DE GRUYTER OPEN

Journal of ELECTRICAL ENGINEERING, VOL 68 (2017), NO1, 39–46

Application of neural models as controllers in mobile robot velocity control loop

Jakub Cerkala, Anna Jadlovska *

This paper presents the application of an inverse neural models used as controllers in comparison to classical PI controllers for velocity tracking control task used in two-wheel, differentially driven mobile robot. The PI controller synthesis is based on linear approximation of actuators with equivalent load. In order to obtain relevant datasets for training of feed-forward multi-layer perceptron based neural network used as neural model, the mathematical model of mobile robot, that combines its kinematic and dynamic properties such as chassis dimensions, center of gravity offset, friction and actuator parameters is used. Neural models are trained off-line to act as an inverse dynamics of DC motors with particular load using data collected in simulation experiment for motor input voltage step changes within bounded operating area. The performances of PI controllers versus inverse neural models in mobile robot internal velocity control loops are demonstrated and compared in simulation experiment of navigation control task for line segment motion in plane.

K e y w o r d s: mathematical model, mobile robot, MLP neural network, direct-inverse control, posture control, DC motor

1 Introduction

In case of planar mobile robots, the differentially driven two-wheel chassis concept is usually used due to its simplicity, stability and high mobility. Almost always, the DC motors with gearboxes are used as an actuators and their design is based on load that mobile robot represents [15, 16]. The mobile robot control in general can be divided into two levels – the internal velocity control loop and navigation control for motion in plane. The main goal of the internal velocity control loop is to achieve best possible tracking of reference wheel angular velocities, which are the outputs of navigation control. In the case of ideal velocity tracking in both internal loops, the dynamic navigation control becomes a pure kinematic control task [2, 9, 17, 18]. Dynamic properties of mobile robot, such as its moment of inertia or friction represents an undesirable impact on its movement properties [6, 7, 10, 19]. The non-ideal reference velocities tracking results into a position error, that is subject of minimization for navigation control [3, 4, 8]. To deal with the velocities tracking problem in internal control loops, the common practice is to use PI controllers, because they can be simply implemented and their design rely on known or approximated mobile robot parameters. However, if the robot parameters are unknown or their approximation is not accurate, the controller synthesis based on analytical model may not be suitable. As an alternative, the controller design may be based on experimentally obtained data. One of the possible approaches is to use neural networks [11–14] trained as an inverse models of mobile robot dynamics corresponding to each wheel [1, 9]. The data for neural network training can be experimentally obtained by measurements on an existing mobile robot, but in order to verify this controller design approach in the early prototyping phase, it may be better to use the sufficiently accurate mobile robot simulation model as an source of training data. The biggest advantage of this procedure is that controller verified in simulation can be directly used as it is in real robot, or the whole training procedure can be repeated with data measured on a real robot.

The main focus of this paper is to present the application possibilities of neural networks trained as inverse neural models used in form of independent wheel angular velocity controllers for the two-wheel, differentially driven mobile robot in comparison to classical PI controllers [16]. In order to achieve relevant training data, the mathematical model that forms the base of simulation model in addition to general dynamic properties includes dynamical effects of actuators and viscous friction [7, 19]. The training data are measured in simulation experiment on a single wheel subsystem and trained neural networks are used in direct-inverse neural control structure for internal wheel velocity tracking loops. To verify and compare performance of classical PI and inverse neural controllers in internal velocity control loop, reference velocities are computed in navigational control loop for motion in plane on a line segment [3, 8].

2 Mobile robot model

The overall mathematical model of mobile robot with two-wheel, differentially driven chassis may be divided into three logical parts – the kinematic model of chassis,

* Department of Cybernetics and Artificial Intelligence, Faculty of Electrical Engineering and Informatics, Technical University of Košice, Slovakia, jakub.cerkala@tuke.sk, anna.jadlovska@tuke.sk

DOI: 10.1515/jee-2017-0005, Print (till 2015) ISSN 1335-3632, On-line ISSN 1339-309X

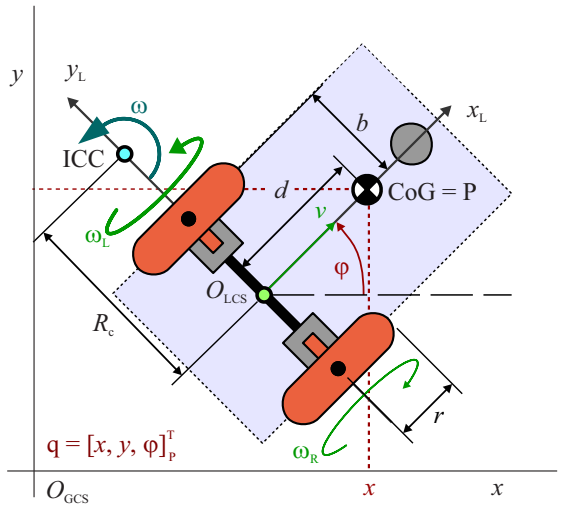

Fig. 1. Mobile robot posture and velocities definition in GCS

the dynamical model of mobile robot relevant to its center of gravity (COG) and to the combined dynamics of DC motors with gearboxes and viscous friction in bearings.

