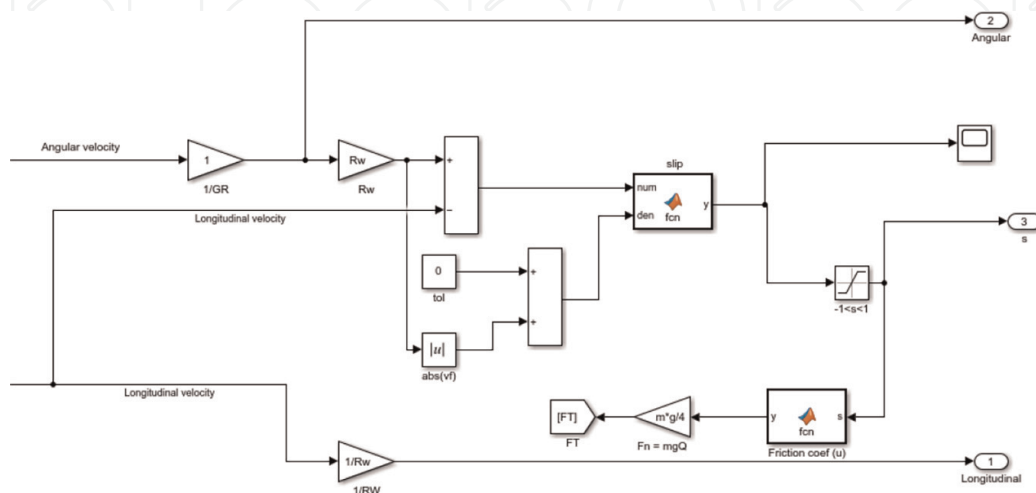


**Figure 5.**
*Block diagram of the motor-wheel system from Eqs. (9)–(13) for the calculation of the traction force from the coefficient of friction.*

behavioral variables that mathematically model the speed adjustment. In order to carry out the previous step, it is required to define a database that protects all the behavior information. This information is the linguistic variable. In addition, the operating range of the controller must be considered, which is translated into modeling the behavior in the decompensation of the wheels.

## 3. PID-like fuzzy controller

The design of controllers based on fuzzy logic plays an essential role in intelligent systems due to the ease of design and implementation, which must be subject to direct collaboration with the person in charge of monitoring the process to be controlled; that is, the design of the controller must be based on the experience of the system operator under certain conditions to establish the linguistic variables that the controller must obey [9, 26]. The fuzzy logic is used to tune the gains of a PID structure controller. The general form of the PID controller is depicted in Eq.(17).

$$u(t) = k_p e(t) + k_i \int_0^t e(\tau) d\tau + k_d \frac{de(t)}{dt} \tag{17}$$

This system is known as a PID-like fuzzy controller [27, 28]. The fuzzification range for computing $k_p$, $k_d$, and $k_i$ gains, in its crisp value form, is selected according to the error, rate of change in error, and the integral of error, respectively. **Figure 6** depicts the structure proposed for the outer loop or master loop. The master loop controls the longitudinal velocity according to the angular velocity.

**Figure 7** presents the structure of the internal loop. It consists of a PI structure capable of controlling the current of the tire coupled to the DC motor. Whether there is an imbalance in the speeds, it means that there is a skid in the wheel. The current must be reduced to equalize the angular velocity with the longitudinal one. That is the aim of the internal structure.

**Figure 8** presents a generalized form to propose the range of operation. As one can see, the three linguistic variables contain seven linguistic values, namely negative big (NB), negative medium (NM), negative small (NS), zero (ZE), positive small (PS), positive medium (PM), and positive big (PB). A "d" and "i" are added at the beginning of each linguistic value to identify if they refer to the derivative or integral linguistic variable. The linguistic values present triangular and Gaussian shapes. For the
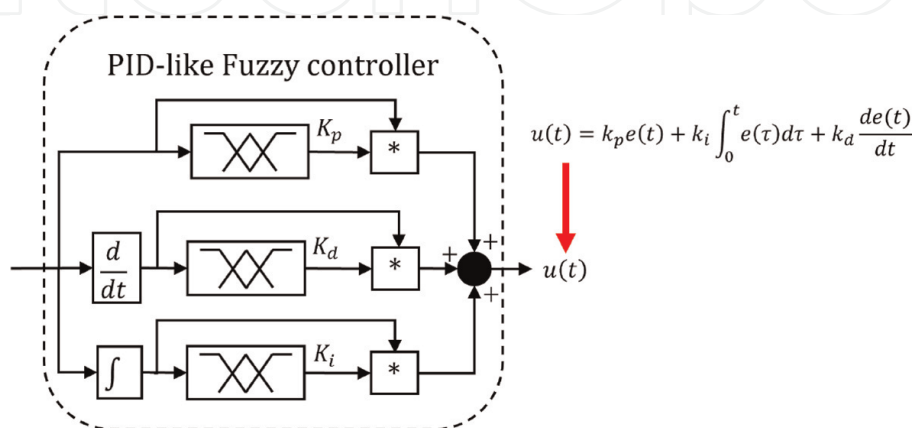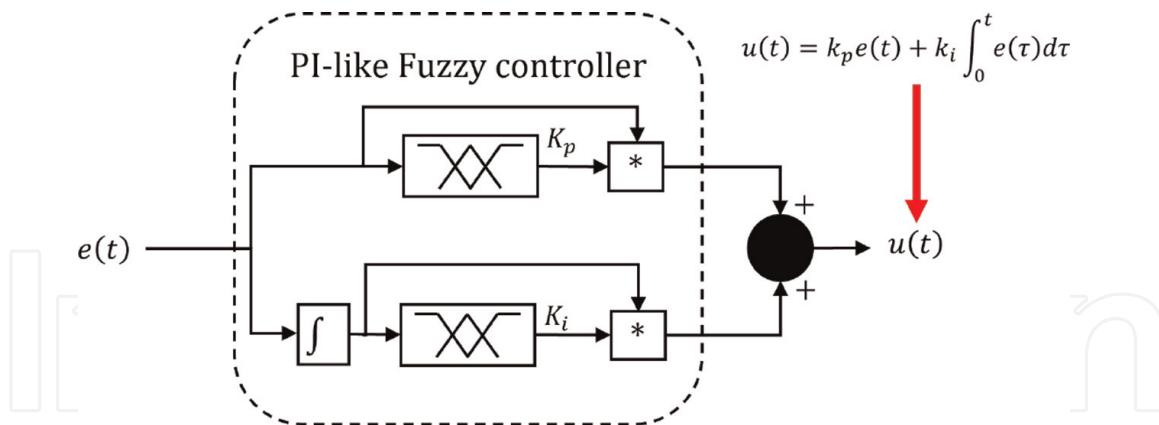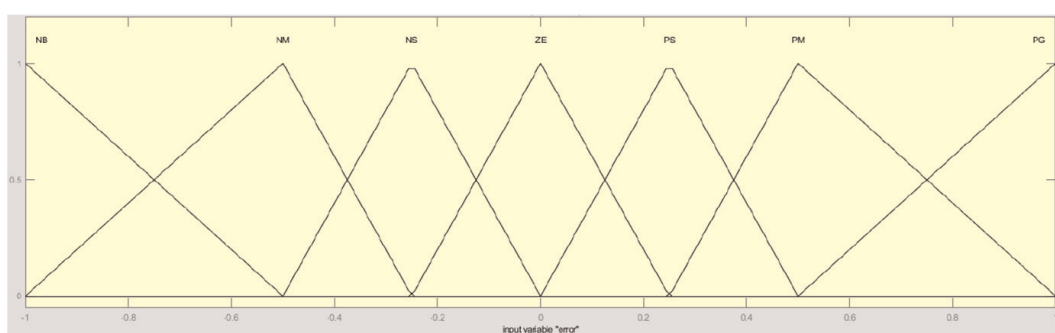


