XVII Reunión de Trabajo en Procesamiento de la Información y Control - RPIC 2017
ISBN: 978-987-544-754-7

# Obstacle avoidance under strict path following

Juan Luis Rosendo*, Fabricio Garelli†, Hernan De Battista‡, and Fernando Valenciaga§
GCA, CONICET and Universidad Nacional de La Plata, C.C.91 (1900), La Plata, Argentina
Email: *juanluisrosendo@gmail.com, †fabricio@ing.unlp.edu.ar, ‡deba@ing.unlp.edu.ar, §fval@ing.unlp.edu.ar

***Abstract*—In this work an obstacle avoidance technique for strict path following task of robotic systems is proposed and tested through simulations. The main idea is to implement an speed adaption loop which based on a discontinuous signal product of an sliding surface in the state space, modifies the speed of reference for a robot. The strength of this approach is that we solve, under certain conditions, the problem of obstacle avoidance when it is forbidden for the robot to leave the given path. Furthermore, we are able to impose a robust desired dynamics in the approach between the robot and the obstacle.**

***Index Terms*—Robotics, Path following, Sliding modes, Speed adaption**

## I. INTRODUCTION

Path planing, path following, and obstacle avoidance are among the most frequent tasks commissioned to robots. Usually these duties are studied in an isolated way passing over their strong coupling. For example during a path following task in an unstructured dynamic environment, when a collision situation is detected a path re-planing must be done reconsidering the new state condition.

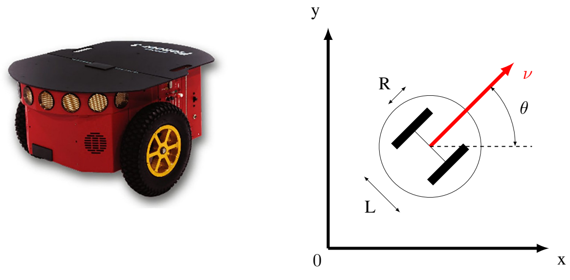
A global path planning algorithm generally use a priori information to build a complete model of the unstructured environment and then try to find the best possible solution. But in unknown or unstructured environments this is not sufficient so it is necessary to combine the path planing method with a local or reactive navigation using on board sensors, so as to locally observe small fragments of the surrounding at each time. In such scenario, the problem of detecting and reacting in the presence of obstacles arises. In the case of mobile robots the most common approaches are: first, to use a belt of proximity sensors (ultrasound, infrared, ...) mounted on the vehicle allowing a discrete scanning of the space around the robot [8]; and in the second place, the use of a rotating laser beam, frequently coupled with a vision system, resulting in continuous estimation of the free region around the vehicle [11].

Once we have the necessary information about our environment, the optimal way to process and take actions will depend on the particular situation where our system will be. Several obstacle avoidance methods exist. Among the most extended it is possible to mention:

1) The Potential Field Methods (PFM): the robot is treated like a particle under the influence of an artificial force field where the obstacles exert repulsive forces, while the target applies an attractive, the sum of all forces determinates the direction and speed of travel. This is the most extended method due to its easy on-line implementation. However, some drawbacks of this method are trap situations due to local minimum, no passage between closely spaced obstacles, and oscillations [1].
2) Vector Field Histogram (VFH): the method uses a polar histogram constructed around the robot, where each component represents the obstacle polar density in the corresponding sector, the set of candidate directions is formed with the components of lower density than a given threshold, and closest to the component that contains the target direction. Finally, through heuristics the robot direction is selected. The VHF is a method formulated to work with probability obstacle distributions and thus is well adapted to work with uncertain sensors such as ultrasonic sonars. One of the drawbacks is the computational cost of the method, although some simplifications have been proposed [2].
3) Velocity obstacles (VO): this method forms a set of candidate control signals that are within the maximum speed of the vehicle. This signals generate safe trajectories considering the obstacle speeds and can be reached in a short period of time given the vehicle acceleration. From this set one control signal is selected as the maximization of an objective function. The main advantage of this method is that it takes into account the obstacle velocities thus it is well suited to dynamic scenarios [4].

