Neural Network Control Based Algorithm for Dual Wheel Mobile Robot

Zulfiqar Ali Soomro*
Directorate of Post-graduate Studies Mehran University of Engg;&Tech; Jamshoro(Pakistan)
*Corresponding author's E-mail: Zulfiqarali_s@yahoo.com

Abstract
The mathematical model of the dual wheel mobile robot is framed through the dynamics of the proposed model. The PI controller is used to control the uncertainties of the wheel mobile robot by drawing its concerned simulink blocks. The neural network algorithm is applied to the model as it deals tuning and optimization properly. The feed forward technique is preferred for the model. Thus various simulation graphics are achieved by Matlab and training the parameters to observe the results and sim-net.

Keyword: WMR, perceptron, PI, feed forward

1. Introduction

Wheel type mobile robots are the most popular transportation mechanisms as the energy efficiency is high, the mechanism is simple and the control system is well investigated. Wheeled Mobile Robots (WMRs) are considered as the most widely used class due to their fast maneuvering, simple controllers and energy saving characteristics. A wheeled mobile robot is a wheeled vehicle which is capable of an autonomous motion without external human drivers, because it is equipped, for its motion, with motors that are driven by an embedded computer. Recently much research has been done on applications of neural networks for control of nonlinear dynamic processes. These works are supported by two of the most important capabilities of neural networks, their ability to learn and their good performance for approximation of nonlinear functions. These two abilities are the main reason behind combining the neural network controller with the back-stepping to fix its disadvantage. There are numerous types of artificial neural networks for addressing many
different types of problems, such as modeling memory, performing pattern recognition, and predicting the evolution of dynamical systems. Most networks therefore perform some kind of data modeling, and they may be split into two broad classes: supervised and unsupervised. One of the most important types of supervised neural networks, called a feed forward multilayer perceptron. The term perceptron is historical, and refers to the function performed by the nodes. Feed forward means that there is a definite input and output, and a flow of data in one direction. This is in contrast to recurrent neural networks in which data flows in a loop. Here feed forward technique of neural network is used to study and control the virtual and mathematical model for wheel mobile robot (WMR) to acquire the concerned simulated results.

2. Dynamically model of WMR

The differential drive mobile robot setup shown in following figure-01 has two wheels with the radius Ra of placed with a distance L from the robot center. Here subscriptions with \( x_m \) and \( y_m \) show coordinates of model while capital X and Y show global coordinates and \( \theta \) as inclination. As intersection of the axis of symmetry with the driving wheels axis, ‘a’ as distance between the center of mass and driving wheels axis in x-direction.
The forward kinematic analysis is used for finding the following function.

\[
\dot{q} = \begin{bmatrix}
\dot{x} \\
\dot{y} \\
\dot{\theta}
\end{bmatrix} = f(\phi_R, \phi_L, L, R_a, \theta) \rightarrow (1)
\]

The speed of each wheel in the robot frame is \(R_a \dot{\phi}\), therefore the translational speed in the robot frame is the average velocity: \(v = R_a \frac{\dot{\phi}_R + \dot{\phi}_L}{2}\) And the rotational velocity is:

\[
\omega = \frac{R_a}{2L} (\dot{\phi}_R - \dot{\phi}_L),
\]

Where \(\dot{\phi}_R\): The rotational velocity of the right wheel, \(\dot{\phi}_L\): The rotational velocity of the left wheel, \(v\): The translational velocity of the platform in the local frame, and \(\omega\): The rotational velocity of the platform in the local and global frames. Given that \(\dot{q}_I = R(\theta)^{-1} \dot{q}_R\), and \(R(\theta)^{-1}\) is rotation matrix, the full model robot velocity in the inertial frame is

\[
\dot{q}_I = R(\theta)^{-1} \frac{R_a}{2} \begin{bmatrix}
\dot{\phi}_R + \dot{\phi}_L \\
0 \\
\dot{\phi}_R - \dot{\phi}_L
\end{bmatrix} (2)
\]
Therefore the robot velocity in the global or inertial frame is as under by putting the values in equation (2) and comparing with equation (1) we get relation as:

\[
\dot{q} = \begin{bmatrix}
\dot{x} \\
\dot{y} \\
\dot{\theta}
\end{bmatrix} = \begin{bmatrix}
\cos(\theta) & -\sin(\theta) & 0 \\
\sin(\theta) & \cos(\theta) & 0 \\
0 & 0 & 1
\end{bmatrix} \begin{bmatrix}
\frac{\phi_R + \phi_L}{2} \\
0 \\
\frac{\phi_R - \phi_L}{2}
\end{bmatrix} = \begin{bmatrix}
\frac{R_a}{2} \frac{\phi_R + \phi_L}{2} \cos(\theta) \\
R_a \frac{\phi_R + \phi_L}{2} \sin(\theta) \\
\frac{R_a}{2L} (\phi_R - \phi_L)
\end{bmatrix}
\]