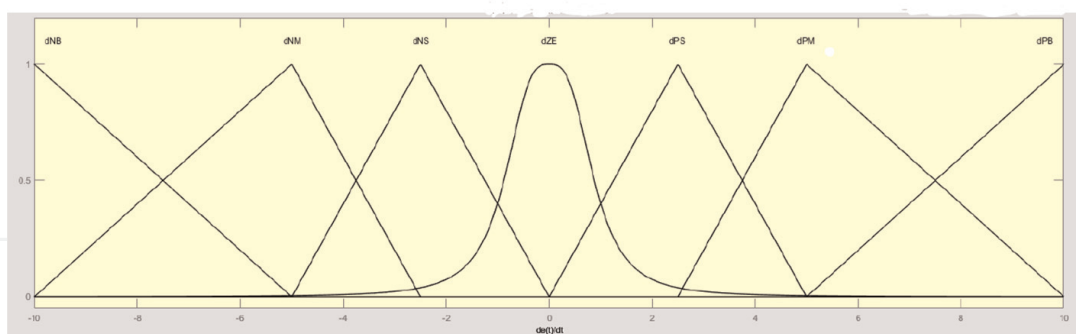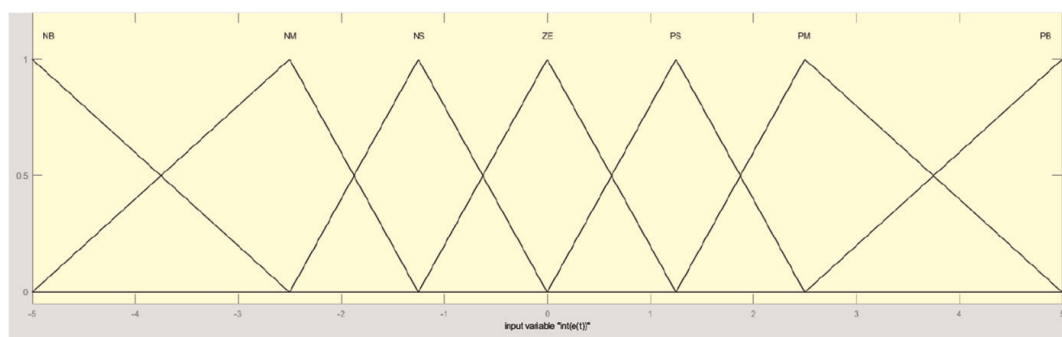**Figure 6.**
*Fuzzy tuner structure.*

**Figure 7.**
*Fuzzy tuner structure.*



(a)



(b)



(c)

**Figure 8.**
*Linguistic variables of the inputs (a) error, (b) derived of the error, and (c) integral of the error.*

linguistic variable of error and integral of error, **Figure 8a** and **b**, respectively, triangular membership functions are used in the design since, according to the control theory, the proportional and integral actions are not as susceptible to noise. On the other hand, in the derivative error linguistic variable, **Figure 8b**, it is used a Gaussian membership function as a zero value to smooth the values of the derivative gain. The ranges of the fuzzyfication stages are $[-1, 1]$, $[-10, 10]$, and $[-5, 5]$, for $k_p$, $k_d$, and $k_i$, respectively.

**Tables 1–3** present the fuzzy associative matrices (FAM) for computing $k_p$, $k_d$, and $k_i$. The inference process corresponds to a one-to-one fuzzy relationship. Likewise, it can be seen that four-linguistic values NM, NS, PS, and PM map to a single value S to calculate the output value for $k_p$. This relationship is used to ensure that the gain values work within a suitable range of values. The value of $K_p$ never takes the zero value in this controller, only an almost zero (AZ) linguistic value. Generally, when the present error is obtained, it enters the fuzzification stage and is evaluated for the rules shown in **Table 1**. The range of values the proportional gain can take is shown in **Figure 9a**. $k_p$ can bring any value between $[1.9, 9.8]$ and depends on the error at an instant of time. It is important to mention that $k_p$ must not be negative for any reason.

Similar to $k_p$, $k_d$ maps four linguistic values, namely dNM, dNS, dPS, and dPM map to a single value S. $k_d$ can be zero (Z) according to the derivative error. The idea of giving a zero value to $k_d$ is to reduce the oscillation caused by the slight variation of the change in error. The range proposed for the $k_d$ gain goes from $[0, 1.4]$.

| e(t)/kp | |
|---|---|
| NB | B |
| NM | S |
| NS | S |
| ZE | AZ |
| PS | S |
| PM | S |
| PB | B |

**Table 1.**
*FAM of $k_P$.*

| $\frac{de(t)}{dt}$/k$_d$ | |
|---|---|
| **dNB** | **B** |
| dNM | S |
| dNS | S |
| dZE | Z |
| dPS | S |
| dPM | S |
| dPB | B |