These methods do not force a particular dynamic behavior over the robot when a collision situation arises, indeed they are designed for environments where the robot is allowed to leave the pre-elaborated path. This work is focused on the application where the path to be followed is strict, i.e. no path deviations are allowed. Although this seems a very strong constraint, it is a situation found in several applications like industrial line following robots or in automated warehouses [12], and not exclusive for robotics: other fields share the interest in this problem as the optimization of railways operations [5] or the recently presented world's first virtual track train. The strict path following has been less studied in literature than the general case where the path is not strict. Usually, the way this topic is addressed is through the analysis of collision situations in multi-robots operation at constant speeds when points of the path are common to more than one vehicle [3].

Here, we propose a computationally non-expensive method for path following that imposes a desired dynamics over the vehicle when a collision situation arrives. The main idea is to adapt a motion parameter which determines the target position over a parametrized path. This idea has already been exploited



(a) Differential robot (b) Reference frame

Fig. 1. Pioneer P3-DX mobile robot

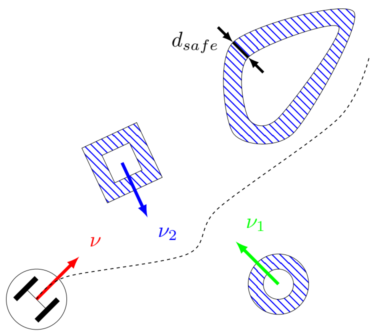


Fig. 2. The environment with the obstacle subset $\Psi$ and the enlarged subset $\hat{\Psi}$ with the $d_{safe}$ neighborhood

in [6] and [9]. In [6] it is used to deal with actuator constraints of marine vehicles, and in [9] collision avoidance is addressed, but in contrast with the present work the trajectory (space and time targets) were modified for respect the given constraints. In this work when a possible obstacle is detected, considering its distance and speed of approximation, the motion adaption is performed by a discontinuous signal which obeys an sliding surface in the state space. This surface is designed to accomplish a desired approach dynamics to the obstacle respecting the pre-elaborated path.

This work is structured in the following way, first in section II a brief introduction about the problem addressed is made. Then in section III an explanation of the proposal is accomplished. Section IV presents the application of the proposal through simulations over the differential robot Pionner, see Fig. 1(a). Finally, some conclusions are given in section V.

## II. PROBLEM DESCRIPTION

We define the general problem of vehicle navigation through a given path with collision avoidance. It is assumed that there exists a dynamic environment that is unknown to the robot and where a pre-elaborated and parametrizable path is defined. This environment is conformed by moving or stationary objects, which are modeled as components of a time variant planar subset $\Psi$. For the practical point of view, we part from the premise that the pre-elaborated path does not contain any collision situation with the stationary components of the environment, in other words we suppose the path is realizable.

We define the distance $d(t)$ from the robot position $\mathbf{p}(t)$ to the environment $\Psi$ as:

$$d(t) := \min_{\mathbf{r}\in\Psi} \| \mathbf{r} - \mathbf{p}(t) \| \qquad (1)$$

where $\| \cdot \|$ denotes the standard Euclidean vector norm, and $\mathbf{r}$ is the nearest obstacle's position belonging to the subset $\Psi$.

As we are going to follow a path we define a time variant target $\eta$ which is going to move through the path. The objective is that our robot follow this target through the path in a safe way, respecting a $d_{safe} > 0$ distance to the obstacles in the environment, see Fig. 2.

Furthermore we are going to demand our robot to follow an specific dynamics when it is approaching to an obstacle, and to stop in case of the $d_{safe}$ constraint is not respected.

In the next section the proposed method to fulfill the problem is briefly presented.

## III. COLLISION AVOIDANCE SPEED ADAPTION (CASA)

To solve the presented problematic an auxiliary loop which modifies the speed of the robot reference is proposed. The scheme in Fig. 3 shows the block diagram of the proposal.

Here we assume that the robot control, as could be a traditional PID control, is implemented inside the block called "Robot + Robot Control". Also it is considered that a first derivative feedforward action takes place inside it, as it is commonly used for robotic reference tracking. In addition, it is supposed that the path is previously generated by a superior control level, and it could be parametrized. The parameter $\lambda$ commands the speed of advance from the path generator represented as the block "$f(\lambda)$". Note however, that the proposed approach can also be applied to those problems in which the path is generated "on-line" as the robot moves, e.g. in the line following robots.

