Stable Switching Control of Wheeled Mobile Robots

Juan Marcos Toibero, Flavio Roberti, Fernando Auat Cheein,
Carlos Soria and Ricardo Carelli
Universidad Nacional de San Juan
Argentina

1. Introduction

This Chapter addresses the wheeled autonomous mobile robots navigation problem starting from very simple solutions which are used to solve more complex tasks. This way, it is considered a hybrid (discrete-time/continuous-time) approach, composed by several individual continuous controllers, each of them solving a particular navigation problem, but when are considered as a whole constitute a new approach for more demanding navigation challenges. The discrete part of this system is compound by a high order supervisor, which gives the logic switching. In particular, this Chapter deals with Switching Systems, which are hybrid system characterized by the fact that only a unique controller could be active at any time. This kind of control architecture presents some benefits as for example the modularity and the possibility to analyze the whole system behavior, allowing the designer to conclude stability. It also allows, conversely, decomposing a given task in simpler subtasks which could be treated separately in order to facilitate the design process.

The interest on asymptotic stability is due to our interest on the control system analysis. For this reason, several well-known concepts such as Lyapunov stability and common Lyapunov functions (Liberzon, 2003) will be treated along this Chapter, in order to prove stability for the individual controllers as well as for the switching controller. So, it will be shown under a control point of view the design and analysis of stable switching controllers and two applications examples to wheeled robots navigation: firstly, developing a switching controller for reactive obstacle avoidance (Toibero et al., 2009); and secondly, its applications at the local navigation level in a simultaneous localization and map building (SLAM) task (Auat Cheeín, 2009).

Robotic autonomy is directly related to the capability the robots have to perform given task without continuous human guidance. Arkin (1998) proposed a way to deal with this autonomy, by decomposing a task into individual behaviors such as e.g. move-to-goal, find-next-point, return-to-start, wander and halt. Next, these behaviors may be selected or fused according to a desired performance. Within the several available behaviours, the interest is on a key feature for robot autonomy: obstacle detection, identification and avoidance.

In this context, there are two main approaches: path planning controllers or reactive controllers. In path planning controllers, the obstacle is to be detected and avoided by defining a new secure path. Instead of that, reactive controllers only react in response to the sensory inputs without major computational effort. In the following, a wall-following (WF) controller already designed is considered as the basis of our obstacle avoider controller. A WF task is characterized by maintaining a constant distance from the robot to a wall (or to an object) and can be properly combined with other tasks in order to obtain higher degrees of autonomy for the mobile platform, e.g. the skill Go-to-goal avoiding obstacles in (Boada et al., 2003). Typically a WF controller has to recognize the distance and orientation errors between the robot and the wall. These two control states can be measured or estimated using sonar, laser rangefinder or information extracted from a video camera. Besides, WF controllers can be easily adapted for navigation along corridors by setting a desired distance relative to the centre of the corridor. Regarding the environment surrounding the robot, a WF controller becomes useful for reactive navigation in unknown environments. In this context, it can be used as an obstacle avoider by considering the obstacle as part of a wall to be followed. Therefore, it is relevant to design controllers for walls with several kinds of contours. The inclusion of such situations leads to a new control problem: to deal with discontinuities on the wall contour. To solve this problem most of the papers in the literature use fuzzy-logic: (Braunstingl & Ezkerra, 1995), (Wang & Liu, 2004), (Ando, 1996), or switching controllers: (Fazli & Kleeman, 2005), (Borenstein & Koren, 1989), (Zhang et al., 2004). The WF problem, as an important component of current robotic research, has been widely considered. In the early work in (van Turennout & Hounderd , 1992) a WF controller was used to avoid obstacles and to follow unknown walls. Then (Ando, 1996) proposes a way to extend the capabilities of this behavior but as a path planning problem, without describing the controller. In (Braunstingl & Ezkerra, 1995) the contour following problem is treated by using a fuzzy logic controller. More recently (Bicho, 2000) has used neural networks in order to estimate the relative orientation of the robot and the wall. In (Zhang et al., 2004) a complete switching controller that allows the robot to track sharp discontinuous trajectories is presented. This last switching approach includes a stability demonstration at the switching times but the controllers need a path to be tracked and the paper does not consider obstacle avoidance while tracking the trajectory. Here, the WF problem is addressed by using a laser rangefinder and odometry in fully unknown environments. Initially, it will be assumed that the wall's contour is smoothly varying and it is proposed an expression relating the contour variations with the saturation of control actions which is useful to design a saturation-free controller. This first part includes the simpler case of straight walls. Then, the contour-following (CF) problem is treated as an extended approach of the WF problem when considering discontinuities such as corners. For this CF controller three individual subsystems were considered, one characterizing each behavior: re-orientation, wall-following and circle-performer. The stability at the switching times is considered and discussed.

In addition to the use of WF in move-to-goal and obstacle avoidance behaviors considered by several authors, other complementary applications of a CF controller are the mapping of interior environments (Auat Cheein, 2009), (Edlinger & von Puttkamer, 1994), (Fazli & Kleeman, 2005) in surveillance tasks and within teleoperation contexts (Wang & Liu, 2004) using a similar behavior for telecommanding a mobile robot.

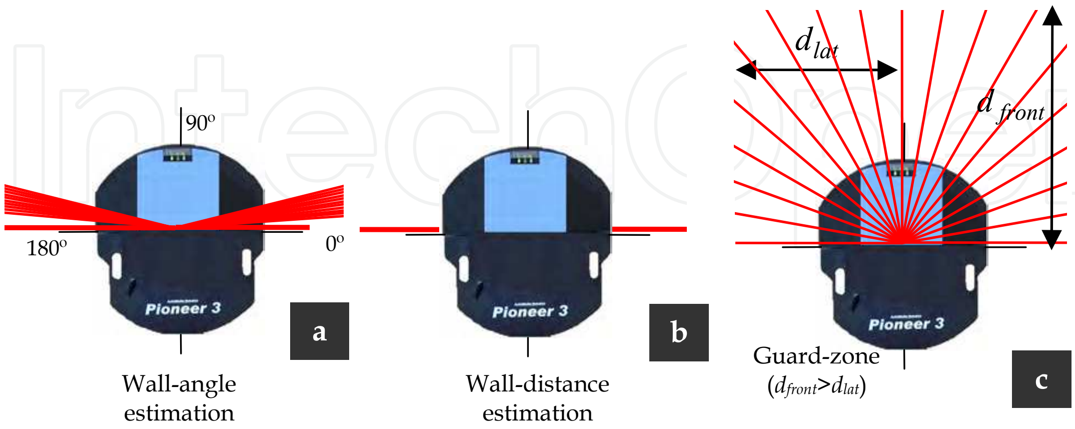
The remaining of this Chapter is organized as follows. Section 2 presents the mobile robot kinematcis. Section 3 proposes a WF controller to maintain this desired distance for straight and smooth varying walls based on laser and odometric information only. Next, in Section 4 some limitations of this approach are shown; and a switching scheme that allows the fully contour-following of the object is presented, including several experimental results in Section 5. As a final example application, in Section 6 it is shown the incorporation of this controller to a SLAM task showing also additional experimental results.

2. Mobile Robot

The unicycle-type wheeled mobile robot is to be considered in this Chapter. Let us set the specific notation, with reference to Figure 1, the global or task coordinate system is denoted by [W] and the coordinate system attached to the robot by [R]. The robot states variables are x, y and θ; where θ denotes the heading of the vehicle relative to the ${}^{W}X$-axis of the world coordinate system. Vector $[x \quad y \quad \theta]^T$ defines the posture of the vehicle. The kinematics of the robot can be modeled by the well known kinematics model.

$$\dot{\chi} = \begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{\theta} \end{bmatrix} = f(\chi, u) = \begin{bmatrix} cos(\theta) & 0 \\ sin(\theta) & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} v \\ \omega \end{bmatrix} \tag{1}$$

The robot is equipped with a laser radar sensor. As it can be seen in Figure 1, some lateral beams are used to estimate the object contour angle, whereas all beams are used to define a safety-zone, which purpose is to detect possible collisions between the robot and the object. The selection of the side of the object to follow is based on the initial conditions. Firstly the robot computes the object orientation and the distance to the object for its right side and also for its left side, then, the robot will take the nearest side to follow and will keep following this side. Each beam is identified by its angle, e.g. for the beam at 0°, the notation *d000* is used to denote the distance measured by this beam. From here on, to simplify the notation, it will be assumed that the robot follows only the wall at the right side of the robot.



Wall-angle estimation

Wall-distance estimation

Guard-zone (d_{front}>d_{lat})

Fig. 1. Laser radar

3. Standard Wall-Following Control

A WF controller will be referred as *standard* in this Chapter, when do not allow to follow discontinuous walls. It could be said that allows the robot to follow continuous (straight) and smooth varying contours. This appreciation is based on its stability proof. Two control states must be defined: the robot-to-wall angle in Section3.1; and the robot-to-wall distance in Section3.2.

