Implementation of a Class of True Digital Control (TDC) in the Navigation of a Ground Vehicle

A. Abdelhamid[a], E. M. Shaban[b,*], K.M. Zied[c], Younes Khalil[a]

[a]Mechanical Design Department, Faculty of Engineering (Mattaria),Helwan University, Cairo – Egypt
[b]Mechanical Engineering Department, Jazan University, Jazan – KSA, On leave from Faculty of Engineering (Mataria), Helwan University – Egypt
[c]Mechanical Engineering Department, Taibah University, Yanbu Branch, Yanbu – KSA
*Corresponding author: Tel.: +966-59 737 5085 or +20-100 900 7574; Fax. +966-7 323 3386
Email address: modern3@hotmail.com or eeshaban@jazanu.edu.sa

Abstract

In this research, experimental work and simulations are carried out to evaluate the Proportional–Integral–Plus (PIP) control methodology which has been applied to a ground vehicle, which may be used in landmines detection. The objective of this work is to evaluate the controller in terms of tracking performance. A primitive route planning algorithm has been proposed to control the vehicle when it encounters an obstacle in its path. This algorithm of collision-free path is automatically generated, given the location of the obstacle. The approach utilizes an experimental data-based model of the vehicle in discrete–time which suitable for PIP control system design. The results of the experimental study have shown that the designed trajectory is comfortably met by using a straightforward fixed gains linear PIP algorithm. The vehicle starts to maneuver and it stops at the desired position at the desired orientation with a small acceptable margin of error which proves the validity of the PIP control algorithm.

Keywords: control system design, system identification, discrete–time transfer function system, non–minimal state space, proportional–integral–plus control

1. Introduction

Ground vehicles can be used for many applications which are inconvenient and dangerous, such as landmines detection. The vehicle should have a set of sensors for navigation, obstacle avoidance, and observing the environment. This is not the case in this study, since the sensors are still in the installation state. The existence of sensors helps the vehicle to make decisions or pass the information to human operator to control the vehicle remotely. This is, consequently, enhances the safety of operators since the ground vehicles do not entail manual operation. Therefore, the full or partial automation of ground vehicles attracts high prospect in research and industry. The current work concerns the navigation of such vehicles; the primary factors considered here are motion planning, modelling and control, and targeting.

The problem of ground vehicles' trajectory planning is extremely challenging. Kanayama et al. [1, 2] introduce continuous paths based on clothoid curves and cubic spirals. A clothoid is a curve whose curvature is proportional to its arc length and it is commonly used for designing highways and railway tracks [3].

Nelson [4] proposed two types of curvature paths; they are polar polynomials representing circular arc segments and Cartesian polynomials representing arc–arc or arc–line–arc segment for lane–change manoeuvres. Lamiraus and Laumond [5] present the generation of a smooth path for car–like vehicles subjected to curvature constraints. They consider the car as 4–D system from a kinematic point of view and as 3–D system from a geometric point of view to compute a collision–free C^2 path.

The paths computed by Fraichard and Scheuer [6] consist of line segments, circular arcs, and clothoid arcs. Although they are not optimal in length, it is shown that they converge towards the optimal *Reeds and Shepp* paths [7], when the upper limit of curvature derivative tends to infinity.

Yamamota et al. [8] study a quasi–time–optimal trajectory planning method for generating the paths of two independently driven wheel type model platform using cubic B–spline curves. In their study, the upper limit on the curvature is not considered.

An attempt of controlling these vehicles in planar motion was accomplished by Mohammed Eghtesad et al. [9], for which each wheel is assumed as a conventional wheel with only two–degrees of freedom (DOF), its third DOF in a planner pair model (wheel and the ground) is restricted when the sideway component of the wheel velocity is

considered zero to represent no sideway slip. This assumption, together with the assumption that the wheel rolls without slip, is called ideal rolling condition. Ideal rolling conditions for wheels impose the nonholonomic constraints on the motion of the vehicle. Nonholonomic systems are found not to be asymptotically stabilizable by smooth state feedback laws [10, 11, 12]. Dynamics–based models are useful not only for verification of ideal rolling assumptions, but also for taking into consideration dynamic constraints like motor–torque limits and for including dynamic components in the model. If the path of the vehicle contains some relatively sharp turns, a dynamic–based controller results in more efficient motion control for avoiding wheel–ground contact loss [13].