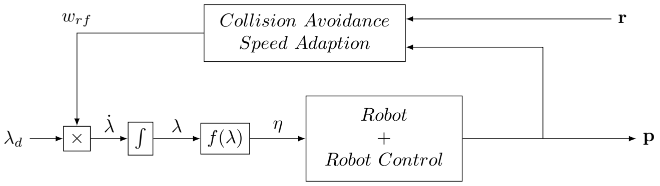


Fig. 3. Auxiliary loop proposed as an obstacle avoidance technique, based in SM

The key of CASA method is to design a sliding surface (Eq. 2) associated to a discontinuous law (Eq. 3), which generates the motion parameter over the path. This ensemble is going to define the dynamics followed during the collision situation. We understand as a collision situation when the distance and the speed of approach between the robot and a potential obstacle overpass a certain desired approach dynamics.

To this end the sliding mode (SM) surface is proposed as:

$$\sigma = d_{safe} - k_d d - k_{dd} \dot{d} \qquad (2)$$

this depends on $d$, $\dot{d}$ and the weighting parameters $k_d$ and $k_{dd}$. The last ones define the desired approaching dynamics to minimal distance $d_{safe}$ constraint. To complete this formulation the associated switching function could be defined as:

$$w_r = \begin{cases} 1 & \sigma \leq 0 \\ 0 & \sigma > 0 \end{cases} \tag{3}$$

From Eq. 2 and Eq. 3 it is possible to see that a discontinuous signal $w_r$ is generated. This signal is then passed through a first-order filter, which generates the signal $w_{rf}$, a soft version of $w_r$. This filter could be described as:

$$\begin{cases} \dot{x_f} = \lambda_f x_f + w_r \\ w_{rf} = -\lambda_f x_f \end{cases} \tag{4}$$

Finally the speed of the commanded reference is produced from this soft version of $w_r$, after being affected by $\lambda_d$ as:

$$\dot{\lambda} = \lambda_d w_{rf} \tag{5}$$

this tuning parameter $\lambda_d$ represents the maximal speed reference of the path in normal conditions, it means without potential collisions. This new signal $\dot{\lambda}$, the adapted motion parameter, is integrated in order to generate $\lambda$ and feed the path generator block.

In the operation of the system there are two situations. In the first situation no collision is detected so, the robot is going to follow a given pre-elaborated path and the speed adaption loop rests inactive, so the $w_r$ signal is equal to “1”. Here the dynamics of the system is governed by the main control of the robot. When a collision situation arrives, it means that the dynamics of approaching is faster than the desired one (defined by Eq. 2), the discontinuous signal $w_r$ changes its value to “0” and then after passing the low-pass filter affects the $\dot{\lambda}$ parameter slowing down the increase of the $\lambda$ parameter. Actually, during this condition a fast commutation of $w_r$ signal forces the system to follow the desired dynamics imposed by the sliding mode (SM) surface $\sigma = 0$. When the collision situation vanishes the system returns to the first situation. This fast commutation is possible due to $\dot{\sigma}$ depends on $w_r$, i.e. $\sigma$ has relative degree one with respect to $w_r$ (see Eq. 2-8), which is a necessary condition for the SM establishment. In consequence, the system will slide over $\sigma = 0$ as long as the discontinuous signal $w_r$ is enough to change the value of $\dot{\sigma}$ from side to side of this surface.

The discontinuous signal $w_r$ is going to slow down the reference speed in function of the distance of approaching and its derivative, in the extreme case when the approach is too fast the speed adaption loop could not respect the desired dynamics (the SM could not be established) and it acts as and auxiliary break which stops the robot.

Some extra considerations over the technique are:

- The choice of the cut frequency in the low pass filter ($\lambda_f$) has important effects in the response of the system, on the one hand a too slow behavior could drive to slow reaction of the system in front of a sudden obstacle, and on the other hand to fast behave results in a non smooth rolling of the robot through the path. Finally, its optimal value depends on the expected speeds of the mobiles involved and the acquisition rate of the distant measures.
- The choice of $k_d$ and $k_{dd}$ is restricted to the desired dynamics. During the SM, the dynamic behaviour imposes is exponential with a time constant $\tau = k_{dd}/k_d$, which must be realizable for the robot. Furthermore the $k_{dd}$ parameter must be different from zero in order to fulfill the necessary condition for the SM establishment.
- It results important to remark that all the high frequency switching in the proposal is restricted to the logical part, thus facilitating their implementation.

