# Low level control of an omnidirectional mobile robot*

R. Comasolivas, J. Quevedo, T. Escobet, A. Escobet and J. Romera

***Abstract*— This paper presents the low level control of an holonomic robot with four omnidirectional wheels. A robust control technique named Quantitative Feedback Theory (QFT), based on an uncertain linear model has been selected to design the PID speed controllers for the four-wheeled robot. A piecewise model has been estimated by means of the least squares estimation approach based on experimental results of the robot in closed loop. In particular, the control is designed using this piecewise model. The performances of the proposed approach are analyzed in real time domain.**

***Keywords**:* Robust Control, Holonomic Mobile Robot, Quantitative Feedback Theory, PID Control.

## I. Introduction

Robots with omnidirectional locomotion have the ability to move in any direction and rotate simultaneously, fact that gives them great freedom of movement and allows them to maneuver in tight and complex spaces. Omnidirectional mobile robots are a kind of holonomic robots. They have better maneuverability compared to non-holonomic robots. This capability of movement is achieved by providing the wheels for mobility in lateral displacement. Today omnidirectional robots are very useful in industrial and home applications as the popular vacuum cleaner robots. In particular, in this work an omnidirectional holonomic mobile robot of four wheels is used.

Different techniques have been applied in order to solve the control problem for omnidirectional mobile robot. Model-predictive control (MPC) techniques are commonly applied to trajectory following [1]. The application of MPC requires a precise model of the system which, in practice, is difficult to achieve due to nonlinearities present in the physical systems. The nonlinearities in the mobile robots are mainly due to the friction. Many authors have addressed the problem of friction compensation, using model based and model free-based techniques [2]. The friction model-based approach consists of feedforward and feedback control that introduced the friction model into the system to cancel out the nonlinear effect. These techniques require a complete model of the friction behavior or adaptive algorithms for estimating the friction parameters online. The friction model-free approach consists in applying linear or nonlinear control for compensating the friction and improves the overall performance of the system. In general, these techniques are based on modifications of PID controllers in order to cope with the nonlinear effects of friction.

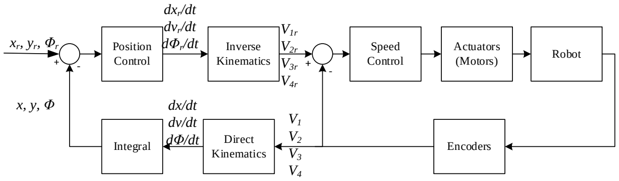

Fig. 1. Cascade control strategy of the robot.

In this paper the model-free approach based on PID structure will be applied to compensate the nonlinearities due to the friction of an omnidirectional robot. More specifically, the cascade control strategy, shown in Fig. 1, will be applied. It consists of two loops: the high level loop (or outer-loop controller) that intends to adjust the control position and orientation to follow the commanded trajectory and a low level loop (or inner-loop) that follows the body rate command given by the high level loop. Both control loops are interconnected by the transformations of direct and inverse kinematics of the robot itself. Thus, the controller output paths setpoints results in four-wheel speeds through the matrix inverse kinematics of the robot and otherwise feedback sensors (encoders) that provides movement of the four wheels is translated into the position and orientation of the robot dynamics by direct kinematics matrix. High-level control with strategies for trajectory planning is discussed in [3]-[4]-[5]. In this paper we focus on the design of a robust low-level controller or speed control that controls the speed of the four wheels, with the objective to compensate nonlinearities.

This work presents the design of a control strategy taking into account the uncertainties and nonlinearities of an omnidirectional mobile robot. This technique of robust control, named Quantitative Feedback Theory (QFT) uses linear interval models.

The paper is organized as follows: in Section II the omnidirectional robot platform is described. Section III shows the method used for estimating the robot model. Section IV describes QFT technology and the design of QFT controller. Section V shows the experimental results. Finally, the conclusions can be found in Section VI.

## II. Robotic platform description

### A. Physical description

The Robotic platform is a four-wheel holonomic mobile robot developed by ST Microelectronics under the i-Sense European project. This platform is a holonomic robot

* This work has supported by CICYT HARCRICS DPI2014-58104-R.

R. Comasolivas, J. Quevedo, T. Escobet and J. Romera are with Research Center for Supervision, Safety and Automatic Control, Technical University of Catalonia (UPC), Rambla Sant, Nebridi, 10, 08222 Terrassa, Spain (e-mails: ramon.comasolivas@upc.edu, joseba.quevedo@upc.edu, teresa.escobet@upc.edu and juli.romera@upc.edu).

