

# Self-Tuning PID Controller for Autonomous Car Tracking in Urban Traffic

Luciano Alonso, Juan Pérez-Oria , Basil M. Al-Hadithi , and Agustín Jiménez

**Abstract—**In this paper an on line self-tuned PID controller is proposed for the control of a car whose goal is to follow another one, at distances and speeds typical in urban traffic. The best-known tuning mechanism is perhaps the MIT rule, due to its ease of implementation. However, as it is well known, this method does not guarantee the stability of the system, providing good results only for constant or slowly varying reference signals and in the absence of noise, which are unrealistic conditions. When the reference input varies with an appreciable rate or in presence of noise, eventually it could result in system instability. In this paper an alternative method is proposed that significantly improves the robustness of the system for varying inputs or in the presence of noise, as demonstrated by simulation.

## I. INTRODUCTION

The biggest concern in the automotive market is perhaps the safety of the passengers, pedestrians and other objects on the road. Cars incorporate more and more intelligent systems to improve safety, comfort and energy efficiency. Most of them, like ABS (Anti-lock Brake System) and ESP (Electronic Stability Program) are designed to maintain the control over the vehicle in extreme circumstances, being very effective at high speeds. However, for distances and speeds typical in urban traffic in which the vast majority of accidents happen, there is still a lack of safety systems. Fortunately, this trend is being reversed in recent times with the emergence of new intelligent systems, such as pedestrian detection, lane change alarm, speed limit alarm or emergency braking system.

All these systems are small steps towards a hypothetical future with fully autonomous and safe vehicles. Authors of this work have a wide trajectory in this field [1] [2] [3]. Here a new step towards this future is presented, whose objective is to attain a car capable by itself of follow another one, maintaining a safe distance depending on the speed, in ranges usual in urban traffic.

The control system uses sensors, which provide the actual speed of the controlled car and the relative position of the car ahead, to calculate the actions on the steering wheel, the accelerator and the brake. For controlling the direction, a proportional (P) controller with a constant gain is used, while for controlling the accelerator-brake a Proportional-Integral-Derivative (PID) self-tuning controller is employed.

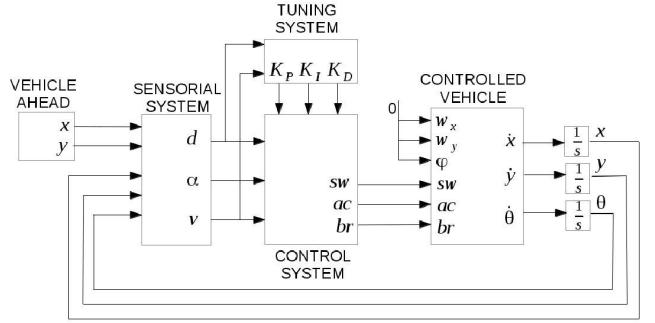


Fig. 1. Block diagram of the mathematical model used for the simulations.

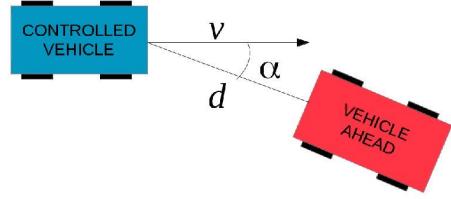


Fig. 2. Variables provided by the sensorial system and used in the control.

The most widely used method for self-tuning controllers is, perhaps, the well known MIT rule (Massachusetts Institute of Technology) [5]. However, as it is well known, with this method, the system can become unstable for time-varying or noisy inputs. Also, there exist other more complicated techniques for online tuning of controllers, as these based for example on the Lyapunov stability theory [7], or on Artificial Intelligence [6]. In this work, a new and easy method is proposed, that significantly reduces the risk of instability compared with the MIT rule.