**Table 2.**
*FAM of $k_d$.*

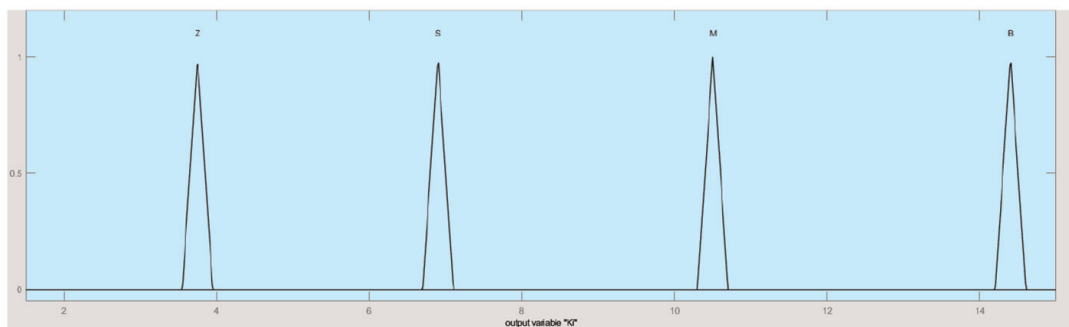| $\int e(t)dt/k_i$ | |
|---|---|
| **iNB** | **B** |
| iNM | M |
| iNS | S |
| iZE | Z |
| iPS | S |
| iPM | M |
| iPB | B |

**Table 3.**
*FAM of $k_i$.*
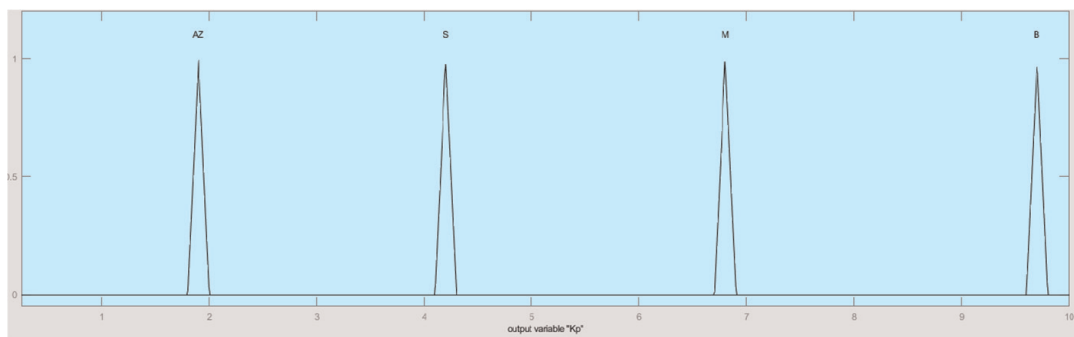


(a)
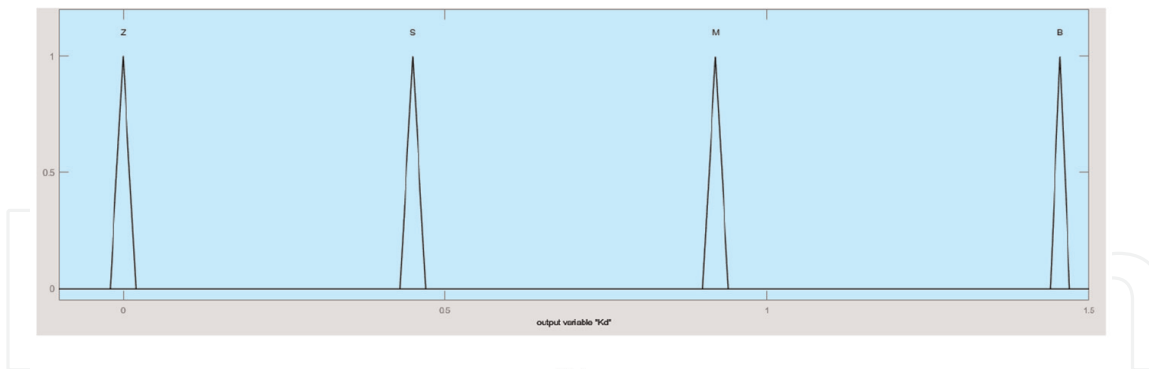


(b)



(c)

**Figure 9.**
*Linguistic variables for the output (a) $k_p$, (b) $k_d$, and (c) $k_i$.*

$k_i$ does not present any alteration in how it computes its value. The range of existence is [3.5,14.5]. The integral action eliminates the steady error; for this particular case, if $k_i$ is big enough, it compensates the error. If the error is small, the integral of the error will also reduce its magnitude considerably.

The singleton is used as a membership function in the defuzzification phase to reduce the computational cost when searching for the profit values. It is important to respect this consideration since there is only one defuzzification phase in the conventional fuzzy controller. In contrast, the proposal for this controller consists of three fuzzy steps, one for each control gain. It is necessary to mention that this process is the same for computing $k_d$ and $k_i$. The centroid method, presented in Eq. 18, is used in the defuzzification stage of each gain.

$$K_{p,d,i}[n] = \frac{\sum_{i=1}^{n} \mu_c(z_i) \cdot z_i}{\sum_{i=1}^{n} \mu_c(z_i)} \tag{18}$$

where $\mu_c(z_i)$ represents degree of membership function, and $z_i$ the position of the singleton.

## 4. Simulations and results

**Figure 10** shows the diagram of the model, which outputs the wheel axis's slip, current, and longitudinal speeds calculated from the wheel's radius and angular velocity. **Tables 4** and **5** display the values of the parameters used for the simulation.
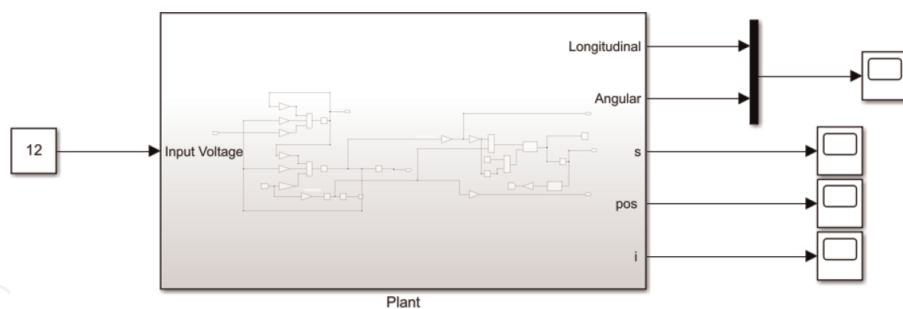


**Figure 10.**
*Diagram model for simulation.*

| Parameter | Value | Units |
|---|---|---|
| $Ra$ | $1.41e^{-1}$ | $\Omega$ |
| $L_a$ | $3e^{-3}$ | $H$ |
| $K_\tau$ | $5.74e^{-3}$ | Volt/rad |
| $K_e$ | $5.74e^{-3}$ | N · m/Amp |
| $b$ | $3.97e^{-6}$ | N · m · s |
| $J$ | $1e^{-4}$ | kg · m |

