Intelligent Tracking Trajectory Design of Mobile Robot

Peter ŠUSTER

Dept. of Cybernetics and Artificial Intelligence, FEI TU of Košice, Slovak Republic

peter.suster@tuke.sk

***Abstract*— This paper introduces a solution to the reference trajectory tracking problem done by a differential wheeled mobile robot Khepera II. The paper includes a mathematical model of mobile robot, which we use for the acquisition of a set training data for creating forward and inverse neural model. The purpose of the control structure was the reference trajectory tracking, which we verified using the Neural Network Toolbox of Matlab/Simulink.**

***Keywords*— mobile robot, MLP neural network, forward neural model, inverse neural model.**

I. Introduction

The primary task of every mobile robot in the industry is to track predefined trajectory form its initial to a final position. Track trajectory of mobile robot is possible by using neuro-fuzzy controller [9]. In our paper, we have used neuro approach for tracking trajectory. Training data, necessary for proposal nonparametric controller, we have obtained from simulation model of the robot, which was controlled the proposed control structure. Simulation model of the mobile robot was used for verify algorithms of tracking defined reference trajectory. Simulation model of the mobile robot is based on a real mobile robot Khepera II of K-team Corp. [8].

II. Mathematical – Physical Model of Mobile Robot

Created a model is based on several assumptions, namely that the robot moves on a perfect flat surface without sliding and also neglects the rolling resistance of the wheels. Position of the mobile robot is given by the coordinates x, y and angle θ, which represents the rotation of the mobile robot in relation to the chosen coordinate system. Mobile robot is controlled by the angular velocities of the wheels ω_L, ω_R. Between the angular velocities ω_L, ω_R and peripheral speeds v_L, v_R there are the following relations

$$v_L = r\omega_L \ , \ \ v_R = r\omega_R \tag{1}$$

where r is radius of the wheel. Position and rotation of the robot in the space can be based on (1) to express the following equations, which form a kinematic model of the mobile robot (Fig.1)

$$\begin{aligned} \dot{x}(t) &= v\cos\theta \\ \dot{y}(t) &= v\sin\theta \\ \dot{\theta}(t) &= \omega \end{aligned} \quad \Rightarrow \quad \begin{aligned} v &= \frac{v_L + v_R}{2} \\ \omega &= \frac{v_L - v_R}{b} \end{aligned} \tag{2}$$

where the inputs into the kinematic model are speeds wheels v_L resp. v_R and the outputs are x, y, θ. The kinematic model (Fig.1) allows us to determine the position and rotation of the robot under the condition that we know the initial state of the robot and we have updated information about the speed of the individual wheel [10].

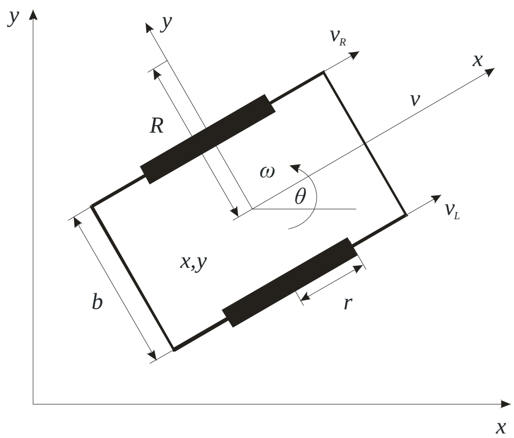

Fig. 1 Kinematic model of mobile robot

The kinematic model does not include friction forces acting on the wheel and the total mass of the mobile robot, so we have extended the mathematical – physical model about the dynamic part (Fig.2), which has the following shape:

$$\begin{aligned} ma_t &= F_L + F_R \\ J\varepsilon &= \frac{(F_L - F_R)b}{2} \end{aligned} \tag{3}$$

where tangent acceleration a_t is given by mass of the robot m and tangent forces F_L a F_R, which acting on the wheels due to change in the rotation speed. Angular acceleration ε is determined by the same forces, the moment of inertia of the robot J and distance between the wheels b [3]. Angular velocities ω_L and ω_R $(\omega_L = \dot{\theta}_L, \omega_R = \dot{\theta}_R)$ of the mobile robot are driven by the voltage U_L and U_R. Differential equations expressing this fact have the following shape [1]

