Designing and simulation a motion Controller for a Wheeled Mobile Robot Autonomous Navigation

Souma M. Alhaj Ali[a] and Ernest L. Hall[b]

[a]The Hashemite University, Department of Industrial Engineering,
B. O. Box: 150459, Zerka 13115 Jordan
[b]University of Cincinnati, Department of Mechanical, Industrial and Nuclear Engineering,
Cincinnati, Ohio

ABSTRACT

This paper describes the development of PD, PID Computed-Torque (CT), and a PD digital motion controller for the autonomous navigation of a Wheeled Mobile Robot (WMR) in outdoor environments. The controllers select the suitable control torques, so that the WMR follows the desired path produced from a navigation algorithm described in a previous paper. PD CT, PID CT, and PD digital controllers were developed using a linear system design procedure to select the feedback control signal that stabilizes the tracking error equation. The torques needed for the motors were computed by using the inverse of the dynamic equation for the WMR.

Simulation software was developed to simulate the performance and efficiency of the controllers. Simulation results verified the effectiveness of the controllers under different motion trajectories, comparing the performance of the three controllers shows that the PD digital controller was the best where the tracking error did not exceed .05 using 20 msec sample period. The significance of this work lies in the development of CT and digital controllers for WMR navigation, instead of robot manipulators. These CT controllers will facilitate the use of WMRs in many applications including defense, industrial, personal, and medical robots.

Keywords: Autonomous navigation, Wheeled Mobile Robots (WMRs), Motion Control, Computed-Torque (CT) Controller, Digital Controller, Motion Analysis, Neuro-Control, Non-Linear Dynamics, Non-Linear Systems and Modeling.

1. INTRODUCTION

Robots and robots manipulators have complex nonlinear dynamics that make their accurate and robust control difficult. On the other hand, they fall in the class of Lagrangian dynamical systems, so that they have several extremely nice physical properties that make their control straight forwarded[1]. Different controllers had been developed for the motion of robot manipulators, however, not until recently where there has been an interest in moving the robot itself, not only its manipulators.

Shim and Sung[2] proposed a WMR asymptotic control with driftless constraints based on empirical practice using the WMR kinematic equations. They showed that with the appropriate selection of the control parameters, the numerical performance of the asymptotic control could be effective. The trajectory control of a wheeled inverse pendulum type robot had been discussed by Yun-Su and Yuta[3], their control algorithm consists of balance and velocity control, steering control, and straight line tracking control for navigation in a real indoor environments.

Rajagopalan and Barakat[4] developed a computed torque control scheme for Cartesian velocity control of WMRs. Their control structure can be used to control any mobile robot if its inverse dynamic model exists. A discontinuous stabilizing controller for WMRs with nonholonomic constraints where the state of the robot asymptotically converges to the target configuration with a smooth trajectory was presented by Zhang and Hirschorn[5]. A path tracking problem was formulated by Koh and Cho[6] for a mobile robot to follow a virtual target vehicle that is moved exactly along the path with specified velocity. The driving velocity control law was designed based on bang-bang control considering the acceleration bounds of driving wheels and the robot dynamic constraints in order to avoid wheel

slippage or mechanical damage during navigation. Zhang, et al.[7] employed a dynamic modeling to design a tracking controller for a differentially steered mobile robot that is subject to wheel slip and external loads.

A sliding mode control was used to develop a trajectory tracking control in the presence of bounded uncertainties[8]. A solution for the trajectory tracking problem for a WMR in the presence of disturbances that violate the nonholonomic constraint is proposed later by the same authors based on discrete-time sliding mode control[9,10]. An electromagnetic approach for path guidance of a mobile-robot-based automatic transport service system with a PD control algorithm was investigated by Wu, et al.[11]. Jiang, et al.[12] developed a model-based control design strategy that deals with global stabilization and global tracking control for the kinematic model with a nonholonomic WMR in the presence of input saturations. An adaptive robust controller was proposed for the global tracking problem for the dynamic of the non-holonomic systems with unknown dynamics[13]. However, real time adaptive controls are not common in practical applications due partly to the stability problems associated with them[14].

A fuzzy logic controller had been tried for WMRs navigation. Montaner and Ramirez-Serrano[15] developed a fuzzy logic controller that can deal with the sensors inputs uncertainty and ambiguity for direction and velocity maneuvers. A locomotion control structure was developed based on the integration of an adaptive fuzzy-net torque controller with a kinematic controller to deal with unstructured unmodeled robot dynamics for a non-holonomic mobile robot cart[16]. Toda, et al.[17] employed a sonar-based mapping of crop rows and fuzzy logic control-based steering for the navigation of a WMR in an agricultural environment. They constructed a crop row map from the sonar readings and transferred it to the fuzzy logic control system, which steers the robot along the crop row. A local guidance control method for WMR using fuzzy logic for guidance, obstacle avoidance and docking of a WMR was proposed by Vázquez and Garcia[18], the method provide a smooth but not necessary optimal solution.

This paper presents the development of .PD CT, PID CT, and PD digital controllers for WMR navigation in unstructured outdoor environments; the controller selects suitable control torques for the motors, which causes the robot to follow the desired path from the navigation algorithm. A dynamic simulation was conducted from a framework developed by Lewis, et al.[1]. The Lewis framework was formulated for a trajectory of robot manipulators. However, this paper is oriented toward robot navigation. Thus, the Lewis framework was adjusted, as needed, to control the robot path, instead of the manipulators trajectories[19]. Simulation software permitting easy investigation of alternative architectures was developed by using MATLAB and C++. The simulation used the Bearcat III dynamic model developed in Alhaj Ali[19].