**Table 4.**
*Parameters of the DC motor.*

| Parameter | Value | Units |
|---|:---:|:---:|
| $G_R$ | 2.5 | — |
| $Q$ | 0.37 | — |
| $r_w$ | $3.2e^{-2}$ | m |
| $tol$ | $1e^{-10}$ | — |
| $m$ | 1.14 | kg |

**Table 5.**
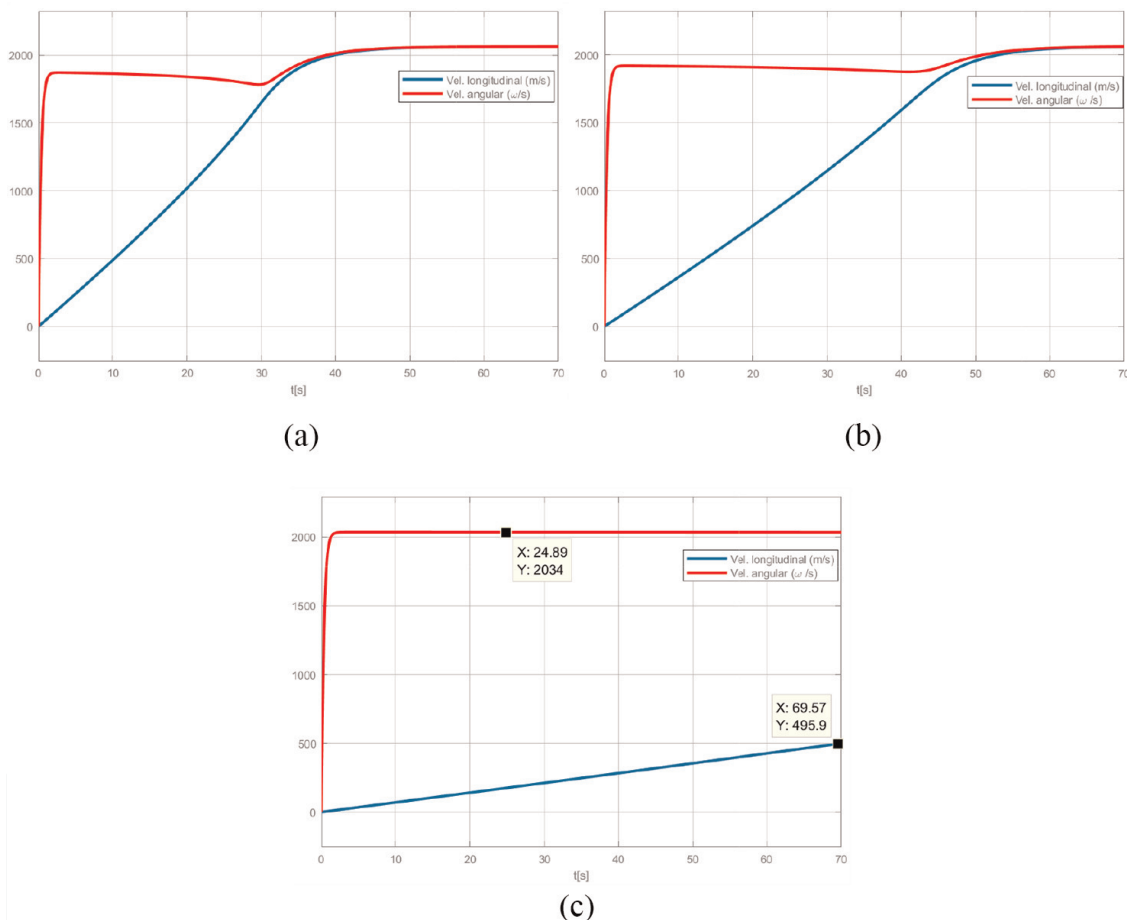*Parameters of the robot's structure.*



**Figure 11.**
*Response of the Quarter-Car robot to surfaces (a) dry asphalt, (b) wet asphalt, and (c) ice.*

The motor values correspond to a DC motor. **Table 5** shows the values that were used for the mechanical structure of the robot.

Assuming that the motor that is coupled to the robot's wheel is fed at 12 V and only the speeds and the slip are measured, we can say that the system is an open-loop representation, whose behavior is shown in **Figure 11** for each of the surfaces presented in **Table 6**. In **Figure 11a**, it can be seen that when the tire crosses a dry asphalt surface, the longitudinal velocity increases considerably and tries to equalize the angular velocity. On the other hand, when the tire crosses a wet asphalt surface, the longitudinal velocity takes longer to equal the angular velocity, as shown in **Figure 11b**. Finally, the angular velocity tends to its maximum permissible value when

| Surface | *B* | *C* | *D* | *E* |
|---------|-----|-----|-----|-----|
| Dry asphalt | 19.25 | 1.65 | 0.92 | 0.6 |
| Wet asphalt | 15.8 | 1.6 | 0.62 | 0.6 |
| Ice | 9.8 | 1.45 | 0.1 | 0.6 |

**Table 6.**
*Parameters for computing the coefficient of friction depending on the surface.*

the tire makes contact with an ice-covered surface. In contrast, the longitudinal rate grows negligibly as a function of time; see **Figure 11c**. This is because the coefficient of friction is very close to zero, which means that the friction is zero, causing the tire to tend to skid without moving considerably longitudinally.

The Algorithm 1 presents the way to compute the friction coefficient used for simulation.

---

**Algorithm 1:**

---

Calculation of the coefficient of friction.
**Función** y = fcn(s)
$B$;
$C$;
$D$;
$E$;
$_{if}$ $s < 0$ **then**
$\quad$| s = abs(s)
$\quad$| $\mu = -D\sin(C\arctan(Bs - E(Bs - \arctan(Bs))))$
**else**
$\quad$| $\mu = D\sin(C\arctan(Bs - E(Bs - \arctan(Bs))))$
**end**
y = $\mu$;

---

The behavior of the robot as a function of longitudinal and radial speed is presented in **Figure 12**, which contains the behavior over time of the difference in rates, better known as sliding. As shown in **Figure 11**, the speeds of the wheel have different magnitudes. This means that the angular speed is much more significant in the first moments than the longitudinal speed. This behavior is without any controller, so there is no device that regulates the speeds from the beginning of the operation. However, in **Figure 12**, it can be seen that the slip tends to zero when the speeds reach the same magnitude on dry and wet asphalt surfaces, **Figure 11**, while for the icy surface, there is always a slip value, and the tire remains rotating without moving longitudinally.

