Control of Mobile Robot Considering Actuator Dynamics with Uncertainties in the Kinematic and Dynamic Models

Nardênio A. Martins, Douglas W. Bertol, Edson R. De Pieri, and Eugênio B. Castelan

Automation and Systems Department, Federal University of Santa Catarina,
P.O. Box 476, 88040-900, Florianópolis, Santa Catarina, Brazil
{nardenio, dwbertol, edson, eugenio}@das.ufsc.br

Abstract. In this paper, a trajectory tracking control for a nonholonomic mobile robot by the integration of a neural kinematic controller (NKC) and neural dynamic controller (NDC) is investigated, where the wheel actuator (e.g., dc motor) dynamics is integrated with mobile robot dynamics and kinematics so that the actuator input voltages are the control inputs, as well as both the kinematic and dynamic models contains parametric and/or nonparametric uncertainties. The proposed neural controller (PNC) is constituted of the NKC and the NDC, and were designed by use of a modelling technique of Gaussian radial basis function neural networks (RBFNNs). The NKC is applied to compensate the uncertainties in the kinematic parameters of the mobile robot. The NDC, based on the sliding mode theory, is applied to compensate the mobile robot dynamics, and parametric and/or nonparametric uncertainties. Also, the PNC are not dependent of the mobile robot kinematics and dynamics neither require the off-line training process. Stability analysis with basis on Lyapunov theory and numerical simulation is provided to show the effectiveness of the PNC.

Keywords: Mobile robot, trajectory tracking, actuator dynamics, uncertainties, kinematic control, dynamic control, neural networks, Lyapunov theory.

1 Introduction

Several control methods have been proposed for motion control of a mobile robot under nonholonomic constraints [1]-[13]. Some researches consider the dynamics of mobile robot to achieve 'perfect velocity tracking' [1], [7]. However, in these methods the perfect knowledge about the parameter values of mobile robot is necessary. Usually, such requirement is unattainable. In practical situations, obtaining exact parameter values on a mobile robot is almost impossible. There are some results on the problems, regarding the integration of both, kinematic and neural dynamic controllers for a mobile robot with uncertainties and/or disturbances in the dynamics [2], [4], [6], [8], [10], [11]. These results are based on the assumption that the kinematics of the system is exactly known or by selecting a special control target (i.e. the forward

S. Omatu et al. (Eds.): IWANN 2009, Part II, LNCS 5518, pp. 1256–1263, 2009.

velocity and the angular velocity of the robot) to avoid this problem. However, in practice, there are uncertainties in both kinematics and dynamics.

As there are few works [3], [5], [9] dealing of both the unknown dynamics and unknown kinematics, the PNC that addresses the problem of integration of the NKC and the NDC (based on the sliding mode theory), considering the presence of parametric and/or nonparametric uncertainties in the kinematic and dynamic models.

In this paper, the wheel actuator (e.g., dc motor) dynamics is integrated with mobile robot dynamics and kinematics so that the actuator input voltages are the control inputs (is more realistic), and differently from other investigations with neural networks in the control of mobile robots [2], [4], [6], [8], [10], [11], the others contributions are: the implementation of the PNC (NKC plus NDC) based on the partitioning of the RBFNN into several smaller subnets in order to obtain more efficient computation; the PNC neither require the knowledge of the mobile robot kinematics and dynamics nor the time-consuming training process; the stability analysis and convergence of the mobile robot control system, and the learning algorithms are proved by using Lyapunov theory.

2 Kinematics and Dynamics of a Mobile Robot

The mobile robot's position (Figure 1) is given by the posture vector and $q = [x_C \ \ y_C \ \ \theta]^T$, containing the center of mass (guidance point) C coordinates and the heading angle θ, with P, d, r, and $2R$ being intersection of the axis of symmetry with the driving wheel axis, distance from the point C to the point P, radius of wheels, and distance between wheels, respectively.

Without considering the effect of surface friction $\bar{F}(\dot{q})$ and gravitational torques $\bar{G}(q) = 0$, the dynamic equations of the nonholonomic mobile robot for control purposes are:

$$\dot{q} = S(q)v\,, \quad \bar{H}(q)\dot{v} + \bar{C}(q,\dot{q})v + \bar{\tau}_d = \bar{\tau}\,, \tag{1}$$

where the properties are maintained, as well as matrices, vectors, and variables are defined as in [2].

