

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

6,900

Open access books available

185,000

International authors and editors

200M

Downloads

Our authors are among the

154

Countries delivered to

TOP 1%

most cited scientists

12.2%

Contributors from top 500 universities



WEB OF SCIENCE™

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

Interested in publishing with us?  
Contact [book.department@intechopen.com](mailto:book.department@intechopen.com)

Numbers displayed above are based on latest data collected.  
For more information visit [www.intechopen.com](http://www.intechopen.com)



## Predictive Control with Local Visual Data

Lluís Pacheco, Ningsu Luo and Xavier Cufí  
*University of Girona  
 Spain*

### 1. Introduction

Last half century depicted an enormous progress in robotics. However, this field is still in its infancy as compared with the human behaviour. Therefore, the ability of a robot is limited in its movement, understanding complex sensorial inputs or performing higher level reasoning. Important clues to enable further progress to be made are the integration of several different fields of knowledge, such as computing, communications and control sciences, with the aim of enabling robots to use higher level reasoning and decision tools with a strong theory base (Murray et al., 2003). The research work presented here integrates control science and robot vision knowledge in a computer science environment. Nowadays, the local visual perception research, applied to autonomous mobile robots, has succeeded in some important objectives, such as feasible obstacle detection and structure knowledge (Campbell, et al., 2004). In this context, the allowed navigation control signals should achieve the obstacle avoidance as well as the final desired coordinates. Scientific community has developed several studies in this field. Based on the dynamic window approach with available robot speeds, the reactive avoidance collisions, safety stop and goal can be achieved using the dynamic constraints of WMR (Fox, et al., 1997). In other works, it is presented the methodologies for the exact motion planning and control, based on the artificial potential fields where the complete information about the free space and goal are encoded (Rimon and Koditschek, 1992). Some approaches on mobile robots propose the use of potential fields, which satisfy the stability in a Lyapunov sense, in a short prediction horizon (Ögren & Leonard, 2005).

The main contribution of this paper is the use of the visual information as a dynamic window where the collision avoidance and safety stop can be planned. Thus, local visual data and local attraction of artificial potential fields are used in order to achieve the Lyapunov stability. This work relates the on-robot visual perception and odometer system information with the nonlinear mobile robot control system, consisting in a differential driven robot with a free rotating wheel. It is developed an easily portable methodology to plan the goal achievement by using the visual data as an available source of positions. The dynamic interactions of the robotic system arise from the knowledge of a set of experimental robot models that allow the development of model predictive control strategies based on the mobile robot platform PRIM available in the Laboratory of Robotics and Computer Vision. Moreover, the local visual information is used as an occupancy grid that allows planning feasible trajectories towards the desired objective. The knowledge of the objective allows the optimal solution of the local desired coordinates based on the acquired images.

Source: Robotics, Automation and Control, Book edited by: Pavla Pecherková, Miroslav Flídr and Jindřich Duník,  
 ISBN 978-953-7619-18-3, pp. 494, October 2008, I-Tech, Vienna, Austria

This chapter is organized as follows: Section 1 gives a brief presentation about the aim of the present work. In the Section 2, the platform is introduced as an electromechanical system. This section also describes the experiments to be realized in order to find the parametric models of the robot suitable for designing and implementing MPC methods. Section 3 presents the MPC strategies used for achieving the path following of the reference trajectories. In the Section 4, the use of visual data is presented as a horizon where optimal trajectories can be planned by using MPC strategies. In the Section 5, the experimental results using MPC strategies are depicted. Finally, in Section 6 some conclusions are made with special attention paid into the future research works.

## 2. The Robot and the control system identification

### 2.1 Introduction

This section gives some description on the main robot electromechanical and sensorial systems of the platform tested in this work. Hence, the WMR PRIM, available in our lab, has been used in order to test and orient the research (Pacheco et al., 2008). The modelling methodology is depicted as an experimental identification. Moreover, the simplified control model and the odometer system are also introduced.

### 2.2 Electromechanical and sensorial system of the robot

Fig. 1.a shows the robot PRIM used in the research work. The mechanical structure of the robot is made of aluminum, with two independent wheels of 16cm diameters actuated by two DC motors. The distance between two wheels is 56.4cm. A third spherical omni-directional wheel is used to guarantee the system stability. The maximum continuous torque of each motor is 131mNm. The proportion of gear reduction for each motor is 86:1 and thus the total force actuating on the robot is near 141N. Shaft encoders with 500 counts/rev are placed at the motor axes, which provide 43000 counts for each turn of the wheel.

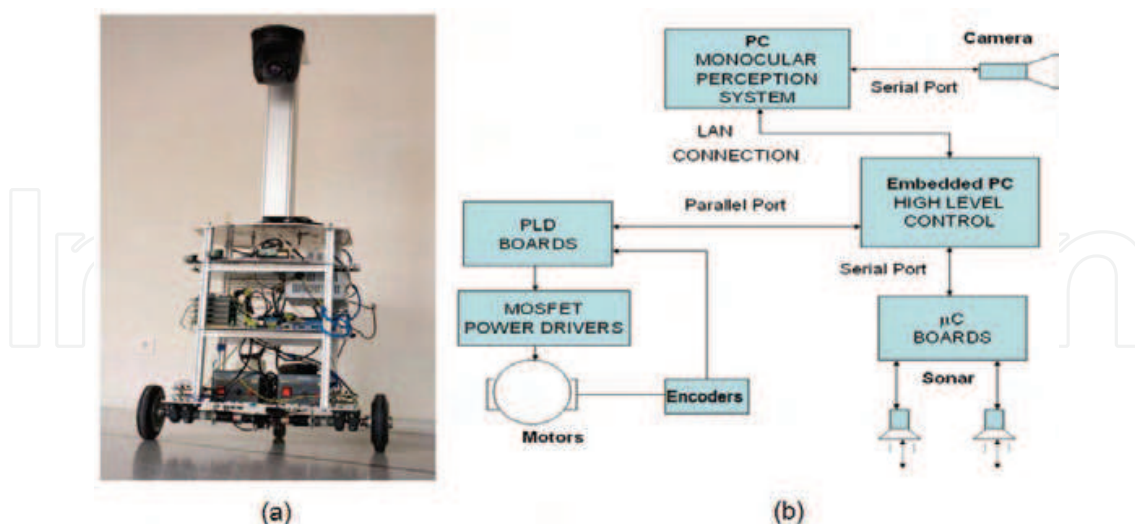


Fig. 1. (a) The robot PRIM used in this work; (b) The sensorial and electronic system blocs

A set of PLD (programmable logic device) boards is connected to the digital outputs of the shaft encoders. The printed circuits boards (PCB) are used to measure the speed of each motor at every 25ms. An absolute counter provides the counts in order to measure the robot position and speed by the odometer system. Another objective of these boards is to generate

a signal of 23kHz PWM for each motor. The communication between the central digital computer and the boards is made through the parallel port. The speed is commanded by a byte and thus it can generate from 0 to 127 advancing or reversing speed commands. The maximal speed is near 0.5m/s. A set of microcontroller boards (MCS-51) is used to read the information available from different connected sensors. The rate of communication with these boards is 9600 b/s. Fig. 1.b shows the electronic and sensorial system blocks. The data gathering and the control by digital computer is set to 100ms. The system flexibility is increased with the possibility of connecting with other computer systems through a local LAN. In this research, it is connected to a machine vision system that controls a colour camera EVI-D70P-PAL through the VISCA RS232-C control protocol. For instance, the camera configuration used in this work is of a horizontal field of view of 48° and a vertical field of 37°. The focus, pan and tilt remain fixed under present configuration. The camera pose is set to 109cm from the floor with a tilt angle of 32°. The local desired coordinates, obtained by the machine vision system, can be transmitted to the control unit connecting the USB port to the LAN.