A primitive algorithm to generate a path in terms of kinematic and geometric constraints of both the vehicle and the obstacle is introduced. The algorithm is presumed to be generated automatically, once an obstacle is observed forwardly. Subsequently, the controller plays its role and the vehicle manoeuvres tracking the proposed trajectory. It is convenient to note here that the obstacle is assumed to have cubic shape–like with side length does not exceed 3 m. The proposed path of the vehicle is shown in Figure 1. The dimensions given in figure for the path are constant.
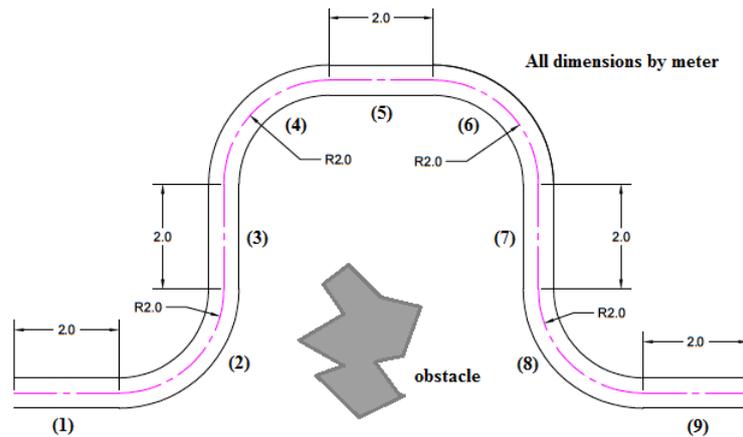

Figure 1. A schematic drawing shows the path planning of the ground vehicle, when an obstacle is observed.

2. The PIP Control Methodology

The proportional–integral–plus (PIP) control is applied as a sort of true digital control (TDC). The PIP controller can be interpreted as a logical extension of conventional PI/PID algorithms, but with inherent model–based prediction control action [14, 15]. Here, non–minimal state–space (NMSS) models are formulated so that full state variable feedback control can be implemented directly from the measured input and output signals of the controlled process without resorting of the design of a deterministic state reconstructor (observer) or a stochastic Kalman filter.

Over the last few years, such NMSS/PIP control system has been successfully employed in a range of practical and simulation studies [e.g. 16–20]. The target of the PIP control system is controlling the speed of the front two wheels (right and left) in order to track the designed trajectory, see Figure 1. The relative angular velocity between the front two wheels is used for manoeuvring the vehicle left or right during the curvature path. However, in case of equal angular velocities for both right and left wheels, the vehicle moves in a straight line path. The general layout of the practical application is shown in Figure 2.

Here, a discrete–time PIP algorithm satisfies all the performance requirements at both straight line and curvature paths with acceptable error. The solution involved a very simple design procedure for the controller, with just one weighting term used to straightforwardly tune the closed loop response, using recursive steady state solution of the Algebraic Riccati Equation (ARE) [21].
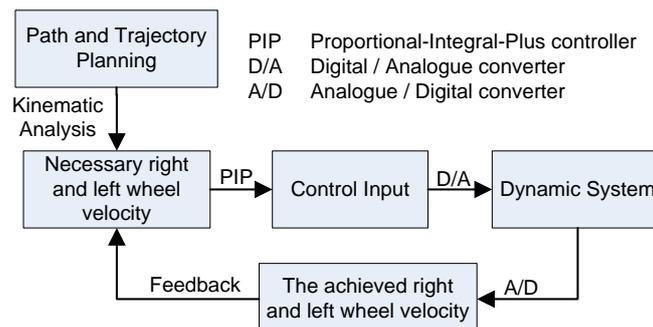

Figure 2. General layout of the ground vehicle practical demonstrator.