A. Escobet is with Department of Electronic System Design and Programming, Technical University of Catalonia (UPC), Av. Bases de Manresa, 61-73, 08240 Manresa, Spain (e-mail: Toni@epsem.upc.edu).

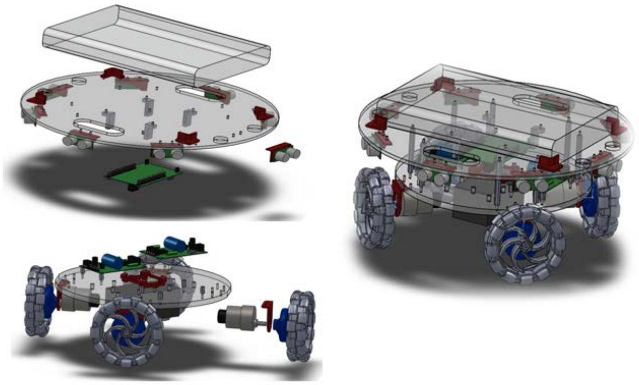

Fig. 2. Platform FWHMR scheme (right) and real picture (left).

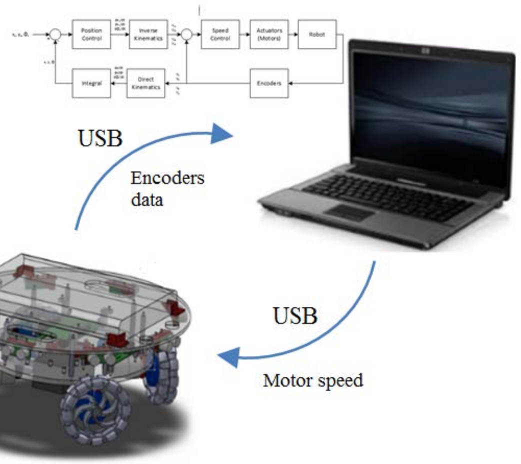

Fig. 3. Control system architecture.

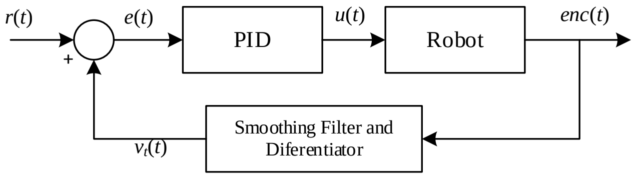

Fig. 4. Low level PID Control strategy.

presenting a great maneuverability and effectiveness. The omnidirectional feature is reached thanks to the characteristics of the wheels, which roll forward like normal wheels but can slide sideways at the same time, allowing almost independent tangential, normal and angular velocities (holonomic property).

It consists of a round structure of 25 cm of diameter supported by four omnidirectional wheels (see Fig. 2). With all components mounted its total weight is about 4.3kg. The robotic platform is equipped with four DC motors, one DC-DC converter, two motor drives board, eight ultrasonic ranges and a 12V deep cycle lead acid battery of 1.2 Ah. Each motor is equipped with incremental encoder counting N = 3600 pulses/turn.

### *B. Control System Architecture*

The platform has two-level control architecture. The main control is conducted by an onboard computer connected via USB communication to the microcontroller on the robot (see Fig. 3).

The steering control module is run in a Net-PC (a Cortex-M4 with CPU working at 168MHz with 192KB of RAM) with Linux. The control algorithms are written in MATLAB and run with a sample time of 40 ms. Three working modes have been implemented: a) speed control for each wheel; b) current control of each motor; and, c) open loop control that allows sending individual PWM set point for each motor. Regarding the high-level algorithms navigation, external communications, algorithms fault tolerance, among others, can be implemented as tasks.

The low-level control layer is in charge of the execution of the high-level velocity commands such as ultrasound sensors acquisition and motor control. It is supported by the electronic device STM32F4D, which is a compact development system with lots of on-board peripherals, where the central part of the system is the 32-bit STM32F407VGT6 6ARM Cortes-M4-microcontroler. The microcontroller performs three basic tasks: 1) to communicate with the higher-level controller through USB; 2) reading encoder counts interrupt driven; and 3) generation of PWM duty cycle.

### *C. Control Strategy*

The control task is to track a given time reference position and orientation with the minimum error. The block diagram of the mobile robot control system is shown in Fig. 1. The control design consists of a cascade velocity and position feedback loop. The control position computes the velocity reference of each wheel. The decided velocity is controlled by the PID feedback control shown in Fig. 4.