3.1 Wall Angle Computation

The wall angle estimation is performed by using only two beams: one perpendicular to the heading of the robot, and the other may be selected depending on the desired robot-wall distance. In order to obtain a more reliable value, it is considered a group of beams at each side of the robot, thus resulting ten different angle values. Each value is obtained according to

$$\theta_{wall}^{10} = atan\left(\frac{-\Delta y}{-\Delta x}\right) = atan\left(\frac{y_{010}-y_{000}}{x_{010}-x_{000}}\right). \qquad (2)$$

These values are then fused together using a decentralized Kalman filter (Brawn & Hwang, 1997) which returns an improved estimation for the wall angle θ_{wall} with respect to the world coordinate system. In Toibero et al. (2009) details and derivations of this filter can be found.

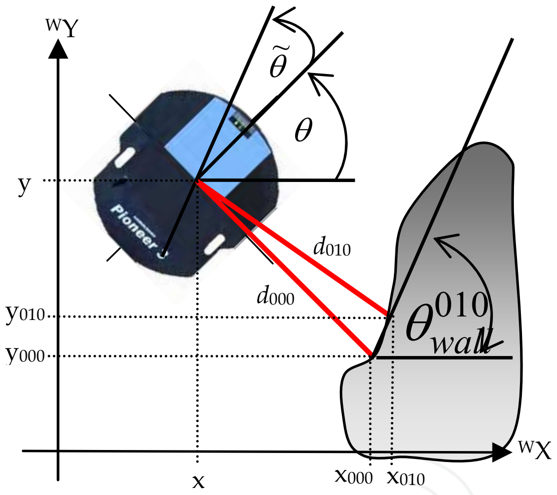


Fig. 2. Wall angle computation. Resulting in the value $\theta_{wall}^{010}$of the wall-angle related to robot posture x, y and θ in the world coordinate system for the beam at 10º

3.2 Robot-wall distance computation

The wall-distance computation to a wall at the right side of the robot is obtained by

$$d_{wall} = d_{000}\, cos(\tilde{\theta}) \qquad (3)$$

where $\tilde{\theta}$ is the difference between the heading angle of the robot and the wall angle. This angle is properly defined in the next paragraphs. It can be obtained a similar expression for the distance to a wall at the left side of the robot by replacing $d_{000}$with d_{180}. Note that according with (3) it is assumed that the distance from the robot to the wall is considered for null orientation error.

3.2.1 Controller description and analysis

This controller renders a control law for wall-following based on the laser sensorial information and the odometry of the robot. The reference for this controller is the desired distance from the robot to the wall d_{des}. As can be seen in Figure 3, the control errors are defined (Toibero et al., 2006b), as follows

$$\tilde{d} = d_{des} - d_{wall} \tag{4.a}$$
$$\tilde{\theta} = \theta_{wall} - \theta \tag{4.b}$$

Equation (4.a) states the distance error between the actual distance to the wall and the desired distance to the wall d_{des} while equation (4.b) states the orientation error between the estimated wall angle and the heading angle of the robot in the world coordinate system [W].

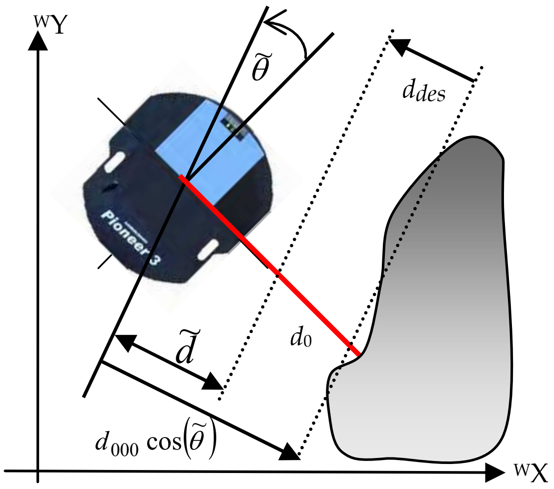


Fig. 3. Wall-following controller scheme. Distance error calculation. Note that the value for the distance error could be negative.

The time variations of the control errors are given by

$$\dot{\tilde{d}} = -v\, sin(\tilde{\theta}) \tag{5.a}$$
$$\dot{\tilde{\theta}} = \dot{\theta}_{wall} - \omega \tag{5.b}$$

where, $\dot{\theta}_{wall}$ is the derivative of the reference for the wall orientation. This derivative is null for a straight wall. Now, taking into consideration the control system analysis, the following - globally positive definite and radially unbounded - Lyapunov candidate function is proposed

$$V = \tfrac{1}{2}\tilde{\theta}^2 + \int_0^{\tilde{d}} k_{\tilde{d}}(\xi)\, \xi\, d\xi, \tag{6}$$

where $k_{\tilde{d}}(\tilde{d})$ was selected as in (Kelly & Carelli, 1996) and ξ is a dumb variable:

$$k_{\tilde{d}} = \tfrac{k_1}{1+|\tilde{d}|} \geq 0. \tag{7}$$

The time derivative of the Lyapunov candidate function is

$$\dot{V} = \tilde{\theta}\dot{\tilde{\theta}} + k_{\tilde{d}}\,\tilde{d}\,\dot{\tilde{d}} \tag{8}$$

Then, the proposed control actions are

$$v = v_{max} \geq 0 \tag{9.a}$$

$$\omega = \dot{\theta}_{wall} + K_{\tilde{\theta}}\,tanh\left(k_{\tilde{\theta}}\,\tilde{\theta}\right) + k_{\tilde{d}}\,\tilde{d}\,v\,\frac{sin(\tilde{\theta})}{\tilde{\theta}} \tag{9.b}$$

where v_{max} is the desired linear velocity; $K_{\tilde{\theta}} \geq 0$ is a positive definite design constant, and $k_{\tilde{\theta}}$ is a design constant that indicates the slope of the *tanh* function which can be adjusted with $K_{\tilde{\theta}}$ in order to bound the control action to its saturation values. By substituting (9) in (8) the derivative of the Lyapunov candidate function becomes

$$\dot{V} = -K_{\tilde{\theta}}\,\tilde{\theta}\,tanh(k_{\tilde{\theta}}\,\tilde{\theta}) \tag{10}$$

Equation (10) states that the control system is Lyapunov stable but not asymptotically stable. To prove the asymptotic stability the autonomous nature of the closed-loop system is considered. The closed-loop equations can be obtained by replacing the control actions in (5)

$$\dot{\tilde{d}} = -v_{max}\,sin\,(\tilde{\theta}) \tag{11.a}$$

$$\dot{\tilde{\theta}} = -K_{\tilde{\theta}}\,\tilde{\theta}\,tanh\left(k_{\tilde{\theta}}\,\tilde{\theta}\right) - k_{\tilde{d}}\,\tilde{d}\,v_{max}\,\frac{sin(\tilde{\theta})}{\tilde{\theta}} \tag{11.b}$$

By applying the Krasovskii-LaSalle theorem (Vidyasagar, 1993) in the Ω region, with

$$\Omega = \left\{\begin{bmatrix}\tilde{d}\\ \tilde{\theta}\end{bmatrix} : \dot{V}\left(\tilde{d},\tilde{\theta}\right) = 0\right\} \Rightarrow \left\{\begin{bmatrix}\tilde{d}\\ \tilde{\theta}\end{bmatrix} = \begin{bmatrix}\tilde{d}\\ 0\end{bmatrix}\right\} \tag{12}$$

the only invariant is $\tilde{d} = 0$. Therefore, by invoking LaSalle theorem, it can be concluded that the origin of the state space is globally uniformly asymptotically stable.

3.2.2 Experimental results

This controller was proved in a real office environment with a Pioneer3-DX mobile robot. The robot length is about 330millimeters. The values of the design parameters were set to: $k_1 = 0.025$, $K_{\tilde{\theta}} = 11.5$, $k_{\tilde{\theta}} = 0.3$ and $d_{des} = 500$millimeters. In order to show the performance of the proposed controller, three different experiments were carried out:

i) The robot follows a straight wall.
ii) The robot follows a wall with smoothly varying contour.
iii) The robot follows an almost-circular contour.