3. System Description

A three–wheeled vehicle system, with two front wheels used for both driving and steering, is used for simulation and experiments. The whole system is composed of a rigid structure, two front wheels, one rear wheel, articulated ground penetrating radar (scraper) installed upon the vehicle, power supply, and $12\,Volt$ battery as a power supply, see Figure 3. The two front wheels drive the whole system forward and backward in a straight line path by means of two independent electric motors. Here, the motors drives the front right and left wheels with the same angular velocity. However, in case of curvilinear path, the two motors drive the front two wheels in distinct angular velocities. The relative velocity between the two front wheels steers the whole system right or left to manoeuvre the vehicle in case of obstacle avoidance.
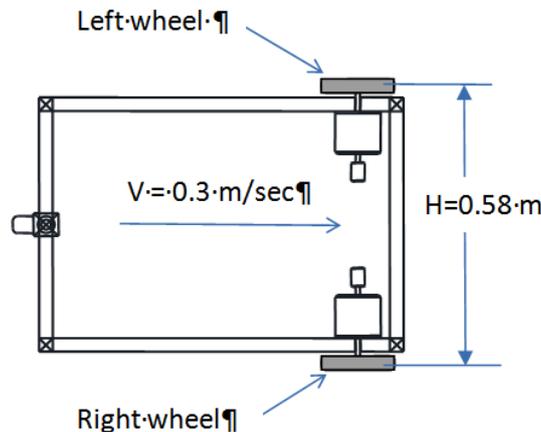
Figure 3. A schematic for the three–wheeled vehicle.

For the sake of manoeuvring and tracking of the designed trajectory, two tachometers are attached to the dc–motors for angular velocities feedback for right and left wheels. The tachometers together with the dc-motors are connected to a PC via LapJack U12 data acquisition unit and interface signal conditioning boards for input/output signal smoothing and amplification. A general layout of the system with its peripherals is shown in Figure 4.

3.1 DC motors

Each motor is itself working as single-input, single-output, SISO, therefore only one DC motor of type DGM-204-2A is used to drive each wheel. The motor specification is: TOSHIBA DC 24 V, 0.85 A, its rated speed is 22 rpm. As shown in Figure 3, only 12 volt is used to drive the system. This 12 volt is supplied via the DAQ system after raising the volt by means of Voltage Rising Board existed in signal conditioning unit.
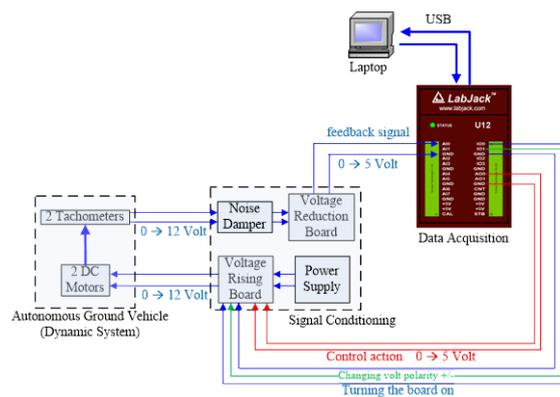
Figure 4. A general layout of the ground vehicle system with its attached hardware peripherals.

3.2 Tachometers

The motor speed is measured directly by means of tachometer. Tachometers do not require power supply; however it produces a voltage which is linearly proportioned to the measured angular velocity. The output signal from the tachometer ranges from $0 \rightarrow 12\,Volt$ after passing through Noise Damper, see Figure 4, and then the signal passes through Voltage Reduction Board for conditioning, which reduces the signal to range from $0 \rightarrow 5\,Volt$, before forwarding to the Data Acquisition DAQ card.

3.3 Data acquisition

Figure 4 shows the top surface of the data acquisition system, LabJack U12, used in the experimental work.

3.4 Software interface

The control aims are achieved by using MATLAB software. The feedback signals are received by MATLAB, and then appropriate control actions are sent to the motors according to the control law programmed in the associated m-file. A flowchart showing the logic of the control system is depicted in Figure 5.

4. Path Planning for the System

Given the path used for obstacle avoidance, see Figure 1, it is possible kinematically to obtain the required consequent angular velocities for the ground vehicle's right and left wheels, i.e. ω_R and ω_L.