The work is organized as follows: in section 2 the mathematical model of the car used in the simulations is described; in section 3 the controllers for both the direction and the speed of the car are explained; section 4 contains the description of the MIT rule and the proposed rule for the adaptation of the parameters of the PID controller for the speed; in section 5 the results of the simulations with both adaptation methods are shown; finally, in section 6 the conclusions are exposed.

## II. MATHEMATICAL MODEL

Fig. 1 shows the block diagram of the full mathematical model used in the simulations. The blocks are:

-**VEHICLE AHEAD**: it provides the position ( $x, y$ ) of the pursued vehicle along the time. This trajectory is randomly generated before the start of the simulation.

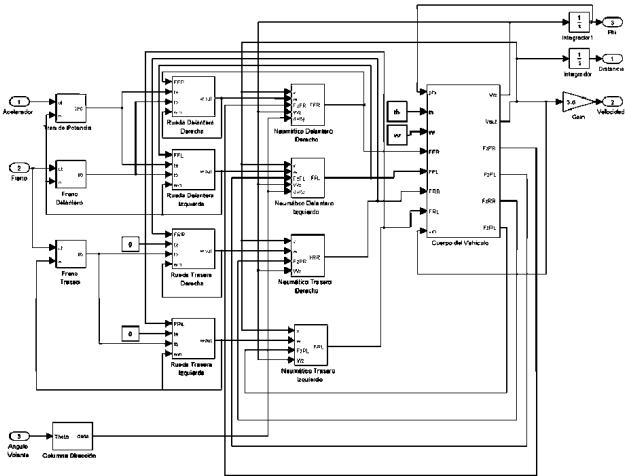


Fig. 3. Block diagram of the mathematical model of the longitudinal and transversal dynamics of the controlled vehicle.

**-SENSORIAL SYSTEM:** it models the sensors providing the speed of the controlled vehicle ( $v$ ), and the position ( $d$ ,  $\alpha$ ) of the pursued vehicle relative to the latter. Fig. 2 shows these variables. Angle  $\alpha$  is saturated at  $\pm 45^\circ$  and distance  $d$  is limited to  $15m$ , which are the ranges of interest. In addition, these signals are filtered by means of a first-order low-pass filter. A sampling time of  $0.1s$  is used.

**-CONTROL SYSTEM:** it models both the P controller for the steering wheel, and the PID for the accelerator-brake. The control signals for the direction and the acceleration are calculated from the inputs provided by the sensorial system, with a  $0.1s$  sampling period. In section III, a detailed explanation of both of them is provided.

**-TUNING SYSTEM:** each  $0.1s$ , it adjusts the parameters of the PID controller of the accelerator-brake, from the signals provided by the sensorial system, using the MIT rule or the proposed method, both of them explained in section IV.

**-CONTROLLED VEHICLE:** it models the longitudinal and transversal dynamics of the controlled vehicle. Its inputs are the signals for the steering wheel, the accelerator pedal and the brake pedal, as well as the wind speed (magnitude and direction) and the road slope. The last three are taken as zero in this work. Its outputs are the velocity of the mass centre and the rotation speed around the vertical axis. Integrating these speeds, the position and orientation of the car are obtained. Figure 3 shows the block diagram of the mathematical model developed. A full description can be seen in (Arce,2012).

### III. CONTROL SYSTEM

The control system is composed of a P type controller for the direction, and a PID type controller for the accelerator-brake.

The direction controller has a constant gain  $K_P = 4$ , which is the reduction of the steering column of the car. Its input is the angle  $\alpha$ , and its output is the angular position of the steering wheel  $sw$ . The control law is therefore:

$$sw(k) = 4 \cdot \alpha(k) \quad (1)$$

$k$  being the sampling instant. In this way, the front wheels of the controlled car are always looking towards the pursued vehicle.