2. CT controllers

The dynamics of any wheeled mobile robot can be formulated as[1,19]:

$$M(q)\frac{d^2q}{dt^2} + N(q, \frac{dq}{dt}) = \tau \tag{1}$$

or, in the case of the existence of unknown disturbances τ_d :

$$M(q)\frac{d^2q}{dt^2} + N(q, \frac{dq}{dt}) + \tau_d = \tau$$

where $M(q)$ is the inertia matrix, $N(q, \frac{dq}{dt})$ represents all the nonlinear terms, and τ is the control input torque (for more details please refer to refer to[19]).

The objective of a motion controller is to move the robot according to the desired motion trajectory $q_d(t)$. The actual motion trajectory is defined as $q(t)$. The tracking error, in this case, can be defined as[1]:

$$e(t) = q_d(t) - q(t) \tag{2}$$

The Brunovsky canonical form can be developed by differentiating $e(t)$ twice and writing it in the terms of the state x [1]:

$$\frac{d}{dt}\begin{bmatrix} e \\ \frac{de}{dt} \end{bmatrix} = \begin{bmatrix} 0 & I \\ 0 & 0 \end{bmatrix}\begin{bmatrix} e \\ \frac{de}{dt} \end{bmatrix} + \begin{bmatrix} 0 \\ I \end{bmatrix} u \tag{3}$$

where:

$$u \equiv \frac{d^2}{dt^2} q_d + M^{-1}(q)(N(q, \frac{dq}{dt}) - \tau),\ x = \begin{bmatrix} e^T \\ \frac{de}{dt}^T \end{bmatrix}$$

To develop the CT controller, a linear system design procedure is used to select the feedback control signal $u(t)$, which stabilizes the tracking error equation. Then the torques needed for the motors are computed by using the inverse of the dynamic equation for the WMR[1]:

$$\tau = M(q)(\frac{d^2}{dt^2} q_d - u) + N(q, \frac{dq}{dt}) \tag{4}$$

Eq. (4) represents a nonlinear feedback control law, which guarantees tracking the desired motion trajectory $q_d(t)$. Two types of CT controllers are developed here, namely, the PD CT controller and the PID CT controller.

2.1 PD CT controller

A PD feedback for $u(t)$, with a derivative gain matrix K_v and a proportional gain matrix K_p, produces the PD CT controller, where the motor torques equal[1]:

$$\tau = M(q)(\frac{d^2}{dt^2} q_d + K_v \frac{de}{dt} + K_p e) + N(q, \frac{dq}{dt}) \tag{5}$$

which has the tracking error dynamics $\frac{d^2}{dt^2} e = -K_v \frac{de}{dt} - K_p e$

The gain matrices need to be selected positive definite to keep the tracking error dynamics stable.

2.2 PID CT controller

The PD CT controller can easily be adjusted to a PID CT controller by adding an integrator gain matrix K_i to $u(t)$, as follows[1]:

$$\tau = M(q)(\frac{d^2}{dt^2} q_d + K_v \frac{de}{dt} + K_p e + K_i(\int e)) + N(q, \frac{dq}{dt}) \tag{6}$$

which has the tracking error dynamics $\frac{d^2}{dt^2} e = -K_v \frac{de}{dt} - K_p e$

The integrator gain cannot be too large, in order to keep the tracking error stable.

2.3 Digital controller

The PD digital controller output can be calculated by using the following equation[1,19]:

$$\tau_\varepsilon = M(q_\varepsilon)(\frac{d^2}{dt^2} q_{d\varepsilon} + K_\upsilon \frac{d}{dt} e_\varepsilon + K_p e_\varepsilon) + N(q_\varepsilon, \frac{d}{dt} q_\varepsilon) \tag{7}$$

The control input can be calculated only at certain sample times, $t_\varepsilon = \varepsilon T$, where T: is the sample period and ε: is the integer value. Care should be taken with some of the problems inherent to digital controllers, such as stability, actuator saturation, and antiwindup[1].

3 Simulation of the controllers

The function of this controller is to select the suitable motor torques, so that the WMR will follow the desired path produced from the navigation system (for more details please refer to refer to20), $q_d(t)$.

3.1 Simulation of the PD CT controller

The simulation program has the following main components:

- The first component computes the desired WMR trajectory from the input from the navigation system. The desired trajectory, $q_d(t)$, is:

$$q_d = \begin{bmatrix} x_{cd} \\ y_{cd} \\ \theta_d \end{bmatrix}$$

where: x_{cd} : is the x-axis component of the desired position of the WMR center of gravity;
y_{cd} : is the y-axis component of the desired position of the WMR center of gravity;
θ_d : is the desired orientation of the WMR.

- The second component calculates the controller input from the tracking error between the desired trajectory , $q_d(t)$, and the actual trajectory , $q(t)$, where $q(t)$ is:

$$q = \begin{bmatrix} x_c \\ y_c \\ \theta \end{bmatrix}$$

where: x_c : is the x-axis component of the actual position of the WMR center of gravity;
y_c : is the y-axis component of the actual position of the WMR center of gravity;
θ : is the actual orientation of the WMR.

Then the inertia term $M(q)$ and the nonlinear term $N(q, \frac{dq}{dt})$ are computed from the WMR dynamic model. Finally, the motors torques are calculated.

- The third component calculates the new position of the WMR by using the state-space equation, $\frac{dx}{dt} = f(x,u)$, where the state-space position/velocity form is used1:

$$x \equiv \begin{bmatrix} q^T \\ \frac{d}{dt} q^T \end{bmatrix}, \quad \frac{d}{dt} x = \begin{bmatrix} \frac{dq}{dt} \\ -M^{-1}(q) N(q, \frac{dq}{dt}) \end{bmatrix} + \begin{bmatrix} 0 \\ M^{-1}(q) \end{bmatrix} \tau \tag{8}$$

This form is used to update the WMR actual position.