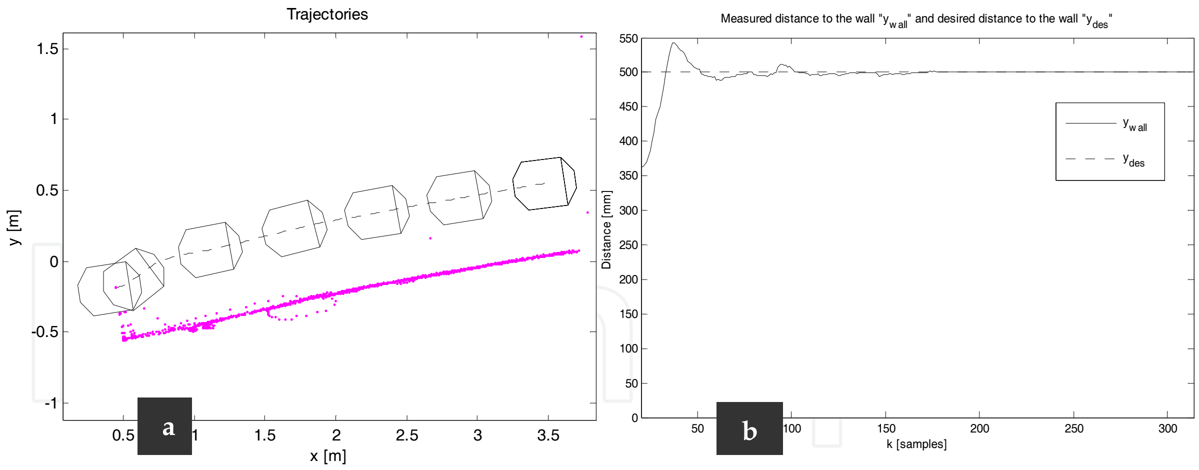


Fig. 4. a) Trajectory described by the robot following a wall with straight contour. The contour of the wall was reconstructed by using laser information. b) Distance between the robot and the straight wall. The linear velocity for this experiment was set to 200millimetres per second.

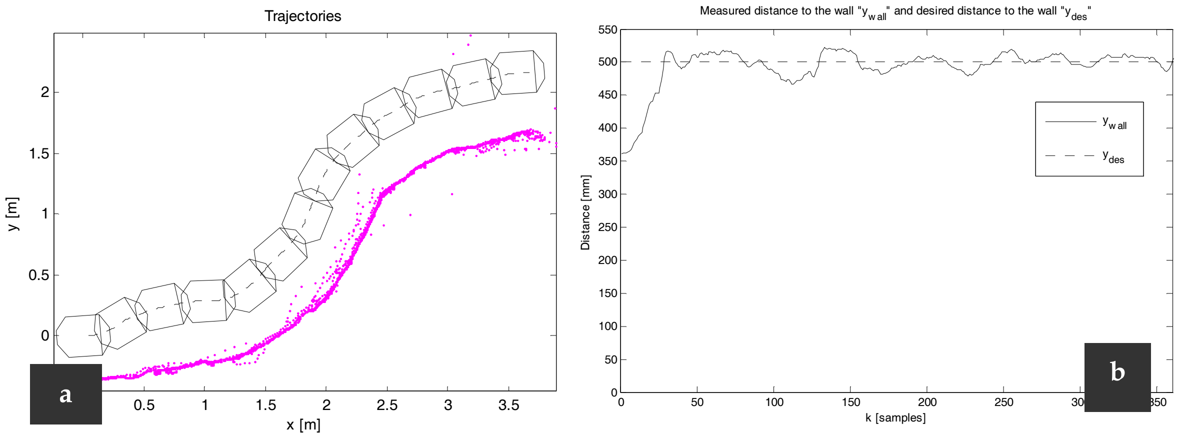


Fig. 5.a) Trajectory described by the robot following a wall of variable contour. As can be seen, the wall presents variations on its contour and on its texture. b) Robot wall distance in the variable contour experiment. The error is less than 25millimetres – a small value when compared to the size of the robot.

From Figures 4 to 6, a good performance of the proposed controller can be concluded. For all the experimental situations the errors are not significant. But it can be noted the impossibility of the controller to follow discontinuous walls' contours. These results inspire the development of the Contour-following controller of the next Section.

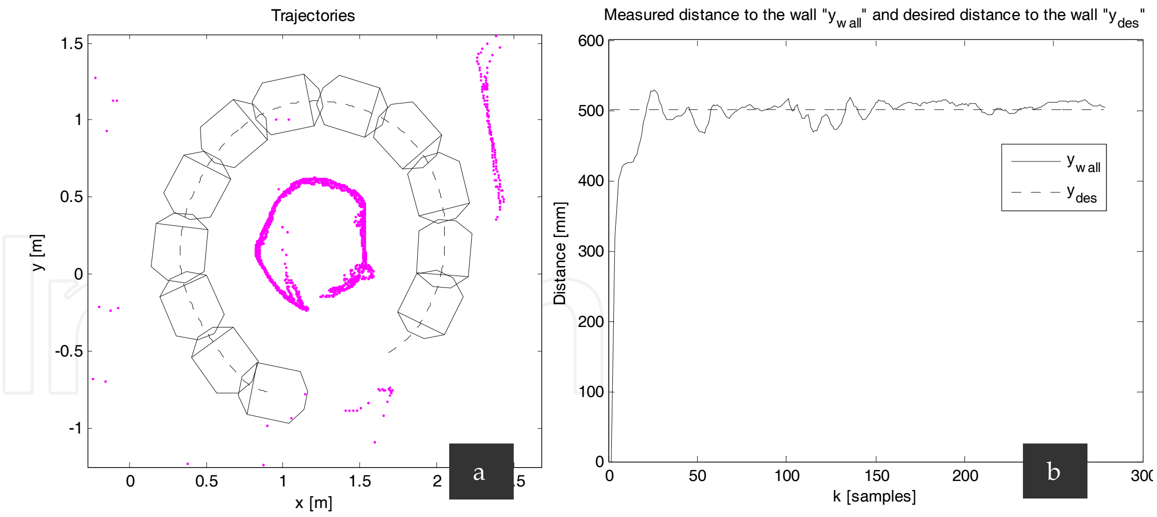


Fig. 6. a) Robot following an almost-circular contour. The linear velocity for this experiment was set to 200millimetres per second. b) Robot wall distance in the circular path experiment. The error is less than 25millimetres.

4. Contour-Following Controller

A stable switching between several controllers is a proper way to improve performance or to achieve control objectives that are difficult or impossible to consider under continuous nonlinear control. The proposed switching controller is designed to follow the contour of objects which have at least a size comparable with that of the robot (Toibero et al., 2006a). The basic idea is to use the wall-following controller described in Section 2. The controller is designed within the context of stability theory for switching systems. The WF controller acting alone shows a good performance when following a wall, but there are two cases which it cannot deal with:

i) Abrupt variations in the contour of the object which makes it difficult to estimate the actual object's angle. In this case (Figure 7.a) the distance measured by the laser beams varies suddenly.

ii) Abrupt variations in the contour due to an interior corner that causes the robot to crash against the object (Figure 7.b).

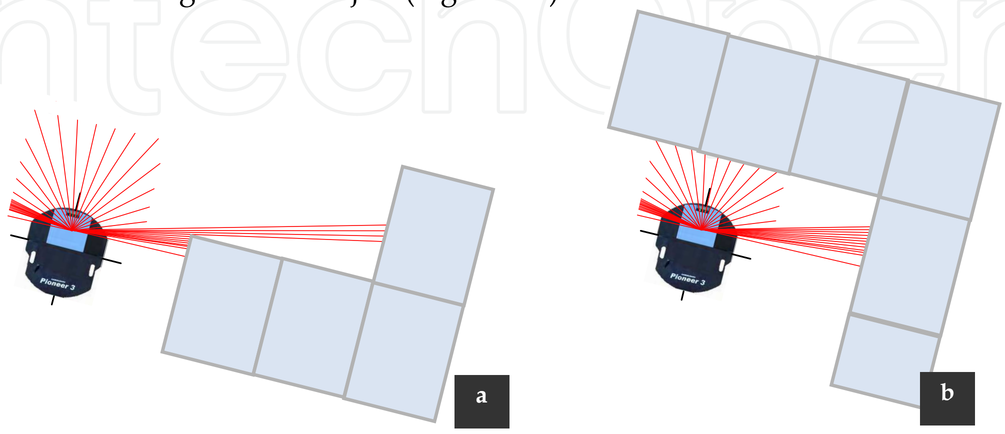


Fig. 7. Main problems when following contours: a) the presence of an open corner and b) the presence of an interior corner

The first step is given by the proper detection of these situations, and fortunatelly, both can be detected by using the laser radar. For example, the first case can be detected by computing the variance of the length of the lateral beams (note that these lateral beams estimates the wall angle). So, this variance would have a value close to zero when the robot is following a wall, but it will become larger when this contour is lost. Therefore, a threshold value can be defined in order to generate a *loss of wall* event. On the other hand, for the second case, collisions can be detected as an invasion of the robot safety-zone. If the length of any beam lies within this predefined safety-zone, then a *possible collision* event is generated.

At this point, we aggregate two new individual subsystems, one to deal with each situation, thus building a switching system.

4.1 Handling the case of missing the wall to follow

In this case, the robot has lost the wall to follow. This behavior is activated when non-consistent measurements of the control states $\tilde{\theta}$ and $\tilde{d}$ are obtained. The proposed solution is to describe a circular path of radius R on the floor (Figure 8) until one of the following conditions is fulfilled:

i) The wall can be followed again using the wall-following behavior provided that $\tilde{d} = \tilde{d}_{lost}$ (Figure 8.a and Figure 8.b); being $\tilde{d}_{lost}$ the last value of $\tilde{d}$ just before switching. This way, it could be assumed that $\dot{\tilde{d}} = 0$ considering the value of $\tilde{d}$ does not change at the beginning and at the end of this behavior.

ii) A possible collision between the robot and the wall is detected (Figure 8.c.), as considered in Section 3.2.