Neglecting motor inductance in the electrical part of the actuator [12], the equations governing the actuator motor can be written as:

$$\tau_m = K_T i\,, \quad u = R_a i + K_b \dot{\phi}_m\,, \tag{2}$$

where τ_m is the torque generated by the motor, K_T is the motor torque constant, i is the current, u is the actuator input voltage, R_a is the resistance, K_b is the counter electromotive force coefficient, and $\dot{\phi}_m$ is the velocity of the actuator motor.

The angular velocity of the actuator motor, $\dot{\phi}_m$, and the corresponding wheel angular velocity v are related by gear ratio N as

$$v = \dot{\phi}_m / N\,, \tag{3}$$

and the motor torque τ_m is related to the wheel torque τ as:

$$\tau = N\tau_m\,. \tag{4}$$

Using (2)-(4), the mobile robot dynamics equation (including actuator dynamics) can be written as:

$$\bar{H}(q)\dot{v} + \bar{C}(q,\dot{q})v + \bar{\tau}_d = (NK_T/R_a)\bar{B}u - (N^2K_TK_b/R_a)\bar{B}v = T\,. \tag{5}$$

From (1), the posture kinematic model of a mobile robot with differential drive can be represented in the cartesian coordinates system, in the matricial form, as:

$$\dot{q} = \begin{bmatrix}\dot{x}\\ \dot{y}\\ \dot{\theta}\end{bmatrix} = \begin{bmatrix}\frac{r}{2R}(R\cos\theta - d\sin\theta) & \frac{r}{2R}(R\cos\theta + d\sin\theta)\\ \frac{r}{2R}(R\sin\theta + d\cos\theta) & \frac{r}{2R}(R\sin\theta - d\cos\theta)\\ \frac{r}{2R} & -\frac{r}{2R}\end{bmatrix}\begin{bmatrix}\omega_r\\ \omega_l\end{bmatrix}, \tag{6}$$

where $v = [\omega_r \quad \omega_l]^T$ is angular velocity of the wheels. This kinematic model (8) is obtained from the description of the angular velocity of the wheels in function of the linear and angular velocities (v_L, ω_A) of the mobile robot through the following relationship:

$$\begin{bmatrix}\omega_r\\ \omega_l\end{bmatrix} = \underbrace{\begin{bmatrix}1/r & R/r\\ 1/r & -R/r\end{bmatrix}\tan\theta\ \cot\theta}_{D(q)}\begin{bmatrix}v_L\\ \omega_A\end{bmatrix}. \tag{7}$$

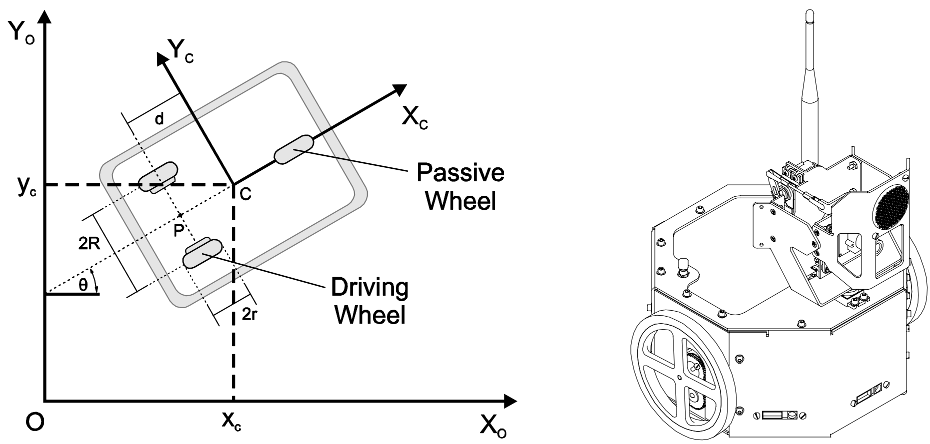

Fig. 1. Model of a nonholonomic mobile robot

3 Neural Networks Modelling by RBFNNs

Based on (5), it can be verified that $\bar{H}(q)$ is function of q only, and $\bar{C}(q,\dot{q}) = \bar{C}(z)$ is a function of q and $\dot{q}$, thus, static neural networks and dynamic neural networks are enough to model them, respectively. Therefore, the stability of the neural networks can be analyzed, where matrix Ge-Lee (GL) [13], defined by $\{.\}$, and its product operator '•' are used. The ordinary matrix and vector are denoted by $[.]$.