$$J\ddot{\theta}_L(t) + F_T\dot{\theta}_L(t) + F_L r = U_L$$
$$J\ddot{\theta}_R(t) + F_T\dot{\theta}_R(t) + F_R r = U_R \tag{4}$$

where F_T is friction force acting on the wheel. From equations (3) and (4), we have obtained dynamic model of the mobile robot in the state space :

$$\dot{x}(t) = Ax(t) + Bu(t)$$
$$y(t) = Cx(t) \tag{5}$$

where the state variables and their derivates have the following physical meaning:

$$x(t) = [x_1(t), x_2(t), x_3(t), x_4(t)] = [v(t), \omega(t), \omega_L(t), \omega_R(t)]$$
$$\dot{x}(t) = [\dot{x}_1(t), \dot{x}_2(t), \dot{x}_3(t), \dot{x}_4(t)] = [a_t(t), \varepsilon(t), \varepsilon_L(t), \varepsilon_R(t)]$$

and inputs into the system are:

$$u = [u_1(t), u_2(t), u_3(t), u_4(t)] = [F_L, F_R, U_L, U_R]$$

And outputs from the system are:

$$y(t) = [y_1(t), y_2(t)] = [x_3(t), x_4(t)].$$

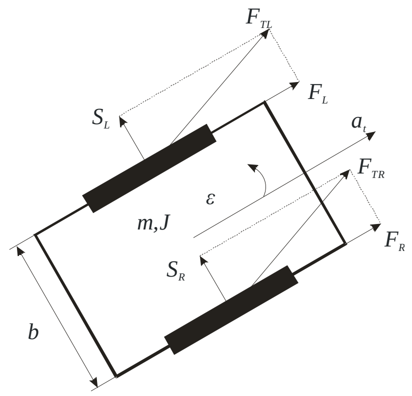

Fig. 2 Dynamic model of mobile robot

We programmed simulation scheme of the mobile robot (Fig.3)(Fig.4) in the Matlab/Simulink, based on the equations of the kinematic (2) and dynamic model (3) (4) :

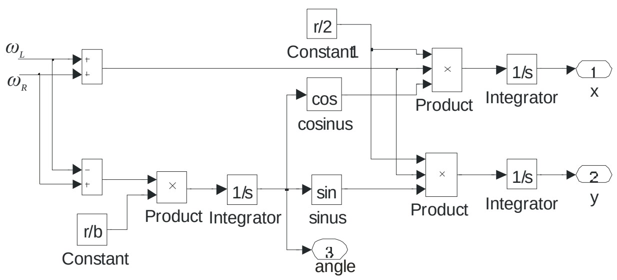

Fig. 3 Simulation scheme of mobile robot - kinematics

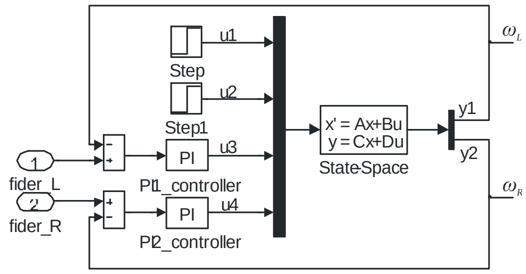

Fig. 4 Simulation scheme of model robot – dynamics

We proposed a control structure to ensure that the mobile robot can track one of the set reference trajectory [2]. The inputs into control structure of model robot are coordinates of current position of model robot *x*, *y* and coordinates of reference trajectory x_{ref}, y_{ref}. We have calculated Euclidean distance between current and desired position of the model robot by means of these coordinates. The outputs from control structure are angular velocities for left and right wheels. Subsystems control structure and model robot (Fig.5) we used in the simulation schemes for acquisition of training data necessary for design forward and inverse neural model. Simulations were carried out in the sample period $T_{vz} = 0{,}01s$.