The inputs to the PD CT controller simulation program are the desired motion trajectory, $q_d(t)$, robot parameters, and the controller parameters k_p and k_v. The outputs of the PD CT controller simulation program are the motor torques, $\tau(t)$ and the actual path, $q(t)$.

3.1.1 PD CT controller simulation results

Several experiments are conducted on the PD CT simulation software. The robot parameters are according to Bearcat III which is a three wheels mobile robot developed in the robotics center of the University of Cincinnati. Different controller parameters are tested by using a sinusoidal desired motion trajectory:

$$q_d = \begin{bmatrix} x_{cd} \\ y_{cd} \\ \theta_d \end{bmatrix} = \begin{bmatrix} c \cdot \sin t \\ c \cdot \cos t \\ c \cdot \sin t \end{bmatrix}$$, where c is a constant.

- In the first set of experiments, the same k_p and k_v are used for the three components of the motion trajectory, q (x , y and θ). Starting with k_p = k_v =0, the tracking errors are in the range of 0.00-6.20, as shown in Fig. 1. The

desired and the actual motion trajectory paths are shown in Fig. 2. As shown in the figures, the tracking errors are very high for both x and y, while the tracking error is about 0.10 for θ.

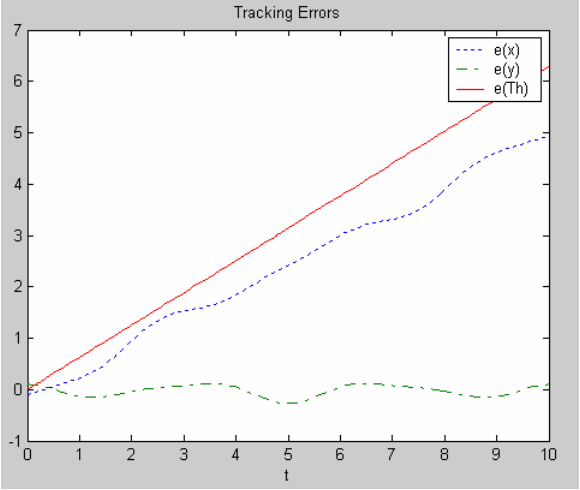

Figure 1: Tracking errors for WMR navigation using a PD CT controller.

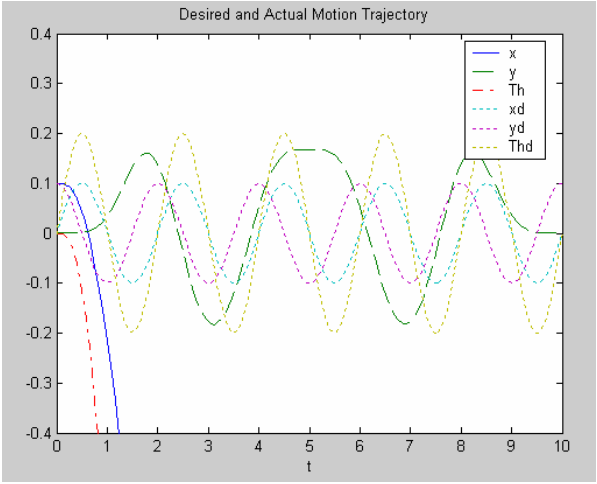

Figure 2: Desired versus actual motion trajectories for WMR navigation using a PD CT controller.

Increasing k_p to 4, and k_v to 10 result in zero tracking error for θ, while the tracking error for x is very small, and oscillating around zero. However, y is observed to have a high tracking error. Further increase k_p to 10, while reduce k_v to 1 reduce the tracking errors as shown in Fig. 3 and Fig. 4.

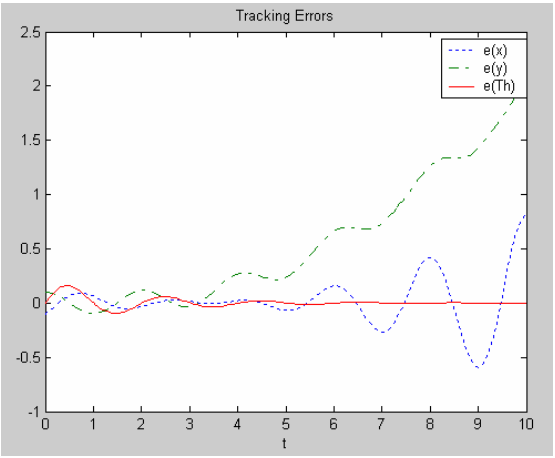

Figure 3: Tracking errors for WMR navigation using a PD CT controller.

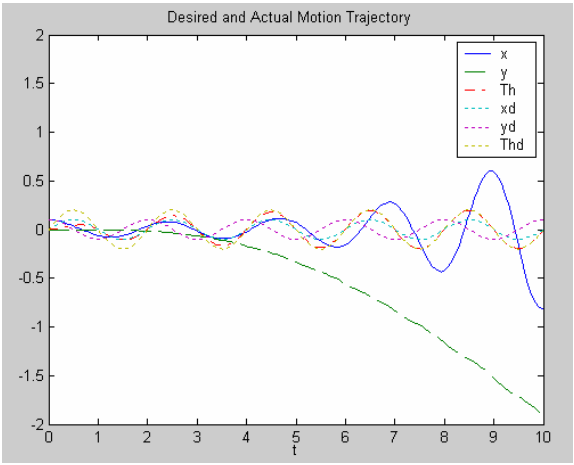

Figure 4: Desired and actual motion trajectories for WMR navigation using a PD CT controller.