Foregrounded in (5), the matrices $\bar{H}(q)$ and $\bar{C}(z)$ of the mobile robot dynamics can be expressed by:

$$\overline{H}(q) = \lfloor \{W_{\overline{H}}\}^T \bullet \{\xi_{\overline{H}}(q)\} \rfloor + E_{\overline{H}}(q)\ , \ \overline{C}(z) = \lfloor \{W_{\overline{C}}\}^T \bullet \{\xi_{\overline{C}}(z)\} \rfloor + E_{\overline{C}}(z)\ , \tag{8}$$

where $\{W_{\overline{H}}\}$, $\{\xi_{\overline{H}}(q)\}$, $\{W_{\overline{C}}\}$, and $\{\xi_{\overline{C}}(z)\}$ are GL matrices, and their respective elements are $W_{\bar{h}_{kj}}$, $\xi_{\bar{h}_{kj}}(q)$, $W_{\bar{c}_{kj}}$, and $\xi_{\bar{c}_{kj}}(z)$. $E_{\overline{H}}(q) \in R^{n\times n}$ and $E_{\overline{C}}(z) \in R^{n\times n}$ are matrices, and their modelling error elements $\varepsilon_{\bar{h}_{kj}}(q)$ and $\varepsilon_{\bar{c}_{kj}}(z)$, respectively.

It is important to emphasize that a $P(.)$ vector can be modeled with static neural networks, since it is a function of a variable only. Thus, $P(.)$ results in:

$$P(.) = \lfloor \{W_P\}^T \bullet \{\xi_P(.)\} \rfloor + E_P(.)\ , \tag{9}$$

where $\{W_P\}$ and $\{\xi_P(.)\}$ are GL vectors, and their respective elements are W_{p_k} and $\xi_{P_k}(.)$. $E_p(.) \in R^n$ is a vector, and their modelling error elements $\varepsilon_{p_k}(.)$.

4 Neural Control of the Kinematic Model

Let v_s be a selected smooth velocity input, that achieves tracking for (6), given by:

$$v_s = \begin{bmatrix} v_f \\ \omega_f \end{bmatrix} = \begin{bmatrix} v_d \cos e_3 + k_1 e_1 \\ \omega_d + k_2 v_d e_2 + k_3 v_d \sin e_3 \end{bmatrix}, \tag{10}$$

where $v_d > 0$ for all t is the reference linear velocity; ω_d is the reference angular velocity; k_1, k_2, and k_3 are positive parameters; and $e_q = [e_1 \quad e_2 \quad e_3]^T$ is the position tracking error vector [14].

From kinematics, (6), if the parameters, r, R, and d are unknown, it is possible to use the estimates of these parameters in (7) and design learning algorithms for the neural controller to compensate these parameters. Assume:

$$a_1(q) = \frac{d}{rd} D(q) = \frac{1}{r} D(q)\ , \quad b_1(q) = \frac{Rd}{rd} D(q) = \frac{R}{r} D(q)\ . \tag{11}$$

Replacing the linear and angular velocities in (7) with (10), and using (9) with $E_p(.) = 0$, assume that the $\hat{A}(q)$ and $\hat{B}(q)$ vectors of (12) can be modeled with static neural networks as:

$$v_c = \begin{bmatrix} v_{c1} \\ v_{c2} \end{bmatrix} = \begin{bmatrix} \omega_r \\ \omega_l \end{bmatrix} = \underbrace{\begin{bmatrix} \hat{a}_1(q) \\ \hat{a}_1(q) \end{bmatrix}}_{\hat{A}(q)} v_f + \underbrace{\begin{bmatrix} \hat{b}_1(q) \\ -\hat{b}_1(q) \end{bmatrix}}_{\hat{B}(q)} \omega_f \ \underbrace{\begin{bmatrix} \widehat{W}_{A1}^T \xi_{A1}(q) \\ \widehat{W}_{A2}^T \xi_{A2}(q) \end{bmatrix}}_{\hat{A}(q)} v_f + \underbrace{\begin{bmatrix} \widehat{W}_{B1}^T \xi_{B1}(q) \\ \widehat{W}_{B2}^T \xi_{B2}(q) \end{bmatrix}}_{\hat{B}(q)} \omega_f\ , \tag{12}$$