For controlling both accelerator and brake pedals, a single PID controller is used. The input is the position error, which is the difference between the distance to the car ahead  $d$  and the safety distance  $d_s$ . The latter is defined as the necessary distance to maintain a gap of a second between vehicles, plus two meters (minimum desired distance), namely:

$$d_s = v \cdot t + 2 \quad (2)$$

being  $t = 1s$ . Taking  $e = d - d_s$ , the control law becomes:

$$\begin{aligned} u(k) &= K_P \cdot e(k) + K_I \cdot \sum_{n=0}^k e(n) + K_D \cdot [e(k) - e(k-1)] \\ &= K_P \cdot e(k) + K_I \cdot I(k) + K_D \cdot D(k) \end{aligned} \quad (3)$$

This signal  $u$  is then saturated between  $-100$  and  $100$  (the limits of the accelerator and brake signals), and finally it is applied on the accelerator or on the brake depending on its sign, according to:

$$\begin{aligned} ac(k) &= \begin{cases} u(k) \cdot \cos^2[sw(k)] & u(k) > 0 \\ 0 & u(k) < 0 \end{cases} \\ br(k) &= \begin{cases} -u(k) \cdot \cos^2[sw(k)] & u(k) < 0 \\ 0 & u(k) > 0 \end{cases} \end{aligned} \quad (4)$$

The factor  $\cos^2[sw(k)]$  avoids big accelerations or decelerations when the front wheels are rotated.

The parameters  $K_P$ ,  $K_I$  and  $K_D$  of the PID controller are tuned on line by the algorithms explained in section IV.

### IV. TUNING ALGORITHMS

For tuning the parameters of the PID controller of the acceleration, two different algorithms had been used: the well known MIT rule, and the proposed rule. To the knowledge of the authors, the proposed rule is a novelty in this field, since they have not found any other work in which this method was used. Both methods are explained below.

#### A. MIT rule

This rule is based on the minimization of a quadratic performance index depending on the parameter to be tuned  $\theta$ :

$$J(\theta) = \frac{e^2(\theta)}{2} \quad (5)$$

If  $J$  must decrease as time goes by, the parameter  $\theta$  must vary in the opposite direction of the gradient:

$$\begin{aligned} \theta(k+1) &= \theta(k) - \gamma \frac{\partial J(\theta)}{\partial \theta} \\ &= \theta(k) - \gamma e(\theta) \frac{\partial e(\theta)}{\partial \theta} \\ &= \theta(k) - \gamma e(\theta) \frac{\partial e(\theta)}{\partial u(\theta)} \frac{\partial u(\theta)}{\partial \theta} \end{aligned} \quad (6)$$

$\gamma$  being a positive number named *learning rate*. A too big value of  $\gamma$  can produce the instability of the system, while a

too small value produces a poor convergence, so this value must be carefully selected.

The first partial derivative in (6) is unknown, since there is no knowledge about the system. However, as we are only interested on the direction of the gradient, its sign is enough to adapt the parameter. It is reasonable to think that increasing the control signal  $u$  raises the speed  $v$ , so the safety distance  $d_s$  increases too, and the error signal  $e$  decreases, therefore this sign must be negative.

The second partial derivative in (6) is obtained from (3), substituting  $\theta$  by each one of the parameters of the controller  $K_p$ ,  $K_I$  and  $K_D$ . Finally, the rules of adaptation of these parameters are:

$$\begin{aligned} K_p(k+1) &= K_p(k) + \gamma_p \cdot e^2(k) \\ K_I(k+1) &= K_I(k) + \gamma_I \cdot e(k) \cdot I(k) \\ K_D(k+1) &= K_D(k) + \gamma_D \cdot e(k) \cdot D(k) \end{aligned} \quad (7)$$

The drawback of this rule is that if the error never vanishes, as is the real case (particularly with time-varying or noisy inputs),  $K_p$  always increases, driving the system towards the instability. This is shown in section V.