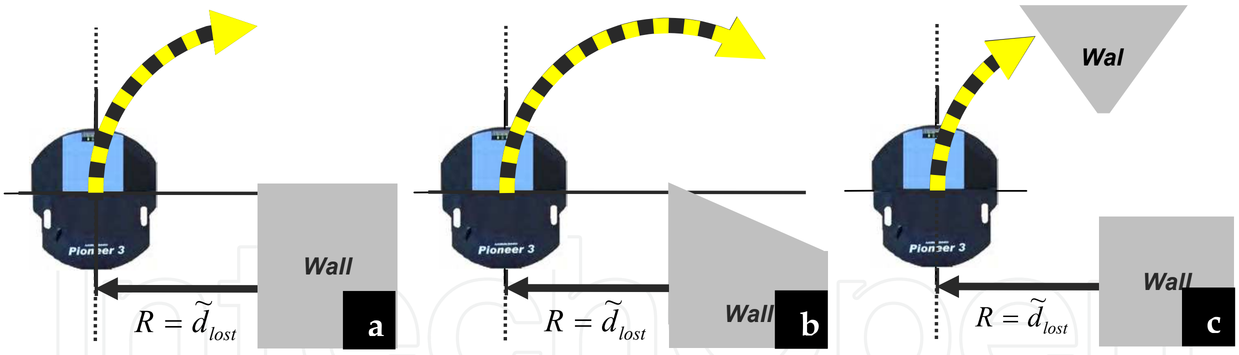


Fig. 8. Handling *"loss of wall"* situations

In order to be able to guarantee stability, a controller must be included, and this controller should share the same control states which the standart WF controller had. For this reason,we included a simple controller to describe a circular path of radius $R > 0$, based on the previous wall-following controller, though the distance error is not considered. Therefore, considering the following errors as in (4)

$$\tilde{\theta} = \theta_{circ} - \theta \tag{13}$$

where θ_{circ} is the heading reference to perform the desired circular path

$$\theta_{circ} = \pm \frac{v_{max}}{R}\, t \tag{14}$$

Here, the plus/minus sign of (14) denotes that the reference θ_{circ} will generate a left or right turn. That is, the robot will turn left (right) if it is following an object at its left (right) side. Then, replacing (14) into (13) and considering its time derivative, we obtain

$$\dot{\tilde{\theta}} = \dot{\theta}_{circ} - \omega = \pm \frac{v_{max}}{R} - \omega. \tag{15}$$

In order to analyze the stability of this control system, consider the following Lyapunov candidate function and its time derivative

$$V_{\tilde{\theta}} = \frac{\tilde{\theta}^2}{2} \tag{16}$$

$$\dot{V}_{\tilde{\theta}} = \tilde{\theta}\, \dot{\tilde{\theta}} = \tilde{\theta}\left(\pm \frac{v_{max}}{R} - \omega\right). \tag{17}$$

Now, to achieve the control objective, the following control actions are proposed

$$v = v_{max} \tag{18.a}$$

$$\omega = \pm \frac{v_{max}}{R} + K_{\tilde{\theta}} \tanh\left(k_{\tilde{\theta}}\tilde{\theta}\right), \qquad K_{\tilde{\theta}} > 0, k_{\tilde{\theta}} > 0. \tag{18.b}$$

Finally, by replacing (18) in (17), the asymptotic stability of the control system can immediately be proved. That is, this controller guarantees that the robot will perform a circular path by acting only on the robot angular velocity.

4.2 Handling possible collisions

This behavior is activated when an obstacle appears in front of the robot at a distance $d_{impact} \le d_{front}$, being d_{impact} the smallest distance to the obstacle measured inside the robot safety-area, which is also characterized by and angle d_{impact} on the laser range finder framework. Also, d_{front} (defined in Section2) is selected to be equal to the actual robot wall distance d_{wall} and $d_{lat} < d_{front}$. The objective of this behavior is to avoid possible collisions by rotating the robot until a free-path condition (characterized by an empty safety-area) is again achieved, and $\tilde{d}_{k+T} = \tilde{d}_k$, where k is the obstacle detection instant, and T is the time during this behavior was active until switching to the wall-following behavior. Under the above mentioned conditions, this behavior will always achieve a free- path condition satisfying $\tilde{d}_{k+T} = \tilde{d}_k$ (see Figure 9). Analogously to the behavior described in Section4.1, it could be assumed that $\dot{\tilde{d}} = 0$, considering that $\tilde{d}$ does not change at the beginning and at the end of this behavior.

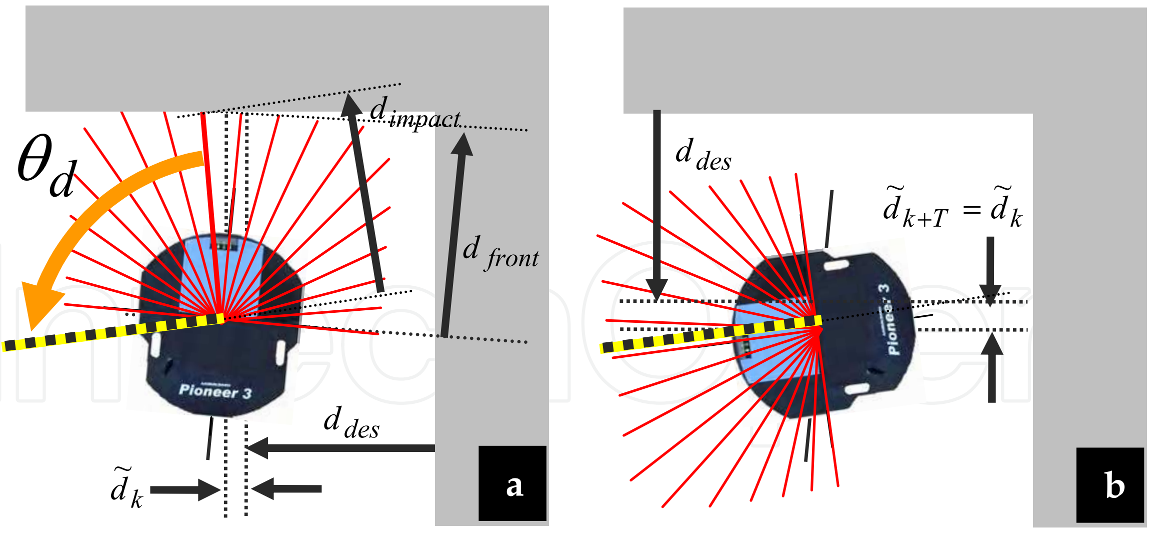


Fig. 9. Handling possible collision situations, a) obstacle detected at instant k, and b) free-path condition achieved with $\tilde{\mathrm{d}}_{\mathrm{k+T}} \leq \tilde{\mathrm{d}}_{\mathrm{k}}$ at instant $k + T$

To this aim, it is proposed a controller to allow the robot to positioning itself at a desired orientation angle θ_d. The orientation error between the heading angle of the robot and the desired orientation is defined as shown in Figure 10

$$\tilde{\theta} = \theta_d - \theta, \tag{19}$$

where $\theta_d = \theta_{impact} \pm 90^{\circ}$ is a constant value that is updated so as to attain an open area.

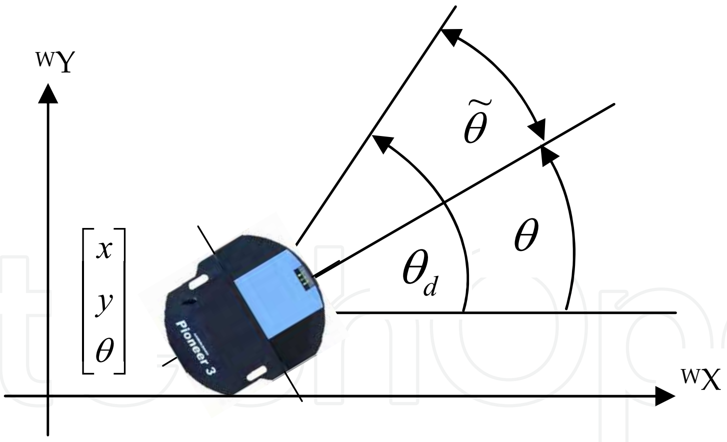


Fig. 10. Angular position controller description

Then, the time derivative of (19) is,

$$\dot{\tilde{\theta}} = -\omega \tag{20}$$

The asymptotic stability analysis of this controller can be proved by considering again (16) along with the following control actions:

$$\nu = 0 \tag{21.a}$$

$$\omega = K_{\tilde{\theta}} \tanh\left(k_{\theta}\, \tilde{\theta}\right), \qquad K_{\tilde{\theta}} > 0 \tag{21.b}$$

As done in the previous section, by replacing (21.b) into (20), the asymptotic stability of this control system, that is $\tilde{\theta}(t) \rightarrow 0$ as $t \rightarrow \infty$, can easily be proved.