where $\hat{a}_1(q)$ and $\hat{b}_1(q)$ are the estimates of $a_1(q)$ and $b_1(q)$; $\tilde{a}_1(q) = a_1(q) - \hat{a}_1(q)$ and $\tilde{b}_1(q) = b_1(q) - \hat{b}_1(q)$ are the estimated errors respectively; $\widehat{W}_{A1}^T \xi_{A1}(q) = \widehat{W}_{A2}^T \xi_{A2}(q)$, $\widehat{W}_{B2}^T \xi_{B2}(q) = -\widehat{W}_{B1}^T \xi_{B1}(q)$, and the wheel angular velocity is defined as the velocity control input v_c.

To design the actuator input voltages, one defines the auxiliary velocity tracking error as:

$$e_c = v_c - v = \begin{bmatrix} v_{c1} - \omega_r \\ v_{c2} - \omega_l \end{bmatrix}. \tag{13}$$

Consider the Lyapunov function candidate in the form:

$$V_1 = \frac{1}{2}\left(e_1^2 + e_2^2 + 2\,\frac{(1-\cos e_3)}{k_2} + \frac{\left(\widetilde{W}_{A1}^T \xi_{A1}(q)\right)^2}{\gamma_1 W_{A1}^T \xi_{A1}(q)} + \frac{\left(\widetilde{W}_{B1}^T \xi_{B1}(q)\right)^2}{\gamma_2 W_{B1}^T \xi_{B1}(q)}\right), \tag{14}$$

with $\gamma_1 > 0, \gamma_2 > 0$.

The parameter learning rules are defined as:

$$\dot{\widehat{W}}_{A1} = \gamma_1 e_1 v_f \left(\xi_{A1}^+(q)\right)^T, \quad \dot{\widehat{W}}_B = \frac{\gamma_2 \sin e_3}{k_2}\,\omega_f \left(\xi_{B1}^+(q)\right)^T, \tag{15}$$

where $\xi_.^+(.)$ is a pseudoinverse matrix.

Performing mathematical manipulations [14], $\dot{V}_1$ becomes:

$$\dot{V}_1 = -k_1 e_1^2 - \frac{k_3 v_d \sin^2 e_3}{k_2}. \tag{16}$$

Apparently, $\dot{V}_1 \le 0$, $\dot{V}_1 = 0$ if only if, $e_1 = 0$ and $e_3 = 0$. Moreover, (10) and (12) show that $v_{c1} = \omega_r$ and $v_{c2} = \omega_l$ are bounded. More details can be found in [14].

5 Neural Control of the Dynamic Model

Let Λ be a symmetric diagonal positive definite matrix, one defines:

$$\begin{aligned} v_r &= v_c + \Lambda \int_0^t e_c\,dt, \quad \dot{v}_r = \dot{v}_c + \Lambda e_c, \\ s &= v_r - v = e_c + \Lambda \int_0^t e_c\,dt, \quad \dot{s} = \dot{v}_r - \dot{v} = \dot{e}_c + \Lambda e_c, \end{aligned} \tag{17}$$

where s is a filtered tracking error term, $\int_0^t e_c\,dt$ is an auxiliary position tracking error, which does not reflect the position tracking error e_q directly, besides not having physical meaning.

One defines the control input to be of the form:

$$\begin{aligned} u &= (R_a/NK_T)\bar{B}^{-1}\left(\left[\{\widehat{W}_{\bar{H}}\}^T \bullet \{\xi_{\bar{H}}(q)\}\right]\dot{v}_r + \left[\{\widehat{W}_{\bar{C}}\}^T \bullet \{\xi_{\bar{C}}(z)\}\right]v_r + \rho - \gamma\right), \\ \rho &= (N^2 K_T K_b/R_a)\bar{B} v_r, \quad \gamma = -G\tanh(\delta s) - (Q + I_n)s, \end{aligned} \tag{18}$$

where $\{\widehat{W}_{\bar{H}}\}$, and $\{\widehat{W}_{\bar{C}}\}$ represent estimates of true parameters of matrices $\{W_{\bar{H}}\}$, and $\{W_{\bar{C}}\}$ of (8), and γ is the robustifying term with the aim of compensating for the bounded unknown disturbances, with $G^T = G > 0$, $\delta > 0$, $(Q + I_n)^T = (Q + I_n) > 0$, and I_n is identity matrix. The $\tanh(.)$ function is adopted for the purpose of deriving closed-loop stability.