### 2.3 Experimental model and system identification

The parametric identification process is based on black box models (Norton, 1986; Ljung, 1989). Thus, the transfer functions are related to a set of polynomials that allow the use of analytic methods in order to deal with the problem of controller design. The nonholonomic system dealt with in this work is considered initially as a MIMO (multiple input multiple output) system, which is composed of a set of SISO subsystems with coupled dynamic influence between two DC motors. The approach of multiple transfer functions consists in making the experiments with different speeds. In order to find a reduced-order model, several studies and experiments have been done through the system identification and model simplification. The parameter estimation is done by using a PRBS (Pseudo Random Binary Signal) as excitation input signal. It guarantees the correct excitation of all dynamic sensible modes of the system along the spectral range and thus results in an accurate precision of parameter estimation. The experiments to be realized consist in exciting two DC motors in different (low, medium and high) ranges of speed.

The ARX (auto-regressive with external input) structure has been used to identify the parameters of the robot system. The problem consists in finding a model that minimizes the error between the real and estimated data. By expressing the ARX equation as a lineal regression, the estimated output can be written as:

$$\hat{y} = \theta \varphi \quad (1)$$

with  $\hat{y}$  being the estimated output vector,  $\theta$  the vector of estimated parameters and  $\varphi$  the vector of measured input and output variables. By using the coupled system structure, the transfer function of the robot can be expressed as follows:

$$\begin{pmatrix} Y_R \\ Y_L \end{pmatrix} = \begin{pmatrix} G_{RR} & G_{LR} \\ G_{RL} & G_{LL} \end{pmatrix} \begin{pmatrix} U_R \\ U_L \end{pmatrix} \quad (2)$$

where  $Y_R$  and  $Y_L$  represent the speeds of right and left wheels, and  $U_R$  and  $U_L$  the corresponding speed commands, respectively. In order to know the dynamics of robot system, the matrix of transfer function should be identified. Fig. 2 shows the speed response

of the left wheel corresponding to a left PBRS input signal. The treatment of experimental data is done before the parameter estimation. In concrete, it includes the data filtering, using the average value of five different experiments with the same input signal, the frequency filtering and the tendency suppression. The system is identified by using the identification toolbox “ident” of Matlab for second order models. The following continuous transfer function matrix for medium speed is obtained:

$$\begin{pmatrix} Y_L \\ Y_R \end{pmatrix} = \begin{pmatrix} \frac{0.35s^2 + 4.82s + 4.46}{s^2 + 5.84s + 4.89} & \frac{0.02s^2 + 0.27s + 0.32}{s^2 + 5.84s + 4.89} \\ \frac{0.26s^2 + 3.41s + 0.28}{s^2 + 5.84s + 4.89} & \frac{0.11s^2 + 1.72s + 5.12}{s^2 + 5.84s + 4.89} \end{pmatrix} \begin{pmatrix} U_L \\ U_R \end{pmatrix} \tag{3}$$

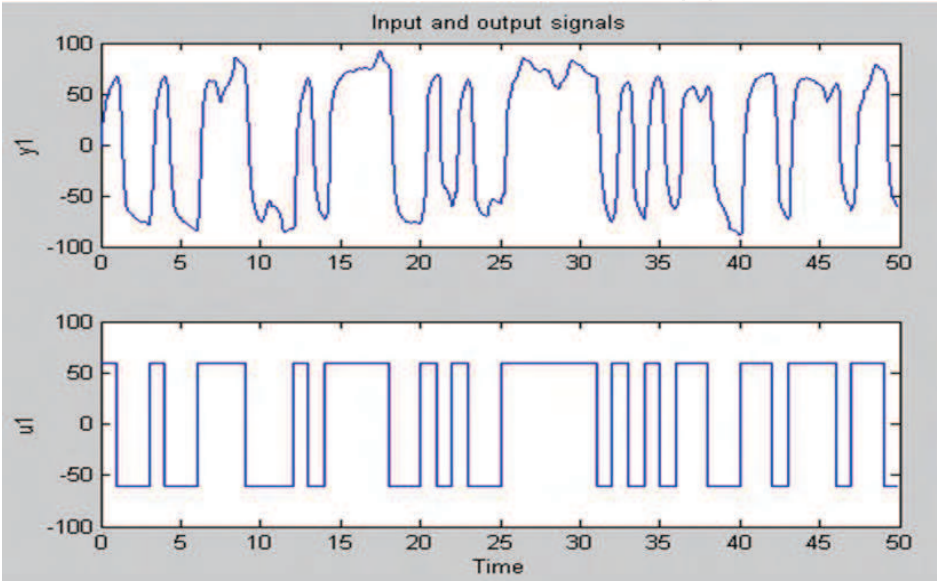


Fig. 2. Left speed output for a left PRBS input signal

It is shown by simulation results that the obtained model fits well to the experimental data.

2.4 Simplified control model and odometer system

This section studies the coupling effects and the way for obtaining a reduced-order dynamic model. It is seen from (3) that the dynamics of two DC motors are different and the steady gains of coupling terms are relatively small (less than 20% of the gains of main diagonal terms). Thus, it is reasonable to neglect the coupling dynamics so as to obtain a simplified model. In order to verify it from real results, a set of experiments have been done by sending non-zero speed commands to one motor while commanding zero speed to the other motor. In Fig. 3, it is shown a response obtained on the left wheel, when a medium speed command is sent to the right wheel. The experimental result confirms the above facts. The existence of different gains in steady state is also verified experimentally. Finally, the order reduction of system model is carried out trough the analysis of pole positions by using the method of root locus. Afterwards, the system models are validated through the experimental data by using the PBRS input signal. A two dimensional array with three different models for each wheel is obtained. Hence, each model has an interval of validity where the transfer function is considered as linear, and classic PID speed control can be developed.



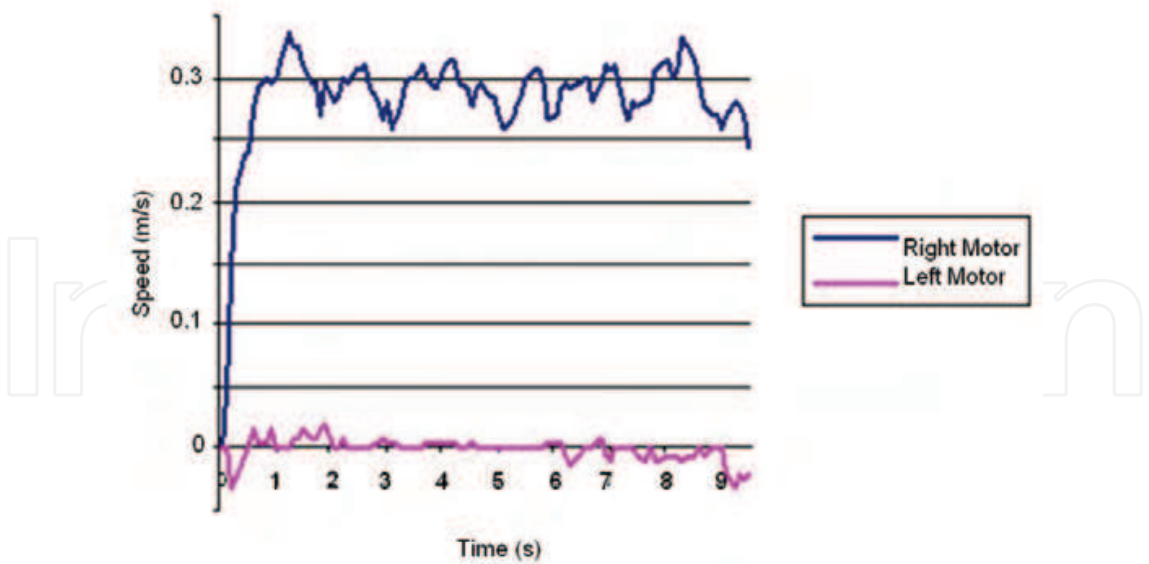


Fig. 3. Coupling effects at the left wheel

The robot speed and position are provided by the odometer system; hence,  $(x, y, \theta)$  denote the coordinates of position and orientation, respectively. The Fig. 4 describes the positioning of robot as a function of the radius of left and right wheels ( $R_e, R_d$ ), and the angular incremental positioning ( $\theta_e, \theta_d$ ), with  $E$  being the distance between two wheels and  $dS$  the incremental displacement of the robot. The position and angular incremental displacements are expressed as:

$$dS = \frac{R_d d\theta_d + R_e d\theta_e}{2} \qquad d\theta = \frac{R_d d\theta_d - R_e d\theta_e}{E}$$

(4)

The coordinates  $(x, y, \theta)$  can be expressed as:

$$\begin{aligned} x_n &= x_{n-1} + dS \cos(\theta_{n-1} + d\theta) \\ y_n &= y_{n-1} + dS \sin(\theta_{n-1} + d\theta) \\ \theta_n &= \theta_{n-1} + d\theta \end{aligned}$$

(5)

Thus, the incremental position of the robot can be obtained through the odometer system with the available encoder information obtained from (4) and (5).