5. Switching System

Once each part was designed it is necessary to desing the supervisor logics. Figure 11 presets the final block diagram, which includes three behaviors: wall-following, orientation and rotation (circular path performer). The switching signal σ generated by the supervisor, takes one of three possible values: a) $\sigma = 0$ if the controller for object-following is active, b) $\sigma = 1$ if the orientation controller is active and c) $\sigma = 2$ if the controller to perform a circular path is active.

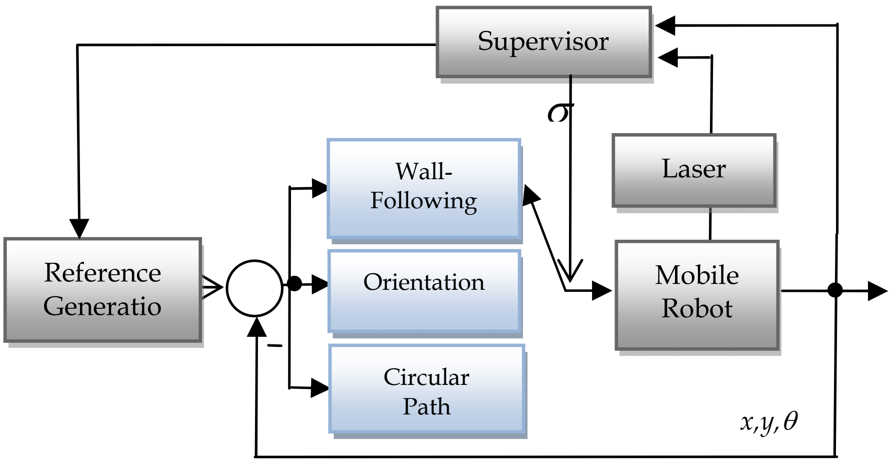


Fig. 11. Block diagram of the Contour-Following controller

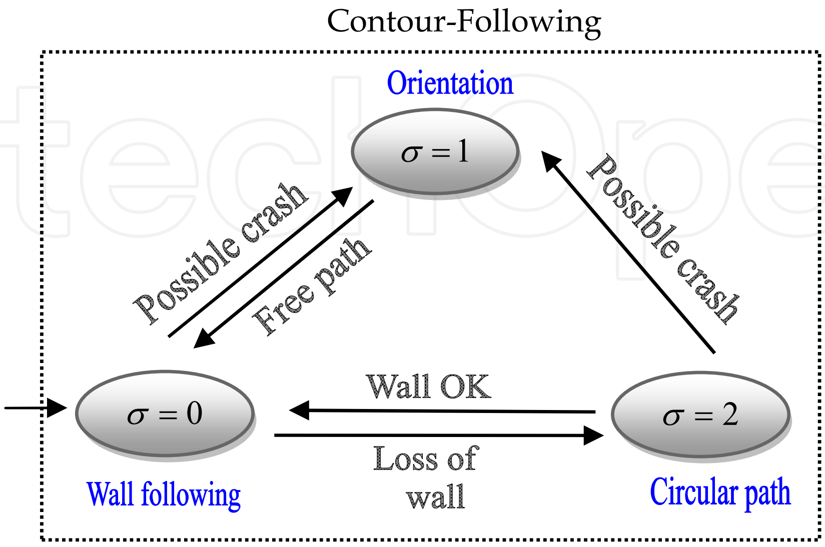


Fig. 12 Supervisor logic

Transitions between these discrete states are ruled by the logic described in Figure 12 (Toibero et al., 2006a). Where, as mentioned before, the “Possible crash” condition is detected as an invasion to the guard-zone shown in Figure 1.c and the “Loss of wall” condition depends on the variance of the length of the laser beams on the side of the robot. It is easy to see that these transitions do not depends on the control states values, but on data provided by the laser range finder. This *a priori* unpredictable data will define the value for the switching signal σ.

5.1 Stability Analysis

Given a family of systems, it is desired that the switched system be stable for every switching signal σ. This is known as stability under arbitrarily switching and has been the subject of several studies (Vu & Liberzon, 2005), (Liberzon, 2003) and (Mancilla-Aguilar, 2000). In fact, common Lyapunov functions are considered in order to prove stability for arbitrary switching. Finding a common Lyapunov function could be a difficult task, but if such function is found, the restrictions over the switching signal disappear, allowing the stable switching between the involved controllers. So, a basic fact that is used in this Section is that the existence of a Common Lyapunov Function with suitable properties guarantees uniform stability, as stated in (Liberzon & Morse, 1999). Let us define a Common Lyapunov Function

Definition I (Liberzon, 2003)

Given a positive definite continuously differentiable function $\mathrm{V}: \mathbb{R}^{\mathrm{n}} \rightarrow \mathbb{R}$, then, it is said that V is a common Lyapunov function for the family of systems $\dot{\chi} = f_p(\chi\,), p \in P$,if there exists a positive definite continuous function $\mathrm{W}: \mathbb{R}^{\mathrm{n}} \rightarrow \mathbb{R}$ such that

$$\frac{\partial v}{\partial \chi}\, f_p(\chi) \leq\, -W(\chi) \qquad \forall \chi, \forall p \in P \tag{22}$$

Where p is some index set. Now, the following theorem can be stated

Theorem I (Liberzon, 2003)

If all systems in the family $\dot{\chi} = f_p(\chi\,), p \in P$ share a radially unbounded common Lyapunov function, then the switched system $\dot{x} = f_\sigma(x)$ is globally uniformely asymptoticaly stable (GUAS).

The main point in this well-known theorem is that the rate of decrease of V along solutions, given by (22), is not affected by switching, hence asymptotic stability is uniform with respect to σ. Now, the purpose is to find a Common Lyapunov Function for the three continuous controllers. Then, it is first necessary to show that the closed-loop systems corresponding to each controller share a common equilibrium point at the origin. From Sections 3, 4.1 and 4.2, the closed-loop equations are

$$\dot{\chi} = f_0(x) = \begin{bmatrix} \dot{\tilde{\theta}} \\ \dot{\tilde{d}} \end{bmatrix} = \begin{bmatrix} -K_{\tilde{\theta}}(\tilde{\theta}) \tanh(k_{\tilde{\theta}}\tilde{\theta}) - k_{\tilde{d}}(\tilde{d})\tilde{d}\; v_{max} \frac{\sin(\tilde{\theta})}{\tilde{\theta}} \\ -v_{max} \sin(\tilde{\theta}) \end{bmatrix} \tag{23}$$

$$\dot{\chi} = f_1(x) = \begin{bmatrix} \dot{\tilde{\theta}} \\ \dot{\tilde{d}} \end{bmatrix} = \begin{bmatrix} -K_{\tilde{\theta}}(\tilde{\theta}) \tanh(k_{\tilde{\theta}}\tilde{\theta}) \\ 0 \end{bmatrix} \tag{24}$$

$$\dot{\chi} = f_2(x) = \begin{bmatrix}\dot{\tilde{\theta}}\\ \dot{\tilde{d}}\end{bmatrix} = \begin{bmatrix}-K_{\tilde{\theta}}(\tilde{\theta})\tanh(k_{\tilde{\theta}}\tilde{\theta})\\ 0\end{bmatrix} \tag{25}$$

It is clear that the origin is a common equilibrium point for the involved controllers. Then, from (6) and (16), a common Lyapunov function for the switching among these controllers is given by (6). Therefore, it can be concluded that the switching control system composed by the three subsystems described in the previous sections is stable for any switching signal σ, because every behavior is at least stable considering the common Lyapunov function (6). However, the proposed application is composed by an asymptotically stable main behavior (wall-following) and two complementary stable behaviors (orientation and circular-path performer). As it was shown along this paper, the complementary behaviors are active only during special situations and the system always returns to the main behavior. Therefore, the GUAS property for the overall switched system can be concluded, provided that the wall following behavior is asymptoticaly stable (AS).

6. Experimental results

The switching contour-follower controller described in this Chapter has been implemented in a Pioneer III mobile robot navigating through a typical office environment at 150millimetres per second. In the following figures it can be seen the trajectory described by the robot when following the interior contour of the Institute of Automatics (INAUT). As mentioned before, the office contour was reconstructed by using the laser sensorial information.

The first experiment (Figure 13) shows a typical situation when following discontinuous contours. The robot follows the outline of a rectangular box at 400millimetres. From this picture it can be appreciated the good performance of the controller when switching between the wall-following and the circular path controller, in this specially discontinuous contour.