Suppose V and Ω denote, respectively, the given linear velocity of the system and its consequent angular velocity during curvilinear path, see Figure 6. Here, in case of curvilinear path with radius R, the vehicle angular velocity becomes $\Omega = V/R$. Consequently, the linear velocity for the ground vehicle's right wheel is

$$V_R = \Omega\left(R \pm \frac{H}{2}\right) \tag{1}$$

Given that H is the distance between right and left wheels, see Figure 3. Also, +ve sign is used in case of turning left, however –ve sign is used in case of turning right. Therefore, the angular velocity for vehicles' right wheel is then

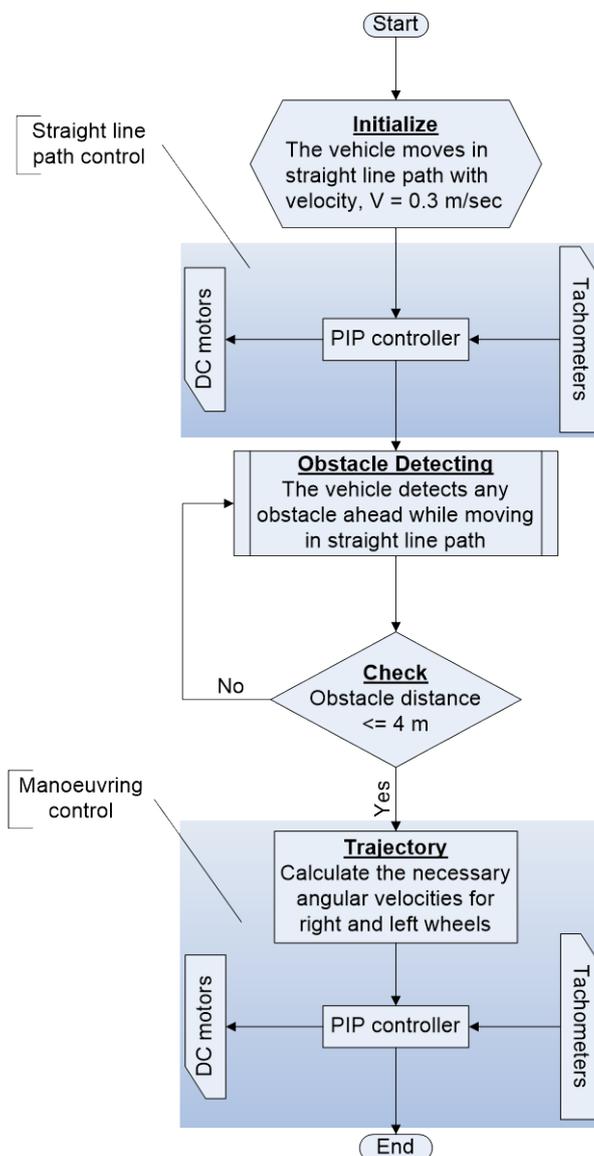$$\omega_R = \frac{\Omega}{r}\left(R \pm \frac{H}{2}\right) \tag{2}$$

Figure 5. Flowchart showing the trajectory following algorithm for the ground vehicle.

It is possible rewrite equation (2) as

$$\omega_R = \frac{V}{r} \pm \frac{V H}{2 R r} \qquad (3)$$

Similarly, the angular velocity for vehicles' left wheel is

$$\omega_L = \frac{V}{r} \mp \frac{V H}{2 R r} \qquad (4)$$

for which –ve sign is used in case of turning left, however +ve sign is used in case of turning right.

During the straight line path, the radius of curvature equals infinity, i.e. $R = \infty$, therefore the vehicles' angular velocity is zero, $\Omega = 0$. Substituting in equations (3 and 4), the angular velocities for vehicles' right and left wheels are then
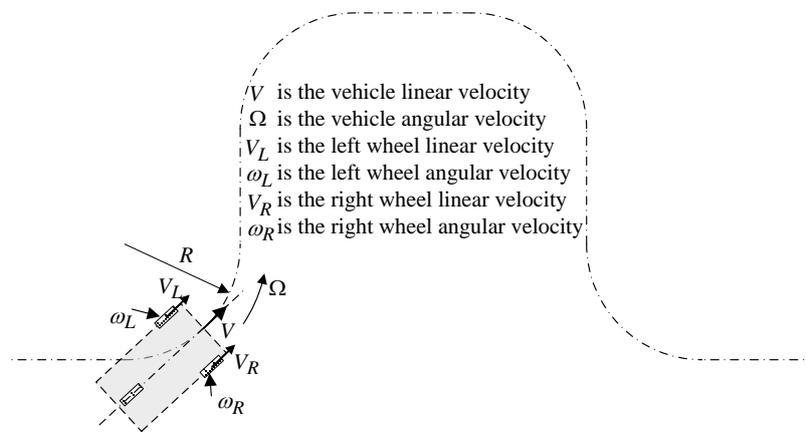
$$\omega_R = \omega_L = \frac{V}{r} \qquad (5)$$