Let us consider Lyapunov function candidate:

$$\begin{aligned} V_2 = \frac{1}{2}&\left(s^T \bar{H}(q)s + \sum_{k=1}^n \widetilde{W}_{\bar{H}k}^T \Gamma_{\bar{H}k}^{-1} \widetilde{W}_{\bar{H}k} + \sum_{k=1}^n \widetilde{W}_{\bar{C}k}^T \Gamma_{\bar{C}k}^{-1} \widetilde{W}_{\bar{C}k}\right) + \\ &+ \left(\int_0^t e_c\,dt\right)^T \Lambda \int_0^t e_c\,dt, \end{aligned} \tag{19}$$

being $\Gamma_{.k}$ are dimensional compatible symmetric positive definite matrices, and $\{\widetilde{W}_{.k}\} = \{W_{.k}\} - \{\widehat{W}_{.k}\}$ is parameter error. Clearly, $V_2 \geq 0$, and $V_2 = 0$ if only if $e_c = 0$, $\int_0^t e_c\, dt = 0$, $s = 0$, and $\{\widetilde{W}_{.}\} = 0$.

The parameter learning laws of neural networks are chosen as:

$$\begin{aligned} \dot{\widehat{W}}_{\bar{H}_k} &= \Gamma_{\bar{H}_k} \bullet \{\xi_{\bar{H}_k}(q)\}\dot{v}_r s_k - K_{\bar{H}_k}\Gamma_{\bar{H}_k}\|s\|\widehat{W}_{\bar{H}_k}\,, \\ \dot{\widehat{W}}_{\bar{C}_k} &= \Gamma_{\bar{C}_k} \bullet \{\xi_{\bar{C}_k}(z)\}v_r s_k - K_{\bar{C}_k}\Gamma_{\bar{C}_k}\|s\|\widehat{W}_{\bar{C}_k}\,, \end{aligned} \tag{20}$$

with $K_{.k} = K_{.} > 0$ are positive constants.

After the necessary mathematical manipulations [15], $\dot{V}_2$ stays:

$$\begin{aligned} \dot{V}_2 \leq -\|e_c\|^2 - \|s\|^2 Q_{\min} - (N^2 K_T K_b / R_a) s^T \bar{B} s - \\ -\|s\|\big(G_{\min}\|\tanh(\delta s)\| - (b_d + e_{NN} + \chi)\big) - \left(\int_0^t e_c\, dt\right)^T \Lambda^T \Lambda \int_0^t e_c\, dt\,, \end{aligned} \tag{21}$$

with $Q_{\min}$, and $G_{\min}$ are the minimum singular values of Q and G, respectively. Because of $\|\tanh(\delta s)\| > \frac{b_d + e_{NN} + \chi}{G_{\min}}$, $\dot{V}_2$ is guaranteed negative.

To ensure that the global system is stable, the Lyapunov function candidate is given as $V = V_1 + V_2$, where V_1 and V_2 refers to (14) and (19) respectively. Since $\dot{V}_1$ in (16) and $\dot{V}_2$ in (21) are guaranteed to be negative, then $\dot{V}$ is also guaranteed negative. More details can be found in [15].

6 Simulations Results

In the realization of the simulations, the kinematic and the dynamic (including actuator dynamics) models described in [16] are used. The model parameters of the prototype wheeled mobile robot estimated in [17] are: $m = 11.0$ kg, $I = 1.057$ kgm^2, $R = 0.265$ m, $r = 0.125$ m, $d = 0.1$ m, $N = 21$, $K_T = [0.057 \quad 0.051]^T$ Vs, $K_b = [0.057 \quad 0.051]^T$ Vs, and $R_a = [0.476 \quad 0.233]^T$ Ω. The reference trajectory is a straight line with initial coordinates $[x_r(0), y_r(0), \theta_r(0)] = [1,2,25.56°]$. The initial position of the robot is $[x_c(0), y_c(0), \theta_c(0)] = [2,1,10°]$. The parameters of the NKC are chosen as $k_1 = 1$, $k_2 = 3$, $k_3 = 7$, $\gamma_1 = 35$, $\gamma_2 = 5$, and $\sigma_{.} = 7$; and the gains of the NDC as $\Lambda = \mathrm{diag}[2]$, $\Gamma_{.k} = 0.1$, $\sigma_{.} = 3$, $K_{.} = 0.001$, $G = \mathrm{diag}[1]$, $Q = \mathrm{diag}[1]$.