2.1 Kinematic model of chassis

The mobile robot chassis posture in respect to Cartesian global coordinate system (GCS) is depicted in Fig. 1 The position vector $\boldsymbol{q}$ defines the robot posture in GCS plane and velocities vector $\boldsymbol{v}$ are defined as

$$\boldsymbol{q} = [\,x \quad y \quad \varphi\,]^\top, \qquad \boldsymbol{v} = [\,v \quad \omega\,]^\top \tag{1}$$

where x, y is the CoG and reference point P position, φ is angular orientation of robot local coordinate system (LCS) in respect to GCS, v is linear and ω is angular velocity components of overall robot motion in plane. The kinematic model of mobile robot is expressed as [9]

$$\dot{\boldsymbol{q}} = \mathbf{S}(\boldsymbol{q})\boldsymbol{v} \implies \begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{\varphi} \end{bmatrix} = \begin{bmatrix} \cos\varphi & -d\sin\varphi \\ \sin\varphi & d\cos\varphi \\ 0 & 1 \end{bmatrix} \begin{bmatrix} v \\ \omega \end{bmatrix} \tag{2}$$

where d is CoG offset from the LCS origin O_{LCS} defined in positive direction of X_L. Kinematic model (2) satisfies the kinematic constraint $\mathbf{A}(\boldsymbol{q}) \in \mathbb{R}^{1\times 3}$ for reference point P motion without lateral slip in form of [17]

$$\mathbf{A}(\boldsymbol{q})\dot{\boldsymbol{q}} \stackrel{!}{=} 0 \implies -\dot{x}\cos\varphi + \dot{y}\sin\varphi - d\dot{\varphi} \stackrel{!}{=} 0\,. \tag{3}$$

The velocities $\boldsymbol{v}$ in (2) are defined for O_{LCS} in respect to an instantaneous center of curvature (ICC) – the center of an arc with changing radius R_{c} on which the robot moves. Since overall robot motion depends on independently mounted wheels, according to right and left wheel angular velocities defined in vector $\boldsymbol{\eta} = [\,\omega_R \quad \omega_L\,]^\top$ the robot overall velocities are obtained as

$$v = \frac{r}{2}(\omega_R + \omega_L)\,, \tag{4}$$

$$\omega = \frac{r}{2b}(\omega_R - \omega_L) \tag{5}$$

where b is half distance between wheels. Based on definition of (4) and (5), an inverse transformation exists – the conversion from overall velocities in $\boldsymbol{v}$ to wheel angular velocities $\boldsymbol{\eta}$ is defined as

$$\boldsymbol{\eta} = \mathbf{D}\boldsymbol{v} \implies \begin{bmatrix} \omega_R \\ \omega_L \end{bmatrix} = \begin{bmatrix} \frac{1}{r} & \frac{b}{r} \\ \frac{1}{r} & -\frac{b}{r} \end{bmatrix} \begin{bmatrix} v \\ \omega \end{bmatrix} \tag{6}$$

where $\mathbf{D} \in \mathbb{R}^{2\times 2}$ is velocity transformation matrix. The overall velocities v and ω are naturally bounded by dynamical properties of mobile robot and its actuators.

2.2 Base dynamic model of robot

To derive the dynamic model for mobile robot, the Newton-Euler approach can be used [5], but as an alternative, the Lagrange approach [2, 7] can be used as well, both approaches lead to same model for vectors $\boldsymbol{q}$ and $\boldsymbol{v}$ defined as in this paper. Based on the second Newton law of motion, the relationships between accelerations $\dot{\boldsymbol{v}}$ of CoG, its velocities $\boldsymbol{v}$ and actual input torques τ_R, τ_L are derived as a differential equations [5]

$$M\dot{v} - dM\omega^2 = \frac{1}{r}(\tau_R + \tau_L)\,, \tag{7}$$

$$(Md^2 + I_{\mathrm{c}})\dot{\omega} + Mdv\omega = \frac{b}{r}(\tau_R - \tau_L) \tag{8}$$

where M is robot overall mass including the wheels and motors, I_{c} is overall robot moment of inertia defined in center of gravity and d is CoG offset. The actual torques τ_i for both wheels $i = R, L$ in (7) and (8) are results of

$$\tau_i = \tau_{\mathrm{m}i} - (\tau_{\mathrm{f}i} - \tau_{\mathrm{z}i}) \tag{9}$$

where $\tau_{\mathrm{m}i}$ are torques, that actuators produce, $\tau_{\mathrm{f}i}$ is friction acting against rotation of particular wheel and external disturbance input is represented by $\tau_{\mathrm{z}i}$. For practical reasons, the equations (7) and (8) can be expressed in matrix form

$$\bar{\mathbf{M}}(\boldsymbol{q})\dot{\boldsymbol{v}} + \bar{\mathbf{V}}(\dot{\boldsymbol{q}}, \boldsymbol{q})\boldsymbol{v} + \bar{\mathbf{F}}\boldsymbol{v} = \bar{\mathbf{B}}(\boldsymbol{q})\boldsymbol{\tau}_{\mathrm{m}} - \bar{\mathbf{B}}(\boldsymbol{q})\boldsymbol{\tau}_{\mathrm{z}} \tag{10}$$

where $\bar{\mathbf{M}}(\boldsymbol{q}) \in \mathbb{R}^{2\times 2}$ is inertia matrix, $\bar{\mathbf{V}}(\dot{\boldsymbol{q}}, \boldsymbol{q}) \in \mathbf{R}^{2\times 2}$ is centripetal forces matrix, $\bar{\mathbf{B}}(\boldsymbol{q}) \in \mathbb{R}^{2\times 2}$ is input transformation matrix and the wheel viscous frictions $\tau_{\mathrm{f}R}, \tau_{\mathrm{f}L}$ are transformed in vector $\bar{\mathbf{F}}\boldsymbol{v} \in \mathbb{R}^{2\times 1}$, matrices are formed as

$$\begin{aligned} \bar{\mathbf{M}}(\boldsymbol{q}) &= \begin{bmatrix} M & 0 \\ 0 & Md^2 + I_{\mathrm{c}} \end{bmatrix}, \ \bar{\mathbf{B}}(\boldsymbol{q}) = \begin{bmatrix} \frac{1}{r} & \frac{1}{r} \\ \frac{b}{r} & -\frac{b}{r} \end{bmatrix} \\ \bar{\mathbf{V}}(\dot{\boldsymbol{q}}, \boldsymbol{q}) &= \begin{bmatrix} 0 & -Md\dot{\varphi} \\ Md\dot{\varphi} & 0 \end{bmatrix}, \ \bar{\mathbf{F}}\boldsymbol{v} = 2\frac{B_{\mathrm{e}}}{r^2} \begin{bmatrix} v \\ b^2\omega \end{bmatrix} \end{aligned} \tag{11}$$

The overall viscous friction coefficient B_{e} and actuator torques $\tau_{\mathrm{m}i}$ are characterized by the parameters of used DC motors, gearboxes and wheel bearings.