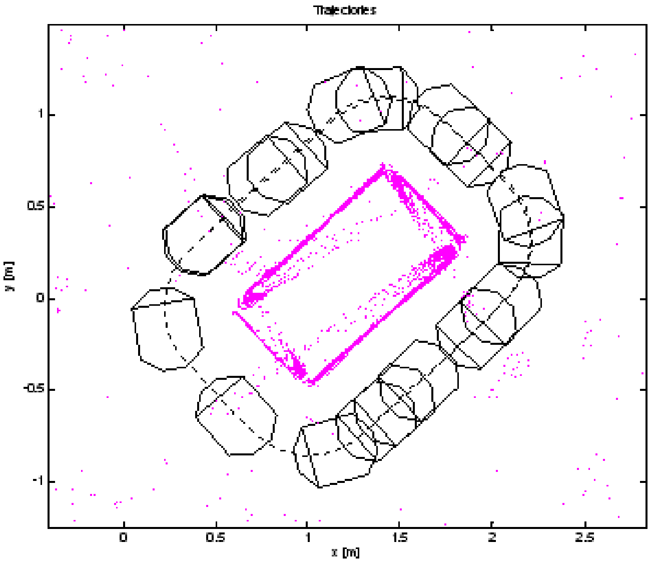


Fig. 13. Robot following a rectangular contour

Figure 14 depicts the performance of the control system when unknown obstacles appear in front of the robot: in this case first avoiding a human being and finally due to a block in the corridor, the robot returns to keep following the corridor the opposite side.

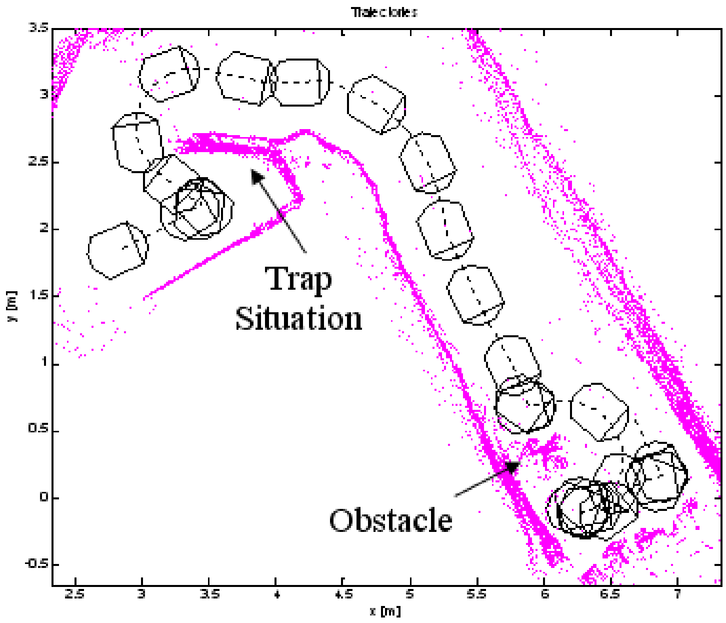


Fig. 14. Robot moving along a corridor avoiding obstacles

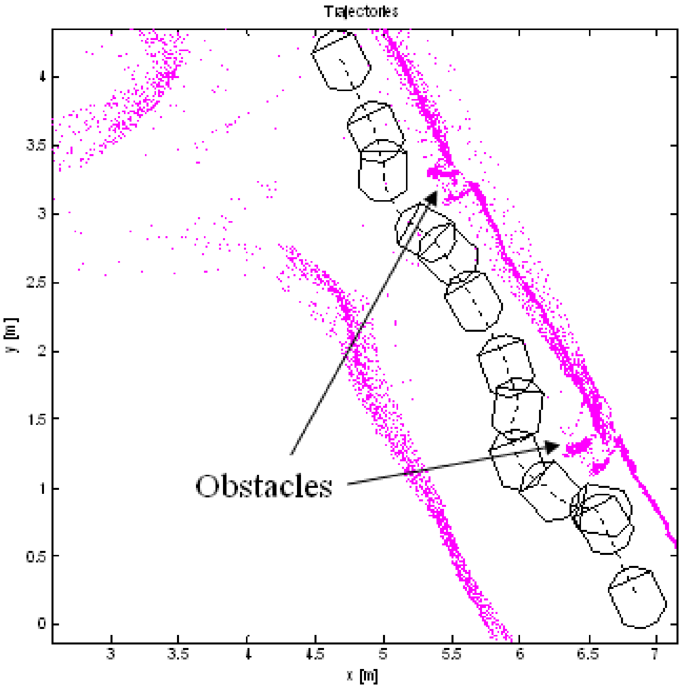


Fig. 15. Robot moving along a corridor avoiding obstacles

The controller constants were set to: v_{max}=150millimeters per second, $k_1 = 0.02$, $K_{\tilde{\theta}} = 0.25$ and $k_{\tilde{\theta}} = 5$. The sample time was of 100milliseconds. The desired distance to the object was selected as 0.38meters, the selection of this value was based on the size of the doors, and larger values for d_{wall} do not allow the robot to pass across them.

7. Mobile robot SLAM algorithm combined with a stable switching controller

Once treated the obstacle avoidance problem, its inclusion to a new system is considered in this Section. In the introductory section, some paragraph dedicated to possible applications to this algorithm mentioned mapping of unknown environments. In fact, simultaneous localization and map building (SLAM) is a challenging problem in mobile robotics that has attracted the interest of an increasing number of researchers in the last decade (Thrun et al., 2005), (Briechle & Hanebeck, 2004). Self-localization of mobile robots is obviously a fundamental issue in autonomous navigation: a mobile robot must be able to estimate its position and orientation (pose) within a map of the environment it is navigating within (Pfister et al., 2003), (Garulli et al., 2005). However, in many applications of practical relevance (like exploration tasks or operations in hostile environments), a map is not available or it is highly uncertain. Therefore, in such cases the robot must use the measurements provided by its sensory equipment to estimate a map of the environment and, at the same time, to localize itself within the map. Several techniques have been proposed to solve the SLAM problem (Thrun et al., 2005), (Durrant-Whyte & Bailey, 2006a), (Durrant-Whyte & Bailey, 2006b).

The SLAM algorithm implemented in this final Section consists on a sequential Extended Kalman Filter (EKF) feature-based SLAM. Prior to the experimental results, it is necessary to devote some paragraphs to its functioning algorithms for completness. This algorithm fuses corners (convex and concave) and lines of the environment in the SLAM system state. Corners are defined in a Cartesian coordinate system whereas lines are defined in the polar system. In addition to the SLAM system state, a secondary map is maintained. This secondary map stores the information on the endpoints of each segment associated with a line of the environment (representing walls). The secondary map and the SLAM system state are updated simultaneously. Once a new feature is added to the SLAM system state, it is also added to the secondary map. The feature extraction method allows the rejection of moving agents of the environment. Equations (26) and (27) show the SLAM system state and its covariance. As it can be seen, the SLAM system state has the robot's pose estimation (χ) and the parameters that define the features of the environment. The covariance matrix has covariance of the robot's pose estimation (P_{vv}), the covariance of the features' parameters (P_{mm}) and the corresponding cross correlations. The elements of the SLAM system state are attached to a global coordinate system determine at the SLAM's first execution (Durrant-Whyte & Bailey, 2006a).

$$\varsigma(k) = \begin{bmatrix} \chi(k) \\ \varsigma_1(k) \\ \vdots \\ \varsigma_n(k) \end{bmatrix}_G \tag{26}$$

$$P(k) = \begin{bmatrix} P_{vv}(k) & P_{vm}(k) \\ P_{vm}^T(k) & P_{mm}(k) \end{bmatrix} \tag{27}$$

The combination of the SLAM algorithm with a strategy for exploration or navigation inside the environment is known as Active SLAM and has been a key poblem in the implementation of autonomous mobile robots. The integration of SLAM algorithms with control strategies to govern the motion of a mobile robot and the ability of selecting feasible destinations on its own will endow the vehicle with full autonomy (Liu et al., 2008), (Liu et al., 2007).

The combination of control strategies with the SLAM algorithm has been addressed from two significantly different points of view. Whereas the first one considers how the control is used to reduce errors during the estimation process (Durrant-Whyte & Bailey, 2006a), (Durrant-Whyte & Bailey, 2006b) the second one concerns exploration techniques providing the best map from the reconstruction perspective (Andrade-Cetto & Sanfeliu, 2006). Despite the duality between regulation and estimation, whatever the control strategy is implemented, it will not be guaranteed that, in general, the mobile robot will follow a specific trajectory inside the environment. In many applications, the control signal is not considered as an input of the SLAM algorithm, and, instead, odometry measurements of the robot are used (Guivant & Nebot, 2001), (Durrant-Whyte & Bailey, 2006a), (Durrant-Whyte & Bailey, 2006b). Thus, most of the associated implementations focus on the low-level, basic control-reactive behavior, leaving the motion planning and control as a secondary algorithm. Thus, albeit restricted to a local reference frame attached to the robot, active exploration strategies for indoor environments are proposed in (Andrade-Cetto & Sanfeliu, 2006), (Liu et al., 2008). As an example, a boundary exploration problem is proposed in (Xi, 2008). In this case, the robot has to reach the best point determined in the boundary of its local point of view. From a global reference perspective, these implementations have a random behavior inside the environment. To solve the lack of global planning, some implementations have included algorithms for searching optimal path based on the information acquired of the environment. These algorithms usually require the map to be gridded and, accordingly, they compute a feasible path to a possible destination (closure of the loop or global boundary points) without specifying the control law implemented on the mobile robot. Despite of the advances made so far, the integration of control strategies based on the SLAM system state (map and vehicle) to guide the robot inside an unknown environment from a local and a global reference frame following a pre-established plan is not quite studied or implemented in real time.