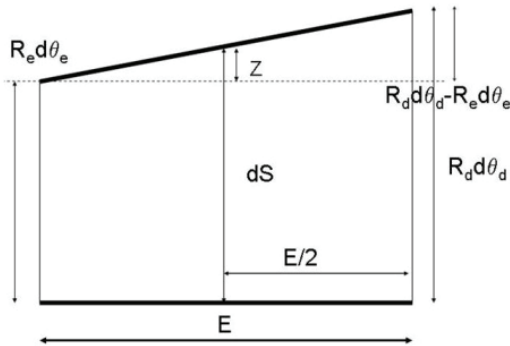


Fig. 4. Positioning of the robot as functions of the angular movement of each wheel

### 3. Model Predictive Control

#### 3.1 Introduction

The MPC (model predictive control) MPC has many interesting aspects for its application to mobile robot control. It is the most effective advanced control technique, as compared to the standard PID control, that has made a significant impact to the industrial process control (Maciejowski, 2002). Recently, real time mobile robot MPC implementations have been developed using global vision sensing (Gupta et al., 2005). In (Küne et al., 2005), it was studied the MPC based optimal control useful for the case when nonlinear mobile robots are used under several constraints, as well as the real time implementation possibilities when short prediction horizons are used. In general, the global trajectory planning becomes unfeasible since the sensorial system of some robots is just local. By using a MPC, the idea of the receding horizon can deal with the local sensor information. In this way, it is proposed a LMPC (local model predictive control) in order to use the available visual data in the navigation strategies for the goal achievement. The MPC is based on minimizing a cost function, related to the objectives, through the selection of the optimal inputs. In this case, the cost function can be expressed as follows:

$$J(n, m) = \min_{\left\{ U(k+i|k) \right\}_{i=0}^{j=m-1}} \left\{ \begin{aligned} & \left[ X(k+n|k) - X_d \right]^T P \left[ X(k+n|k) - X_d \right] \\ & + \sum_{i=1}^{n-1} \left[ X(k+i|k) - X_d \right]^T Q \left[ X(k+i|k) - X_d \right] \\ & + \sum_{i=0}^{m-1} U^T(k+i|k) R U(k+i|k) \end{aligned} \right\} \quad (6)$$

where  $X_d = (x_d, y_d, \theta_d)$  denotes the desired trajectory coordinates. The first term of (6) is referred to the final desired coordinate achievement, the second term to the trajectory to be followed, and the last one to the input signals minimization.  $P$ ,  $Q$  and  $R$  are weighting parameters.  $X(k+n|k)$  represents the terminal value of the predicted output after the horizon of prediction  $n$  and  $X(k+i|k)$  represents the predicted output values within the prediction horizon. The system constraints are also considered:

$$\left\{ \begin{aligned} & \left| U(k+i|k) \right| \leq G_1 \quad \alpha \in [0, 1) \\ & \left| [x_{k+i}, y_{k+i}] - [x_o, y_o] \right| \geq G_2 \\ & \left| [x_{k+n}, y_{k+n}] - [x_d, y_d] \right| \leq \alpha \left| [x_k, y_k] - [x_d, y_d] \right| \end{aligned} \right\} \quad (7)$$

The limitation of the input signal is taken into account in the first constraint. The second constraint is related to the obstacle points where the robot should avoid the collision. The last one is just a convergence criterion.

#### 3.2 LMPC Algorithms

This section gives the LMPC algorithms by using the basic ideas presented in the MPC introduction. The LMPC algorithm is run in the following steps:

1. To read the actual robot position;
2. To minimize the cost function and to obtain a series of optimal input signals;

3. To choose the first component of the obtained input signals as the command signal;
4. To go back to the step 1 in the next sampling period.

The minimization of the cost function is a nonlinear problem in which the following equation should be verified:

$$f(\alpha x + \beta y) \leq \alpha f(x) + \beta f(y) \quad (8)$$

It is a convex optimization problem caused by the trigonometric functions used in (5), (Boyd & Vandenberghe, 2004). The use of interior point methods can solve the above problem (Nesterov et al., 1994). Among many algorithms that can solve the optimization, the descent methods are used, such as the gradient descent method among others, (Ortega, et al. 2000). The gradient descent algorithm has been implemented in this work. In order to obtain the optimal solution, some constraints over the inputs are taken into account:

- The signal increment is kept fixed within the control horizon
- The input signals remain constant during the remaining interval of time

The input constraints present advantages such like the reduction in the computation time and the smooth behavior of the robot during the prediction horizon. Thus, the set of available input is reduced to one value. In order to reduce the optimal signal value search, the possible input sets are considered as a bidimensional array, as shown in Fig. 5. Then, the array is decomposed into four zones, and the search is just located to analyze the center points of each zone. It is considered the region that offers better optimization, where the algorithm is repeated for each sub-zone, until no sub-interval can be found. Once the algorithm is proposed, several simulations have been carried out in order to verify the effectiveness, and then to make the improvements. Thus, when only the desired coordinates are considered, the robot could not arrive in the final point. Fig. 6.a and Fig. 6.b show the simulated results corresponding to the use of two different horizons (one with a prediction horizon  $n=5$  and a control horizon  $m=3$ , and the other with a prediction horizon  $n=10$  and a control horizon  $m=5$ ), where it is depicted that the inputs can minimize the cost function by shifting the robot position to the left.

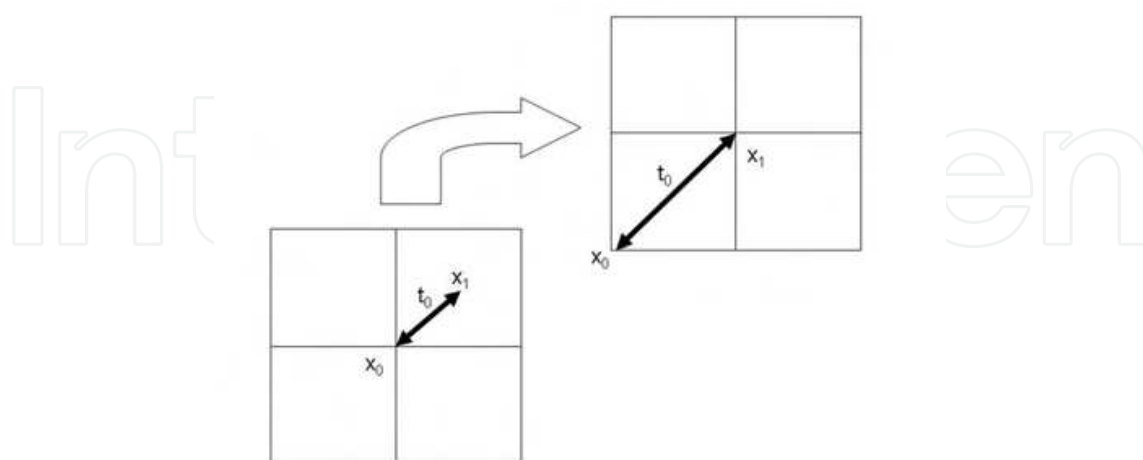


Fig. 5. Optimal interval search.

The reason can be found in (3), where the right motor has more gain than the left. This problem can be easily solved by considering a straight-line trajectory from the actual point



of the robot to the final desired point. Thus, the trajectory should be included into the LMPC cost function. The possible coordinates available for prediction, as shown in Fig. 6.a, depict a denser horizon due to the shorter prediction horizon. Therefore, prediction horizons between 0.5s and 1s were proposed and the computation time for each LMPC step was set to less than 100ms, running in an embedded PC of 700MHz. Trajectory tracking and final goal achievement are other interesting aspects to be analyzed. Fig. 7.a shows the simulated results obtained in tracking a straight line of 2m using two different prediction horizons. Fig. 7.b shows the velocities of both wheels with the above strategies. The wide prediction strategy shows a softer behaviour due to the larger control horizon.