2.3 DC motor with gearbox and friction model

The electro-mechanical principal structure of DC motor with gearbox and attached load is depicted in Fig. 2. The DC motor, which is considered in this paper drives the robot via an ideal gearbox with input/output ratio of $1 : N$.

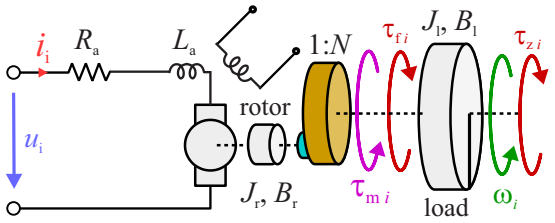

Fig. 2. Electro-mechanical scheme of single actuator

Motor torques produced by both motors are expressed as vector $\boldsymbol{\tau}_\mathrm{m}$ in load rotation axis

$$\boldsymbol{\tau}_\mathrm{m} = NK_\tau \boldsymbol{i} \implies \begin{bmatrix} \tau_{\mathrm{m}R} \\ \tau_{\mathrm{m}L} \end{bmatrix} = N \begin{bmatrix} K_\tau & 0 \\ 0 & K_\tau \end{bmatrix} \begin{bmatrix} i_R \\ i_L \end{bmatrix} \quad (12)$$

where K_τ is the motor torque constant and i_R, i_L are rotor coil currents. From the mechanical point of view, considered two moments of inertia can be taken into account – J_r for the motor rotor and J_l for the load attached after the gearbox. The overall equivalent moment of inertia J_e, which the motor have to overcome when changing velocity can be expressed in load axis according to gearbox ratio as

$$J_\mathrm{e} = N^2 J_r + J_l \,. \quad (13)$$

Similar equivalence apply for the friction or disturbance torque. When the rotor rotates in stator, the acting viscous friction can be computed by usually known coefficient B_r, but for viscous friction in load axis, the coefficient B_l is unknown because it depends on used gearbox and wheel bearings. The value of overall equivalent viscous friction coefficient B_e can be obtained experimentally. Since viscous friction can be expressed as a linear rising function, for steady-state velocities of load at constant voltage is the viscous friction equal to coil current.

The load attached to a single actuator can be approximated in form of a full wheel with radius r and half robot mass, based on the second Newton law of motion is the approximation defined as

$$J_\mathrm{e}\dot{\omega}_i + B_\mathrm{e}\omega_i = \tau_{\mathrm{m}i} - \tau_{\mathrm{z}i} \quad (14)$$

The relationship between motor input voltage u_i and rotor coil current i_i is expressed as differential equation

$$L_\mathrm{a}\frac{\mathrm{d}i_i}{\mathrm{d}t} + R_\mathrm{a} i_i + N\frac{60}{2\pi}K_e\omega_i = u_i \quad (15)$$

where L_a is rotor coil inductance, R_a is terminal resistance, K_e is constant for back electro-motive force and ω_i is angular velocity of load. Actuator dynamics according to (6) and (15) can be combined in matrix form

$$\begin{gathered} \mathbf{L}_\mathrm{a}\frac{\mathrm{d}}{\mathrm{d}t}\boldsymbol{i} + \mathbf{R}_\mathrm{a}\boldsymbol{i} + N\mathbf{K}_\tau\mathbf{D}\boldsymbol{v} = \boldsymbol{u} \\ \text{where } \boldsymbol{u} = \begin{bmatrix} u_R \\ u_L \end{bmatrix} \end{gathered} \quad (16)$$

while $K_\tau = \frac{60}{2\pi}K_e$. Based on (2), (6), (10), (12) and (16) is the state-space model in form of

$$\frac{\mathrm{d}}{\mathrm{d}t}\begin{bmatrix} \boldsymbol{q} \\ \boldsymbol{v} \\ \boldsymbol{i} \end{bmatrix} = \begin{bmatrix} \mathbf{0} & \mathbf{S} & \mathbf{0} \\ \mathbf{0} & -\bar{\mathbf{M}}^{-1}[\bar{\mathbf{V}} + \bar{\mathbf{F}}] & \bar{\mathbf{M}}^{-1}\bar{\mathbf{B}}N\mathbf{K}_\tau \\ \mathbf{0} & -\mathbf{L}_\mathrm{a}N\mathbf{K}_\tau\mathbf{D} & -\mathbf{L}_\mathrm{a}\mathbf{R}_\mathrm{a} \end{bmatrix} \begin{bmatrix} \boldsymbol{q} \\ \boldsymbol{v} \\ \boldsymbol{i} \end{bmatrix} + \begin{bmatrix} \mathbf{0} \\ \boldsymbol{0} \\ \mathbf{L}_\mathrm{a} \end{bmatrix} \boldsymbol{u} + \begin{bmatrix} \mathbf{0} \\ -\bar{\mathbf{M}}^{-1}\bar{\mathbf{B}} \\ \boldsymbol{0} \end{bmatrix} \boldsymbol{\tau}_\mathrm{z} \quad (17)$$

The state space model (17) is implemented in Simulink environment and used for simulational experiments.