The above equation is the general form of forward kinematic equation for a differential drive mobile robot. A mobile robot system having an n-dimensional configuration space L with generalized coordinates \((q_1, q_2, ..., q_n)\) and subject to m constraints can be described by the following general dynamic equation:

\[
M(q)\ddot{q} + V(q, \dot{q}) + F(\dot{q}) + G(q) + \tau_d = B(q)\tau - A^T(q)\lambda
\]  

Where: \(M(q)\) is the symmetric positive definite inertia matrix, \(V(q, \dot{q})\) is the centripetal and coriolis matrix, \(F(\dot{q})\) is the surface friction matrix, \(G(q)\) is the gravitational vector, \(\tau_d\) Denoted bounded unknown disturbances including unstructured un-modeled dynamics, \(B(q)\) is the input transformation matrix, \(\tau\) is the input vector, \(A^T(q)\) is the matrix associated with the constraints, \(\lambda\) is the vector of the constraint force. By applying some mathematical and Lagrangian Dynamics approach to the wheel robot model we get the derivations to complete the form of the differential drive mobile robot dynamic equation in eq-4 as follow

\[
\begin{bmatrix}
m & 0 & \frac{m\sin\theta}{L_c + 2ma^2} \\
0 & m & \frac{-m\cos\theta}{L_c + 2ma^2} \\
\frac{m\sin\theta}{L_c + 2ma^2} & \frac{-m\cos\theta}{L_c + 2ma^2} & 1
\end{bmatrix} \ddot{q} + \begin{bmatrix}
ma^2\cos\theta \\
ma^2\sin\theta \\
0
\end{bmatrix} = \begin{bmatrix}
\frac{1}{R_a} \frac{\cos\theta}{L_c + 2ma^2} \\
\frac{1}{R_a} \frac{\cos\theta}{L_c + 2ma^2} \\
\frac{1}{R_a} \frac{\sin\theta}{L_c + 2ma^2} \\
\frac{1}{R_a} \frac{\sin\theta}{L_c + 2ma^2} \\
\frac{1}{R_a} \frac{\tau_R}{L_c + 2ma^2} \\
\frac{1}{R_a} \frac{\tau_L}{L_c + 2ma^2}
\end{bmatrix}
\]

\[
\begin{bmatrix}
m(\dot{x}_c \cos\theta + \dot{y}_c \sin\theta) \dot{\theta} \sin\theta \\
-m(\dot{x}_c \cos\theta + \dot{y}_c \sin\theta) \dot{\theta} \cos\theta \\
ma(\dot{x}_c \cos\theta + \dot{y}_c \sin\theta) \dot{\theta}
\end{bmatrix}
\]
Where \( F(q) = 0 \) considered to be zero in this derivation, 
\( G(q) = 0 \) motion is constrained to the ground, 
\( \tau_d = 0 \) considered to be zero in this derivation, 
\( \lambda = -m(\dot{x}_c \cos \theta + \dot{y}_c \sin \theta) \dot{\theta} \), 
\( A^T(q) = \begin{bmatrix} -\sin \theta & \cos \theta & -a \\ \cos \theta & \sin \theta & a \\ -a & 0 & 1 \end{bmatrix} \), 
the \( S(q) \) matrix is the modified forward kinematic matrix which has two velocity terms related to the distance between the robot centroid and wheel axis.

3. Wheel mobile robot control

The primary shortcoming of the P controller is the lack of its tolerance for high gains, which motivates adding an integral gain \( K_I \) to the controller. The integral gain will eliminate
the signal drop, and generally the system reacts better with a PI controller than a P controller, because the integral gain will grow ever larger even with small errors. Integral gain provides stiffness to the signal. That means when the error occurs, the integral gain will move to correct it. The higher the gain is, the faster the correction. Tuning the PI controller was very sensitive for high $K_p$ gains.