### B. Proposed rule

This alternative method tries to avoid the continuous increasing of the parameters, using new rules that allow their decreasing. Along with  $e$  and  $D$ , two new signals are calculated. They represent a filtered version of the error  $e_m$  (a simple low-pass first order filter is used in this work), and its derivative  $D_m$ . The proposed adaptation rules are:

$$\begin{aligned} K_p(k+1) &= K_p(k) + \gamma_p \cdot [e(k) - e_m(k)] \\ K_I(k+1) &= K_I(k) + \gamma_I \cdot e_m(k) \\ K_D(k+1) &= K_D(k) + \gamma_D \cdot [D(k) - D_m(k)] \end{aligned} \quad (8)$$

In this way, the three parameters can increase and decrease along the time, so this method will be, at least initially, more robust than the MIT rule when referring to the stability of the system. This is particularly true for controlling the system under study, as it is shown in section V.

## V. SIMULATION RESULTS

In this section, the results obtained by simulation of the described system are presented. Both tuning methods have been used. A quadratic performance index is defined that allows the numerical comparison of the results:

$$J = \frac{1}{T} \int_0^T [d(t) - d_s(t)]^2 dt \quad (9)$$

$T$  being the total simulated time. The used learning rates for the three parameters are  $\gamma_p = 0.05$ ,  $\gamma_I = 0.005$  and  $\gamma_D = 0.05$ . The initial values of the parameters of the controller are  $K_p = 20$  and  $K_I = K_D = 0$ . The trajectory followed by the vehicles has been the same in all simulations.

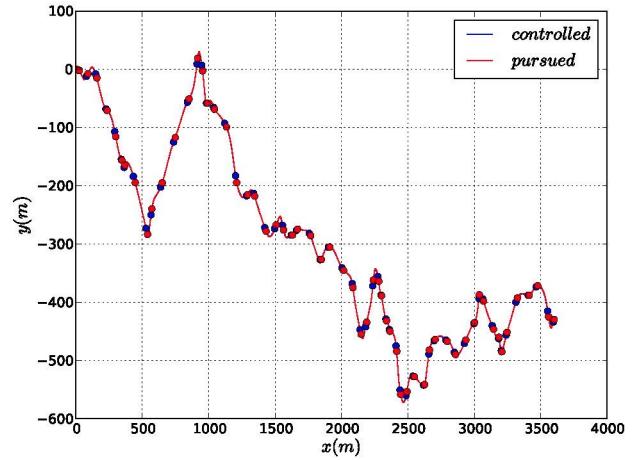


Fig. 4. Path followed by the cars over a period of 10 minutes, using the MIT rule. Dots show their positions every ten seconds.

### A. Results of simulation using MIT rule

Fig. 4 shows the path followed by the cars over a period of ten minutes, using the MIT rule as adaptation algorithm. The dots indicate the positions of the cars every ten seconds. As it can be seen they are nearly identical, indicating that the controlled car has been capable of following the car ahead.

In Fig. 5, the evolution along the time of the most important variables is represented. The graph (a) shows the signal acting on the throttle (when positive) or on the brake (when negative). In graph (b), the signal controlling the steering wheel is shown, along with the angle of the position of the car ahead relative to the longitudinal axis of the controlled vehicle, scaled by a factor of 4 for simplifying the comparison. The speeds are presented in graph (c). Finally, the distance between vehicles and the safety distance obtained from the speed of the controlled car are shown in graph (d). As it can be seen, particularly in graphs (c) and (d), the controlled car adjusts its speed to that of the vehicle ahead, and maintains a safe distance in spite of not being following a straight path.

The evolution of the position error is shown in Fig. 6, along with the parameters of the controller. A continuous increasing in  $K_D$  can be observed, as it is expected from the quadratic dependence on the error in (8). This increase causes a decrease in the error signal. The obtained value for the performance index in this case is  $J = 3.3029$ .

To check the effect of the presence of noise in the measured signals, a random noise has been added to the distance and angle, with a maximum value of a 20% of the magnitude of the measurement, and the simulation has been repeated. Fig. 7 shows the magnitudes of interest, and the error as well as the parameters of the controller can be viewed in Fig. 8. The most important thing to observe is that the control signal of the throttle-brake becomes quickly saturated, indicating that the system is unstable. This is caused by a fast increase in the parameters of the controller, specially in  $K_p$ , as seen in Figure 8. In this case, the performance index has a value of  $J = 5.5916$ .