A Coulomb friction and a bounded periodic disturbance term are added to the robot system as:

$$\bar{\tau}_d = \left[\big(f_1 + f_1(t)\big)\mathrm{sgn}(\omega_r) + 0.1\sin 2\pi t \quad \big(f_2 + f_2(t)\big)\mathrm{sgn}(\omega_l) + 0.1\cos 2\pi t\right]^T, \tag{22}$$

where $f_1 = 1.0$, $f_2 = 1.0$. Function $f(t)$ is nonlinear, defined as: $[f_1(t) \quad f_2(t)] = [0.0 \quad 0.0]^T$ if $t < 3$; $[f_1(t) \quad f_2(t)] = [0.0 \quad 2.0]^T$ if $3 \leq t \leq 8$; $[f_1(t) \quad f_2(t)] = [2.0 \quad 2.0]^T$ if $t \geq 8$, respectively. Thus, $\bar{\tau}_d$ is subject to a sudden change at time goes to 3s and 8s. Moreover, in 8s, the mobile robot suddenly dropped of an object of 30.0 kg, that is, a quarter of its original mass.

The tracking performance of the PNC can be observed in the: Figure 2, since the mobile robot naturally describes a smooth path tracking over the desired trajectory and that the tracking errors tend to zero; Figure 3 that the robot velocities tend to desired values and that shows the behavior of the wheel actuator input voltages.