More mathematical formalism related to proposed technique and the SM necessary and sufficient conditions can be found in [10] and [9]. In the next section this technique is tested over different situation where it is possible to get a general idea of its behaviors, and the possible applications covered by it.

## IV. APPLICATION TO A DIFFERENTIAL MOBILE ROBOT

In this section several simulations are explained in order to show the strength of the proposal, the implementation of simulations has been made through Matlab environment and V-REP simulator [7]. The latter offers not only realistic simulation graphics but also the capability of considering the real dynamics properties of the robots. We employ the Pioneer P3-DX robot which is available in the library of the program and a validated model.

### A. Robot and inner controller description

The Pioneer P3-DX mobile robot (Fig. 1(a)) respond to a differential robot model, which could be modeled as follows:

$$\begin{aligned} \dot{x} &= \nu \cos\theta \\ \dot{y} &= \nu \sin\theta \\ \dot{\theta} &= \omega \end{aligned} \tag{6}$$

here $\mathbf{p}(t) = [x(t), y(t), \theta(t)]^\top$ is the vector of the vehicle's Cartesian coordinates and $\theta(t)$, its heading. The angle $\theta(t) \in (-\pi, \pi]$ is measured in the counter-clockwise direction from the x-axis, see Fig. 1(b). $\nu$ and $\omega$ are the speed and angular velocity respectively, both bounded variables.

The main control proposed is conformed by two independent proportional actions for the $\nu$ and $\omega$ command signals, and a corresponding feedforward action. Fig. 4 shows the proposed configuration where $\eta = [x_r, y_r, \theta_r]^\top$ is the path reference, $\mathbf{e} = [x_r - x, x_y - y, \theta_r - \theta]^\top = [e_x, e_y, e_\theta]^\top$ is error vector, and $\mathbf{u} = \mathbf{u_c} + \mathbf{u_f} = [\nu, \omega]^\top$ is the control signal.

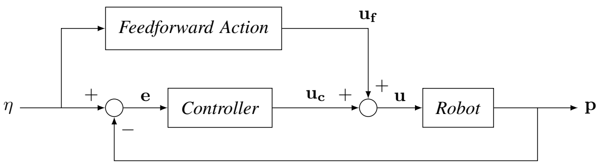


Fig. 4. Inner control for path following

The equation that govern the control is establish as:

$$\mathbf{u_c} = \begin{bmatrix} k_{p\nu}\sqrt{e_x^2 + e_y^2} \\ k_{p\omega}\,\mathrm{atan2}(sin(e_a), cos(e_a)) \end{bmatrix} \tag{7}$$

$$e_a = \mathrm{atan2}(e_x, e_y) - \theta$$

and the corresponding feedforward action as:

$$\mathbf{u_f} = \begin{bmatrix} k_{f\nu}\sqrt{\dot{e_x}^2 + \dot{e_y}^2} \\ k_{f\omega}\dot{\theta}_r \end{bmatrix} \tag{8}$$

here, $k_{p\nu}$, $k_{p\omega}$, $k_{f\nu}$ and $k_{f\omega}$ are the tuning parameters for the control set. Note that the function $\mathrm{atan2}(x, y)$ is the arctangent function where its result is the angle in radians between the positive x-axis of a plane and the point given by the coordinates (x, y) on it.

No further details will be given about the controller, the interested reader can consult [13] where more details about the control and model can be found.

### *B. Simulation results*

*1) Fixed obstacle:* The first simulation as scenario shows the robot following a straight path which drives it against a collision with a fixed obstacle, see Fig. 5. At the beginning of the simulation our robot is too far from the obstacle so the speed adaptation loop rests inactive and the main controller governs the system. As the robot advances in the path the distance to the obstacle decreases, at the moment that the distance and its derivative break the maximum desired approaching dynamics the SM is established. The auxiliary loop signals over the time can be seen in Fig. 6. It is possible to observe at time 18 [s] that the SM starts, and consequently the $\lambda$ parameter slows down its increase. Once the auxiliary loop is active, it forces the system to follow the desired dynamic slowing down the robot speed up to stop the robot just in the border of the safety condition, where $d = d_{safe}$. For this simulation the tuning parameters were $d_{safe} = 1$, $k_d = 1$, $k_{dd} = 1$, sampling time $T_s = 10[ms]$, cut-off frequency of the low pass filter $fc = 0.4[Hz]$, and $\lambda_d = 0.2$.