4. Neural Net Work Application

Since the Neural networks (NN) are complex nonlinear distributed systems, and as a result they have a wide range of applications, hence NN technique is applied to the proposed ‘Dual wheel mobile robot’. A feed forward neural network and PID controller are introduced to design the control system of WMR and their performances are compared through simulations. As in linear algebra the term vector is often used in neural network jargon. The values of the input nodes are often called the input vector. Similarly, the list of activation values of the output layer is called the output vector.

The method used here is the default training (optimization) i-e Levenburg-Marquardt (which is the default- it is a combination of gradient descent and Newton’s Method). In the next training/testing session, we will use a different method. The network of the neurons is connected through synapses or weights. Each neuron performs a simple calculation that is a function of the activations of the neurons that are connected to it. Through feedback mechanisms and/or the non-linear output response of neurons, the network as a whole is capable of performing extremely complicated tasks, including universal computation and universal approximation. A neural network is trained using a training set. A training set comprises information about the problem to be solved as input stimuli.
5. Simulation Results of the model

In the 1st quadrant of figure-3, The Input data achieved from Kc controller used for the dual wheel mobile robot are used for neural network analysis. These input parameters denoted by ‘red-cross’ through certain training are used to obtain the required outputs denoted by ‘blue circles’ through concerned net. Here each hidden layer proposed consists of 24 units. While inputs and outputs initially are coinciding with each other at certain limit, then they are shattered from each other and then they meet each other before going to zero. This conceives that trained outputs give same response after training net parameters of 25 units to apply suitable targets to inputs. The 2nd quadrant of the figure-3 shows the linearly spaced vectors based on feed forward network 0 to 20 units with length of net. Here Straight line starts below than -1 with two humps at nearly 2 to -0.6 and other from Horizontal plane 14 to -0.6 on vertical resembles as Gaussian function in radial basis function in neural network.

![Network based linearly spaced vectors](image1)

![Network based logarithmically spaced elements](image2)

**Figure 3: neural network based simulation graphs**

In the 3rd quadrant of figure-3, the network logarithmically spaced vectors with size 0 to 20 units equally to the length of another neutrally net is shown. Here feed forward is varied in
straight line below than -1 with log spaced parameters. In the 4th quadrant of the figure-3 shows 3-D effect of the input elements shown along x-axis, and the targets are shown along with y-axis, while the z-axis represents the inputs combined along with their concerned targets.

Figure 4: (A) Neural network training tool
Figure 4: (B) Regression analysis neural network
Figure 4: (C) Validation performance NNT

Figure 4: (D) Neural network training states
The figure-4(A), the basic view of the neural network training specimen is shown. Here basic tools represent the neural network algorithm consisting upon two layers having one weight and one bias each. First layer is having ‘tansig’ nature, while second layer represents as ‘purelin’ in neural network which has epochs of 10000000. The figure-4(B), Regression analysis is shown which is statistical tool for the investigation of relationships between variables. Here targets are compared with verified various outputs in the stages of training, validation, testing and all of the regression through T, data, fit. In the linear regression model, the dependent variable is assumed to be a linear function of one or more independent variables plus an error introduced to account for all other factors. The figure-4(C), the ‘mean square error (mse)’ is described by performance neural net with 10 epochs each. Here estimation is determined by train, validation, test, best and goal. The figure-4(D), here the training state is determined in neural net by the gradient, mu and validation checked by 10 epochs.

**Conclusion**

The mathematical model of the wheel mobile robot is framed with certain formulae. The Proportional integral (PI) controller is used to control the uncertainties created by WMR, but it does not perfectly removes the full created noise of the model. The PID (Proportional-Integral Derivative) Controllers are widely used in industries but these are linear controllers. Robot control is a non-linear Controller. Optimal results are usually not possible. Since the turning and optimization is very difficult chore and are not adaptive as they may not work under varying conditions, hence neural network algorithm is proposed to apply on this dual wheel mobile. It is well-established fact that feed forward neural networks are suitable for engineering applications; therefore we propose to use feed forward neural network for the applications. The simulink, Matlab and various neural network techniques are used to obtain required simulation results and concerned graphs.
References