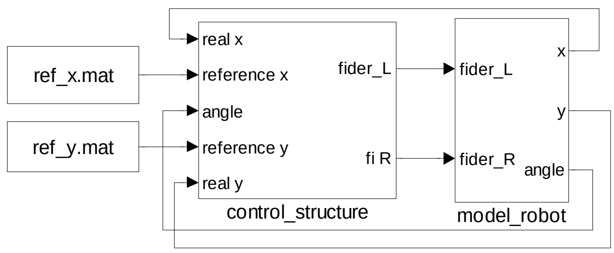

Fig. 5 Simulation scheme is designed for to simulate movements of the mobile robot

III. FORWARD NEURAL MODEL OF MOBILE ROBOT

Neural model, which approximates dynamic of the system is called forward model. Neural network is placed in parallel with identification system and error between output of the neural network $\hat{y}(k+1)$ and output of the dynamic system $y(k+1)$, the so-called prediction error, is used as training signal for neural network (Fig.6). Forward network of MLP type was used as neural network.

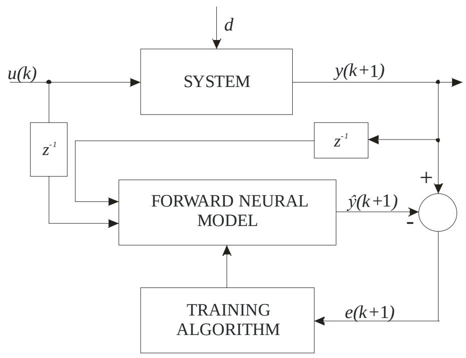

Fig. 6 Identification scheme based on output prediction error

If the output of the neural model is $\hat{y}(k+1)$ then we can express the equation by approximation

$$\hat{y}(k+1)=\hat{f}\left[y(k),\ldots,(k-n+1),u(k),\ldots,u(k-m+1)\right] \tag{6}$$

where $\hat{f}$ is represents the non-linear input-output representation by the neural model and $y(k)$ resp. $u(k)$ is n - output resp. m - input of the previous values [4].

Training data for proposal forward neural model, we obtained from simulation scheme to simulate the movement of the robot along defined reference trajectory. Reference trajectory was represented by vectors x and y coordinate. For training the forward neural model, we used a forward neural network of Multi Layer Perceptron (MLP) type with ten neurons in the input layer, with ten neurons in the hidden layer and with two neurons in the output layer (Fig.7). The training of forward neural model was carried out by the Levenberg-Marquardt algorithm using Neural Network Toolbox.

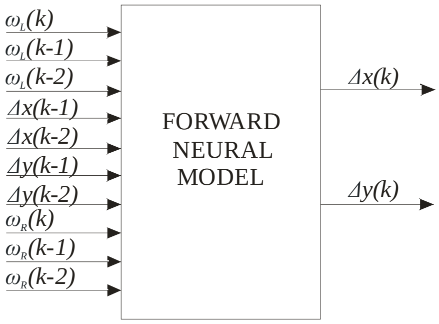

Fig. 7 Forward neural model of the mobile robot

The validation of the model is the next step after training of the neural model. The result of testing of trained forward neural model (Fig.7) is shown in the Fig.8. From picture (Fig.8) shows that forward neural model can approximate with accuracy, which meets for its further use at the tracking defined reference trajectory.