V is the vehicle linear velocity
Ω is the vehicle angular velocity
V_L is the left wheel linear velocity
ω_L is the left wheel angular velocity
V_R is the right wheel linear velocity
ω_R is the right wheel angular velocity

Figure 6. The kinematic representation of the ground vehicle system during curvilinear path.

5. True Digital Control (TDC)

The True Digital Control (TDC) approach to control system design is carried out entirely in discrete time, starting from the identification and estimation of a suitable linearized model to the real-time implementation of the final control law. Therefore, as a first step, it is required to find a linearized single-input single output (SISO) representation of the dynamic system based on a transfer function (TF) model, with parameters identified and estimated from the measured data. In terms of backward shift operator, z^{-1}, this TF model takes the form of (6).

$$y_k = \frac{B(z^{-1})}{A(z^{-1})} u_k \qquad (6)$$

where y_k is the output variable, u_k the control input and,

$$A(z^{-1}) = 1 + a_1 z^{-1} + a_2 z^{-2} + \cdots + a_n z^{-n}$$
$$B(z^{-1}) = b_1 z^{-1} + b_2 z^{-2} + \cdots + b_m z^{-m} \qquad (7)$$

in which $a_1 \cdots a_n$ and $b_1 \cdots b_m$ are the TF parameters, while z^{-1} is the backward shift operator, i.e. $z^{-i} y_k = y_{k-i}$.

An appropriate structure for the transfer function (6) needs to be defined, i.e. the triad $\{n, m, \delta\}$ where δ is the pure time delay, typically represented by setting $b_1 \cdots b_{\delta-1} = 0$. The two main statistical measures employed to help determine these values are the coefficient of determination R_T^2, based on the response error, which is a simple measure of model fit; and the Young Identification Criterion (YIC), which provides a combined measure of fit and parametric efficiency, with large negative values indicating a model which explains the output data well, without over-parameterization [22]. The present work utilizes the Simplified Refined Instrumental Variable (SRIV) algorithm to estimate the model parameters [22]–[23].

These statistical tools and associated estimation algorithms have been assembled in the Matlab® software as the CAPTAIN toolbox. It could be found at http://www.es.lancs.ac.uk/cres/captain/.

It is possible to show that the SISO model (6) can be represented by the following linear Non–Minimal State Space (NMSS) equations,

$$\mathbf{x}_k = \mathbf{F}\mathbf{x}_{k-1} + \mathbf{g}u_{k-1} + \mathbf{d}y_{d,k}$$
$$y_k = \mathbf{h}\mathbf{x}_k$$

(8)

for which the matrix $\mathbf{F}$, and the vectors $\mathbf{g}$, $\mathbf{d}$ and $\mathbf{h}$ are defined by [24]. Here, the $n+m$ dimensional *non-minimal* state vector $\mathbf{x}_k$, consists of the present and past sampled values of the input and output variables, i.e.,

$$\mathbf{x}_k = \begin{bmatrix} y_k & y_{k-1} & \cdots & y_{k-n+1} & u_{k-1} & \cdots & u_{k-m+1} & z_k \end{bmatrix}^T$$

(9)

Here, $z_k = z_{k-1} + \{r_k - y_k\}$ is the *integral-of-error* between the reference r_k and the sampled output y_k. Type 1 servomechanism performance is introduced by means of the *integral-of-error* state z_k. If the closed-loop system is stable, then this ensures that steady-state tracking of the command level is inherent in the basic design. The control law associated with the NMSS model (8) takes the usual State Variable Feedback (SVF) form,
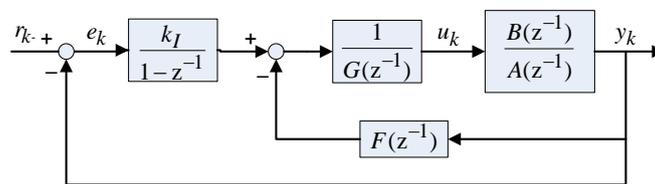
$$u_k = -\mathbf{k}\, x_k$$