## 4.1 PID-like fuzzy controller results

For the design of the PID-like fuzzy controller, the methodology presented in the section 3 was used; see **Figure 13**.

In order to solve the problem of different speeds, a slip controller will be designed whose reference input is from a seven segments motion profile. The wheel–motor
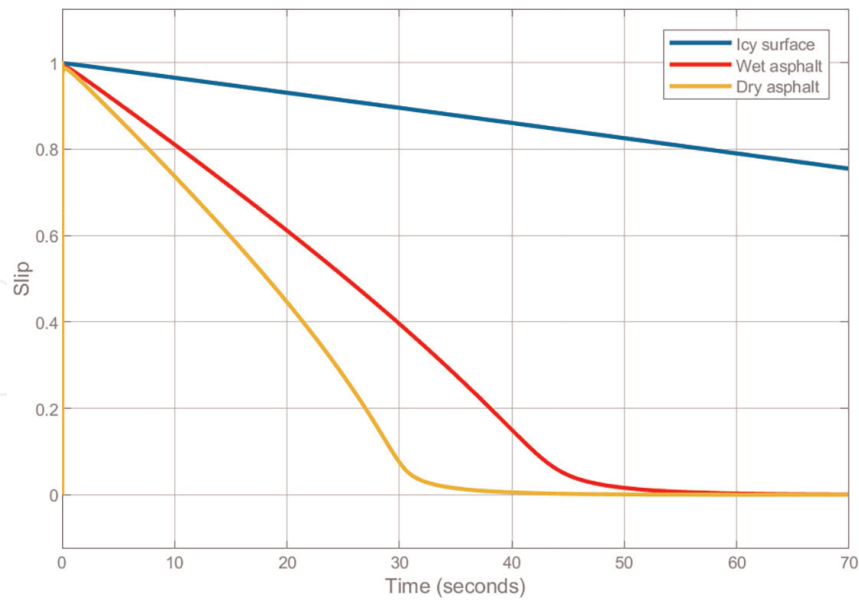
**Figure 12.**
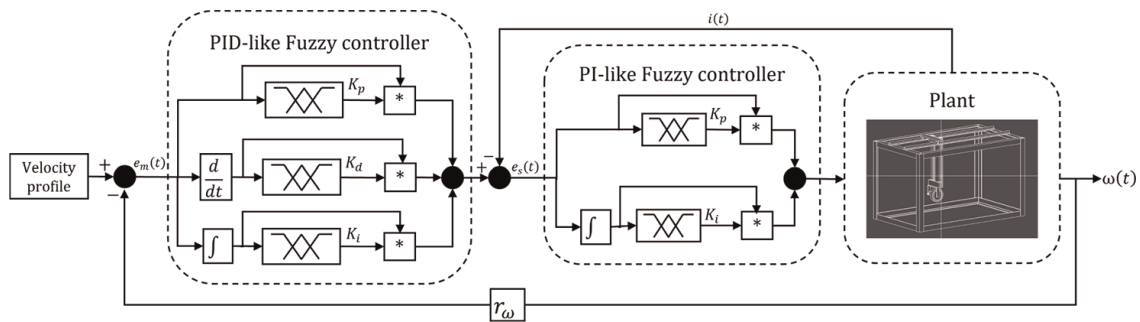*Slip response to a 12 V input.*



**Figure 13.**
*PID-like fuzzy controllers and the Quarter-Car robot structure.*

system must ensure the same angular and longitudinal speed by following the change in reference speed at any time. In **Figure 14**, the velocity profile is shown as input. It goes through two integrators because the jerk profile is being generated, so when integrating once, you get the reference acceleration, and the second time you get the input velocity. The longitudinal velocity is compared concerning the motion profile. The idea is that the rim moves according to the reference and can work with the surfaces presented in **Table 6**. The difference between the reference speed and the longitudinal speed produces the error, which is the input to the master controller. The
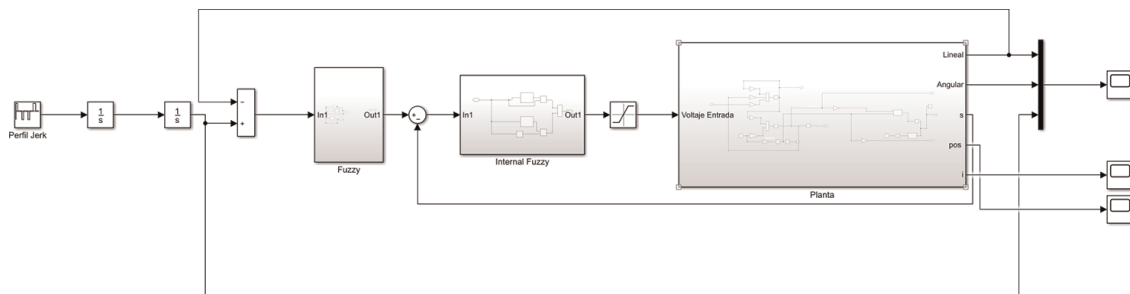


**Figure 14.**
*Fuzzy controller.*

output of this controller is compared to the slip to generate an internal control loop to prevent slippage.

The Quarter-Car and the control system used for the simulation are shown in **Figure 14**. The diagram shows a cascade controller with a longitudinal speed master loop and a current slave loop. Both control structures are based on fuzzy logic. For the master loop, a self-tuning PID controller is used. This structure is shown in detail in **Figure 15**. The same rules were used and the same number of linguistic values. The difference lies in the range of operation of the linguistic variables of gains $k_p$, $k_d$, and $k_i$. These ranges were proposed from the PID controller answer from the previous section. For the slave loop, a PI structure with self-tuning was proposed, and the same strategy as the one mentioned above was used. Only the range of values of the input to the internal $k_p$ and $k_i$ profit parts has been modified.

The ranges of values are smaller and are intended to compare the response of the master controller with the slip produced by the tire and the selected surface to evaluate.