As mentioned in the previous subsection, to measure the motor speed we used the incremental encoders coupled to the wheel axes. In order to filter the noise inherent to the measurement process (encoders) and performing its differentiation for obtaining a more reliable estimation of the wheels speed, a digital smoothing polynomial filter has been selected. The filter is based on a quadratic polynomial proposed by Savitzky-Golay [6] where each 4 raw data of each encoder at every $t_s' = 0.01$ seconds ($f_s' = 100$ Hz) of sampling time are replaced by only one smoothed data using a second order polynomial. From this it is easy to obtain the derivative of the polynomial as an estimation of each wheel speed. In this case, the new sampling time is $t_s = 4t_s'$ ($f_s =$ 25Hz) which is enough for the control loop. The formula of the Savitzky-Golay smoothing filter to estimate the speed of each wheel, $i$, is:

$$v_i(kt_s) = \frac{1}{2t_s}\sum_{j=1}^{4} h_j enc_i(kt_s - (j-1)t_s') \tag{1}$$

where $v_i$ is the wheel speed, $enc_i$ is the measure of the encoder with a sampling time of $t_s'$ seconds and

$h = [1.05 \quad -0.65 \quad -0.85 \quad 0.45]$.

## III. Interval Model Identification

In many literature works [7], mobile robots are modeled taking into account the inertia, traction, viscous friction and Coulomb friction forces, giving a non-linear and multivariable system. But in practice, the parameters of the resulting non-linear model are difficult to quantify accurately, or the model itself is in error. Many techniques have been used to solve the problem of estimating the parameters or states of nonlinear systems. The most basic approach is to linearize the system around an operating point and to develop a first order set of linear state equations. Other techniques use least squares methods directly to try to minimize the errors in the non-linear estimation problem.

The technique used in this paper extends the linearization technique by using piece-wise linear models that cover the expected range of the state variables [8]. The only restriction is that the non-linearities must be able to be approximated as piecewise linear functions.

The non-linear model of the mobile robot platform can be written in the form:

$$\dot{\mathbf{x}}(t) = f\big(\mathbf{x}(t), \mathbf{u}(t)\big)$$
$$\mathbf{y}(t) = h(\mathbf{x}(t)), \qquad (2)$$

where $f$ and $h$ are non-linear functions, $\mathbf{x}(t) \in \mathcal{R}^4$ are the system state, $\mathbf{u}(t) \in \mathcal{R}^4$ are the motor's voltage and $\mathbf{y}(t) \in \mathcal{R}^4$ are the wheels velocities. The system is modeled as a set of piecewise linear equation of the form:

$$\dot{\hat{\mathbf{x}}}(t) = \mathbf{A}_{\boldsymbol{i}}\hat{\mathbf{x}}(t) + \mathbf{B}_{\boldsymbol{i}}\mathbf{u}(t)$$
$$\hat{\mathbf{y}}(t) = \mathbf{C}_{\boldsymbol{i}}\hat{\mathbf{x}}(t)\text{ , for } x \in S_i, i = 1, \cdots, q. \qquad (3)$$

assuming that the non-linear model (2) is decomposed into $q$ individual linear state space models (3) with matrices $\mathbf{A}_{\boldsymbol{i}}, \mathbf{B}_{\boldsymbol{i}}$ and $\mathbf{C}_{\boldsymbol{i}}$. Each linear model (3) is defined as a polyhedric partition $S_i \subseteq \mathcal{R}^4$ of the state space. And, the equivalent input-output representation is given by:

$$\hat{\mathbf{y}}(t) = \mathbf{P}_{\boldsymbol{i}}(s)\mathbf{u}(t), \qquad (4)$$

where $\mathbf{P}_{\boldsymbol{i}}(s) = \mathbf{C}_{\boldsymbol{i}}(sI - \mathbf{A}_{\boldsymbol{i}})^{-1}\mathbf{B}_{\boldsymbol{i}}$

In our study case, the set of linear systems has been identified performing system identification experiments around different equilibrium position under closed-loop control. Fig. 5 shows the voltage and velocity of one wheel when the closed-loop reference signal is a sequence of steps. Each step has been analyzed using the well-known Auto-Regressive eXogenous (ARX) method, allowing the estimation of a set of individual systems. A first order model, with two parameters gain ($k$) and time constant ($\tau$) (5), provides a good fit for the 20th individual linear models for each wheel.

$$\mathbf{P}_i(s) = \frac{k_i}{\tau_i s + 1}, \qquad (5)$$

The estimated parameters of (5) are given in Fig. 6. Each color identifies a wheel of the robot. Notice that, every wheel has a different behavior, which is uncorrelated with the reference signal.