(10)

Figure 7. The PIP control system implementation.

where $\mathbf{k} = [f_0\ f_1 \cdots f_{n-1}\ g_1 \cdots g_{m-1}\ -K_I]$ is the SVF control gain vector. In more conventional block-diagram terms, the SVF controller (10) can be implemented as shown in Figure 7, where it is clear that it can be considered as one particular extension of the ubiquitous PI controller, where the PI action is enhanced by the higher order forward path and feedback compensators $1/G(z^{-1})$ and $F(z^{-1})$ where,

$$F(z^{-1}) = f_0 + f_1 z^{-1} + \cdots + f_{n-1} z^{-(n-1)}$$
$$G(z^{-1}) = 1 + g_1 z^{-1} + \cdots + g_{m-1} z^{-(m-1)}$$

(11)

However, because it exploits fully the power of SVF within the NMSS setting, PIP control is inherently much more flexible and sophisticated, allowing for well-known SVF strategies such as closed loop pole assignment, with decoupling control in the multivariable case; or optimization in terms of a Linear-Quadratic (LQ) cost function of the form,

$$J = \frac{1}{2} \sum_{i=0}^{\infty} \left\{ \mathbf{x}_i^{\mathrm{T}} \mathbf{Q} \mathbf{x}_i + R u_i^2 \right\}$$

(12)

where $\mathbf{Q} = diag[q_1 \ldots q_n\ q_{n+1} \ldots q_{n+m-1}\ q_{n+m}]$ is a diagonal state weighting matrix and R is an additional scalar weight on the input. The resulting SVF gains are then obtained recursively from the steady state solution of the Algebraic Riccati Equation (ARE), derived from the standard LQ cost function (12) as follows [26],

$$\mathbf{k} = \left[\mathbf{g}^T\, \mathbf{P}^{(i+1)}\, \mathbf{g} + R\right]^{-1} \mathbf{g}^T\, \mathbf{P}^{(i+1)}\, \mathbf{F}$$
$$\mathbf{P}^{(i)} = \mathbf{F}^T\, \mathbf{P}^{(i+1)} \left[\mathbf{F} - \mathbf{g}\mathbf{k}\right] + \mathbf{Q}$$

(13)

for which $\mathbf{P}$ is a symmetrical positive-definite matrix with the initial value, $\mathbf{P}^{(i+1)}$, equal to the weighting matrix $\mathbf{Q}$ and $\mathbf{k}$ is the control gain vector.

6. System Identification and Estimation

The LabJack U12 and its hardware peripherals limit the sampling rate to 20 samples/second. This shows sufficient sampling rate for the system regarding similar dynamic systems [17–20].

In the first instance, Pseudo Random Binary Signal (PRBS) has been used to excite both dc motors which drive and manoeuvre the AVG. Therefore, the data were collected at different magnitudes to average the TF parameters. The SRIV algorithm together with the *YIC* and R_T^2 identification criteria reveal that a first order TF model with one numerator having one sample delay provides the best estimated model and most optimum fit to the data across a wide range of operating conditions. Therefore, the TF for both dc motors, based on (6), is

$$y_k = \frac{b_1 \, z^{-1}}{1 - a_1 \, z^{-1}} u_k \tag{14}$$

where y_k is the angular velocity of the dc motor, in rpm, and u_k is the normalized voltage applied to the dc motor scaled from 0 to 100. The parameters $a_1 = -0.2507$ and $b_1 = 0.9572$ for the right dc motor yields the best overall fit for the full range of operating conditions. Similarly, the parameters $a_1 = -0.25$ and $b_1 = 1.2639$ are used for the left dc motor, see Figure 8.