As shown in these figures, the tracking errors are in the range of 0.00-2.00. The tracking error of θ is still zero. However, x has a greater oscillation about zero. In this case, the tracking error of y is reduced to 2.00, compared to 4.25 in the previous trial. Increasing the value of k_p to 100 and the value of k_v to 10, does not make the solution any better.

- In this set of experiments, different values for k_p and k_v are used for each components of the motion trajectory q. The two parameters k_{p1} and k_{v1} are used to control x. While the two parameters k_{p2} and k_{v2} are used to control y, k_{p3} and k_{v3} are used to control θ. Starting with k_{p1}=2, k_{v1}=1, k_{p2}=0, k_{v2}=10, k_{p3}=2, and k_{v3}=1, the tracking errors are in the range of 0.00-0.04, as shown in Fig. 5. The desired versus the actual motion trajectory is shown

in Fig. 6. These figures display good results, because the tracking errors are zero for both x and θ. Hence, the actual follow the desired motion trajectories for these two components. The tracking error for y oscillates around 0.02.

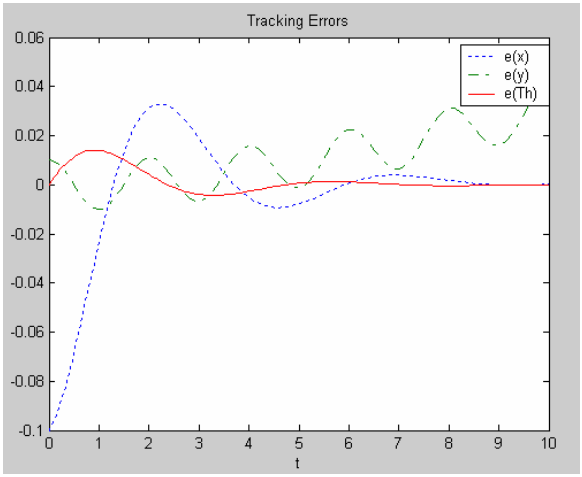

Figure 5: Tracking errors for WMR navigation using a PD CT controller.

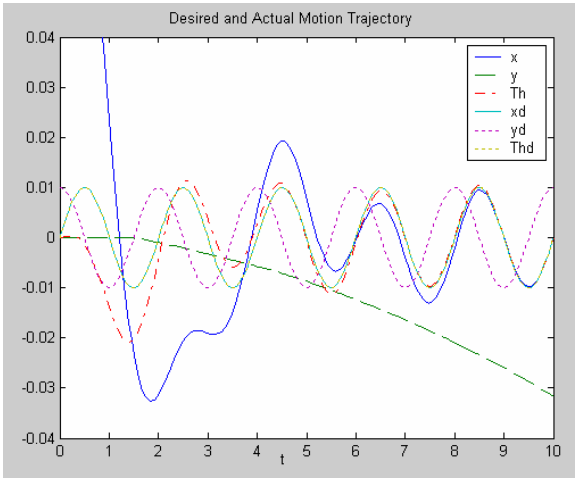

Figure 6: Desired and actual motion trajectories for WMR navigation using a PD CT controller.

Keeping k_{p1}=2, k_{v1}=1, k_{p2}=20, k_{p3}=2, and k_{v3}=1 and further increasing k_{v2} to 1000, increase the tracking errors to the range of 0.00 to 0.08.

3.2 Simulation for the PID CT controller

The PID CT controller simulation program for the WMR motion is developed in the same way that the PD CT controller simulation program was developed, except that Eq. (6) is used in calculating the motor torques. Several experiments are conducted on the PID CT simulation software. The robot parameters are according to Bearcat III. Different trajectories and controller parameters are tried, and the results are:

Case I: Using a sinusoidal desired motion trajectories:

$$q_d = \begin{bmatrix} x_{cd} \\ y_{cd} \\ \theta_d \end{bmatrix} = \begin{bmatrix} c \cdot \sin t \\ c \cdot \cos t \\ c \cdot \sin t \end{bmatrix}$$

where c is a constant.

Setting the value of k_p at 0 and both k_v and k_i at 10. The tracking errors are in the range of 0.00-2.75 as shown in Fig. 7. The desired versus the actual motion trajectories are shown in Fig. 8. As shown in these figures, the tracking error oscillates around zero for θ. The tracking error oscillates between 0.00 and -0.90 for x, and between 0.00 and 2.75 for y.

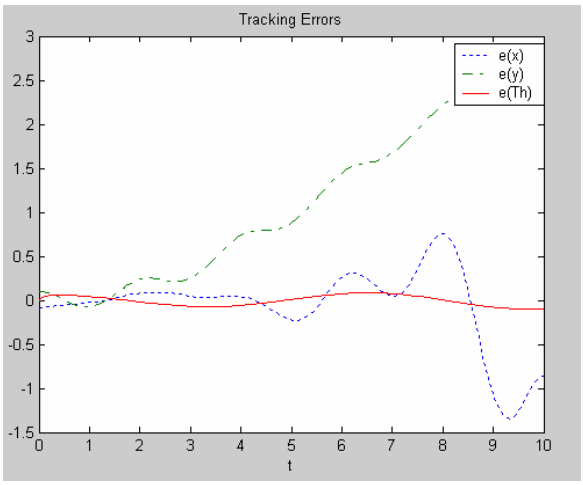

Figure 7: Tracking errors for WMR navigation using a PID CT controller.