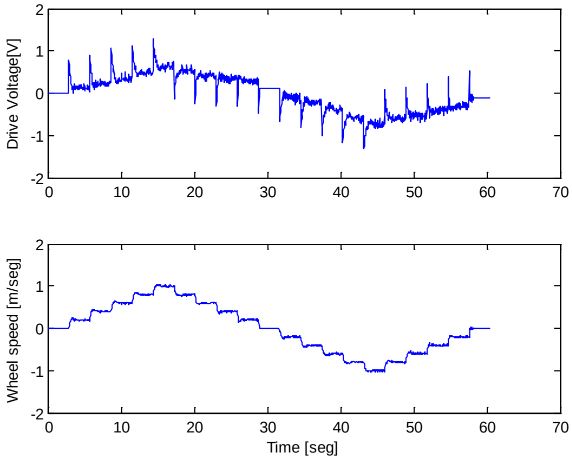

Fig. 5. Voltage and velocity of one wheel.

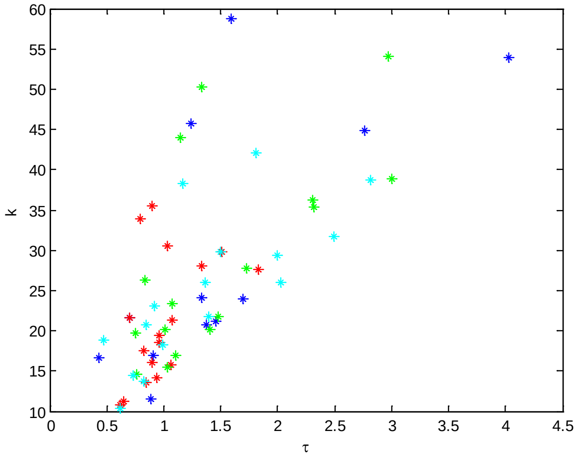

Fig. 6. Estimated parameters for each individual linear function, the colors correspond to a wheel of the robot.

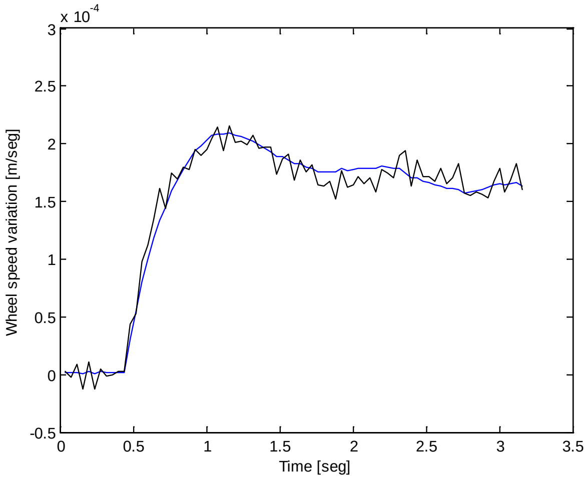

Fig. 7. Measured (black line) and simulated (blue line) model output of one linear system

Fig. 7 shows the simulated output of one of the estimated models together with the measured output.

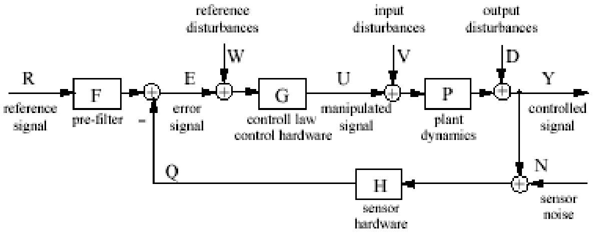

Fig. 8. QFT generic control structure.

## IV. Robust design using QFT

As it has been mentioned in section III, the robot platform is characterized by multiple linear systems which makes it difficult to maintain good stability margins and performance properties for the closed-loop system. The purpose of this section is to introduce robust feedback control design in order to achieve improved robust performance of the controller.

### A. QFT description

Quantitative feedback theory, developed by Isaac Horowitz [9], (Fig. 8), is a frequency domain design technique using the Nichols chart (NC) in order to achieve a desired robust design over a specified region of plant uncertainty. Specifications have to be expressed in frequency domain, thus desired time-domain responses have also to be translated into frequency domain tolerances which implies some constraints (bounds) on the loop transfer function in the NC.