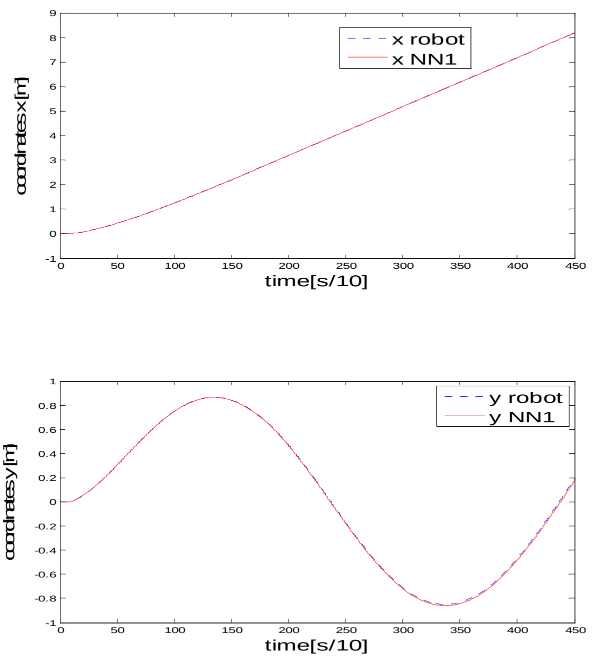

Fig. 8 Comparison outputs of the system and the forward neural model

IV INVERSE NEURAL MODEL OF MOBILE ROBOT

Inverse neural model of the system is an important part of the theory of control. If the forward neural model was described by the equation (6), then the inverse model can be expressed in the form:

$$u(k)=f^{-1}\begin{bmatrix} r(k+1),y(k),..,y(k-n+1),... \\ ....u(k),....,u(k-m+1) \end{bmatrix} \tag{7}$$

where $y(k+1)$ is an unknown value, therefore it is substituted by the reference value of the control variable $r(k+1)$. To obtain inverse neural model, we have chosen General training architecture (Fig.9), which requires a known reference trajectory $r(k)$. Signal $u(k)$ is applied to the inputs of structure based on input predictive error with the aim of to obtain a corresponding system output $y(k)$, while the neural network is trained by the error $e_u(k)$, which is obtained as the difference of the neural model output $\hat{u}(k)$ and input signal $u(k)$ into the system [4].

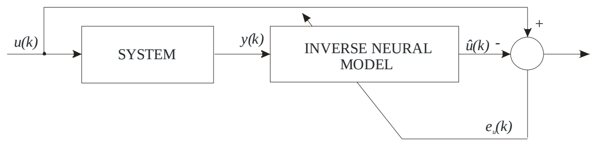

Fig. 9 General training structure

For training the forward neural model, we used forward neural network of Multi Layer Perceptron (MLP) type with fourteen neurons in the input layer, with five neurons in the hidden layer and with two neurons in the output layer (Fig.10). Training of forward neural model was carried out by the Levenberg-Marquardt algorithm.

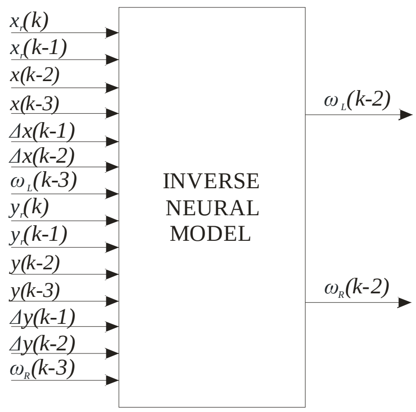

Fig. 10 Inverse neural model of mobile robot

We applied inverse neural model (Fig.10) together with forward neural model (Fig.7) into control structure IMC (Fig.12), which we have used for tracking defined reference trajectory. We proposed the IMC filter into control structure for better tracking trajectory. The goal of the tracking is to control the movement of the mobile robot from the point A to the point B along the chosen reference trajectory (Fig.11).

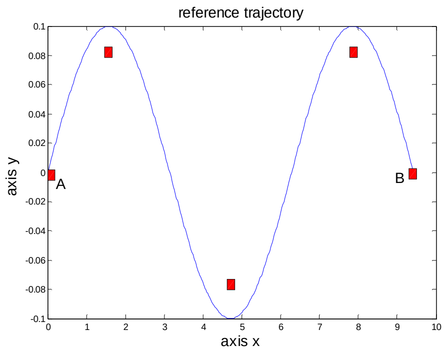

Fig. 11 Reference trajectory

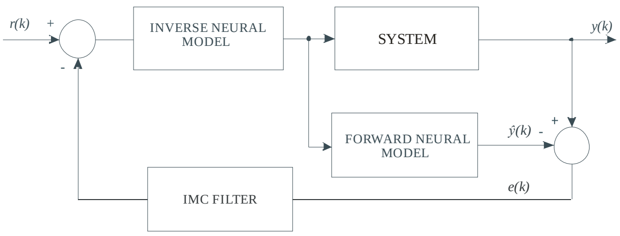

Fig. 12 Control structure Internal Model Control

The output from control structure IMC is current trajectory of simulation model of mobile robot (Fig.13), which is controlled nonparametric neural controller.