In this work, the robot starts at an unknown position inside an unknown environment and by an active exploration, searches for boundary points from a local reference frame attached to the robot. From now on, these boundary points will be designated by local uncertainty points. Once a local uncertainty point is determined, a trajectory connecting this point and the robot's current pose is generated. The trajectory is continued by one resulting from the execution of a switching adaptive controller (De la Cruz et al., 2008) articulated with an avoiding obstacle algorithm to prevent the collision with moving agents. Once a neighborhood of the local uncertainty point is reached by the robot, the vehicle searches for a new boundary point. This process is repeated until no additional local uncertainty point can be determined. Once this stage is reached - mainly due to the fact that the environment is occupied with already mapped features -, the robot searches for global destination regions. The global destination regions are represented by global uncertainty points. These points are determined by using the Gaussian distribution of each random variable involved in the EKF-SLAM system state. Thus, according to the geometrical information given by the secondary map of the environment, the entire map is circumscribed by four virtual segments that determine the limits of the known environment. Then, a set of points contained inside the limits of the virtual features is generated by a Monte Carlo experiment, and those that are not navigable are rejected. The probability attached to each one of the remaining point is estimated based on the concept of sum of Gaussians, which yields an estimate of the occupational probability of each point. Those points whose probability values do not fall near zero (free space point) or near one (non-empty point) will be considered as a global uncertainty point. This kind of points can be attached to the boundary of the map - unexplored region - or to badly mapped features. Once a global uncertainty point is determined -according to the needs of the navigation purposes - a hybrid contour-following control (Toibero et al., 2007) is implemented to drive the robot to that point. Once a neighborhood of the global uncertainty point is reached, the exploration switches to the mode of searching for local uncertainty points in a local reference frame. The entire navigation system stops when no global uncertainty points are found, what will happen when the map is complete. During the entire navigation or exploration phase, the SLAM algorithm continues been executed Additional information about this topic can be found in (Auat Cheein, 2009). Figure 16 shows the general architecture of the SLAM algorithm with the non-reactive behavior controllers (the adaptive switching controller and the hybrid contour following) implemented on the mobile robot.

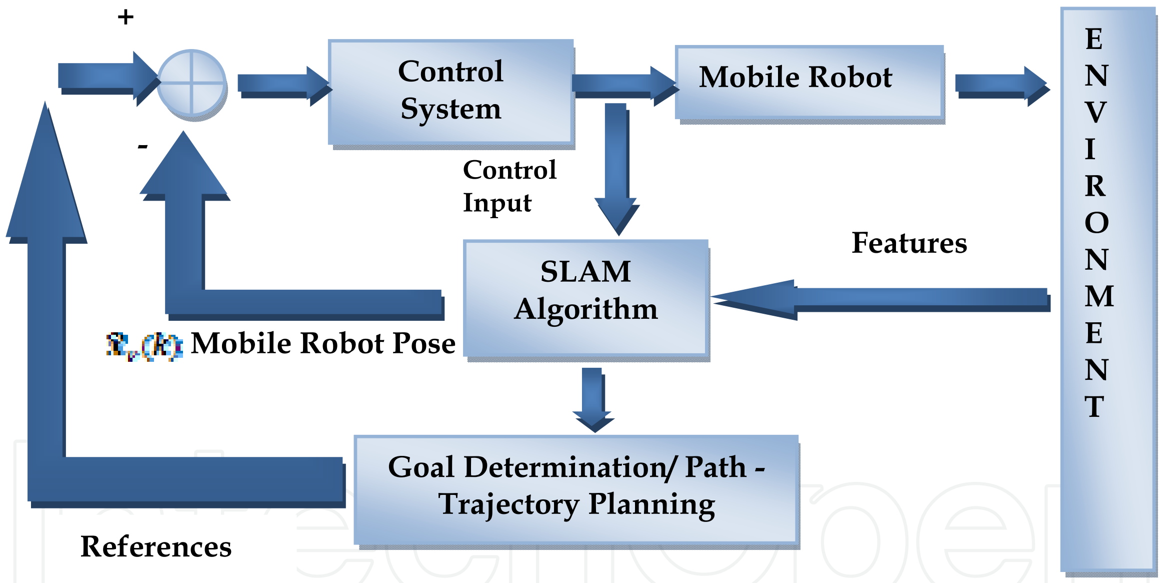


Fig. 16. General SLAM - Control system architecture where the control uses SLAM system state information for planning purposes

Figure 17 shows the real time experimentation of the SLAM algorithm combined with the two control strategies. The experimentation was carried out at the facilities of the Instituto de Automatica. As it can be seen, the map obtained by the SLAM was built consistently. The local controller (adaptive switching control) and the global controller (hybrid contour following) have allowed the entire navigation of the mobile robot within the Instituto de Automatica.

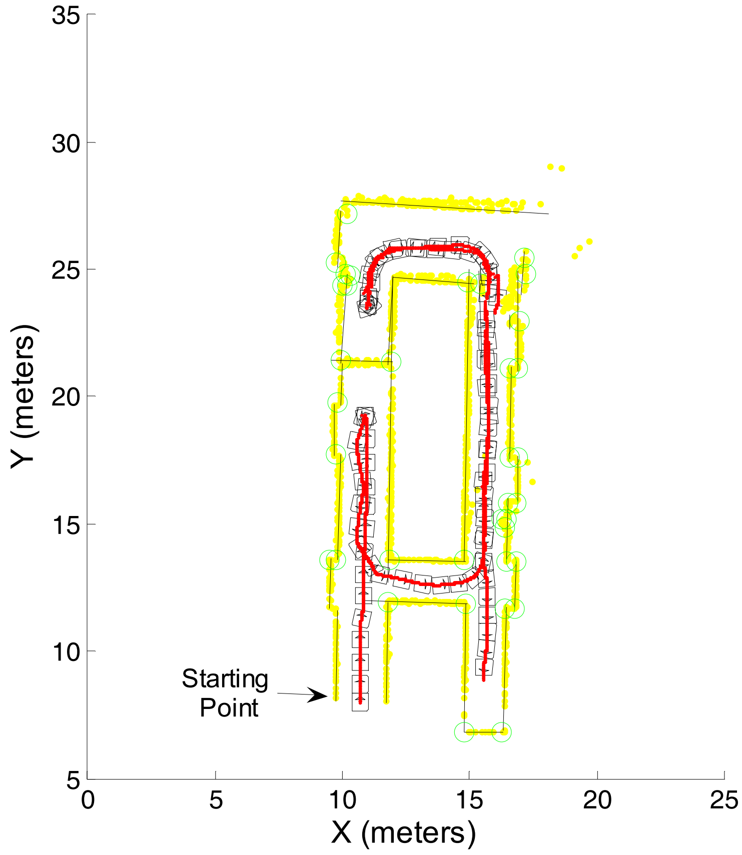


Fig. 17. Map reconstruction of the Institute of Automatics

In Figure 17, the green circles are the corners detected in the environment, the solid dark lines are the segments associated with walls whereas red crosses are the beginning and ending points of such segments; the red points are the path travelled by the mobile robot and the yellow points are raw laser data.

8. Conclusions

In this Chapter it has been presented a switched countour-following controller, which allows a wheeled mobile robot to follow discontinuous walls' contours. This controller has been developed by considering a standard (stable) wall-following controller and aggregating two complementary (also stable) controllers. One risponsible for avoid collisions between the robot and the object which is being followed, and the other responsible for find a lost contour of this same object. This new swiching controller proved to be stable with respect to its swiching signal, guaranteing that the robot will be able to stay at a desired robot-to-object distance, and with the same object orientation.

Next, this controllers was included into a SLAM algorithm, in order to deal with the exploration and map construction of unknown environments, exposing the modularity capability of this kind of control architecture.

In designing the control system, the asymptotic stability of the individual controllers as well the asymptotic stability at the switching times (for the switching controller) were considered and proved.
Several experimental results in a Pioneer III mobile robot with odometry and laser radar sensor have been included; showing the good performance of the proposed control strategy in a laboratory setting, in the first attemps, and later, in a large scale setting for the SLAM experiment.

Acknowledgments

Authors thank to the National Council of Scientific and Technical Research of Argentina (CONICET) for partially supporting this research.

9. References

Ando, Y. & S. Yuta. (1996). A reactive wall-following algorithm and its behaviors of an autonomous mobile robot with sonar ring. Journal of Robotics and Mechatronics, Vol. 8(1), pp.33-39.

Andrade-Cetto, J. & Sanfeliu, A. (2006). Environment Learning for Indoors Mobile Robots, ISBN: 978-3-540-32795-0, Series: Springer Tracts in Advanced Robotics , Vol. 23, Springer

Arkin, R.C. (1998). Behavior-based Robotics, MIT Press: Cambridge, M A.