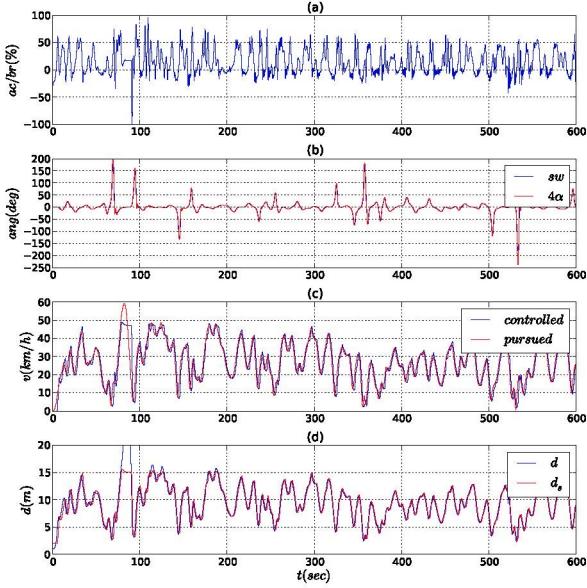


Fig. 5. Evolution of the more important variables of the vehicles along the simulation using MIT rule. (a) Control signal for the throttle and brake pedals. (b) Steering wheel control signal, and relative angle of the vehicle ahead, scaled for comparison. (c) Speeds of the vehicles. (d) Separation between vehicles, and safety distance obtained from the current speed of the controlled car.

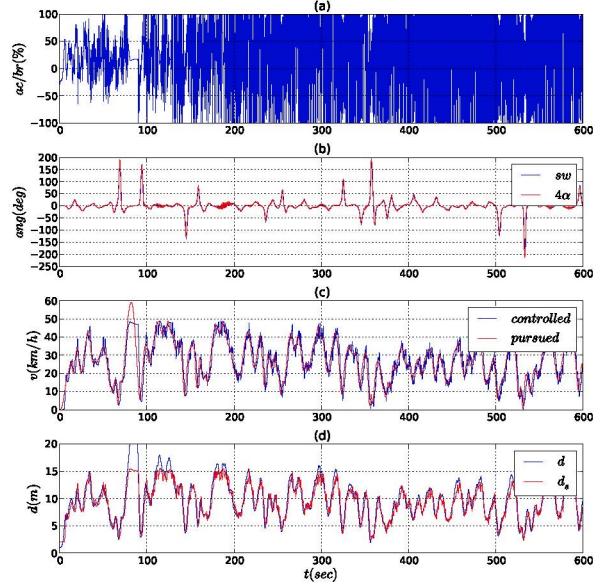


Fig. 7. Evolution of the important variables of the vehicles along the simulation using MIT rule, in presence of noise. (a) Control signal for the throttle and brake pedals. (b) Steering wheel control signal, and relative angle of the vehicle ahead, scaled for comparison. (c) Speeds of the vehicles. (d) Separation between vehicles, and safety distance obtained from the current speed of the controlled car.