The simulation result is shown in **Figure 15**, where each surface shows a similar behavior when following the reference and where the angular and longitudinal velocities do not offer a significant displacement. This is to the slide controller function. For example, in **Figure 15**, a dry asphalt surface is used, and it can be seen that the speeds adhere to the reference. In 15–*b*, it can be seen that the longitudinal velocity tends to
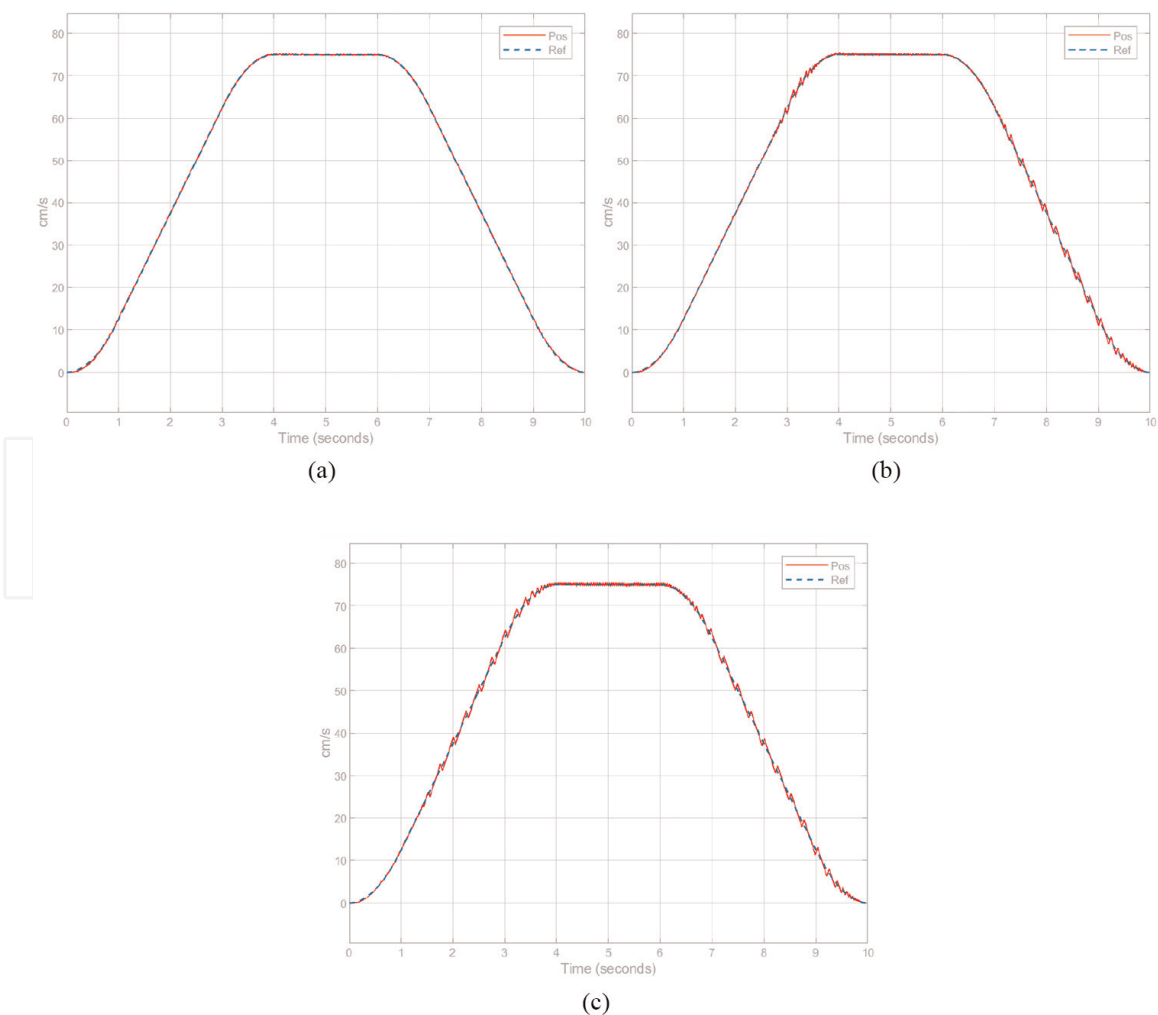


(a)

(b)

(c)

**Figure 15.**
*Response of the tire-motor plant to surfaces (a) dry asphalt, (b) wet asphalt y, (c) ice.*

decrease in magnitude, although the change in it is not significant. Finally, **Figure 15c** shows the behavior of the angular and longitudinal velocities on an ice surface. It is shown that the controller is reacting well, the longitudinal speed reduces its magnitude, and the angular speed follows the profile. The latter makes sense since the slip functions the angular and longitudinal velocities. Therefore, the magnitude of the velocities must be similar depending on the surface over which the plant moves.

The slip measured by the simulation is shown in **Figure 16**. As can be seen, there is less slip for dry asphalt. This was to be expected since this type of surface provides a better grip on the tires to the surface. On the other hand, we see that slip tends to increase in magnitude when the robot's wheel moves on a wet surface, similar to a larger vehicle when it rains. Finally, when the surface contains ice, the landslide tends to increase in magnitude since surfaces with ice have a minimal coefficient of friction, causing the landslide to increase in magnitude if it is not controlled.

The current measured in the simulation is shown in **Figure 17**. The dry asphalt surface presents a greater magnitude of current mainly due to the coefficient of friction since there is always a grip; that is, there is a considerable friction force. When the tire runs on a wet surface, it tends to decrease the coefficient of friction, making this one turn more efficiently, thus reducing the current supplied to the motor. Finally, when the surface is ice, the current tends to decrease since the wheel will tend to slip, which means that the amount of current that is going to be injected into the motor is less, and in order to compensate for the speed in the wheel or on the motor shaft.

The final position or target position is deduced from the 7-segment profile. **Figure 18** shows the position of the system with its respective reference. **Table 7** displays the values of the desired and measured positions of the different surfaces.