In general, an uncertain plant is described by the interval model:

$$P(\theta, s) = \frac{N(\theta, s)}{D(\theta, s)}, \tag{6}$$

where $N$ and $D$ are polynomials in the $s$ domain and $\theta$ is the vector of uncertain parameters of dimension $n_\theta$ with their values bounded by a compact set $\theta \in \Theta$ of box type, i.e., $\Theta = \{\theta \in \mathcal{R}^{n_\theta} | \underline{\theta} \leq \theta \leq \overline{\theta}\}$. The feedback controller is defined as following

$$G(\kappa, s) = \frac{N_c(\kappa, s)}{D_c(\kappa, s)}, \tag{7}$$

where $N_c$ and $D_c$ are polynomials in the $s$ domain; and $\kappa$ is a vector controller parameters of dimension $n_\kappa$. Design process allows a designer to see what trade-offs are necessary to achieve a desired performance level. QFT technique procedure implies the next 4 basic steps: plant templates, frequency bounds, loop shaping and prefilter design (see [10]).

*Plant templates*: The uncertain plant $P(s)$ is first translated in the frequency domain (using a discrete grid of frequencies $\omega_1, \omega_2, \cdots, \omega_N$, typically chosen to cover adequately the system's bandwidth), resulting in $N$ uncertainty plant templates.

*Frequency Bounds*: The description of the desired closed-loop system performances (as robust stability, tracking, disturbance rejection, control effort) are required. In QFT design methodology, these requirements are described as frequency constraints on the nominal open-loop function and represented on the NC as curves, called frequency bounds.

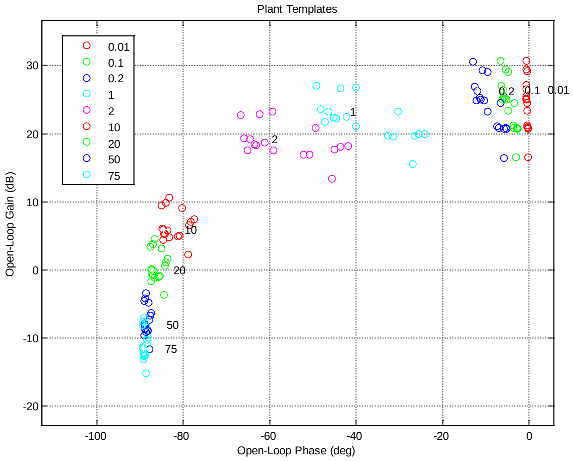

Fig. 9. Plant templates for the piecewise model.

*Loop Shaping*: In this step, the controller design is undertaken on the NC considering the frequency constraints and the nominal open loop, $L_0(s)$, of the system. The nominal loop is given by $L_0(s) = G(s)P_0(s)$, where the nominal model $P_0(s)$ is one model of the set $P(s)$ and the controller function $G(s)$ is chosen interactively by the designer in accordance with the frequency constraints.

*Prefilter Design*: Finally, in the case of tracking performances, a prefilter, $F(s)$, design is conducted using a Bode diagram. There exist different Computer Aided Design (CAD) packages to make the controller tuning easier. Some examples of these CAD packages are QFT Toolbox for Matlab [11], QFT Control Toolbox (QFTCT) [12] and QFT Interactive Tool [13].

### B. Templates generation

There exist several approaches to solve the problem of generating conservative template boundaries [11]. In general these procedures try to envelope the plant uncertainty using some geometrical surface. In this paper, we propose to compute the templates taking into account the set of linear models (5) estimated in section III.

The plant templates are plotted on the Nichols chart at the useful desired frequency $\omega = \{0.01, 0.1, 0.2, 1, 2, 10, 20, 50, 75\}$ rad/seg for the estimated piecewise models (5).

The templates obtained for the family of plants and for the set of frequencies are as illustrated in Fig. 9, where the nominal plant is for $k$ =21.08 and $\tau$ = 1.053.

### C. Performance specifications

Two performance specifications are formulated.

*Robust stability*. To ensure robust stability in the closed loop system the next constraint must be satisfied:

$$\left|\frac{L(j\omega)}{1 + L(j\omega)}\right| \leq M_R(j\omega), \tag{8}$$

where $M_R(j\omega)$ is the bounded magnitude of the closed-loop frequency response and $L(j\omega) = G(j\omega)P(j\omega)$, is the open

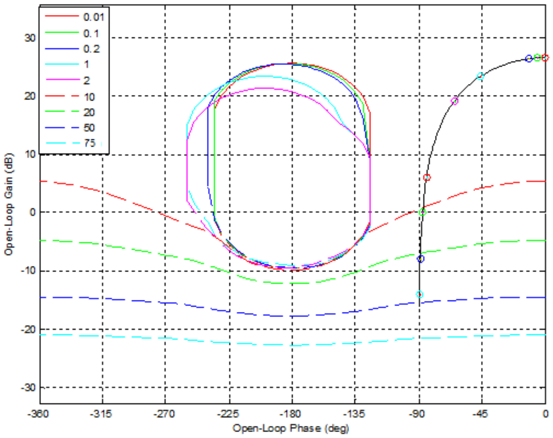

Fig. 10. Open-loop frequency response and QFT bounds with the piecewise models with *G*=1.