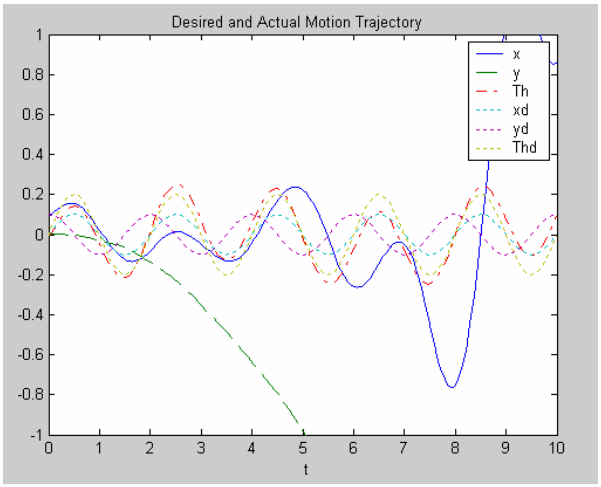

Figure 8: Desired versus actual motion trajectories for WMR navigation using a PID CT controller.

Increasing the value of k_p to 10, while keeping the same values for k_v and k_i ($k_v = k_i = 10$). Increases the tracking errors to the range of 0.00-3.15. Noticeable is an improvement in the tracking error of θ and x, and a small increase in the tracking error of y. However, the tracking error of y oscillates less than in the first case. Further increases k_p to 100, while keeping k_v at 10 and reducing k_i to 0 increases the tracking errors to the range of -17.00-2.50.

Case II: Another experiment is conducted by using $k_p = 2$, $k_v = 1$, and $k_i = 1$ for the following desired motion trajectories:

$$q_d = \begin{bmatrix} x_{cd} \\ y_{cd} \\ \theta_d \end{bmatrix} = \begin{bmatrix} c_1 \cdot t^2 \\ c_1 \cdot t^2 + c_2 \cdot t \\ c_3 \cdot \sin t \end{bmatrix}, \text{ where } c_1, c_2, \text{ and } c_3 \text{ are constants.}$$

The tracking errors are in the range of -0.25-0.00, as shown in Fig. 9. The desired versus the actual motion trajectories are shown in Fig. 10. As shown in these figures, the tracking errors for x and θ are very small, oscillating around zero. For y, the tracking error starts at zero and increases to around 0.25. However, it is still small.

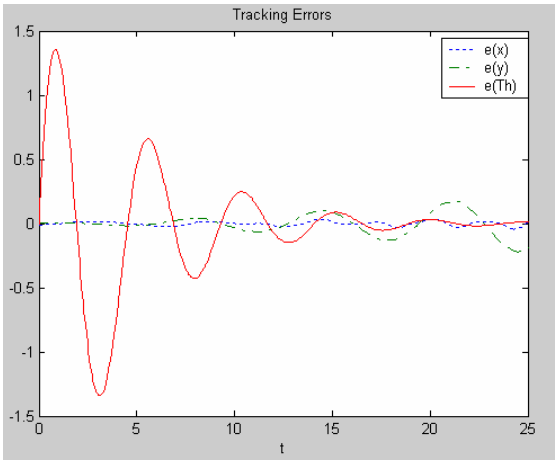

Figure 9: Tracking errors for WMR navigation using a PID CT controller.

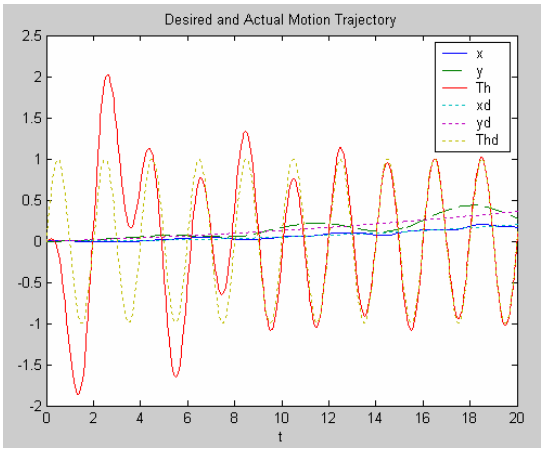

Figure 10: Desired versus actual motion trajectories for WMR navigation using a PID CT controller.

Case III: An experiment is conducted by using the same controller parameters as those in case II (k_p=2, k_v=1, and k_i=1) and using the following desired motion trajectory:

$$q_d = \begin{bmatrix} x_{cd} \\ y_{cd} \\ \theta_d \end{bmatrix} = \begin{bmatrix} c_1 \cdot t \\ c_2 \cdot t \\ c_3 \cdot \sin t \end{bmatrix}$$, where c_1, c_2, and c_3 are constants.

The tracking errors are in the range of -0.01-0.35, as shown in Fig. 11. The desired versus the actual motion trajectories are shown in Fig. 12. As shown in these figures, the tracking error for θ is zero. For x it oscillates around zero, and for y it starts at zero and increases to 0.35.

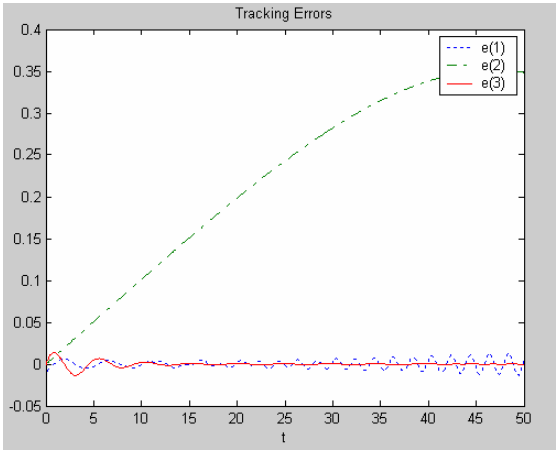

Figure 11: Tracking errors for WMR navigation using a PID CT controller.