3 Velocity control loop

As mentioned in introduction, the main goal for the robot internal control loops is the tracking of reference wheel velocities and transform the task of dynamic control to simpler kinematic control problem [17], internal control loops are illustrated in Fig. 3. The control errors $\boldsymbol{e}_\eta$ are defined in terms wheel angular velocities as

$$\boldsymbol{e}_\eta = \boldsymbol{\eta}_\mathrm{ref} - \boldsymbol{\eta} \implies \begin{bmatrix} e_R \\ e_L \end{bmatrix} = \begin{bmatrix} \omega_{R\mathrm{ref}} \\ \omega_{L\mathrm{ref}} \end{bmatrix} - \begin{bmatrix} \omega_R \\ \omega_L \end{bmatrix} \quad (18)$$

The first internal velocity control loop design approach uses classical PI controllers and the second is based on experimental data and neural models.

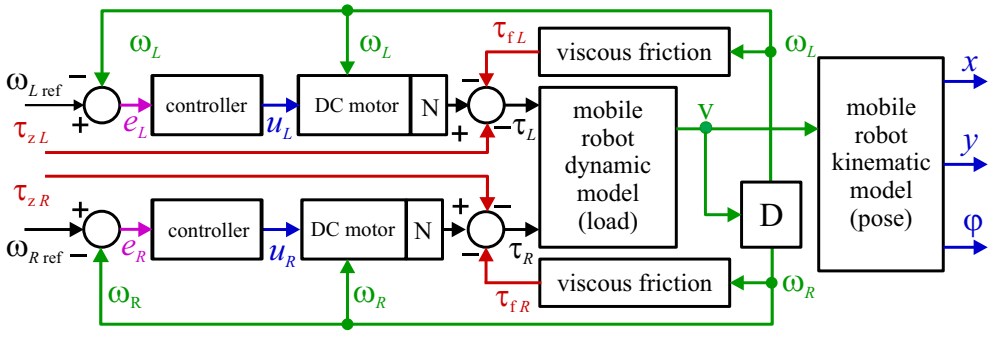

Fig. 3. Mobile robot internal velocity control loop with DC motors

3.1 PI controllers in feedback control loop

From the controller synthesis point of view, the mobile robot can be considered as a pair of motors, that move attached load. Using Laplace transformation, single DC motor with gearbox, viscous friction and attached equivalent load can be expressed as a linear transfer function

$$F_{\omega_i/u_i}(s) = \frac{\frac{NK_\tau}{J_\mathrm{e}L_\mathrm{a}}}{s^2 + \frac{J_\mathrm{e}R_\mathrm{a}+B_\mathrm{e}L_\mathrm{a}}{J_\mathrm{e}L_\mathrm{a}}s + \frac{B_\mathrm{e}R_\mathrm{a}+N^2K_\tau^2}{J_\mathrm{e}L_\mathrm{a}}} \tag{19}$$

while for the controller synthesis is it preferred in form of

$$F_{\omega_i/u_i}(s) = \frac{Z}{(T_1 s+1)(T_2 s+1)} \tag{20}$$

where Z is system gain and T_1, T_2 are time constants. Based on the transfer function (20), the parameters r_0, r_{-1} of PI controllers defined in continuous form

$$u_i(t) = r_0 e_i(t) + r_{-1}\int_0^t e_i(t)\mathrm{d}t \tag{21}$$

can be obtained using pole-placement method. Also, continuous controller (21) have to be discretized with an achievable sample rate Δt_PI in order to be implemented into mobile robot. Discrete PS controller action is in form

$$u_i(k) = u_i(k-1) + q_0 e_i(k) + q_1 e_i(k-1) \tag{22}$$

where q_0, q_1 are equivalent discrete controller parameters.

3.2 Direct-inverse neural control structure

An alternative approach of internal velocity control loop design is to use the inverse neural models as velocity controllers. A multi layer perceptron neural network (MLP) with delayed samples of angular velocities feed to input and predicted control action as output can be used, considered neural model is depicted in Fig. 4.

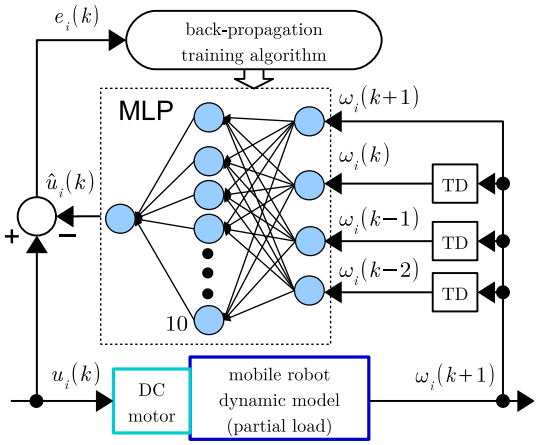

Fig. 4. Neural network model for inverse controller

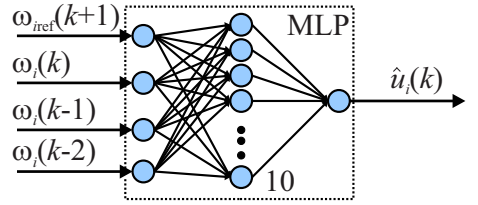

Fig. 5. Neural network general training setup

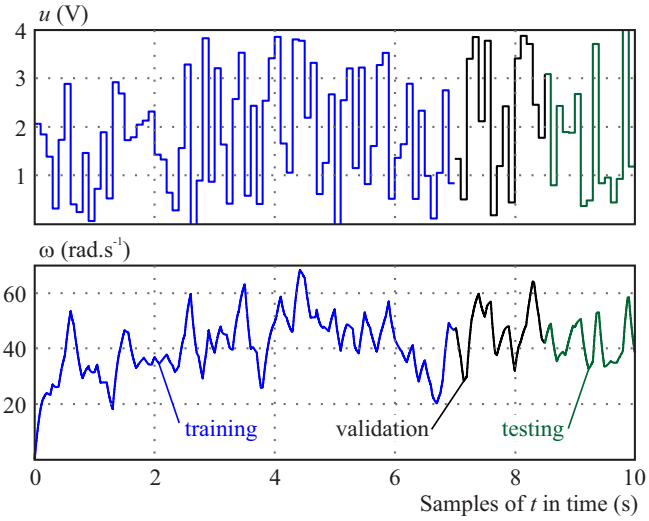

Fig. 6. Training, validation and testing dataset

The training of neural model can be realized for step changes of input voltage in range of expected operating area at Δt_INN sample rate, the neural MLP network [13] is connected in General Training structure to single motor as Fig. 5 shows.