It results interesting to observe the plane $d$ vs. $\dot{d}$ in Fig. 7, it shows how the system evolves for different path speed set-points. Lets pay attention to the "blue line". First, the robot starts from a rest condition in the "A" point and evolves with the main controller dynamics. When the robot reaches the desired approaching dynamics, "C" point, the speed adaption loop is activated an forces the system to follow the sliding surface, which is represented by the line which crosses the "E-C" points. The system continues with the desired dynamics up to the point "E" which represents the limit of the allowed distance with the approaching speed equals to zero. To show the strength of the purpose also in Fig. 7 is possible to see other two simulations where the speed of approach are different, one slower than the previous one with $\lambda_d = 0.1$ and other faster with $\lambda_d = 0.3$. Both cases show the same behavior. Note that the greater the speed the greater the distance to the obstacle at which the adaptive loop starts acting, see Fig. 5.

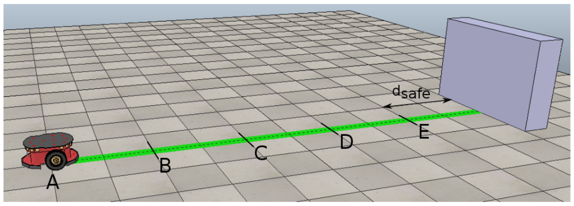


Fig. 5. Fixed obstacle situation

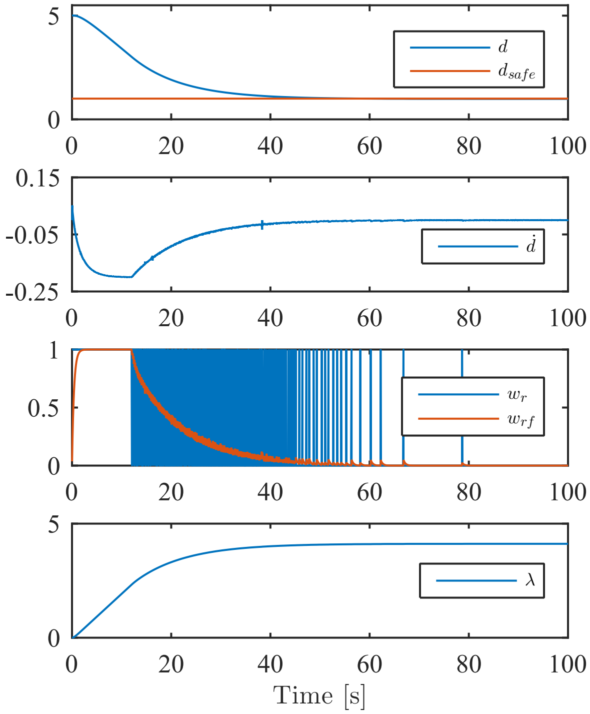


Fig. 6. Speed adaption loop's signals for a fixed obstacle

One of the main characteristic of the proposal is the possibility of imposing different desired dynamics. Fig. 8 shows the same simulation but with different dynamics imposed through the change of the $k_{dd}$ parameter. In this figure it is also possible to observe a technique limit: in the case of the green curve the demanded approaching dynamic (dotted line) is to fast for the system so the auxiliary loop works as a emergency break, the stopping dynamics being the fastest which the robot can follow. Naturally, this situation can be avoided as long as $k_{dd}$ is properly chosen according to the robot dynamics features and the expected obstacle speed in the case of moving scenario.