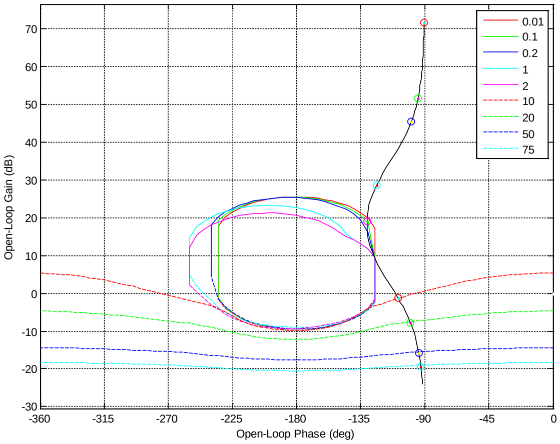

Fig. 11. Open-loop frequency response with $G_1$.

loop transfer function, including the uncertain plant $P(j\omega)$ and the controller $G(j\omega)$ to design.

*Control effort*. The control effort specification is given by:

$$\left|\frac{G(j\omega)}{1+L(j\omega)}\right| \leq M_U(j\omega), \tag{9}$$

where $M_U(\omega)$ is the bounded magnitude.

For controller design, two numerical specifications have been considered to obtain a good compromise between robust stability and control effort by trial and error: $M_R = 1.2$ (1.6 dB) and $M_U = 0.42$. (-7.5 dB).

The intersection of the QFT bounds (Fig. 9) with the inequalities (8) and (9) was computed on the NC at the same frequencies of plant templates for the first specification, as depicted in Fig. 10. The vertical curve line represents the frequency response of the nominal open-loop transfer function with the controller $G(j\omega) = 1$ at several frequency points marked with the same colors as the bounds. The closed round lines represent the worst case of robust stability bounds and the lines in the lower area represent the worst case of control effort. The bound plotted with a solid line implies that $L_0(s)$ must lie above it in order to meet the performance specifications whereas the bound plotted with a dashed line implies that $L_0(s)$ must lie below [14].

### D. *Loop-shaping controller*

The objective is to design a controller $G(s)$ with a PI structure modelled by

$$G(s) = K_P + \frac{K_I}{s}, \tag{10}$$

being $K_P$the proportional gain and $K_I$the integration gain. The controller is synthesized in NC by changing the gain and adding poles and zeros until the nominal plant lies near its bounds.

The transfer function of the controller designed considering both specifications is:

$$G_1(s) = \frac{0.4\ (s + 4.552)}{s} \tag{11}$$

The PID parameters are $K_P = 0.4$ and $K_I = 1.8$. The resulting open-loop frequency responses with this controller are illustrated in Fig. 11. The controller obtained is robust, that is, it provides good results for all the family of plants defined, not only for the nominal plant used in the loop-shaping stage.

## V. Performance Validation

The validation of the obtained results are made checking the specifications in the frequency and temporal domains. The analysis of the closed loop system performances in frequency domain is illustrated in Fig. 12, showing that the worst case closed loop frequency response and control effort magnitudes in both scenarios is below the specified values ($M_R$=1.2 = 1.584 dB and $M_U$ = 0.42=-7.5 dB, dashed line).

Fig. 13 shows angular speed response and speed error of one wheel after introducing a set of step changes as speed reference in the four DC motors. In order to achieve a more detailed view of the angular speed response, Fig. 14 shows the four wheels angular speed evolution during two steps. Notice that the motors reach the reference speed rapidly with an overshoot. In all the scenario, the maximum overshoot obtained is less than 20% satisfying the design specification ($M_R$=1.2). The rise time is between 0.16 and 1.12s.

The performance of the designed controller, $G_1$, is compared with the original controller PID proposed by the platform developers, with parameters: $K_P$ =0.6 and $T_I$=0.1, named as $G_2$. The Figs. 15 and 16 show the results with the original controller, $G_2$, applying the same set of step changes as speed reference in the four DC motors. In these results, there is no overshoot but the rise time exceeds 10 seconds. Finally, Table 1 displays a summary of the comparison regarding two well known criteria: integral absolute error (IAE) and integral absolute of controlled action (IAU) for the 4 wheels. The IAU is very similar in both controllers but the IAE is clearly reduced around 20% using the QFT controller with respect to the original one.