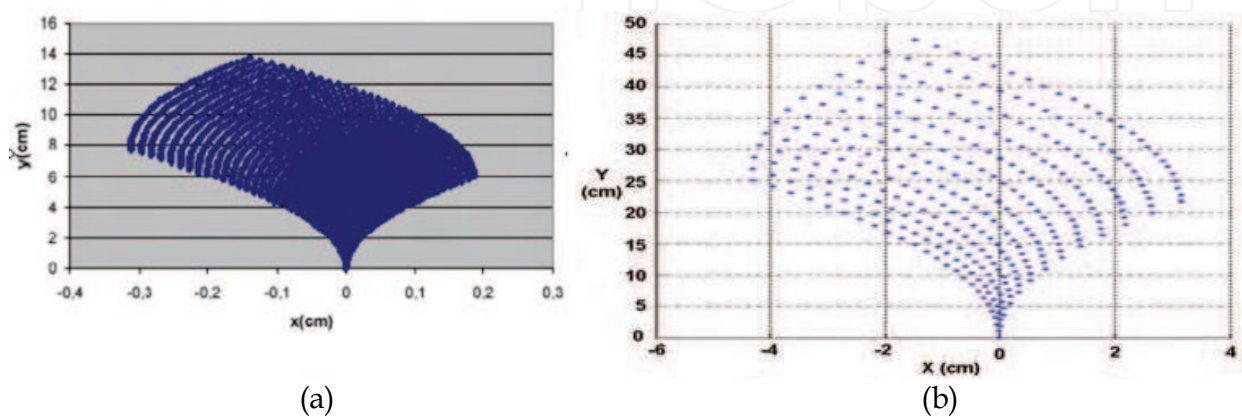


Fig. 6. Predicted coordinates from speed zero (a)  $n=5$   $m=3$  (b)  $n=10$   $m=5$

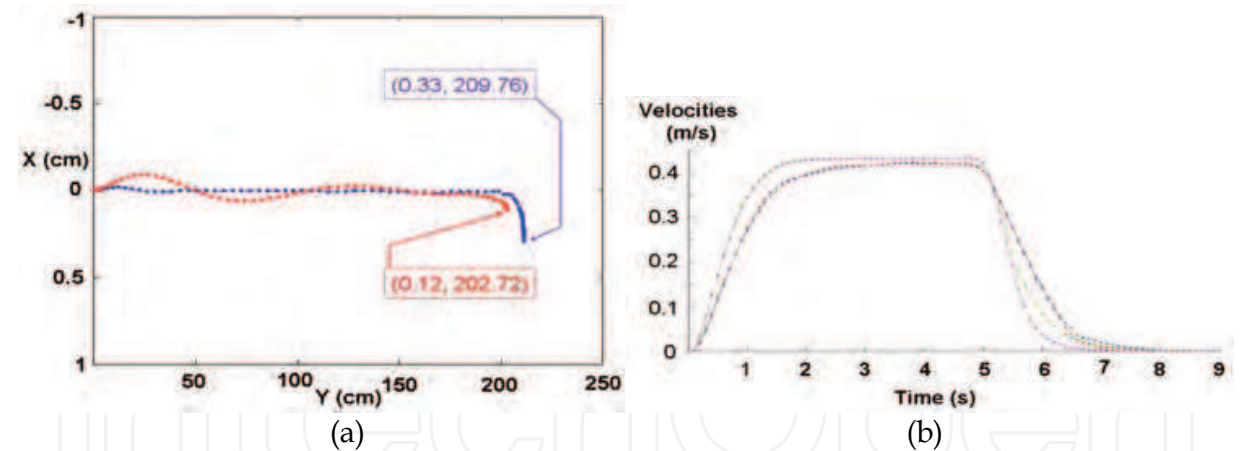


Fig. 7. (a) Trajectory tracking in red ( $n=10$ ,  $m=5$ ) and in blue ( $n=5$ ,  $m=3$ ). The larger prediction horizon shows a closer final goal achievement and worse trajectory tracking. (b) Wheel speeds during the 2m straight line tracking. The red and blue dots show the right and left speeds respectively, with  $n=10$  and  $m=5$ . The magenta and green dot lines depict the right and left speeds with  $n=5$  and  $m=3$ .

#### 4. The Horizon of local visual perception

##### 4.1 Introduction

The computer vision techniques applied to WMR have solved the problem of obstacle detection by using different methods as stereo vision systems, optical flow or DFF (depth

from focus). Stereo vision systems seem to provide the easiest cues to infer scene depth (Horn, 1998). The optical flow techniques used in WMR result in several applications as i.e. structure knowledge, obstacle avoidance, or visual servoing (Campbell et al., 2004). The DFF methods are also suitable for WMR. For example, three different focused images were used, with almost the same scene, acquired with three different cameras (Nourbakhsh et al., 1997). In this work, it is supposed that available obstacle positions are provided by using computer vision systems. The use of sensor information as a useful source to build 2D environment models consists of a free or occupied grid proposed by (Elfes, 1989). The knowledge of occupancy grids has been used for static indoor mapping with a 2D grid (Thrun, 2002). In other works of multidimensional grids, multi target tracking algorithms are employed by using obstacle state space with Bayesian filtering techniques (Coué et al., 2006). In this work it is proposed the use of the local visual information available from the camera as a local map that has enough information in order to achieve a desired objective. The present research assumes that the occupancy grid is obtained by a machine vision system. It is proposed an algorithm that computes the local optimal desired coordinate as well as the local trajectory to be reached. The research developed assumes indoor environments as well as flat floor constraints. However, it can be also applied in outdoor environments.

This section presents firstly the local map relationships with the camera configuration and poses. Hence, the scene perception coordinates are computed. Then, the optimal control navigation strategy is presented, which uses the available visual data as a horizon of perception. From each frame, it is computed the optimal local coordinates that should be reached in order to achieve the desired objective. Finally, the WMR dynamics constraints and the strategy of navigation drawbacks are also commented.

#### 4.2 Scene perception

The local visual data provided by the camera are used in order to plan a feasible trajectory and to avoid the obstacle collision. The scene available coordinates appear as an image, where each pixel coordinates correspond to a 3D scene coordinates. In the case attaining to this work, flat floor surface is assumed. Hence, scene coordinates can be computed using camera setup and pose knowledge, and assuming projective perspective. The Fig. 8 shows the robot configuration studied in this work. The angles  $\alpha$ ,  $\beta$  and  $\varphi$  are related to the vertical and horizontal field of view, and the tilt camera pose, respectively. The vertical coordinate of the camera is represented by  $H$ . Using trigonometric relationships, the scene coordinates can be computed:

$$y_j = H \tan(\varphi - \alpha/2 + \Delta\alpha) \quad \Delta\alpha = K_j \frac{\alpha}{R} \quad (0 \leq K_j \leq R) \quad (9)$$

$$x_{i,j} = \pm \frac{H}{\cos(\varphi - \alpha/2 + \Delta\alpha)} \tan(\Delta\beta) \quad \Delta\beta = K_i \frac{\beta}{C} \quad (0 \leq K_i \leq C/2) \quad (10)$$

The  $K_i$  and  $K_j$  are parameters used to cover the image pixel discrete space. Thus,  $R$  and  $C$  represent the image resolution through the total number of rows and columns. It should be noted that for each row position, which corresponds to scene coordinates  $y_j$ , there exist  $C$  column coordinates  $x_{i,j}$ . The above equations provide the available local map coordinates when no obstacle is detected. Thus, considering the experimental setup reported in Section 2, the local on-robot map depicted in Fig. 9 is obtained.