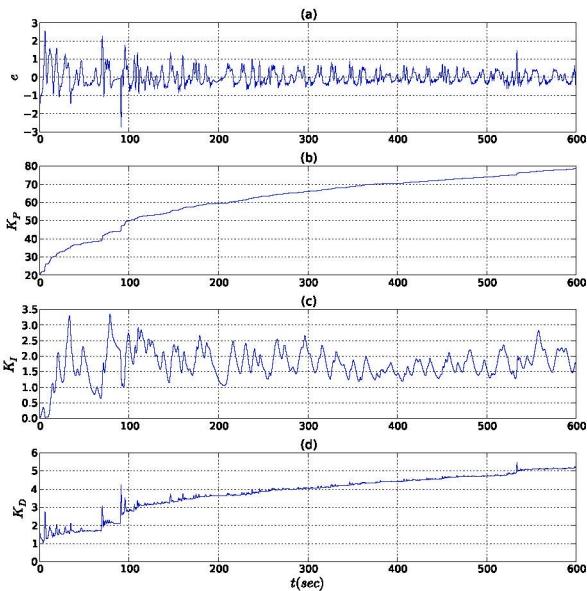


Fig. 6. Distance error and PID parameters using MIT rule. (a) Distance error. (b)  $K_P$ . (c)  $K_I$ . (d)  $K_D$ .

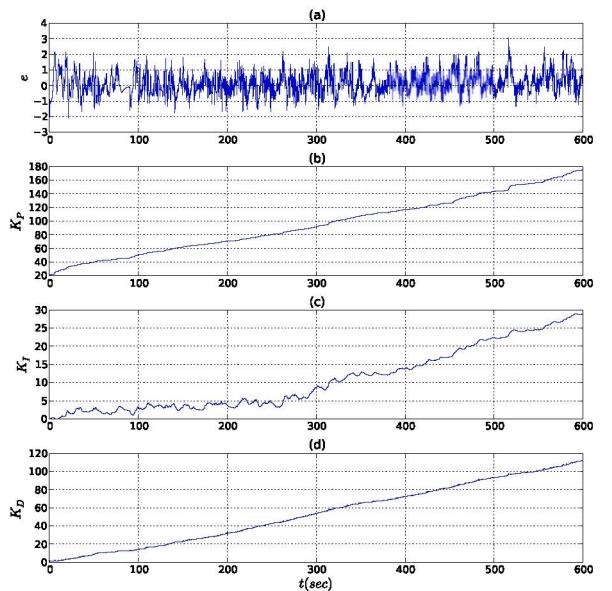


Fig. 8. Error signal and PID parameters using MIT rule with noise added to the distance and angle. (a) Distance error. (b)  $K_P$ . (c)  $K_I$ . (d)  $K_D$ .

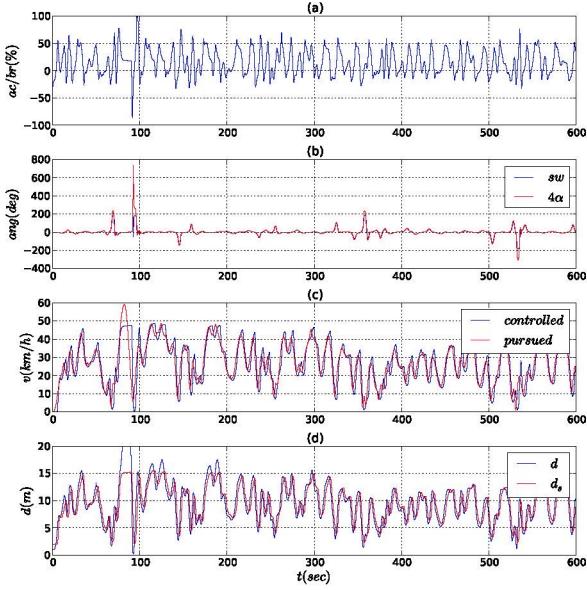


Fig. 9. Evolution of the important variables of the vehicles along the simulation using the proposed rule, without noise. (a) Control signal for the throttle and brake pedals. (b) Steering wheel control signal, and relative angle of the vehicle ahead, scaled for comparison. (c) Speeds of the vehicles. (d) Separation between vehicles, and safety distance obtained from the current speed of the controlled car.

### B. Results of simulation using the proposed rule

The simulations have been carried out using the proposed rule, using the same values for the learning rates and the initial values of the parameters of the controller, with and without noise in the signals coming from the sensorial system. Again, the trajectories of the vehicles are indistinguishable, so they have been not shown.