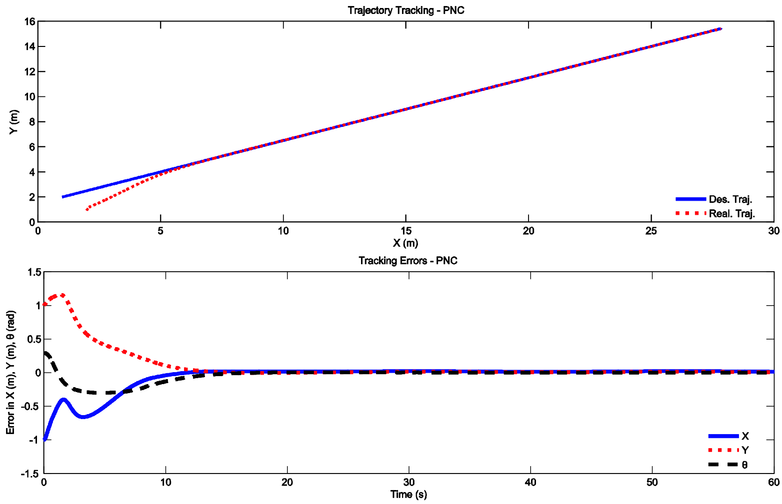

Fig. 2. PNC - Trajectory tracking and tracking errors

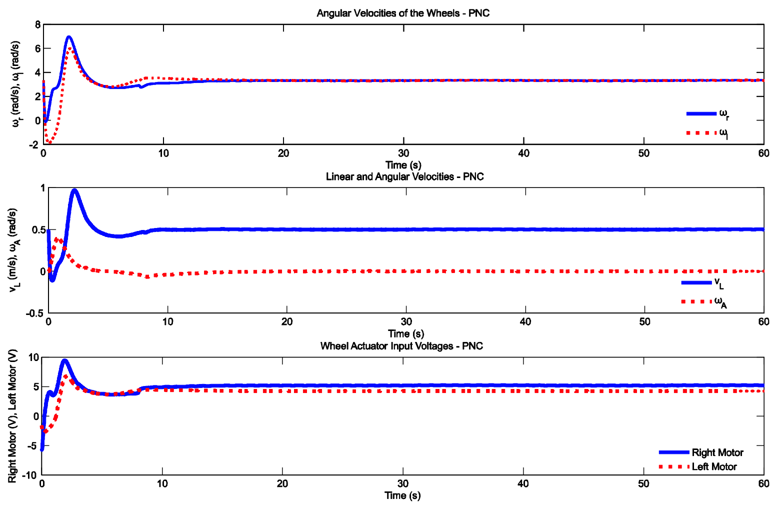

Fig. 3. PNC - Velocities and wheel actuator input voltages

References

1. Fierro, R., Lewis, F.L.: Control of a Nonholonomic Mobile Robot: Backstepping Kinematics into Dynamics. In: Proc. Conf. on Decision and Control, pp. 3805–3810 (1995)
2. Fierro, R., Lewis, F.L.: Control of a Nonholonomic Mobile Robot using Neural Networks. IEEE Trans. on Neural Networks 9(4), 589–600 (1998)
3. Fukao, T., Nakagawa, H., Adachi, N.: Adaptive Tracking Control of a Nonholonomic Mobile Robot. IEEE Trans. on Robotics and Automation 16(5), 609–615 (2000)

4. de Oliveira, V.M., De Pieri, E.R., Lages, W.F.: Feedforward Control of a Mobile Robot using a Neural Network. In: Proc. IEEE Int. Conf. on Systems, Man, and Cybernetics, pp. 3342–3347 (2000)
5. Hu, T., Yang, S.X., Wang, F., Mittal, G.S.: A Neural Network Controller for a Nonholonomic Mobile Robot with Unknown Robot Parameters. In: Proc. IEEE Int. Conf. Robotics and Automation, pp. 3540–3545 (2002)
6. de Sousa Júnior, C., Hemerly, E.M., Galvao, R.K.H.: Adaptive Control for Mobile Robot using Wavelet Networks. IEEE Trans. on Systems, Man, and Cybernetics, Part B 32(4), 493–504 (2002)
7. Chwa, D.: Sliding Mode Tracking Control of Nonholonomic Wheeled Mobile Robots in Polar Coordinates. IEEE Trans. on Control Systems Technology 12(4), 637–644 (2004)
8. Oh, C., Kim, M.-S., Lee, J.-J.: Control of a Nonholonomic Mobile Robot using an RBF Network. Journal Artificial Life and Robotics 8(1), 14–19 (2004)
9. Dong, W., Kuhnert, K.-D.: Robust Adaptive Control of Nonholonomic Mobile Robot with Parameter and Nonparameter Uncertainties. IEEE Trans. Robotics 21(2), 261–266 (2005)
10. Liu, S., Yu, Q., Lin, W., Yang, S.X.: Tracking Control of Mobile Robots based on Improved RBF Neural Networks. In: Proc. IEEE/RSJ Int. Conf. Intel. Robots and Systems, pp. 1879–1884 (2006)
11. Peng, J., Wang, Y., Yu, H.: Neural Network based Robust Tracking Control for Nonholonomic Mobile Robot. In: Advances in Neural Networks. LNCS, pp. 804–812. Springer, Heidelberg (2007)
12. Mills, J.K.: Hybrid Actuator for Robot Manipulators: Design, Control and Performance. In: Proc. IEEE Int. Conf. Robotics and Automation, pp. 1872–1878 (1990)
13. Ge, S.S.: Robust Adaptive NN Feedback Linearization Control of Nonlinear Systems. Int. J. Systems Science, 1327–1338 (1996)
14. Martins, N.A., Bertol, D.W., Lombardi, W.C., De Pieri, E.R., Castelan, E.B.: Trajectory Tracking of a Nonholonomic Mobile Robot with Parametric and Nonparametric Uncertainties: A Proposed Neural Control. Int. J. of Factory Automation, Robotics and Soft Computing 2, 103–110 (2008)
15. Martins, N.A., Bertol, D.W., Lombardi, W.C., De Pieri, E.R., Castelan, E.B.: Neural Dynamic Controllers for the Trajectory Tracking of a Nonholonomic Mobile Robot Including the Actuator Dynamics. Int. J. of Factory Automation, Robotics and Soft Computing 1, 39–44 (2009)
16. Coelho, P., Nunes, U.: Path-Following Control of Mobile Robots in Presence of Uncertainties. IEEE Trans. Robotics 21(2), 252–261 (2005)
17. Huang, C.-L.: A motion platform of autonomous mobile robot. Master thesis, Department of Electrical Engineering, National Taiwan University, pp. 27–30 (2003)