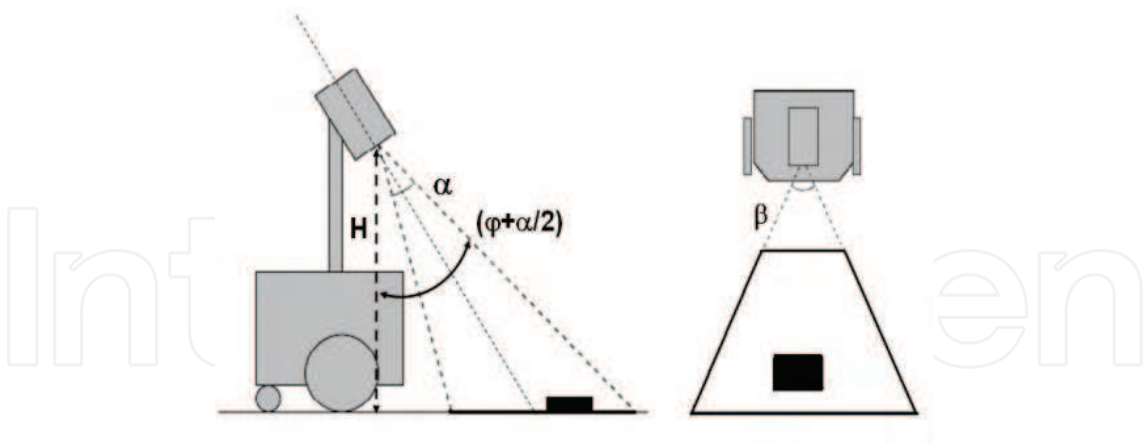


Fig. 8. Fixed camera configuration including vertical and horizontal field of view, and vertical tilt angle

4.3 Local optimal trajectory, algorithms and constraints

The available information provided by the camera is considered as a local horizon where the trajectory is planned. Hence, a local map with free obstacle coordinates is provided. In this sense, the available local coordinates are shown in Fig. 9. It is noted that low resolution scene grids are used in order to speed up the computing process. The minimization of a cost function, which consists of the Euclidean distance between the desired coordinates and the available local scene coordinates, can be optimally solved by finding the local desired coordinates. Hence, the algorithm explores the image pixels,  $IMAGE(i,j)$ , considering just the free obstacle positions. Once the local desired point is obtained, a trajectory is planned between the robot coordinates, at the instant when the frame was acquired, and the optimal scene coordinates. Thus, the current robot coordinates are related to this trajectory, as well as to the control methods.

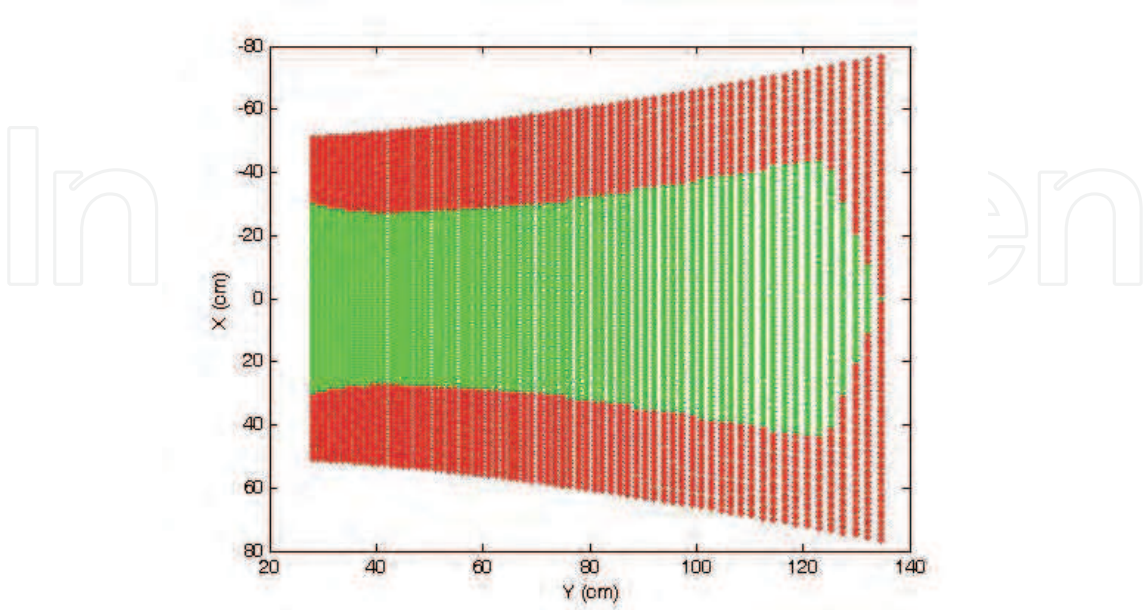


Fig. 9. Local visual perception free of obstacles, under 96x72 low resolution grids. Available local map coordinates (in green), the necessary wide-path (in red).

WMR movements are planned based on the local visual data, and always in advancing sense. The change in the WMR orientation can be done in advancing sense by the use of the trajectory/robot orientation difference as the cost function computed over the available visual data. The proposed algorithm, concerning to obtaining the local visual desired coordinates, consists of two simple steps:

- To obtain the column corresponding to the best optimal coordinates that will be the local desired  $X_i$  coordinate.
- To obtain the closer obstacle row, which will be the local desired  $Y_j$  coordinate.

The proposed algorithm can be considered as a first order approach, using a gross motion planning over a low resolution grid. The obstacle coordinates are increased in size with the path width of the robot (Schilling, 1990). Consequently, the range of visually available orientations is reduced by the wide-path of WMR. Other important aspects as dynamic reactive distance and safety stop distance should also be considered. The dynamic reactive distance, which should be bigger than the safety stop distance, is related to the robot dynamics and the processing time for each frame. Moreover, the trajectory situated in the visual map should be larger than a dynamic reactive distance. Thus, by using the models corresponding to the WMR PRIM, three different dynamical reactive distances are found. As i.e. considering a vision system that processes 4 frames each second; the table 1 shows these concepts.

The local minimal failures will be produced when a convergence criterion, similar to that used in (7), is not satisfied. Thus, the local visual map cannot provide with closer optimal desired coordinates, because obstacles block the trajectory to the goal. In these situations, obstacle contour tracking is proposed. Hence, local objectives for contour tracking are used, instead of the goal coordinates, as the source for obtaining a path until the feasible goal trajectories are found. The Fig. 10 shows an example with local minimal failures.

Model	Safety stop distance	Obstacle reactive distance	Robot displacement	Minimal allowable distance
Low velocity (0.15 m/s)	0.085m	0.125m	0.038m	0.248m
Medium velocity (0.3 m/s)	0.178m	0.125m	0.075m	0.378m
High velocity (0.45 m/s)	0.310m	0.125m	0.113m	0.548m

Table 1. Reactive criterion and minimal allowable distances

It is seen that in A, the optimal trajectory is a straight line between A and E. However, an obstacle is met at B, and local minimal failure is produced at B. When this situation occurs, no trajectory can approach to the desired goal,  $(X_d, Y_d)$ . Then, obstacle contour tracking is proposed between B and C. Once C is attained, local minimization along coordinates Y is found and the trajectory between C and D is planned. From D to E local minimums are reached until the final goal is achieved. It should be noted that once B is reached, the left or right obstacle contour are possible. However, the right direction will bring the robot to an increasing  $Y_j$  distance.

The robot follows the desired goals except when the situation of obstacle contour tracking is produced, and then local objectives are just the contour following points. The local minimal



failures can be considered as a drawback that should be overcome with more efforts. In this sense, by taking into account the vision navigation strategies (Desouza & Kak, 2002), it is proposed in this work the use of feasible maps or landmarks in order to provide local objective coordinates that can be used for guiding the WMR to reach the final goal coordinates. The use of effective artificial attraction potential fields should also be considered.