Firstly, the simulation without noise has been repeated. In Fig. 9, the interesting signals are shown, and the distance error  $e$  and filtered error  $e_m$  along with the parameters of the PID controller can be viewed in Fig. 10. As it can be seen, these parameters remain bounded, so the error signal is bigger than for the MIT rule, but still in acceptable values. The resulting value for the performance index is  $J = 9.9935$ .

To verify the robustness of the method against noise, a new simulation is carried out with the same noise level in the signals from the sensorial system, as in the case for the MIT rule. The results can be observed in Figs. 11 and 12. The resultant performance index is  $J = 9.7306$ . The most remarkable feature is that, despite the presence of noise, the system remains stable, and the error as well as the parameters of the controller are bounded. The noise affects the control signal, which is not so clean as without noise, but it is not saturated as in the case of the MIT rule, so this adaptation rule is most robust against noise than the former one.

Finally, to verify that the system remains stable for larger times using the proposed rule, several simulations of ten hours of duration, with noise in the signals and with different paths were carried out. In all of them the parameters of the controller were maintained in bounded values, so the controlled car was able to follow the car ahead without turning unstable.

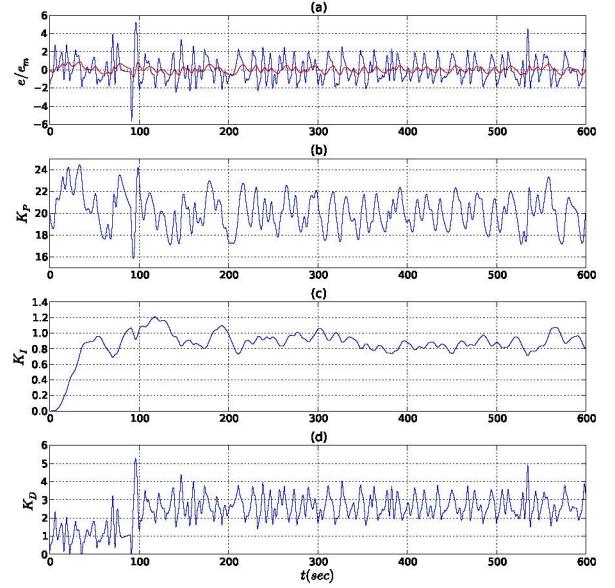


Fig. 10. Error signals and PID parameters using the proposed rule without noise. (a) Distance error  $e$  (blue) and filtered error  $e_m$  (red). (b)  $K_P$ . (c)  $K_I$ . (d)  $K_D$ .

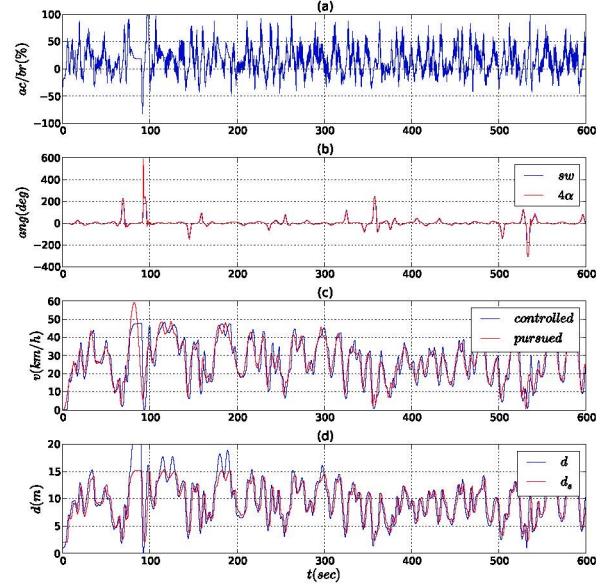


Fig. 11. Evolution of the important variables of the vehicles along the simulation using the proposed rule, with noisy signals. (a) Control signal for the throttle and brake pedals. (b) Steering wheel control signal, and relative angle of the vehicle ahead, scaled for comparison. (c) Speeds of the vehicles. (d) Separation between vehicles, and safety distance obtained from the current speed of the controlled car.