An inverse neural model to DC motor with equivalent load is defined in form of predicted control as

$$\hat{\boldsymbol{u}}(k) = \hat{f}^{-1}\begin{bmatrix} \boldsymbol{w}(k+1) & \boldsymbol{\omega}(k) & \boldsymbol{\omega}(k-1) \\ \boldsymbol{\omega}(k-2) & \boldsymbol{u}(k-1) & \end{bmatrix} \tag{23}$$

where value $\boldsymbol{\omega}(k+1)$ known during training is replaced with reference velocity $\boldsymbol{w}(k+1)$, which then allows to use neural model (23) as a velocity controller. While offline training procedure, the Levemberg-Marquardt algorithm is used and the criterion

$$J_\mathrm{GT} = \frac{1}{2n}\sum_{k=1}^{n}[\boldsymbol{u}(k) - \hat{\boldsymbol{u}}(k)]^2 \qquad \longrightarrow \min \tag{24}$$

is minimized, dataset used for training is in Fig. 6. Since the robot is symmetric in its X_L axis – the load and friction is applied equally to both wheels, it is sufficient to train inverse neural model only on single wheel and use its copy for the second wheel.

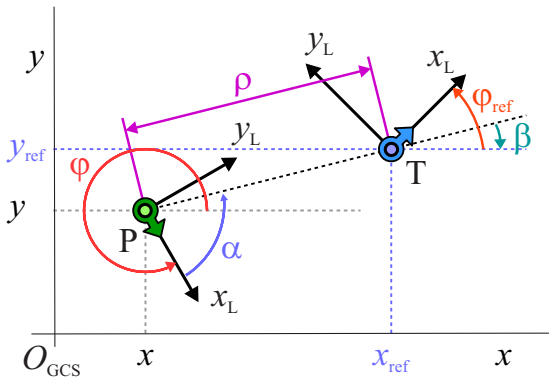

Fig. 7. Definition of polar posture errors

4 Navigation control loop

In order to compare both internal velocity control loops concepts, it is appropriate to use a plane navigation control structure as a source of relevant reference velocities. In this paper, the reference posture navigation control concept is used [3]. The mobile robot posture in plane is defined in its reference point P, which is chosen in robot center of gravity with. The reference posture Tq is characterized by position coordinates $x_{\mathrm{ref}}, y_{\mathrm{ref}}$ and final heading angle φ_{ref}. The navigation feedback control task is to ensure convergence of position errors $\boldsymbol{e_q}$ to zero as a function of time

$$\frac{\mathrm{d}\boldsymbol{e_q}}{\mathrm{d}t} + \boldsymbol{K}_{\mathrm{fb}}\boldsymbol{e_q} = 0 \tag{25}$$

where $\boldsymbol{K}_{\mathrm{fb}} \in \mathbb{R}^{2\times 3}$ is a matrix of appropriately chosen positive gains. The control input is required in form of

$$\mathbf{v}_{\mathrm{ref}} = \boldsymbol{K}_{\mathrm{fb}}\boldsymbol{e_q} \tag{26}$$

where values of $\boldsymbol{K}_{\mathrm{fb}}$ depends on the definition of errors in $\boldsymbol{e_q}$. From the practical aspect, it is convenient to express position errors in polar coordinates

$$\rho = \sqrt{(x_{\mathrm{ref}} - x)^2 + (y_{\mathrm{ref}} - y)^2}\,, \tag{27}$$

$$\alpha = \tan^{-1}\left(\frac{y_{\mathrm{ref}} - y}{x_{\mathrm{ref}} - x}\right) - \varphi + \pi\,, \tag{28}$$

$$\beta = \varphi_{\mathrm{ref}} - \alpha - \varphi \tag{29}$$

where ρ is euclidean distance between points P and T, α is heading error and β is error according to desired final orientation in point T, situation is illustrated in Fig. 7.

Based on the definition of polar errors in (27), (28), (29) and kinematic model of oriented point, the changes of polar errors in respect to overall robot velocities are

$$\begin{bmatrix}\dot\rho\\ \dot\alpha\\ \dot\beta\end{bmatrix} = \begin{bmatrix}-\cos\alpha & 0\\ \frac{1}{\rho}\sin\alpha & -1\\ -\frac{1}{\rho}\sin\alpha & 0\end{bmatrix}\begin{bmatrix}v\\ \omega\end{bmatrix}. \tag{30}$$

For navigation controller chosen in form of

$$v_{\mathrm{ref}} = k_\rho \rho\,, \tag{31}$$

$$\omega_{\mathrm{ref}} = k_\alpha \alpha + k_\beta \beta \tag{32}$$

is the minimization of position errors $\boldsymbol{e_q}$ in closed feedback loop characterized as

$$\begin{bmatrix}\dot\rho\\ \dot\alpha\\ \dot\beta\end{bmatrix} = \begin{bmatrix}-k_\rho\cos\alpha\\ k_\rho\sin\alpha - k_\alpha\alpha - k_\beta\beta\\ -k_\rho\sin\alpha\end{bmatrix} \tag{33}$$

while for the case of $\rho = 0$ there is no singularity and equilibrium occurs when

$$[\rho \quad \alpha \quad \beta]^\top = [0 \quad 0 \quad 0]^\top. \tag{34}$$

According to requirement for local exponential stability is using linearizion of (33) in equilibrium (34) possible to define limits [3]

$$k_\rho > 0\,, \qquad k_\beta < 0\,, \qquad k_\alpha - k_\rho > 0 \tag{35}$$

which need to be taken into account when choosing controller (31) gains [3]. In navigation control implementation, it is necessary to ensure that errors α and β will be always expressed in interval $\langle -\pi; \pi\rangle$. The sample rate may be different from Δt_{PI} or Δt_{INN}, but it depends on the maximal velocities of the robot.