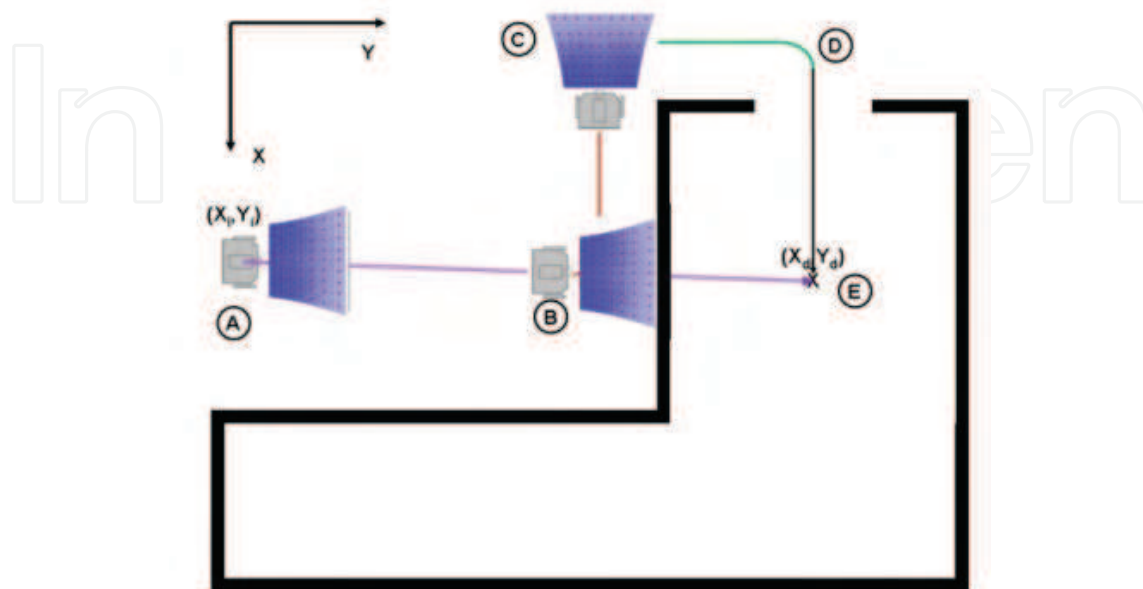


Fig. 10. Example of local minimal failures produced at B with A being the starting point and E the desired goal.

## 5. Testing local visual trajectories using LMPC strategies

### 5.1 Introduction

The minimization of path following error is considered as a challenging subject in mobile robotics. The main objective of highly precise motion tracking consists in minimizing the error between the robot and the desired path. Real-time implementation of MPC in the mobile robotics has been developed using global vision sensing (Gupta et al., 2005). In (Küne et al, 2005), MPC based optimal control was studied, which is useful for cases when nonlinear mobile robots are used under several constraints. In general, real-time implementation is possible when a short prediction horizon is used. By using MPC, the idea of receding horizon can deal with the local sensor information. MPC is based on minimizing a cost function related to the objectives for generating the optimal inputs.

The LMPC (local model predictive control) is proposed to use the available visual data in the navigation strategies for the goal achievement (Pacheco & Luo, 2007). Define the cost function as follows:

$$J(n, m) = \min_{\left\{ U(k+i|k) \right\}_{i=0}^{m-1}} \left\{ \begin{aligned} & \left[ X(k+n|k) - X_{ld} \right]^T P \left[ X(k+n|k) - X_{ld} \right] \\ & + \sum_{i=1}^{n-1} \left[ X(k+i|k) - \overline{X_{ld} X_{l0}} \right]^T Q \left[ X(k+i|k) - \overline{X_{ld} X_{l0}} \right] \\ & + \sum_{i=0}^{m-1} U^T(k+i|k) R U(k+i|k) \end{aligned} \right\} \quad (11)$$



The first term of (11) refers to reaching the local desired coordinates,  $X_{ld}=(x_d, y_d, \theta_d)$ . The second one is related to the distance between the predicted robot positions and the trajectory segment  $\overline{X_{ld}X_{l0}}$  given by a straight line between the initial robot coordinates  $X_{l0}=(x_{l0}, y_{l0}, \theta_{l0})$  from where the local perception frame was acquired, and the desired local position  $X_{ld}=(x_d, y_d, \theta_d)$  belonging to the local perception. The last one is related to the input signals denoted as  $U$ .  $P$ ,  $Q$  and  $R$  are weighting parameters that express the importance of each term.  $X(k+n|k)$  represents the terminal value of the predicted output after the horizon of prediction  $n$  and  $X(k+i|k)$  represents the predicted output values within the prediction horizon. The system constraints are also considered:

$$\left\{ \begin{array}{l} |U(k+1|k)| \leq G_1 \quad \alpha \in (0,1] \\ |[X_{k+n}] - [X_{ld}]| \leq \alpha |[X_k] - [X_{ld}]| \end{array} \right\} \quad (12)$$

Where  $X_{k+n}$  denotes the predicted coordinates and  $X_k$  the actual coordinates. The limitation of the input signal is taken into account in the first constraint. The last one is a contractive constraint (Wan, 2007). Contractive constraint arises in a convergence towards the desired trajectory; hence until a new trajectory is not commanded the control system will achieve the objective coordinates. Therefore, path planning consists in a set of points, obtained within the available field of view, and tracked by the LMPC strategy.

## 5.2 Experimental results

The trajectory tracking accuracy and time performance are two important aspects to be considered. In this context, the odometer system performance was analyzed by measuring the accuracy of the system. It was done by commanding long trajectories along lab corridors. After calibrating the odometer, the results showed that a commanded trajectory of 22m provided averaged final distance errors of less than 0.5m, and angular orientation errors of less than 5°. Hence in this research, it is analyzed local trajectories of less than 1.5m accordingly with the narrow visual perception provided. Thus, the odometer system errors can be neglected when local trajectories are considered. Therefore, the odometer system is locally used to compute LMPC trajectory tracking errors. The tested trajectories are obtained from the available set of local map coordinates as shown in Fig. 9. The LMPC results are analyzed when different trajectories tracking are commanded, as it is depicted in Fig. 11.

Denote  $E_1$  as the average final error,  $E_2$  the maximal average tracking error,  $E_3$  the average tracking error, and  $E_4$  the standard deviation of average tracking error. Table 2 presents the statistics concerning about the error obtained in cm testing the trajectories shown in Fig. 11.

It can be seen that the accuracy of trajectory tracking, when straight line is commanded, has a deviation error of 0.54cm. However, when a turning action is performed, the error in straight line tracking is bigger as consequence of the robot dynamics when it is moving forward. Hence, the forward movement consists in usually a steering action. Fig. 11 gives a clue about what is happening. Thus, the major turning angle will produce the major deviation distance. Usually, it is very difficult to reduce the approaching distance to zero, due to the control difficulty of dead zone for the WMR and to the fact that the final target is considered in the present work as being reached by the robot when the Euclidean approaching distance is less than 5cm.