*2) Moving obstacles:* In this situation we are more concerned about the response of the system in the condition when moving obstacles are in the environment. In Fig. 9 the setup of the simulation is presented. Here we have a main robot with the proposed technique implemented which must follow a straight path (green one). Also we have two mobile obstacles which follow perpendicular paths to the first one. One of the obstacles is another Pioneer robot with a classical path following controller, which must follow the red path. The other

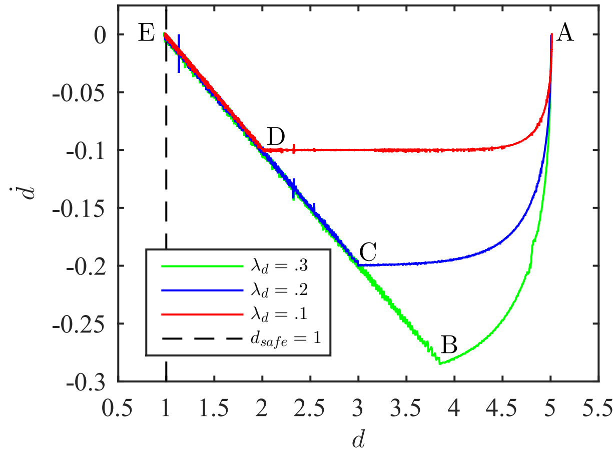


Fig. 7. Different speeds during the approach to obstacle

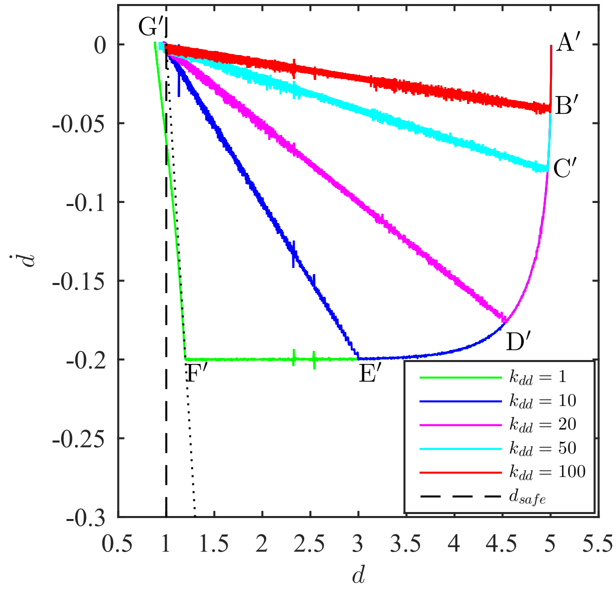


Fig. 8. Response of different dynamics imposed to the robot

obstacle is a human being who must follow the blue path, the last one moves faster than the robot. Fig. 10 shows the

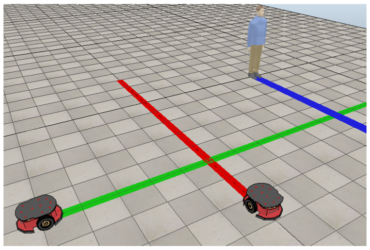


Fig. 9. Moving obstacles situation

speed adaptation loops' signals. It is possible to see that the main robot starts its movement following the path according to the main controller dynamics. But soon at time 2 [s] the speed of approach and the distance to the first obstacle shoots the auxiliary loop slowing down the speed of the main robot, giving time to the first obstacle to cross the path. Once the obstacle begins to increase its distance, the auxiliary loop, still active, allows the main robot to accelerate always following the desired dynamics up to the moment where the loop changes again to the inactive state ($w_{rf} = 1$, $\dot{\lambda} = \lambda_d$). Then the main robot faces another collision situation in time 18 [s], in this case is the human who approximates much faster than the previous obstacle so the speed adaption loop does not have time to establish an SM. In consequence, it acts as an emergency break which stops the robot allowing the human to cross the path up to the time 20 [s] where the human is moving away from the robot position. In this last condition a SM is established when the robot starts to move again, and finally the inactive condition of the loop is reached. The main objective of this simulation is to observe the behavior in front of two obstacles moving at different speeds.