5 Experimental results

The training of inverse neural models and simulation of navigation control is realized in Simulink environment. Mobile robot parameters used in simulation are listed in Tab. 1. The Faulhaber DC micromotor 2224006SR and ideal gearbox of ratio 12 : 80 serves as an actuators.

The coefficient of overall equal viscous friction B_{e} is identified by practical experiment, based on steady-state current for selected motor input voltage levels within operating range. The linear transfer function for the approximation of DC motor with equivalent load (19) used in PI controller synthesis is

$$F_{\omega_i/u_i}(s) = \frac{21.23}{(2.3216\times 10^{-5}s + 1)(0.1394s + 1)} \tag{36}$$

while poles, controller gains and used controller sample rates for internal control loops are summed in Tab. 2. The sample rate Δt_{INN} used for neural model training is also used as neural controller sample rate. For the navigation controller, that acts on different sample rate $\Delta t_{\mathrm{nav}} = 0.01\,\mathrm{s}$, the controller gains are chosen in respect to maximal expected position errors as

$$k_\rho = 2\,, \quad k_\alpha = 5\,, \quad k_\beta = -1 \tag{37}$$

while their values satisfy conditions in (35).

Table 1. Mobile robot parameters used in simulation

symbol	description	value	unit
r	wheel radius	0.025	m
b	half-distance between wheels	0.035	m
d	CoG X axis offset	0.005	m
M	overall robot mass	0.5	kg
I_c	overall robot moment of inertia	0.0005	kg.m^2

Table 2. Controller settings used in simulation

symbol	description	value	unit
p_1, p_2	poles used in PI synthesis	$-15 \pm 15j$	-
r_0	PI proportional gain	0.1598	s
r_{-1}	PI integral gain	2.9529	s
Δt_{PI}	PI controller sample	0.002	s
Δt_{INN}	INN controller sample	0.005	s

Table 3. Motor and controller limits used in simulation

symbol	description	value	unit
U_{limit}	DC motor voltage	$0-4$	V
I_{limit}	DC motor current	± 1	A
$v_{\mathrm{ref\ limit}}$	reference linear velocity	± 1	m.s^{-1}
$\omega_{\mathrm{ref\ limit}}$	reference angular velocity	± 3	rad.s^{-1}

In the case of DC motor models, internal velocity controllers and outputs of navigation controller, the considered limits are summed in Tab. 3.

To get a better illustration of used internal velocity tracking control loops properties, the navigation control task is the robot motion along a 1.5 m line segment, that origins in point $[0 \quad 0]$. The reference posture at time $t = 0$ moves along line by 0.5 m.s^{-1} constant speed with $\frac{\pi}{4}$ heading angle and after 3 seconds the reference posture stops, maintaining its heading direction. Both mobile robots, with different internal loops at $t = 0$ begins in the same posture $[0.5 \quad 0 \quad \pi]$ and their motion during a simulation experiment is depicted in Fig. 8.

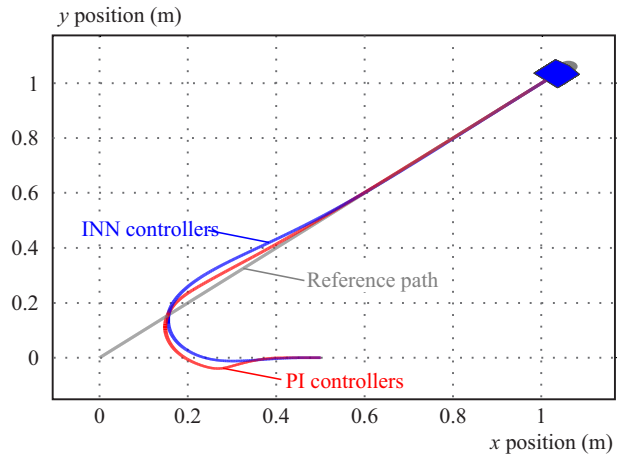

Fig. 8. Navigation control to line motion with stop

The motion of robots due navigation control can be divide into three phases – initial closing in phase, line following phase with a constant error ρ and slowing down to final position. Reference velocities $\mathbf{v}_{\mathrm{ref}}$ tracking in case of PI controllers based internal loop features a overshoot followed by undershoot, caused by the nature of selected complex conjugate poles p_1, p_2 and limits of $\mathbf{v}_{\mathrm{ref}}$, the real velocities tracking is shown in Fig. 9.

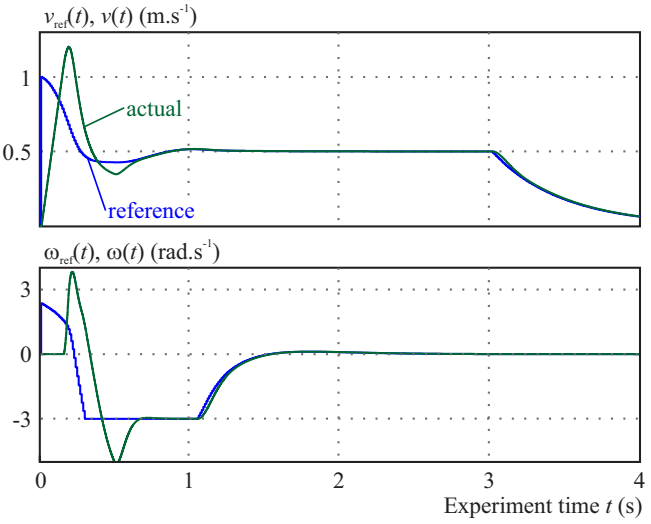

Fig. 9. Internal loop velocities tracking – PI controllers

On the contrary, in the case of internal control loop based on inverse neural controllers, the tracking of reference linear velocity v_{ref} is almost ideal, however there are oscillations around reference angular velocity ω_{ref} from the start, Fig. 10.