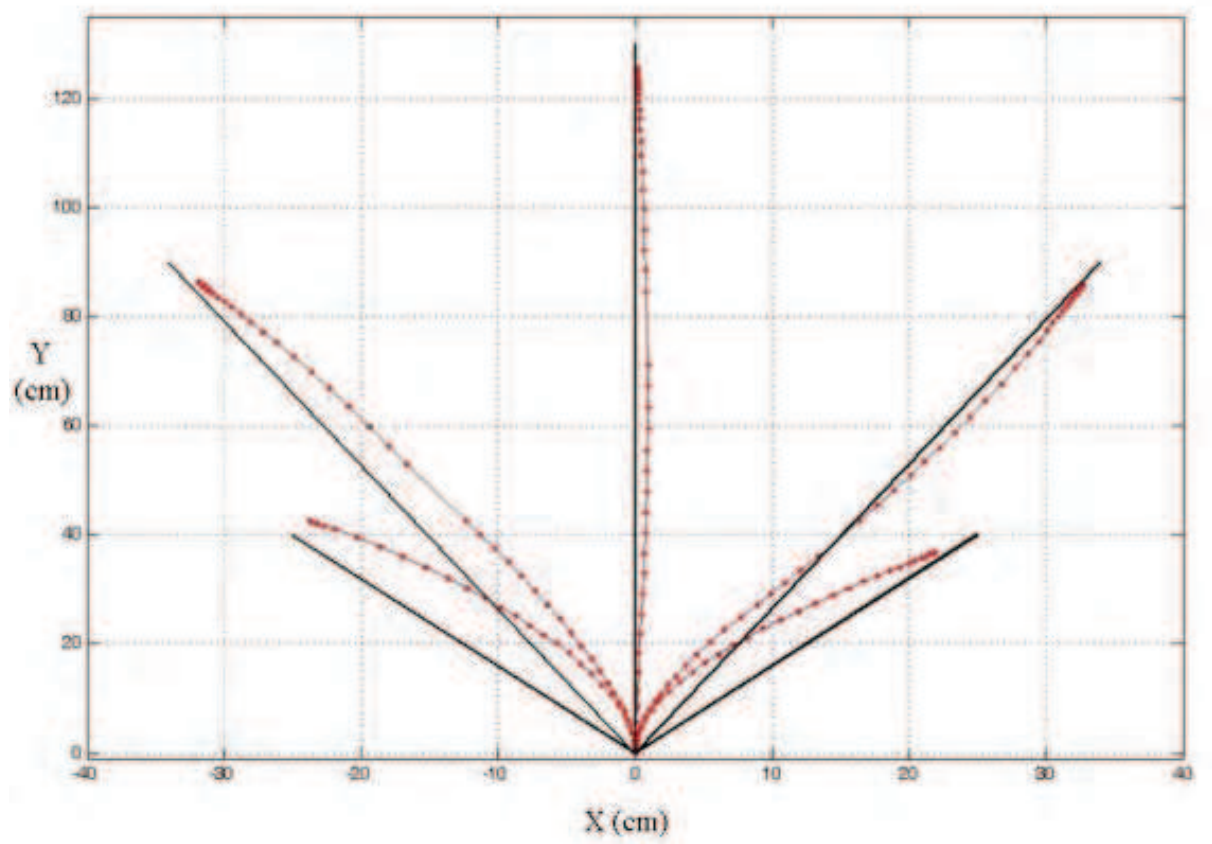


Fig. 11. Trajectory tracking tested from point to point by using the available local map coordinates provided by the monocular perception system.

Trajectory	E <sub>1</sub>	E <sub>2</sub>	E <sub>3</sub>	E <sub>4</sub>
From (0,0) to (0,130)	4.4cm	0.9cm	0.54cm	0.068
From (0,0) to (34,90)	3.8cm	3.9cm	2.3cm	0.82
From (0,0) To (25,40)	4.5cm	5.3cm	3.9cm	1.96

Table 2. Point to point trajectory tracking statistics

Other interesting results consist in testing the LMPC performance when the trajectory is composed of a set of points to be tracked. In this sense, when it is regarded to the kind of robot used, a pure rotation is possible by commanding the same speed with different sense to each wheel motor. Hence, when a trajectory is composed of many points, two possibilities exist: *continuous movement* in advancing sense, or *discontinuous movement* in which the robot makes the trajectory orientation changes by turning around itself at the beginning of the new straight segment. Fig. 12 shows the tracking performance of the robot by tuning around itself, when the robot follows a trajectory composed of a set of points (0,0), (-25,50), (-25,100), (0, 150) and (0,200). The reported trajectory deviations are less than 5 cm. However, the tracking time may reach up to 25s.

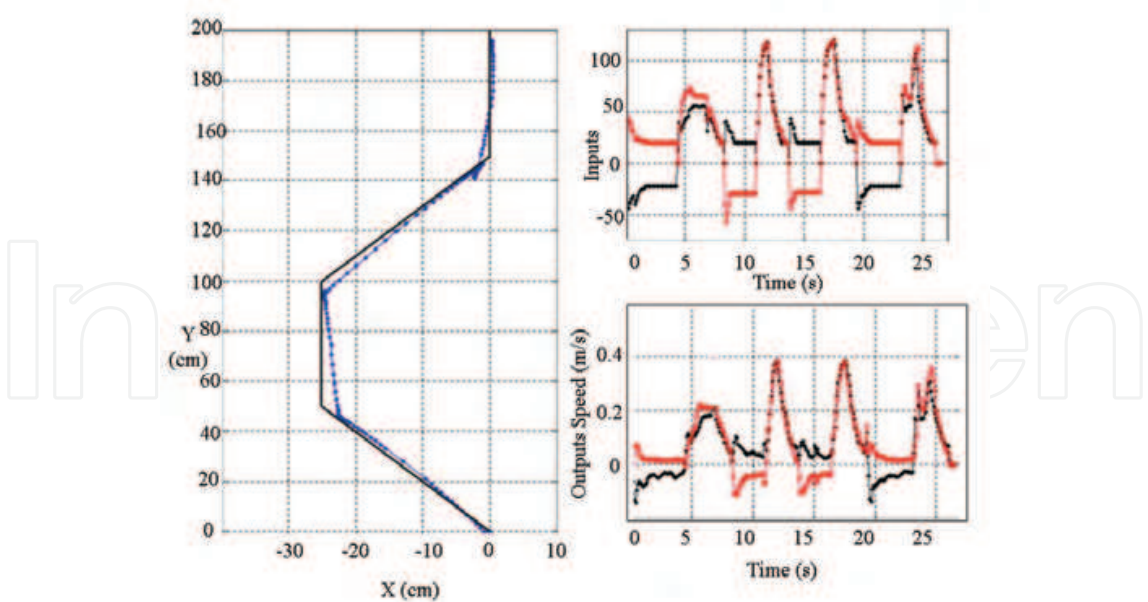


Fig. 12. Trajectory tracking with discontinuous movement

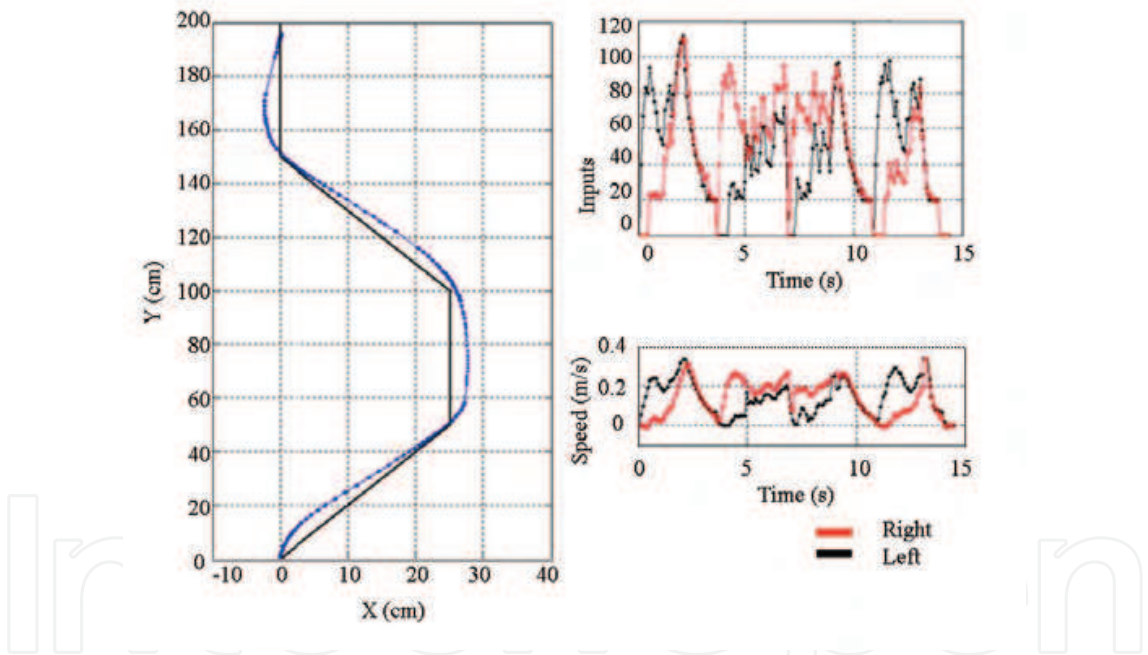


Fig. 13. Trajectory tracking with continuous movement

The trajectory tracking strategy with *continuous movement*, for a set of points (0,0), (25,50), (25,100), (0,150) and (0,200), is represented in Fig. 13. In this case, it is reported a bigger trajectory deviation, due to the WMR’s mechanical dynamics. The trajectory tracking is performed much faster ( $\leq 15s$ ). Hence, in the continuous moving case, it needs a turning action with a minimum radius, once the direction is attained the robot deviation is very small. Thus, trajectories following straight lines have reported errors less than 1cm. When time performance is analyzed the *continuous movement* presents a better behaviour. Moreover, in the research reported in this work, just *continuous movement* is possible, due to the narrow available field of view showed in Fig. 9.