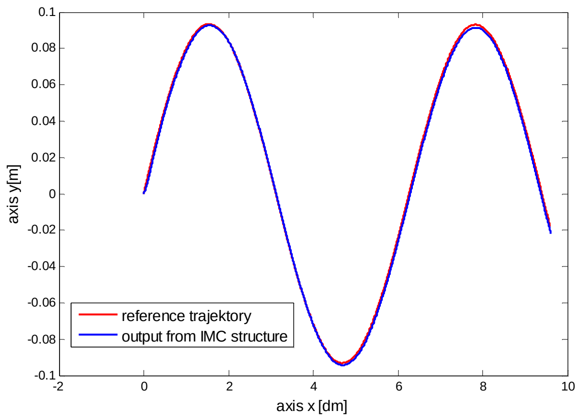

Fig. 13 Comparison the defined reference trajectory and output from IMC structure

From Fig.13, we can see that simulation model of mobile robot tracks the defined reference trajectory. We verified the functionality for other sinus trajectories with other amplitudes. When we have changed trajectory is necessary training the new inverse and forward neural model.

V. CONLUSION

We have analyzed the problem of tracking predefined reference trajectory of the mobile robot in the paper. As solution to the problem, we have proposed nonparametric neural controller, which we implemented into the control structure IMC together with forward neural model. Training data, necessary for proposal nonparametric controller, we have obtained from simulation model of the robot. The obtained knowledge in the field tracking reference trajectory of the mobile robot, we want to use for real mobile robot Khepera III, which are in our laboratory at the Department of Cybernetics and Artificial Intelligence.

ACKNOWLEDGMENT

This contribution is the results of the Vega project implementation: Multiagent Network Control System with Automatic Reconfiguration (No.1/0617/08), supported by the Scientific Grant Agency of Slovak Republic.

REFERENCES

[1] D. DOMINGUEZ, VRML and Simulink Interface for the Development of 3-D Simulator for Mobile Robots, *Proceedings of world academy of science, Engineering and Technology*, volume 25, November 2007. ISSN 1307 - 6884

[2] J. FIC, *Riadenie mobilného robota Khepera II s využitím metód umelej inteligencie.* Master Thesis, TU Košice, Faculty of Electrical Engineering and Informatics, Košice, Slovakia, 2009.

[3] M. GAJDUŠEK, F. ŠOLC, Generovaní časově optimální trajektorie pro mobilního robota s diferenciálním řízením. *AT&P Journal*, 2006, No. 2, pp 86-89.

[4] A. JADLOVSKÁ, J. SARNOVSKÝ, Aplikácia inverzného neurónového modelu nelineárneho procesu v štruktúre priameho inverzného riadenie. *AT&P Journal*, 2002, No.10, pp.75-77, No. 11, pp.84-86, No.12, pp. 46, 2002, ISSN 1335-2237

[5] J. KAJAN, Comparison of some neural control structures for nonlinear systéme. *Journal of Cybernetics and Informatics*, 2009, volume 8 2009. ISSN 1336-4774

[6] B. KIM, P. TSIOTRAS, Controllers for Unicycle-Type Wheeled Robots: Theoretical Results and Experimental Validation. *IEEE Transactions on robotics and automation*, 2002, No. 3, pp. 294-307.

[7] F. KÜHNE, W. F. LAGES, J. M. GOMES da SILVA, Mobile robot trajectory tracking using model predictive control. In.: *VII SBAI/II IEEE LARS*, São Luís, september 2005.

[8] K-TEAM. Dostupné na: <http://www.k-team.com >.

[9] I. MASÁR, Inteligentný regulátor na sledovanie trajektórie mobilným robotom. *Automatizace*, 2007, No 2, pp. 80-85.

[10] J. ŠEMBERA, F. ŠOLC, Modelovaní a řízení mobilního robotu s diferenciálním podvozkem. *AT&P Journal PLUS*, 2007, No. 1, pp 203-207.