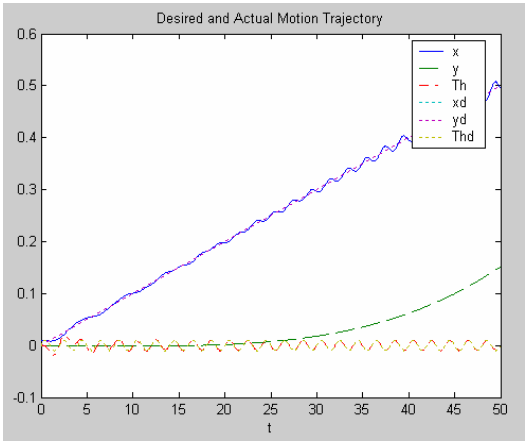

Figure 12: Desired versus actual motion trajectories for WMR navigation using a PID CT controller.

3.3 Simulation of the digital controller

Simulation software for the digital controller using Eq. (8) and the WMR dynamics as follows:

- The controller input is only updated at times εT .
- The controller computes the first control output by using the initial robot dynamics state values. The integrator then holds this value for its calculations over one sample time.
- For the next sample period, the final robot dynamics state values are assigned as the new initial robot dynamics state values. The integrator holds this value for its calculations over this sample time period, etc.

Case I: In the first experiment, the following desired motion trajectory is used:

$$q_d = \begin{bmatrix} x_{cd} \\ y_{cd} \\ \theta_d \end{bmatrix} = \begin{bmatrix} 0.01 \sin t \\ 0.01 \cos t \\ 0.01 \sin t \end{bmatrix}$$

k_p is set to 2, while k_v is set to 1, sample period selected is 20 msec, the results are shown in Figs. 13, 14. As shown in Fig. 13 the tracking errors for x and θ are almost zero. For y, the tracking error oscillates between 0.000 and -0.025. The actual motion trajectories are smooth and match the desired motion trajectories for x and θ. For y, the actual motion trajectory is also smooth, but higher than the desired motion trajectory.

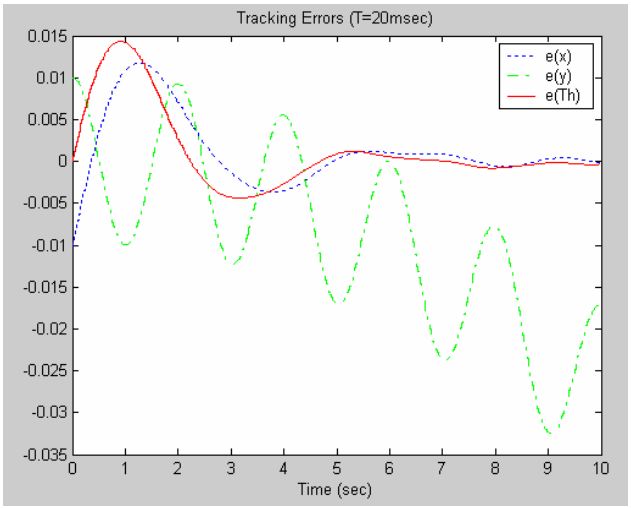

Figure 13: Tracking errors for WMR navigation, using a digital controller.

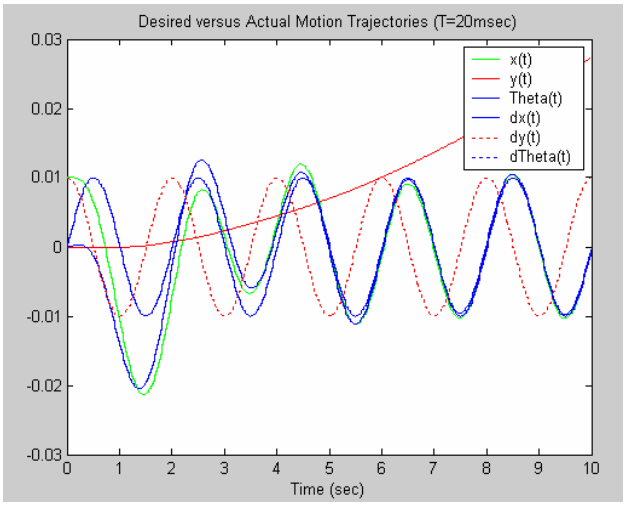

Figure 14: Desired versus actual motion trajectories for WMR navigation, using a digital controller.

Further increasing the derivative gain, k_v, from 1 to 100 does not improve the results.

To study the effect of the different amplitudes for the desired motion trajectories, another experiment is conducted with the following desired motion trajectory, and with the same controller parameters:

$$q_d = \begin{bmatrix} x_{cd} \\ y_{cd} \\ \theta_d \end{bmatrix} = \begin{bmatrix} 0.001\sin t \\ 0.001\cos t \\ 0.001\sin t \end{bmatrix}$$

The results are shown in Figs. 15, 16. As shown in Fig. 15, the tracking errors for x and θ are almost zero, while the tracking error for y oscillates around zero. The actual motion trajectories are smooth and match the desired motion trajectories for x and θ. For y, the actual motion trajectory is equal to zero. This result is actually interesting, since it shows that even the amplitudes of the desired motion trajectory have an effect on the tracking errors.

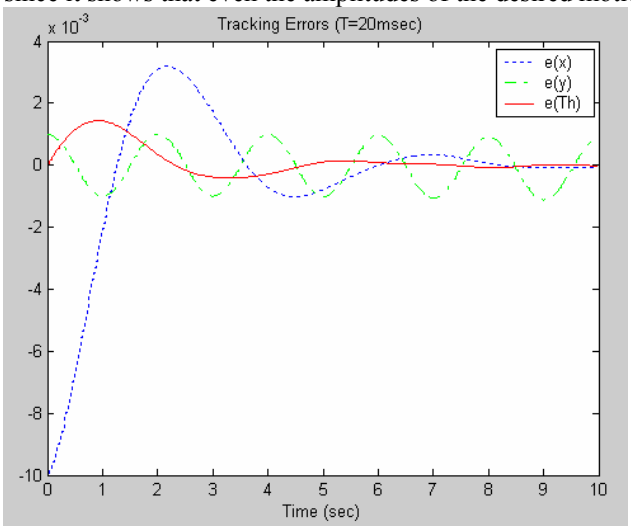

Figure 15: Tracking errors for WMR navigation, using a digital controller.