**Table 8** shows the performance of the controllers against their respective surfaces using the root mean square error (RMSE). As can be seen, the PID controller shows a better performance than the fuzzy controller on a dry asphalt surface. The adaptive
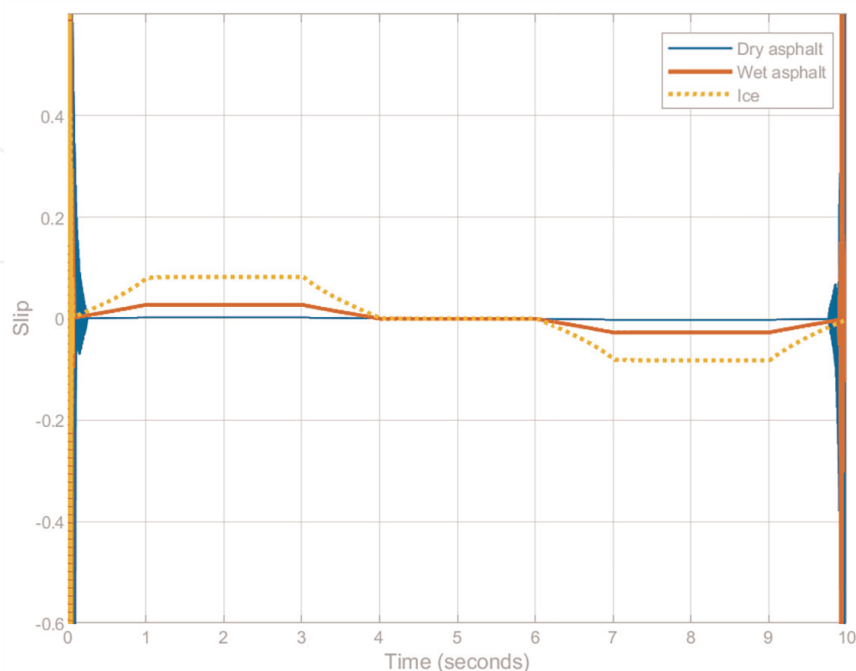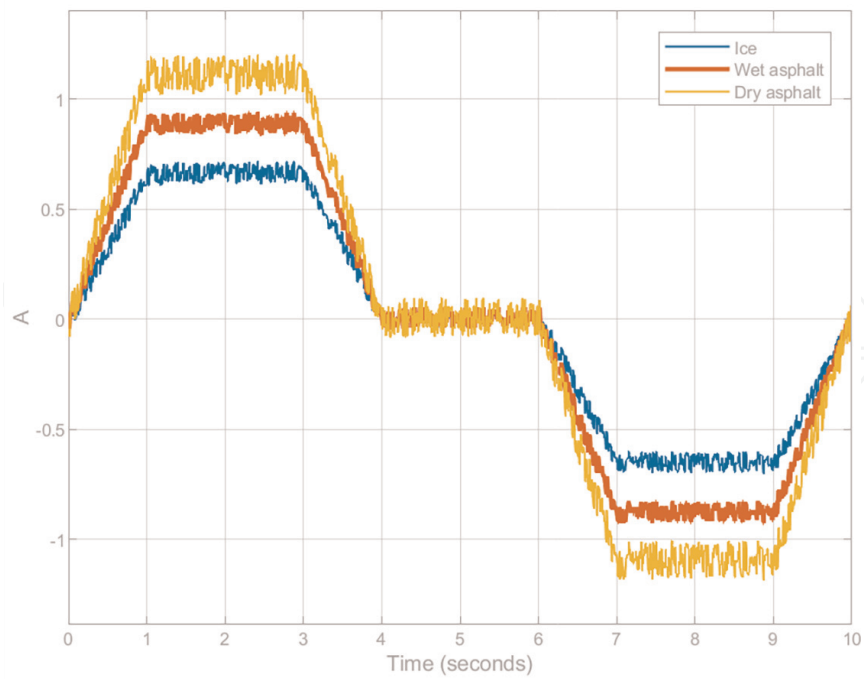


**Figure 16.**
*Slip behaviour.*
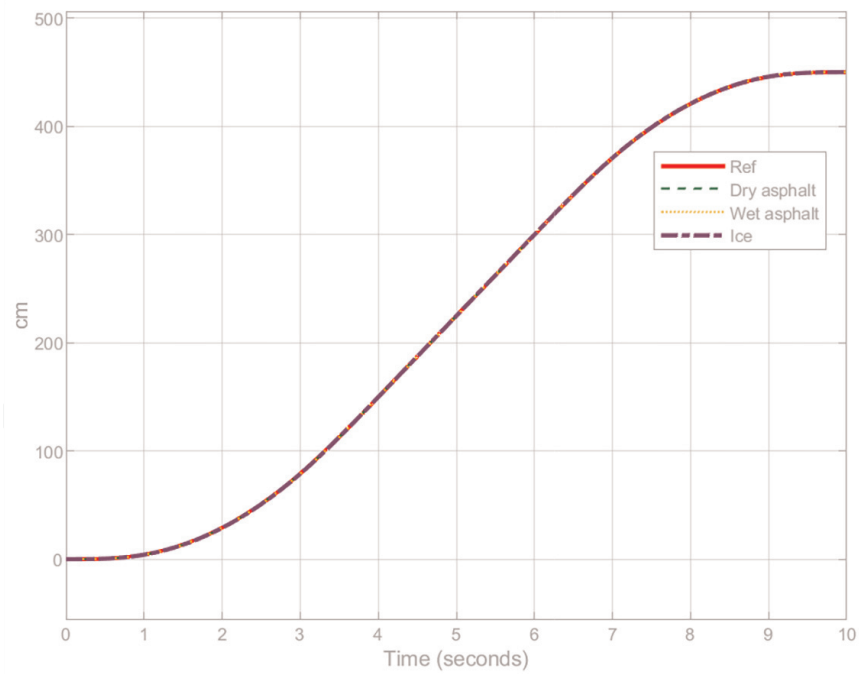
**Figure 17.**
*Current consumption.*



**Figure 18.**
*Perfil de 7 segmentos.*

| Reference | Dry asphalt | Wet asphalt | Ice |
|-----------|-------------|-------------|-----|
| 450 cm | 450 cm | 451.05 cm | 455.2 cm |

**Table 7.**
*Final position values.*

| Structure | Dry asphalt | Wet asphalt | Ice |
|---|---|---|---|
| PID | 0.01498 | 0.01673 | 593.9 |
| Fuzzy controller | 0.01577 | 0.01658 | 0.01913 |

**Table 8.**
*Performance values.*

PID-like fuzzy controller starts to increase its performance when there is a change of surface, for example, for the wet surface. Finally, better performance is appreciated when the surface is icy since the traction controller allows longitudinal speed to be controlled by controlling the wheel's speed, making the latter follow the reference.

## 5. Conclusions

The mathematical model of the motor and the wheel are used for the traction controller. The angular speed, longitudinal speed, current, and slip are obtained from this model. The current is used in the slave loop, whose objective is to brake the wheel to compensate for the angular longitudinal speed through the motor shaft. The master loop aims to follow an S-curve velocity profile. Three different surfaces were used for the simulations: dry asphalt, wet asphalt, and ice.

A PID controller was implemented to make the comparisons, before the surfaces mentioned above, with the fuzzy controller. The fuzzy controller design followed the same methodology as the one used for the motion controller. This makes the adaptive PID-like fuzzy controller master–slave design methodology easy to reproduce. The PID controller worked well on wet and dry asphalt surfaces, but the controller no longer compensated for speeds when the Quarter-Car robot was presented on an icy surface. The same gains were used for all three trials. The fuzzy controller worked well for all three surfaces, showing robustness to changing surfaces. For future work, a type-2 fuzzy logic controller will be implemented to prove the type reduction algorithms in embedded systems in the slip control applications.