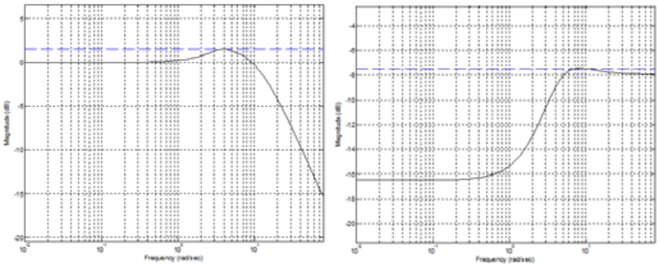

Fig. 12. Closed loop robust stability margins (right) and closed loop effort control margin (left).

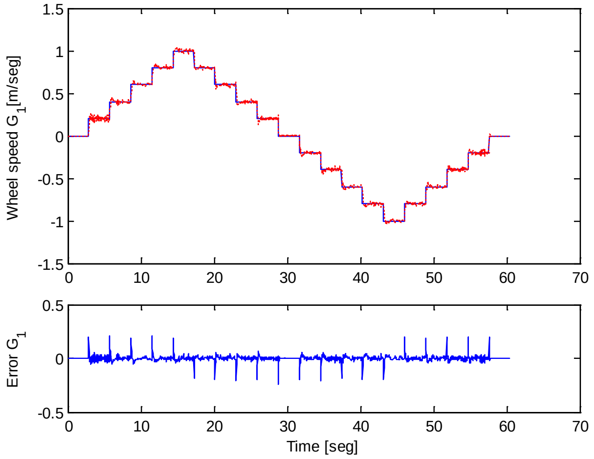

Fig. 13. Angular speed (red) and speed reference (blue) (over) and error speed (below) of one wheel with $G_1$.

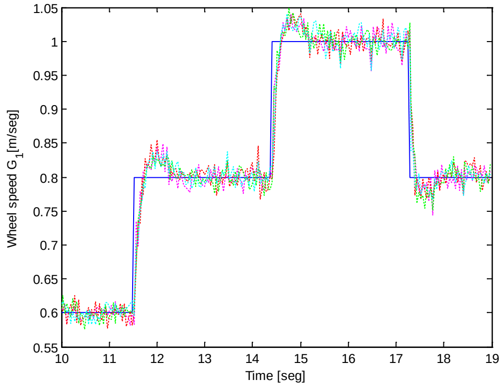

Fig. 14. Zoom of angular speed response of the four wheels with $G_1$.

Table 1. Results of the robot scenario with the two controllers G1 and G2 using IAE and IAU criteria for each wheel.

| Criteria | Robot Wheel | $G_1$ | $G_2$ |
|---|---|---|---|
| IAE | 1 | 18.6506 | 26.5345 |
| | 2 | 20.0840 | 24.5890 |
| | 3 | 19.0484 | 24.0632 |
| | 4 | 17.1635 | 24.8780 |
| IAU | 1 | 9.7830 | 8.2625 |
| | 2 | 4.7066 | 6.5278 |
| | 3 | 7.2056 | 7.0192 |
| | 4 | 7.2809 | 8.1058 |

## VI. Conclusion

In this paper the design and implementation of a robust speed controller for a four-wheeled omnidirectional mobile robot using QFT technique has been presented. The robot platform has been modeled with a piecewise model that has been estimated by means of least-squares identification using data collected from the real robot. The real experiments show the expected performances of the controller design and better dynamical performances with respect to the original controller of this mobile robot.

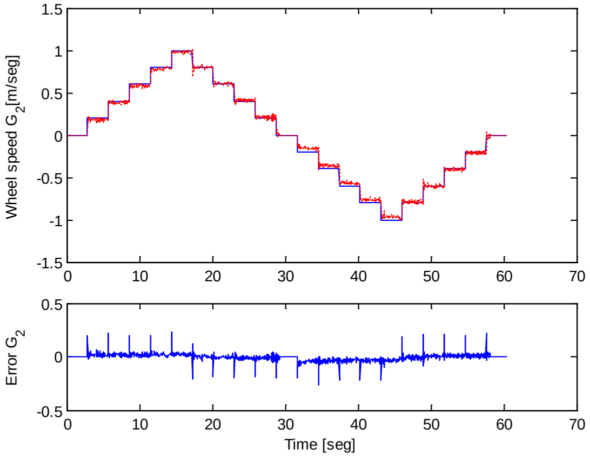

Fig. 15. Angular speed (red) and speed reference (blue) (over) and error speed (below) with $G_2$.