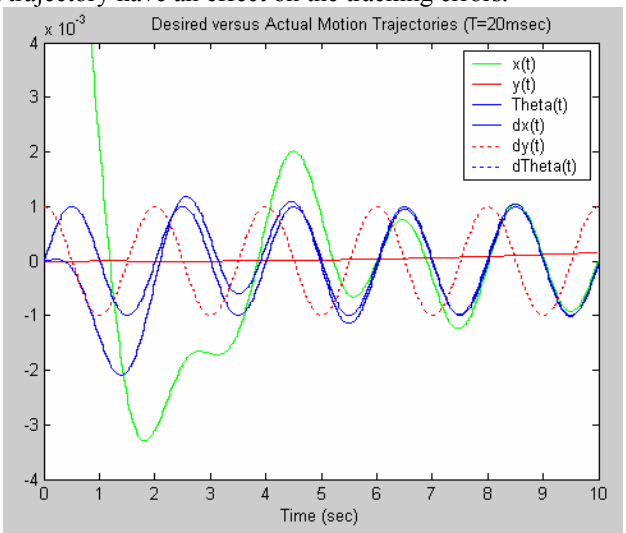

Figure 16: Desired versus actual motion trajectories for WMR navigation.

Case IV: To study the effect of different desired motion trajectories, another experiment is conducted, with the following desired motion trajectory is used:

$$q_d = \begin{bmatrix} x_{cd} \\ y_{cd} \\ \theta_d \end{bmatrix} = \begin{bmatrix} 0.005t^2 \\ 0.005t^2 + 0.008t \\ 0.001\sin t \end{bmatrix}$$

As shown in Fig. 17, the tracking errors for x and θ are zero, while the tracking error for y increases from 0.00 to 0.13. The actual motion trajectories are smooth and match the desired motion trajectories for x and θ. For y, the actual motion trajectory equals zero.

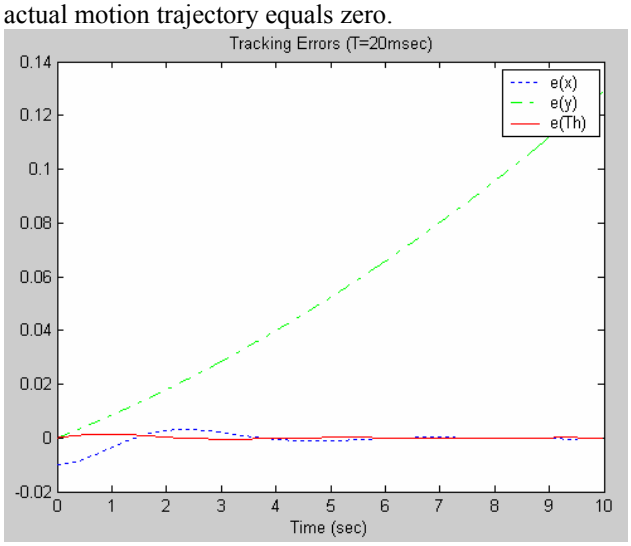

Figure 6.57: Tracking errors for WMR navigation, using a digital controller.

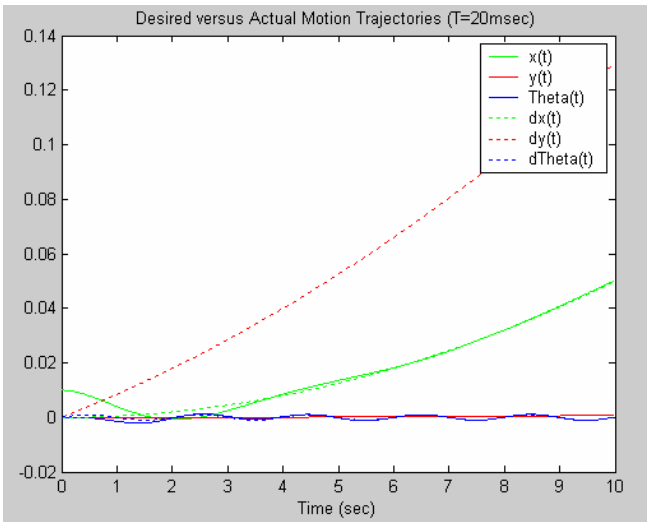

Figure 6.58: Desired versus actual motion trajectories for WMR navigation, using a digital controller.

Further increasing the proportional gain to 100 makes the system unstable were the tracking errors became very high, and the actual motion trajectories are very far from those desired.

3. Conclusions

In this paper, the development of a PD CT and PID CT controller for WMR autonomous navigation in an outdoor environment is presented. The controllers select suitable control torques, so that the WMR will follow the desired path from a navigation algorithm described in a previous paper. The controllers are tested under various control parameters and motion trajectories. Simulation results show that, for a PD CT controller, k_p and k_v need to be different for each component of the motion trajectory q. Small positive values are needed in order to get good results. The gain parameters k_{p1}=2, k_{v1}=1, k_{p3}=2, and k_{v3}=1 are observed to give reasonable results. The values of parameters k_{p2} and k_{v2} should not be too large. However, setting the values of k_{p2} at 0 and k_{v2} at 10 produces the best results.