## Conflict of interest

The authors declare no conflict of interest.

## Abbreviations

| | |
|---|---|
| PID | Proportional-integral-derivative |
| DC | Direct current |
| TCS | Traction constrol system |
| FAM | Fuzzy associative matrices |
| NB | Negative big |
| NM | Negative medium |
| NS | Negative small |
| ZE | Zero |
| PS | Positive small |

| | |
|---|---|
| PM | Positive medium |
| PB | Positive big |
| S | Small |
| AZ | Almost zero |
| B | Big |
| RMSE | Root mean square error |

## Author details

José R. García-Martínez[1*†], Edson E. Cruz-Miguel[1†], Juvenal Rodríguez-Reséndiz[2†], Luis D. Ramírez-González[1†] and Miguel A. Rojas-Hernández[1†]

1 Universidad Veracruzana, Poza Rica, Ver., Mexico

2 Universidad Autónoma de Querétaro, Querétaro., Mexico

*Address all correspondence to: romangarcia@uv.mx

† These authors contributed equally.

## IntechOpen

# References

[1] Zadeh LA. A Fuzzy-Algorithmic Approach to the Definition of Complex or Imprecise Concepts. In: Systems Theory in the Social Sciences. Interdisciplinary Systems Research / Interdisziplinäre Systemforschung. Birkhäuser, Basel: Springer; 1976. pp. 202-282

[2] Alavala CA. Fuzzy Logic and Neural Networks: Basic Concepts & Application. New Delhi: New Age International; 2008

[3] Hanss M. Applied Fuzzy Arithmetic. Berlin Heidelberg: Springer; 2005

[4] Cruz PP, Herrera A. Inteligencia artificial con aplicaciones a la ingeniería. México: Alpha Editorial; 2011

[5] Distefano JJ, Stubberud AR. Retroalimentación y sistemas de control. Vol. 2. Colombia: McGraw-Hill; 1995

[6] Kuo BC. Automatic Control Systems. United States: Prentice Hall PTR; 1987

[7] Ogata K. Ingeniera de control moderna. España: Pearson Educación; 2003

[8] Nise NS. Control Systems Engineering. United States: John Wiley & Sons; 2020

[9] Mandal AK. Introduction to Control Engineering: Modeling, Analysis and Design. New Delhi: New Age International; 2006

[10] Jager R. Fuzzy Logic in Control. Amsterdam: Rene Jager; 1995

[11] Kacprzyk J. Studies in fuzziness and soft computing 295. Berlin, Heidelberg: Springer; 2013

[12] Albagul A, Martono W, Muhida R. Dynamic modelling and adaptive traction control for mobile robots. In: Kordic V, Lazinica A, Merdan M, editors. Cutting Edge Robotics. London, UK, Rijeka: IntechOpen; 2005 Chapter 1

[13] Kuntanapreeda S. Traction control of electric vehicles using sliding-mode controller with tractive force observer. International Journal of Vehicular Technology. 2014;**2014**:1-10

[14] Mu J, Yan X-G, Spurgeon SK, Mao Z. Nonlinear sliding mode control of a two-wheeled mobile robot system. International Journal of Modelling, Identification and Control. 2017;**27**(2): 75-83

[15] Konduri S, Orlando E, Torres C, Pagilla PR. Effect of wheel slip in the coordination of wheeled mobile robots. IFAC Proceedings Volumes. 2014;**47**(3): 8097-8102

[16] Alhaj Ali SM, Hall EL. Design and simulation of a motion controller for a wheeled mobile-robot autonomous navigation. In: Intelligent Robots and Computer Vision XXIII: Algorithms, Techniques, and Active Vision. Vol. 6006. International Society for Optics and Photonics; Proc. SPIE. 2005. p. 60060M

[17] Iossaqui JG, Camino JF, Zampieri DE. A nonlinear control design for tracked robots with longitudinal slip. IFAC Proceedings Volumes. 2011;**44**(1): 5932-5937

[18] Ryu J-C, Agrawal SK. Differential flatness-based robust control of mobile robots in the presence of slip. The International Journal of Robotics Research. 2011; **30**(4):463-475

[19] Kim J, Lee J. Intelligent slip-optimization control with traction-energy trade-off for wheeled robots on rough terrain. In: 2014 IEEE/RSJ International Conference on Intelligent Robots and Systems, IEEE, 2014. pp. 1938–1943

[20] Kim D-E, Yoon HN, Kim KS, Sreejith MS, Lee J-M. Using current sensing method and fuzzy pid controller for slip phenomena estimation and compensation of mobile robot. In: 2017 14th International Conference on Ubiquitous Robots and Ambient Intelligence (URAI), IEEE, 2017 pp. 397–401

[21] Falsafi MH, Alipour K, Tarvirdizadeh B. Tracking-error fuzzy-based control for nonholonomic wheeled robots. Arabian Journal for Science and Engineering. 2019;**44**:881-892

[22] Falsafi M, Alipour K, Tarvirdizadeh B. Fuzzy motion control for wheeled mobile robots in real-time. Journal of Computational & Applied Research in Mechanical Engineering (JCARME). 2019;**8**(2):133-144

[23] Pacejka HB, Sharp RS. Shear force development by pneumatic tyres in steady state conditions: a review of modelling aspects. Vehicle System Dynamics. 1991;**20**(3-4):121-175

[24] Bakker E, Nyborg L, Pacejka HB. Tyre modelling for use in vehicle dynamics studies. SAE Transactions. 1987:190-204

[25] Jin L, Chen P, Zhang R, Ling M. Longitudinal velocity estimation based on fuzzy logic for electronic stability control system. Advances in Mechanical Engineering. 2017;**9**(5): 16878140176986s62

[26] De Silva CW. Intelligent Control: Fuzzy Logic Applications. Boca Raton: CRC Press; 2018

[27] García-Martínez JR, Cruz-Miguel EE, Carrillo-Serrano RV, Mendoza-Mondragón F, Toledano-Ayala M, Rodríguez-Reséndiz J. A pid-type fuzzy logic controller-based approach for motion control applications. Sensors. 2020;**20**(18):5323

[28] Doan D-V, Nguyen K, Thai Q-V. Load-frequency control of three-area interconnected power systems with renewable energy sources using novel pso pid-like fuzzy logic controllers. Engineering, Technology & Applied Science Research. 2022;**12**(3):8597-8604