Auat Cheein, F. A. (2009). Simultaneous Localization and Mapping of a Mobile Robot based on Uncertainty Zones Navigation, PhD Thesis, ISBN: 978-987-05-6727-1, INAUT: National University of San Juan

Bailey, T.; Nieto, J.; Guivant, J.; Stevens, M. & Nebot, E. (2006). Consistency of the EKF-SLAM algorithm, in IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), pp. 3562–3568, ISBN: 1-4244-0259-X, Beijing, China

Bicho, E. (2000). Detecting, representing and following walls based on low-level distance sensors, Proc. of the 2nd Int. Symposium on Neural Computation.

Boada, M.J.L.; Rodriguez, F.J.; Barber, R. & Salichs, M.A. (2003). A control system based on reactive skills for autonomous mobile robots, IEEE Int. Conf. on Advanced Robotics.

Borenstein, J. & Koren, Y. (1989). Real-time obstacle avoidance for fast mobile robots, IEEE Trans. on Systems, Man and Cybernetics. Vol. 19(5), pp.1179-1187

Braunstingl, R. and Ezkerra, J.M. (1995). Fuzzy logic wall following of a mobile robot based on the concept of general perception, Int. Conf. on Advanced Robotics, pp. 367-376

Brawn, R. & Hwang, P. (1997). Introduction to Random Signals and Applied Kalman Filtering 3rd ed., John Wiley & Sons: New York, pp. 371-375

Briechle, K. & Hanebeck, U. D. (2004). Localization of mobile robot using relative bearing measurements, IEEE Transactions on Robotics and Automation, vol. 20, no. 1, pp. 36–44, ISSN: 1042-296X

De la Cruz, C.; Carelli, R. & Bastos, T. F. (2008). Switching Adaptive Control of Mobile Robots, in: International Symposium on Industrial Electronics, pp. 835-840, ISBN: 978-1-4244-1665-3, June 30 2008-July 2, Cambridge, UK

Durrant-Whyte, H. & Bailey, T. (2006a). Simultaneous localization and mapping (SLAM): part I essential algorithms. IEEE Robotics and Automation Magazine, vol. 13, pp. 99-108, ISSN: 1070-9932

Durrant-Whyte, H. & Bailey, T. (2006b). Simultaneous localization and mapping (SLAM): part II state of the art, IEEE Robotics and Automation Magazine. Vol.13, No. 3, pp. 108-117, Sept., ISSN: 1070-9932

Edlinger, T. & von Puttkamer, E. (1994). Exploration of an indoor-environment by an autonomous mobile robot, IEEE/RSJ/GI Int. Conf. on Intelligent Robots and Systems 2, pp.1278-1284

Fazli, S. & Kleeman, L. (2005). Wall following and obstacle avoidance results from a multi-dsp sonar ring on a mobile robot, In Proc. of the IEEE Int. Conf. on Mechatronics & Automation, 5, pp.432-437

Garulli, A.; Giannitrapani, A.; Rosi, A. & Vicino, A. (2005). Mobile robot SLAM for line-based environment representation, Proceedings of the 44th IEEE Conference on Decision and Control, pp. 2041- 2046, ISBN: 0-7803-9567-0, December 2005

Guivant, J.E. & Nebot, E.M. (2001). Optimization of the simultaneous localization and map-building algorithm for real-time implementation, IEEE Transactions on Robotics and Automation, vol. 17, pp. 242–257, ISSN: 1042-296X

Kelly, R. & Carelli, R. (1996). A class of nonlinear PD-type for robot manipulator, Journal of Robotic Systems, 13(12), pp.793-802

Liberzon, D. & Morse, A.S. (1999). Basic problems in stability and design of switched systems, IEEE Control Systems Magazine, 19(5):pp.59-70

Liberzon, D. (2003). Switching in Systems and Control, Birkhauser

Liu, Y. & Sun, F. (2007). A solution to active simultaneous localization and mapping problem based on optimal control, in: Proceedings of IEEE International Conference on Mechatronics and Automation, pp. 314-319, ISBN: 978-1-4244-0828-3, 5-8 August, Harbin, China, 2007

Liu, Y; Dong, J. & Sun, F. (2008). An Efficient Navigation Strategy for Mobile Robots with Uncertainty Estimation, in: Proc. of the World Congress on Intelligent Control and Automation, pp. 5174-5179, ISBN: 978-1-4244-2113-8, 25-27 June, Chongquing, China

Mancilla-Aguilar, J. L. (2000). A condition for the stability of Switched Nonlinear Systems, IEEE Trans. on Automatic Control, 45, pp.2077-2079

Pfister, A. T.; Roumeliotis, S. I. & Burdick, J. W. (2003). Weighted line fitting algorithms for mobile robot map building and efficient data representation, in Proceedings of the 2003 IEEE International Conference on Robotics and Automation, pp. 1304–1311, ISBN: 0780377362, Taiwan, September 2003

Thrun, S.; Burgard, W. & Fox, D. (2005). Probabilistic Robotics, ISBN: 978-0-262-20162-9, MIT Press, Cambridge.

Toibero, J.M.; Carelli, R. & Kuchen, B. (2006a). Stable Switching Contour-Following Controller for Wheeled Mobile Robots, IEEE Int. Conf. on Robotics and Automation

Toibero, J.M.; Carelli, R. & Kuchen, B. (2006b). Wall-following stable control for wheeled mobile robots, Proc. of the 8th Int. IFAC Symposium on Robot Control

Toibero, J.M.; Carelli, R. & Kuchen, B. (2007). Switching control of mobile robots for autonomous navigation in unknown environments, in: IEEE International Conference on Robotics and Automation, pp. 1974-1979, ISBN: 1-4244-0601-3, 10-14 April, Rome, Italy

Toibero, J.M.; Roberti, F. & Carelli, R. (2009). Stable Switching Contour-Following Control of Wheeled Mobile Robots. ROBOTICA, 27, 1-12

van Turennout, P. & Hounderd, G. (1992). Following a wall with a mobile robot using ultrasonic sensors, In Proc. of the 1992 IEEE/RSJ Int. Conf. on Intelligent Robots and Systems 2, pp.1451-1456

Vidyasagar, M. (1993). Nonlinear Systems Analysis 2nd edition, Prentice Hall, New Jersey

Vu, L. & Liberzon, D. (2005). Common Lyapunov Functions for families of commuting nonlinear systems, Systems & control letters, 54: pp.405-416

Wang, M. & Liu, J. (2004). Autonomous robot navigation using fuzzy logic controller. Proc. of the Int. Conf. on Machine Learning and Cybernetics, pp.26-29, 2004

Xi, B.; Guo, R.; Sun, F. & Huang, Y. (2008). Simulation Research for Active Simultaneous Localization and Mapping Based on Extended Kalman Filter, in: Proc. of the IEEE International Conference on Automation and Logistics, pp. 2443-2448, ISBN: 978-1-4244-2502-0, 1-3 Sept, Quingdao, China.

Zhang, Z.; Sarkar, N. & Yun, X. (2004). Supervisory control of a mobile robot for agile motion coordination, IEEE Int. Conf. on Robotics and Automation, Vol.3, pp.2196-2203

Mobile Robots Navigation

Edited by Alejandra Barrera

ISBN 978-953-307-076-6
Hard cover, 666 pages
Publisher InTech
Published online 01, March, 2010
Published in print edition March, 2010

Mobile robots navigation includes different interrelated activities: (i) perception, as obtaining and interpreting sensory information; (ii) exploration, as the strategy that guides the robot to select the next direction to go; (iii) mapping, involving the construction of a spatial representation by using the sensory information perceived; (iv) localization, as the strategy to estimate the robot position within the spatial map; (v) path planning, as the strategy to find a path towards a goal location being optimal or not; and (vi) path execution, where motor actions are determined and adapted to environmental changes. The book addresses those activities by integrating results from the research work of several authors all over the world. Research cases are documented in 32 chapters organized within 7 categories next described.

Juan Marcos Toibero, Flavio Roberti, Fernando Auat Cheein, Carlos Soria and Ricardo Carelli (2010). Stable Switching Control of Wheeled Mobile Robots, Mobile Robots Navigation, Alejandra Barrera (Ed.), ISBN: 978-953-307-076-6, InTech, Available from: http://www.intechopen.com/books/mobile-robots-navigation/stable-switching-control-of-wheeled-mobile-robots

INTECH
open science | open minds

InTech Europe
University Campus STeP Ri
Slavka Krautzeka 83/A
51000 Rijeka, Croatia
Phone: +385 (51) 770 447
Fax: +385 (51) 686 166
www.intechopen.com

InTech China
Unit 405, Office Block, Hotel Equatorial Shanghai
No.65, Yan An Road (West), Shanghai, 200040, China
中国上海市延安西路65号上海国际贵都大饭店办公楼405单元
Phone: +86-21-62489820
Fax: +86-21-62489821