## 6. Conclusions and future work

This work has integrated the control science and the robot vision knowledge into a computer science environment. Hence, local path planning by using local information is reported. One of the important aspects of the paper has been the simplicity, as well as the easy and direct applicability of the presented approaches. The proposed methodology has been attained by using the on-robot local visual information, acquired by a monocular camera system, and the techniques of LMPC. The use of sensor fusion, especially the odometer system information, is of a great importance. The odometer system uses are not just constrained to the control of the velocity of each wheel. Thus, the absolute robot coordinates have been used for planning a trajectory to the desired global or local objectives. The local trajectory planning has been done using the relative robot coordinates, corresponding to the instant when the frame was acquired. The available local visual data provide a local map, where the feasible local minimal goal is selected, considering obstacle avoidance politics.

Nowadays, the research is focused to implement the presented methods through developing flexible software tools that should allow to test the vision methods and to create locally readable virtual obstacle maps. The use of virtual visual information can be useful for testing the robot under synthetic environments and simulating different camera configurations. Further studies on LMPC should be done in order to analyze improvements such as changing the tracking set-point when the WMR is not close to the desired point or its relative performance with respect to other control laws. The influence of the motor dead zones is also an interesting aspect that should make further efforts to deal with.

## 7. Acknowledgments

This work has been partially funded by the Commission of Science and Technology of Spain (CICYT) through the coordinated project DPI-2007-66796-C03-02, and by the Government of Catalonia through the Network Xartap and the consolidated research group's grant SGR2005-01008.

## 8. References

- Boyd, S. & Vandenberghe, L. (2004). *Convex Optimization*, Cambridge University Press, ISBN-13: 9780521833783
- Campbell, J.; Sukthankar, R. & Nourbakhsh, I. (2004). Techniques for Evaluating Optical Flow in Extreme Terrain. *Proceedings of Intelligent Robots and Systems*, 3704-3711, ISBN 0-7803-8463-6, Sendai (Japan), September 2004.
- Coué, C.; Pradalier, C.; Laugier, C.; Fraichard, T. & Bessière, P. (2006). Bayesian Occupancy Filtering for Multitarget Tracking: An Automotive Application. *International Journal of Robotics Research*, Vol. 25, No. 1, (Jan 2006) 19-30
- DeSouza, G.N. & Kak, A.C. (2002). Vision for Mobile Robot Navigation: a survey. *Pattern Analysis and Machine Intelligence*, Vol. 24, No. 2, (Feb 2002), 237-267
- Elfes, A. (1989). Using occupancy grids for mobile robot perception and navigation, *IEEE Computer*, Vol. 22, No. 2, (Jun 1989) 46-57, ISSN: 0018-9162



- Fox, D.; Burgard, W. & Thrun, S. (1997). The dynamic window approach to collision avoidance. *IEEE Robotics & Automation Magazine*, Vol. 4, No. 1, (Mar 1997) 23-33, ISSN 1070-9932
- Gupta, G.S.; Messom, C.H. & Demidenko, S. (2005). Real-time identification and predictive control of fast mobile robots using global vision sensor. *IEEE Trans. On Instr. and Measurement*, Vol. 54, No. 1, (February 2005) 200-214, ISSN 1557-9662
- Horn, B. K. P. (1998). Robot Vision. *MIT press*, Ed. McGraw-Hill, ISBN 0-262-08159-8, London (England)
- Küne, F.; Lages, W. and Da Silva, J. (2005). Point stabilization of mobile robots with nonlinear model predictive control. *Proc. IEEE Int. Conf. On Mech. and Aut.*, 1163-1168, ISBN 0-7803-9044, Ontario (Canada), July 2005
- Ljung, L. (1991). Issues in System Identification. *IEEE Control Systems Magazine*, Vol. 11, No.1, (Jan 1991) 25-29, ISSN 0272-1708
- Maciejowski, J.M. (2002). *Predictive Control with Constraints*, Ed. Prentice Hall, ISBN 0-201-39823-0, Essex (England)
- Nourbakhsh, I. R.; Andre, D.; Tomasi, C. & Genesereth, M. R. (1997). Mobile Robot Obstacle Avoidance Via Depth From Focus. *Proc. IEEE Robotics and Aut. Systems*, 151-158.
- Murray R. M.; Aström K. J.; Boyd S. P.; Brockett R. W. & Stein G. (2003). Future Directions in Control in an Information-Rich World. *IEEE Control Systems Magazine*, Vol. 23, No. 2, (April 2003) 20-33, ISSN 0272-1708
- Nesterov, I. E.; Nemirovskii, A. & Nesterov, Y. (1994). Interior-Point Polynomial Methods in Convex Programming. *Siam Studies in Applied Mathematics*, Vol 13, Publications, ISBN 0898713196
- Norton, J. (1986). An Introduction to Identification. *Academic Press*, London and New York, 1986
- Ögren, P. & Leonard, N. (2005). A convergent dynamic window approach to obstacle avoidance. *IEEE Transaction on Robotics*, Vol. 21, No. 2., (April 2005) 188-195
- Ortega, J. M. & Rheinboldt, W.C. (2000). Iterative Solution of Nonlinear Equations in Several Variables. *Society for Industrial and Applied Mathematics*, ISBN-10: 0898714613
- Pacheco, L. & Luo, N. (2007) Mobile Robot Local Predictive Control Using a Visual Perception Horizon, *Int. Journal of Factory Autom. Robotics and Soft Comp.*, Vol. 1, No. 2, (Dec 2007) 73-81
- Pacheco, L.; Luo, N.; Ferrer, I. and Cufí, X. (2008). Control Education within a Multidisciplinary Summer Course on Applied Mobile Robotics. *Inter. 17th IFAC World Congress*, 11660-11665, Seoul (Korea), July 2008
- Rimon, E. & Koditschek, D. (1992). Exact robot navigation using artificial potential functions. *IEEE Transaction Robotics and Automation*, Vol. 8, No. 5, (Oct. 1992) 501-518
- Schilling, R.J. (1990). Fundamental of Robotics. *Prentice-Hall (Ed.)*, New Jersey (USA) 1990, ISBN 0-13-334376-6



- Thrun, S. (2002). Robotic mapping: a survey. Exploring Artificial Intelligence in the New Millennium, *Exploring Artificial Intelligence in the New Millennium*, G. Lakemeyer and B. Nebel, editors. Morgan Kaufmann, San Mateo, CA
- Wan, J. (2007) Computational reliable approaches of contractive MPC for discrete-time systems, *PhD Thesis, University of Girona*

IntechOpen

IntechOpen



## **Robotics Automation and Control**

Edited by Pavla Pecherkova, Miroslav Flidr and Jindrich Dunik

ISBN 978-953-7619-18-3

Hard cover, 494 pages

**Publisher** InTech

**Published online** 01, October, 2008

**Published in print edition** October, 2008

This book was conceived as a gathering place of new ideas from academia, industry, research and practice in the fields of robotics, automation and control. The aim of the book was to point out interactions among various fields of interests in spite of diversity and narrow specializations which prevail in the current research. The common denominator of all included chapters appears to be a synergy of various specializations. This synergy yields deeper understanding of the treated problems. Each new approach applied to a particular problem can enrich and inspire improvements of already established approaches to the problem.

### **How to reference**

In order to correctly reference this scholarly work, feel free to copy and paste the following:

Lluís Pacheco, Ningsu Luo and Xavier Cufí (2008). Predictive Control with Local Visual Data, Robotics Automation and Control, Pavla Pecherkova, Miroslav Flidr and Jindrich Dunik (Ed.), ISBN: 978-953-7619-18-3, InTech, Available from:

[http://www.intechopen.com/books/robotics\\_automation\\_and\\_control/predictive\\_control\\_with\\_local\\_visual\\_data](http://www.intechopen.com/books/robotics_automation_and_control/predictive_control_with_local_visual_data)

**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](http://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

© 2008 The Author(s). Licensee IntechOpen. This chapter is distributed under the terms of the [Creative Commons Attribution-NonCommercial-ShareAlike-3.0 License](https://creativecommons.org/licenses/by-nc-sa/3.0/), which permits use, distribution and reproduction for non-commercial purposes, provided the original is properly cited and derivative works building on this content are distributed under the same license.

IntechOpen

IntechOpen