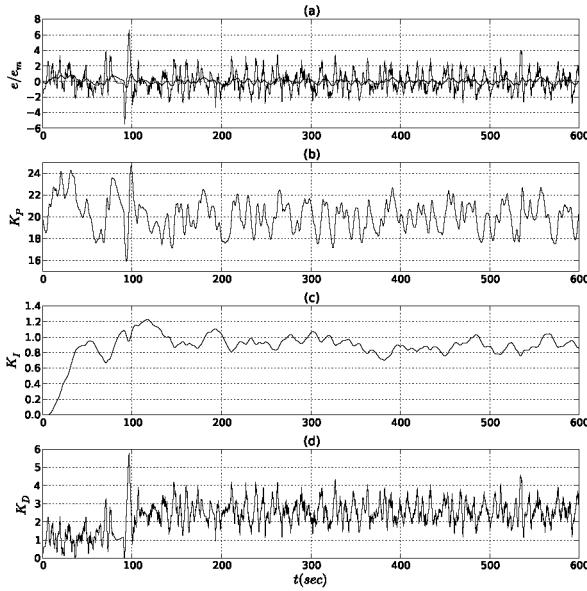


Fig. 12. Error signals and PID parameters using the proposed rule with noise. (a) Distance error  $e$  (blue) and filtered error  $e_m$  (red). (b)  $K_P$ . (c)  $K_I$ . (d)  $K_D$ .

## VI. CONCLUSION

An alternative method to the well known MIT rule has been proposed, for the online tuning of a PID controller. It has been successfully applied to the control of a car which tries to follow another vehicle ahead. The robustness of the method against noisy measurement signals has been validated by means of several simulations. A better performance than with the MIT rule is obtained.

The experimental validation in the test vehicle available to the research team is still pending. Also in the next future, the authors will seek to formally establish the conditions to be met by a system to ensure its stability when using the proposed method.

## ACKNOWLEDGMENT

This work was supported by the Ministry of Economy and Competitiveness of the Spanish Government (project DPI2012-36959).

## REFERENCES

- [1] L. Alonso, V. Milanés, C. Torre, J. Godoy, J.M. Pérez-Oria, T. de Pedro, Ultrasonic sensors in urban traffic driving-aid systems, Sensors 2011, 11(1), 661-673; doi:10.3390/s110100661.
- [2] V. Milanés, L. Alonso, J. Villagrá, J. Godoy, T. de Pedro, J.M. Pérez-Oria, Traffic jam driving with nmv avoidance, Mechanical Systems and Signal Processing, Volume 31, August 2012, Pages 332-344.
- [3] L. Alonso, J.M. Pérez-Oria, M. Fernández, C. Rodríguez, J. Arce, M. Ibarra, V. Ordóñez, Genetically tuned controller of an Adaptive Cruise Control for urban traffic based on ultrasounds, Artificial Neural Networks - ICANN 2010, Lecture Notes in Computer Science, Volume 6353, 2010, pp 479-485.
- [4] J. Arce, Contribución a sistemas inteligentes de seguridad activa basados en ultrasonidos, PhD thesis, University of Cantabria, Spain, 2012.
- [5] K.J. Astrom, B. Wittenmark, Adaptive control, Addison-Wesley Longman Publishing Co., Inc. Boston, MA, USA 1994, ISBN:0201558661
- [6] Z. Li, Model reference adaptive controller design based on fuzzy inference system, Journal of Information & Computational Science, 8: 9(2011) 1683-1693.
- [7] G. Bortolotto, A. Desages, J. Romagnoli, Optimal tuning of discrete pid controllers using Lyapunov's direct method robustness analysis, Chemical Engineering Communications Volume 109, Issue 1, 1991, pages 1-18, DOI: 10.1080/00986449108910970.