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 \(\theta\), 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 \(\omega_L, \omega_R\). Between the angular velocities \(\omega_L, \omega_R\) and peripheral speeds \(v_L, v_R\) there are the following relations

\[ v_L = r \omega_L, \quad v_R = r \omega_R \]  \hspace{1cm} (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)

\[
\begin{align*}
x(t) &= v \cos \theta \\
y(t) &= v \sin \theta \\
\dot{\theta}(t) &= \omega
\end{align*}
\]  \hspace{1cm} (2)

where the inputs into the kinematic model are speeds wheels \(v_L, v_R\) and the outputs are \(x, y, \theta\). 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](image-url)
robot are driven by the voltage $U_L$ and $U_R$. Differential equations expressing this fact have the following shape [1]

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

(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:

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

(5)

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

$$x(t) = [x_1(t), x_2(t), \xi_1(t), \xi_2(t), \xi_3(t), \xi_4(t)] = [\theta(t), \dot{\theta}(t), \omega_L(t), \omega_R(t)]$$

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:

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

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. 2 Dynamic model of mobile robot](image)

**Fig. 2 Dynamic model of mobile robot**

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 $T_{ic} = 0.01s$.

![Fig. 3 Simulation scheme of mobile robot - kinematics](image)

**Fig. 3 Simulation scheme of mobile robot - kinematics**

![Fig. 4 Simulation scheme of model robot – dynamics](image)

**Fig. 4 Simulation scheme of model robot – dynamics**

We presented 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 $T_{ic} = 0.01s$.

![Fig. 5 Simulation scheme is designed for to simulate movements of the mobile robot](image)

**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](image)

**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
\[ y(k+1) = \hat{f}(y(k),...,y(k-n+1),u(k),...,u(k-m+1)) \]  
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 five 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.

The validation of the model is the next step after training of the neural model. The result of testing of trained forward neural model was carried out by the Levenberg-Marquardt algorithm.

\[ u(k) = f^{-1}(r(k+1), y(k),...,y(k-n+1),...,u(k),...,u(k-m+1)) \]  

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 \( e_y(k) \), which is obtained as the difference of the neural model output \( \hat{u}(k) \) and input signal \( u(k) \) into the system [4].

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.

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

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. CONCLUSION

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