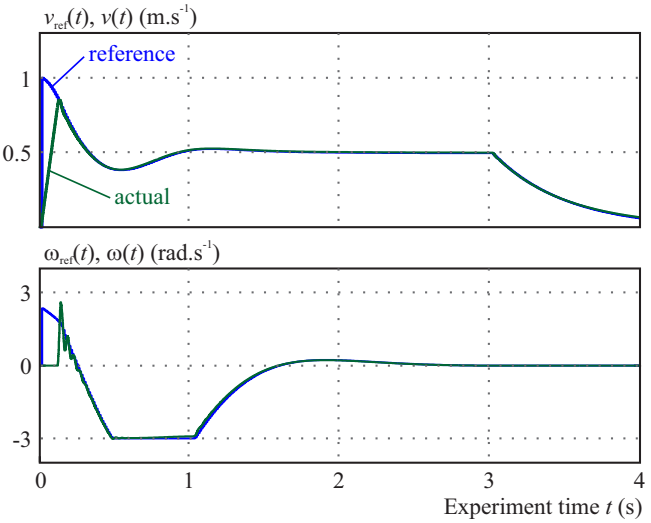

Fig. 10. Internal loop velocities tracking - INN controllers

The minimization of posture error with the same navigation controller is better in case of inverse neural controllers used in internal velocity control loop in comparison to PI controllers, the convergence of polar errors for both mobile robots is illustrated in Fig. 11.

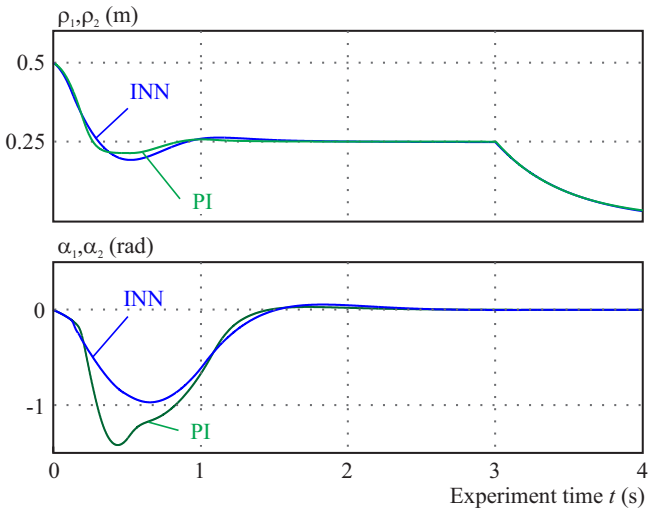

Fig. 11. Comparison of error minimalization for INN and PI controllers

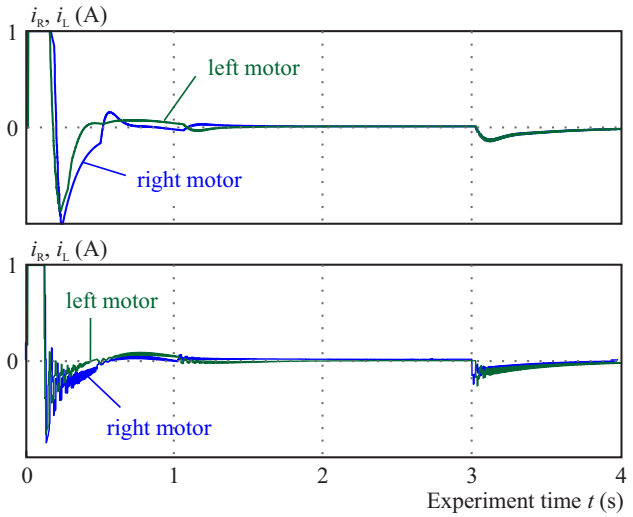

Fig. 12. Right and left wheel armature currents

In the terms of DC motor coil currents, the changes are smoother in case of classical PI controllers, the produced torques dependent on coil currents are in case of inverse neural controllers oscillating within limited range when there is a reference velocity change $\boldsymbol{v}_{\text{ref}}$ that have a negative impact the control quality, the comparison of wheel motor currents for both internal velocity control loops is shown in Fig. 12.

6 Conclusions

As the simulation experiment in this article demonstrated, the neural models can be successfully trained as an inverse dynamics models and used as controllers in mobile robot internal velocity tracking control loop. The linear reference velocity tracking is in case of inverse neural controllers almost ideal, similar performance is difficult to achieve using simpler PI controllers. In order to train neural networks used as neural controllers, the analytical model of mobile robot is not necessary when the real mobile robot operating limits are known. The independently trained inverse neural model controllers can incorporate uneven load acting on particular actuators. The PI controllers offer smoother motor coil currents for smooth reference velocities changes in comparison to inverse neural controllers. In addition, the PI controller require only one previous sample of wheel angular velocity error, while the inverse neural controller used three past samples of wheel angular velocities. For the navigation control task, where the reference velocities are not constant, it is better to use PI controllers in internal velocity tracking control loops. On the contrary, in case of constant reference velocities the inverse neural controller may have a better performance.