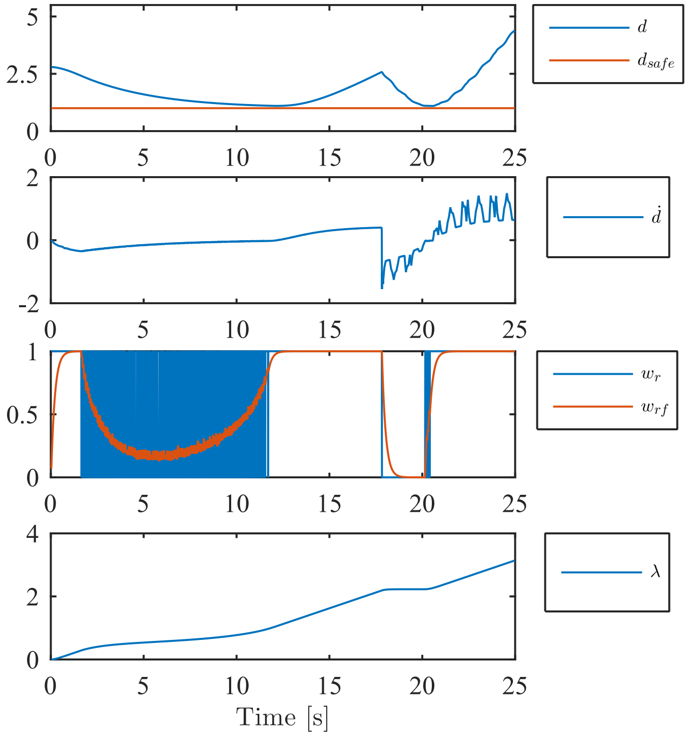


Fig. 10. Speed adaption loop's signals for a mobile obstacle

*3) Corridor situation:* In this case the idea of the simulation is to test the ability of the proposed technique to adapt the robot's speed to any other speed vehicle which must share the path followed. The setup of the simulation can be appreciated in Fig. 11. Here we have two vehicles, the main robot which have the proposed technique implemented and must follow the green path, and another robot (from here called the obstacle) with a traditional path following controller which must follow the red one. Both robots share a section of the path, that could be a corridor in a real factory situation. In addition to the previous description, the main robot can move faster than the obstacle, but is the second one who first reach the corridor.

In Fig. 12 the speed adaption loops' signals can be observed. Here at the beginning the robot starts to reduce its speed due to the proximity to the obstacle (time 1[s]). As soon as the obstacle gets into the corridor (time 8[s]), it increases the distance to the robot, so the auxiliary loop allows increasing the reference speed. The robot and the obstacle speeds can be seen in Fig. 12 as $\| \mathbf{v_r} \|$ and $\| \mathbf{v_{ro}} \|$ respectively. Once both vehicles are in the corridor it is possible to see how the auxiliary loop modifies the speed of the robot to follow the obstacle just in the border condition of $d_{safe}$. As the obstacle must do a 90 degrees turn at the end of the corridor it must reduce its speed, and the speed adaption loop breaks the main robot (time 27 [s]), then it starts again the movement (time 30 [s]) respecting the desired dynamics up to the conditions of activation for the loop vanish (time 34 [s]).