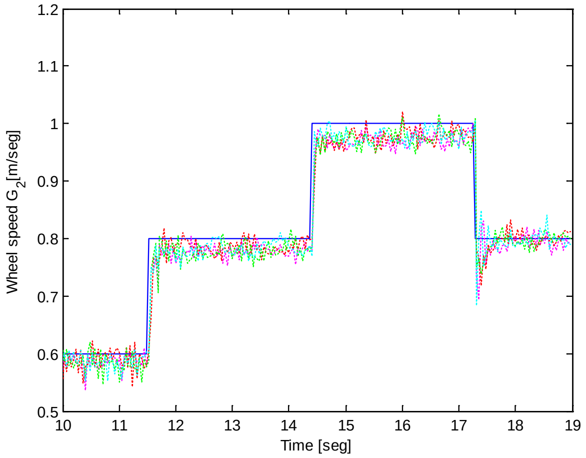

Fig. 16. Zoom of angular speed response of the fourth wheels with $G_2$.

## References

[1] F. Nejjari, D. Rotondo, V. Puig, M. Innocenti, "Quasi-LPV modelling and non-linear identification of a Twin Rotor System," in *Proc Control & Automation (MED)*, 2012 20th Mediterranean Conference on , vol., no., pp.229,234, 3-6 July 2012.

[2] J. C. Lins Barreto S, A. G. Scolari Conceicao, C. E. T. Dorea, L. Martinez, E. R. De Pieri, "Design and Implementation of Model-Predictive Control With Friction Compensation on an Omnidirectional Mobile Robot," *Mechatronics, IEEE/ASME Transactions on* , vol.19, no.2, pp.467,476, April 2014

[3] E. Hashemi, M. G. Jadidi, B. Babarsad, "Trajectory planning optimization with dynamic modeling of four wheeled omni-directional mobile robots," in *Proc* CIRA *2009 IEEE Int. Symp*, pp. 272-277.

[4] Zhiwen Zeng; Huimin Lu; ZhiQiang Zheng, "High-speed trajectory tracking based on model predictive control for omni-directional mobile robots," in *Proc. 25th Control and Decision Conf.*, Chinese, 2013, pp. 3179-3184.

[5] R. Oftadeh, R. Ghabcheloo, J. Mattila, "Time optimal path following with bounded velocities and accelerations for mobile robots with independently steerable wheels," in *Proc. IEEE Int. Conf. Robotics and Automation*, 2014, pp. 2925-2931.

[6] A. Savitzky, M.J.E. Golay, "Smoothing and differentiation of data by simplified least-squares procedures", *Anal. Chem.*, 36 (8) (1964), pp. 1627-1639.

[7] H. P. Oliveira, A. J. Sousa, A. P. Moreira and P. J. Costa, "Modeling and Assessing of Omni-Directional Robots with Three and Four Wheels", in *Contemporary Robotics - Challenges and Solutions*, A D Rodi (Ed.), 2009.

[8] G.K. Lowe, M.A. Zohdy, "Modeling nonlinear systems using multiple piecewise linear equations", *Nonlinear Analysis: Modelling and Control: Issues*, vol. 15, no. 4, pp. 451–458, 2010.

[9] Horowitz, I., *Synthesis of Feedback Systems*, Academic Press, New York, 1963

[10] Houpis, C.H., and S.J. Rasmussen. Quantitative Feedback Theory: Fundamentals and Applications. NY, USA: Marcel Dekker, 1999.

[11] C. Borghesani, Y. Chait, O. Yaniv: "QFT Frequency Domain Control Design Toolbox", 2003 Available: http://pcsl.ecs.umass.edu/cbs/software.html

[12] M. Garcia-Sanz, A. Mauch, C. Philippe, The QFT Control Toolbox (QFTCT) for Matlab, European Space Agency ESA-ESTEC. Public University of Navarra, Case Western Reserve University, 2008-2014. Available: http://cesc.case.edu.

[13] J. M. Díaz, S. Dormido, J. Aranda, "An Interactive Software Tool for Learning Robust Control Design Using Quantitative Feedback Theory Methodology", International Journal of Engineering Education, Volume 23, Number 5, October 2007, pp. 1011-1023(13). Tool link: http://www.uned.es/itfe/QFTIT/

[14] M. Gil-Martínez and M. García-Sanz, "Simultaneous meeting of robust control specifications in QFT", Int. J. Robust Nonlinear Control 2003; 13:643-656