Acknowledgement

This work has been supported by the Research and Development Operational Program for project: University Science Park Technicom for innovative applications with knowledge technology support, ITMS code 26220220182, co-financed by the ERDF (80%) and by grant KEGA – 001TUKE-4/2015 (20%).

References

[1] I. Bala Sateesh and G. Satis, *Inverse Control of DC Motor using Artificial Neural Networks*, 2015.

[2] G. Campion, B. d'Andrea Novel and G. Bastin, *"Advanced Robot Control"*, Proceedings of the International Workshop on Nonlinear and Adaptive Control: Issues in Robotics Grenoble, France, Nov. 21–23, 1990, Springer Berlin Heidelberg, 1991 pp. 106–124, Ch. Controllability and state feedback stabilizability of non holonomic mechanical systems.

[3] E. Cuevas, D. Zaldivar and M. Pérez, "Low-Cost Commercial LEGO Platform for Mobile Robotics", *CoRR* abs/1407.0049, 2014.

[4] A. De Luca and G. Oriolo, "Modelling and Control of Non-Holonomic Mechanical Systems", *Kinematics and dynamics of multi-body systems*, Springer, 1995 pp. 277–342.

[5] R. Dhaouadi and A. A. Hatab, "Dynamic Modelling of Differential-Drive Mobile Robots using Lagrange and Newton-Euler Methodologies: A Unified Framework", *Advance in Robotics & Automation*, vol. l2, no. 2, 2013.

[6] J. Čerkala and A. Jadlovská, "Dynamics with Friction in Mobile Robot Simulink Model", *Technical Computing Bratislava 2014*, Proceedings, 2015, Lap Lambert Academic Publishing, pp. 65–81.

[7] J. Čerkala and A. Jadlovská, "Nonholonomic Mobile Robot with Differential Chassis Mathematical Modelling and Implementation in Simulink with Friction in Dynamics", *Acta Electrotechnica et Informatica*, vol. 15, no. 3, 2015 pp. 3–8.

[8] J. Čerkala, T. Klein, and A. Jadlovská, "Modeling and Control of Mobile Robot with Differential Chassis", *Electrical Engineering and Informatics 6*, Proceedings of the Faculty of Electrical Engineering and Informatics of the Technical University of Košice, September 2015, Faculty of Electrical Engineering and Informatics, pp. 651–656.

[9] R. Fierro and F. Lewis, "Control of a Nonholonomic Mobile Robot: Backstepping Kinematics into Dynamics", *Decision and Control*, 1995, Proceedings of the 34-th IEEE Conference, Dec 1995, vol. 4, pp. 3805–3810.

[10] C. Iurian, F. Ikhouane, J. Rodellar Benedé, and R. Griñó Cubero, *Identification of a System with Dry Friction*, 2005.

[11] A. Jadlovská, J. Čerkala, and M. Tomčák, "Neural Model in Mobile Robot Trajectory Following Task", *Electrical Engineering and Informatics 6*, Proceedings of the Faculty of Electrical Engineering and Informatics of the Technical University of Košice, September 2015, Faculty of Electrical Engineering and Informatics, pp. 739–744.

[12] A. Jadlovská, "Using Forward and Inverse Neural Models for Solving Optimal Tracking Problem of Non-linear System", *Journal of Electrical Engineering*, vol. 55, 2004, pp. 150–155.

[13] A. Jadlovská and S. Jadlovská, *Modern Methods of Modelling and Control of Nonlinear Systems*, elfa press, 2013, pp. 279, ISBN 978-8086-228-2 (in Slovak).

[14] S. Kajan, "Comparison of Some Neural Control Structures for Nonlinear System", *Journal of Cybernetics and Informatics*, vol. 8, 2009.

[15] R. Rojas, *Models for DC Motors*, 2010, Free University of Berlin, Institute of Computer Scienc.

[16] N. Sarkar, Y. Xiaoping and V. Kumar, "Systems Dynamics and Control, Proposed Course Overview and Education Oriented Approach for Mechatronics Engineering Curricula: Case Study", *American Journal of Educational Science*, vol. 1, no. 4, 2015, pp. 135–151.

[17] N. Sarkar, Y. Xiaoping and V. Kumar, "Control of Mechanical Systems with Rolling Constraints: Application to Dynamic Control of Mobile Robots", *The International Journal of Robotics Research*, vol. 13, no. 1, 1994, pp. 55–69.

[18] R. Siegwart and I. R. Nourbakhsh, *Introduction to Autonomous Mobile Robots*, MIT Press, 2004.

[19] I. Virgala, P. Frankovský and M. Kenderová, "Friction Effect Analysis of a DC Motor", *American Journal of Mechanical Engineering*, vol. 1, no. 1, 2013, pp. 1–5.

Received 7 December 2016

Jakub Čerkala was born February 4th, 1988. In 2012 he graduated (MSc) with distinction at the Department of Cybernetics and Artificial Intelligence of the Faculty of Electrical Engineering and Informatics at Technical University in Košice. Since September 2012 he has been internal PhD student at the Department of Cybernetics and Artificial Intelligence.

Anna Jadlovská was born October 29th, 1960. She received her MSc degree in the field of Technical Cybernetics at the Faculty of Electrical Engineering of the Technical University in Košice in 1984. She defended her PhD thesis in the domain of Automatization and Control in 2001 at the same University; her thesis title was Modelling and Control of Non-linear Processes Using Neural Networks. Since 1993 she worked in the Department of Cybernetics and Artificial Intelligence Faculty of Electrical Engineering and Informatics Technical University in Košice as an Associate Assistant and since 2004 she has been working as an Associate Professor. Her main research activities include the problems of adaptive and optimal control in particular predictive control with constraints for non-linear processes using neural networks and methods of artificial intelligence (Intelligent Control Design). She is the author of scientific articles and contributions to various journals and international conference proceedings, as well as being the co-author of some monographs.