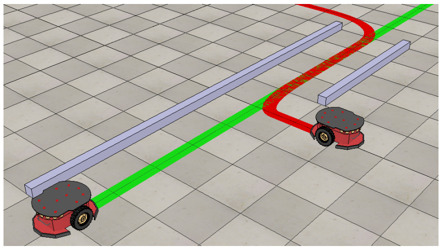


Fig. 11. Corridor situation

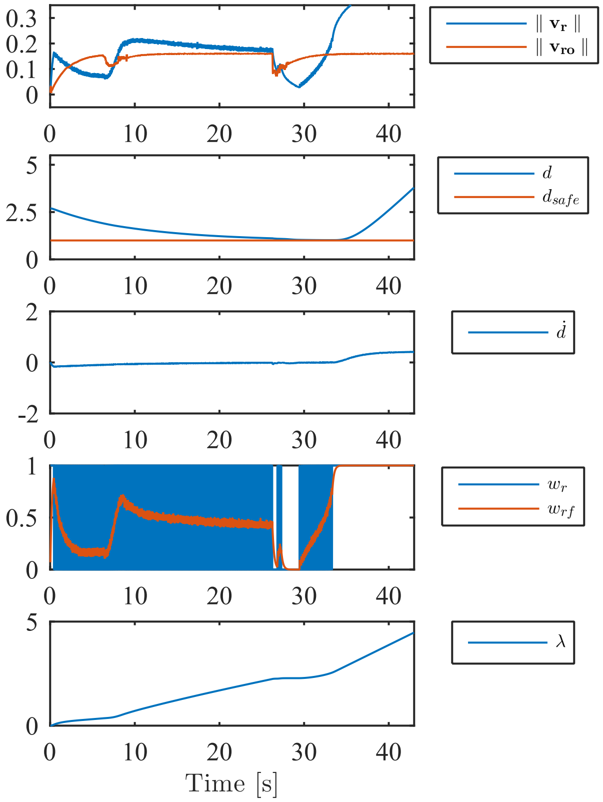


Fig. 12. Speed adaption loop's signals for a mobile obstacle in a hall condition

## V. CONCLUSION

In this work an obstacle avoidance technique for strict path following tasks has been set up, and tested through different simulations. Simulations prove that the technique is realizable and has interesting value for practical applications. The main characteristics are the simplicity of its implementation (just a few lines of code) and the ability to imposed a desired dynamic in a collision situation. Furthermore, the feature to adapt the speed of reference in path shared situation is also remarkable.

In future works it is expected to implement this technique in real robots in order to have a complete validation. Another two aspects that require future work is to bound the parameters involved in the operation of the technique to derive the necessary and sufficient condition for the SM establishment, and to consider modifications in the technique towards a more intelligent reaction when a the collision dynamics overpass that one which is allowed, for example in a frontal collision situation.

## ACKNOWLEDGMENT

The authors would like to thank the partial support of CONICET (PIP 112-201501-00837), MINCYT and UNLP(I216).

## REFERENCES

[1] Y. Koren and J. Borenstein, *Potential Field Methods and Their Inherent Limitations for Mobile Robot Navigation*. Proceedings of the 1991 IEEE International Conference on Robotics and Automation Sacramento.

[2] Y. Zhu, T. Zhang, J. Song and X. Li, *A new method for mobile robots to avoid collision with moving obstacle*. Artif Life Robotics (2012) 16:507510 DOI 10.1007/s10015-011-0975-z

[3] L. Oliveira da Costa, M. B. Arantes de Souza, E. J. de Oliveira and L. Willer de Oliveira, *Optimal Speed of Multiple UAVs based on Nonlinear Programming to Avoid Collision in Periodic Paths*. EngOpt 2016 - 5th International Conference on Engineering Optimization

[4] D. Wilkie, J. Van den Berg and D. Manocha, *Generalized Velocity Obstacles*. The 2009 IEEE/RSJ International Conference on Intelligent Robots and Systems October 11-15, 2009 St. Louis, USA

[5] Y. Zhao and P. Ioannou, *Positive Train Control With Dynamic Headway Based on an Active Communication System*. IEEE Transactions on intelligent transportation systems, Vol. 16, No. 6, December 2015

[6] J. L. Rosendo, B. Clement and F. Garelli, *Sliding mode reference conditioning for path following applied to an AUV*. CAMS 2016 (IFAC Conference on Control Applications in Marine Systems), Trondheim, Norway, DOI: 10.1016/j.ifacol.2016.10.314.

[7] E. Rohmer, S. P. N. Singh, M. Freese, "V-REP: a Versatile and Scalable Robot Simulation Framework," IEEE/RSJ Int. Conf. on Intelligent Robots and Systems, 2013

[8] James L. Crowley *World Modeling and Position Estimation for a Mobile Robot Using Ultrasonic Ranging*. 1989. Proceedings. IEEE International Conference on Robotics and Automation, DOI: 10.1109/ROBOT.1989.100062.

[9] L. Gracia, F. Garelli, and A. Sala, *Reactive Sliding-Mode Algorithm for Collision Avoidance in Robotic Systems*. IEEE Transactions on Control Systems Technology (Volume: 21, Issue: 6, Nov. 2013), DOI: 10.1109/TCST.2012.2231866.

[10] F. Garelli, L. Gracia, A. Sala, P. Albertos, *Sliding mode speed auto-regulation technique for robotic tracking*. Robotics and Autonomous Systems Volume 59, Issues 78, JulyAugust 2011, Pages 519529.

[11] P. Moghadam, W. S. Wijesoma and D. J. Feng , *Improving Path Planning and Mapping Based on Stereo Vision and Lidar*. 2008 10th Intl. Conf. on Control, Automation, Robotics and Vision, Hanoi, Vietnam, DOI: 10.1109/ICARCV.2008.4795550.

[12] S. Y. Nof, *Handbook of Industrial Robotics*. Wiley, 1999. ISBN: 9780471177838

[13] Siciliano, Bruno, Khatib, Oussama, *Handbook of Robotics*. Springer, 2008. ISBN: 9783540239574