Simulation results for a PID CT controller shows that increasing k_p and fixing k_v and k_i, or increasing k_v and fixing k_p and k_i, reduces the tracking errors for θ and x, while it increases the tracking error of y. However, there is a limit to this increase, which is about 10. Using very large, or zero, values for k_p, k_v, or k_i is not recommended. k_p=2, k_v=1, and k_i=1 seem to give very reasonable results.

For a PD digital controller, it is noticed that k_p =2 and k_v =1 are the best parameters. Increasing the value of k_v does not improve results. However, increasing the value of k_p does, provided its value does not exceed 50.

Comparing the performance of the three controllers shows that the digital controller provides the best results. Hence, its use is recommended for this application.

The selection process of proper parameters for the controller is a challenging task, since there are too many parameters to be changed at each trial. Although x , y , and θ are independent of each other, they are, nevertheless, used to update the new values of the actual motion trajectories. Hence, they have an impact upon each other. Therefore, keeping the same value of k_p and k_v for a particular motion trajectory component does not imply that the actual motion trajectory will remain the same. Finding a good method for calculating the best values for the controller parameters is recommended.

4. References

1. F. L. Lewis, S. Jagannathan, and A. Yesildirek, *Neural Network Control of Robot Manipulators and Nonlinear Systems*, pp. 147-167, Padstow, UK: Taylor and Francis Ltd, T. J. International Ltd, 1999.
2. H.-S. Shim and Y.-G. Sung, "Asymptotic control for wheeled mobile robots with driftless constraints," Robotics and Autonomous Systems, Vol. **43**, Issue 1, pp. 29-37, 2003.
3. H. Yun-Su and S. Yuta, "Trajectory tracking control for navigation of the inverse pendulum type self-contained mobile robot," Robotics and Autonomous Systems, Vol. **17**, Issue 1-2, pp. 65-80, 1996.
4. R. Rajagopalan and N. Barakat, "Velocity control of wheeled mobile robots using computed torque control and its performance for a differentially driven robot," Journal of Robotic Systems, Vol. **14**, Issue 4, pp. 325-340, 1997.
5. M. Zhang and R. M. Hirschorn, "Discontinuous feedback stabilization of nonholonomic wheeled mobile robots," Dynamics and Control, Vol. **7**, Issue 2, pp. 155-169, 1997.
6. K. C. Koh and H. S. Cho, "A smooth path tracking algorithm for wheeled mobile robots with dynamic constraints," Journal of Intelligent and Robotic Systems, Vol. **24**, Issue 4, pp. 367-385, 1999.
7. Y. Zhang, J. H. Chung, and S. A. Velinsky, "Variable structure control of a differentially steered wheeled mobile robot," Journal of Intelligent and Robotic Systems, Vol. **36**, Issue 3, pp. 301-314, 2003.
8. M. L. Corradini and G. Orlando, "Robust tracking control of mobile robots in the presence of uncertainties in the dynamical model," Journal of Robotic Systems, Vol. **18**, Issue 6, pp. 317-323, 2001.
9. M. L. Corradini and G. Orlando, "Control of mobile robots with uncertainties in the dynamical model: a discrete time sliding mode approach with experimental results," Control Engineering Practice, Vol. **10**, Issue 1, pp. 23-34, 2002.
10. M. L. Corradini, T. Leo, and G. Orlando, "Experimental testing of a discrete-time sliding mode controller for trajectory tracking of a wheeled mobile robot in the presence of skidding effects," Journal of Robotic Systems, Vol. **19**, Issue 4, pp. 177-188, 2002.
11. S.-F. Wu, J.-S. Mei, and P.-Y. Niu, "Path guidance and control of a guided wheeled mobile robot," Control Engineering Practice, Vol. **9**, Issue 1, pp. 97-105, 2001.
12. Z.-P. Jiang, E. Lefeber, and H. Nijmeijer, "Saturated stabilization and tracking of a nonholonomic mobile robot," Systems and Control Letters, Vol. **42**, Issue 5, pp. 327-332, 2001.
13. W. Dong, W. Liang Xu, and W. Huo, "Trajectory tracking control of dynamic non-holonomic systems with unknown dynamics," International Journal of Robust and Nonlinear Control, Vol. **9**, Issue 13, pp. 905-922, 1999.
14. P. J. Werbos, "New designs for universal stability in classical adaptive control and reinforcement learning," Proceedings of the International Joint Conference on Neural Networks IJCNN, USA, Vol. **4**, pp. 2292-2295, 1999.
15. M. B. Montaner and A. Ramirez-Serrano, "Fuzzy knowledge-based controller design for autonomous robot navigation," Expert Systems with Applications, Vol. **14**, Issue 1-2, pp. 179-186, 1998.
16. A. V. Topalov, J.-H. Kim, and T. P. Proychev, "Fuzzy-net control of non-holonomic mobile robot using evolutionary feedback-error-learning," Robotics and Autonomous Systems, Vol. **23**, Issue 3, pp. 187-200, 1998.
17. M. Toda, O. Kitani, T. Okamoto, and T. Torii, "Navigation method for a mobile robot via sonar-based crop row mapping and fuzzy logic control," Journal of Agricultural Engineering Research, Vol. **72**, Issue 4, pp. 299-309, 1999.

18. F. Vázquez and E. Garcia, “A local guidance method for low-cost mobile robots using fuzzy logic,” Annual Review in Automatic Programming, Vol. **19**, pp. 203-207, 1994.
19. S. M. Alhaj Ali, *Technologies for autonomous navigation in unstructured outdoor environments*, Ph.D. dissertation, University of Cincinnati, Cincinnati, OH, 2003.
20. S. M. Alhaj Ali, and E. L. Hall, “Technologies for autonomous operation in unstructured outdoor environments, Part I: Navigation,” Vol. 12, pp. 57-62, Proceedings of the Artificial Neural Networks in Engineering Conference, 2002.