7. PIP Controller Design and Simulation

In this case, the linear NMSS equations are given by (8) with the state vector $x_k = \begin{bmatrix} y_k & z_k \end{bmatrix}^T$. This NMSS representation is controllable since the two conditions of controllability are applied for the whole range of linearized input, [26]. The SVF–PIP control law for the two dc motors is given by

$$\begin{aligned} u_k &= -\mathbf{k}\,\mathbf{x}_k \\ &= -\begin{bmatrix} k_p & -k_I \end{bmatrix} \begin{bmatrix} y_k \\ z_k \end{bmatrix} \end{aligned} \tag{15}$$
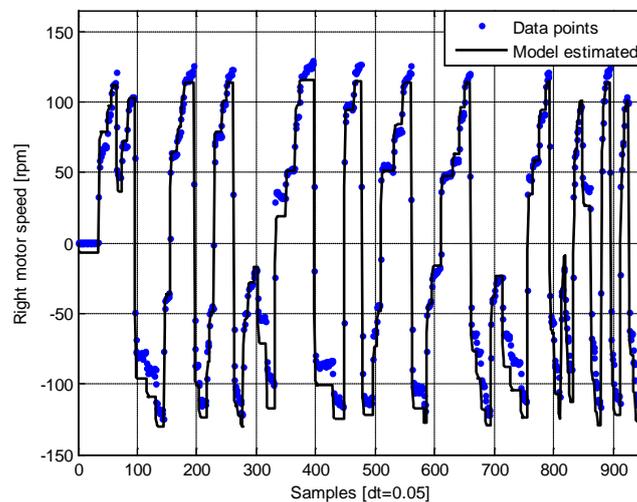
(a) Right hand side DC motor, $R_T^2 = 0.97$
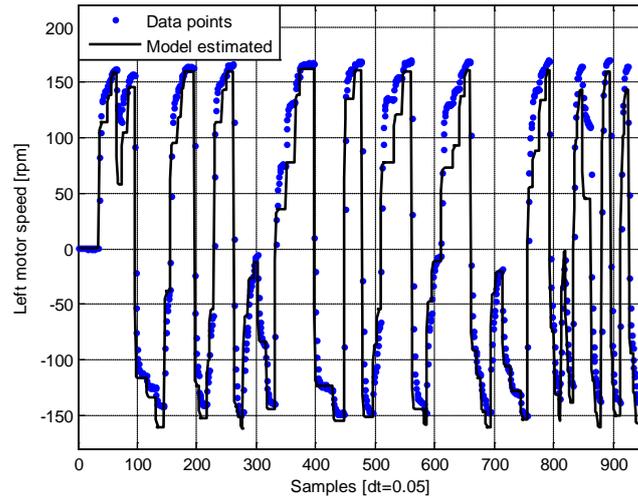
(b) Left hand side DC motor, $R_T^2 = 0.94$

**Figure 8. The response of the model (14) [solid], with respect to the data collected, using PRBS input.
The sampling rate is 20 samples/second.**

The problem of optimization is defined as follows: for a linear SISO discrete NMSS form defined in (8), it is required to find the control law in equation (15) with optimal SVF control gain vector that minimizes the quadratic cost function (12). The resulting SVF gains are then obtained recursively from the steady state solution of the ARE (13) derived from the standard LQ cost function (12). Trial and error experimentation suggests that setting $Q = diag \begin{bmatrix} 700 & 1 \end{bmatrix}$ and $R = 10$ for right dc motor, and $Q = diag \begin{bmatrix} 50 & 1 \end{bmatrix}$ and $R = 1$ for left dc motor yield a suitably fast PIP–SVF gain vectors,

$$\begin{aligned} \mathbf{k} &= \begin{bmatrix} 0.2580 & -0.0384 \end{bmatrix} \quad \text{for right dc motor} \\ \mathbf{k} &= \begin{bmatrix} 0.1957 & -0.1036 \end{bmatrix} \quad \text{for left dc motor} \end{aligned} \tag{16}$$

However, the two DC motors used are the same, the two gain vectors defined in (16) are different because of the differences existing in the elements of the diagonal weighting matrix, $\mathbf{Q}$, and scalar input weight, R., as well as the differences in their TF parameters. This may arise due to the differences in their inherent dynamic performance.

The simulation of the model (14) for both right and left motors, using gain vectors in (16), is shown in Figure 9, for which the simulation shows satisfactory tracking performance with acceptable settling time for both left and right wheels
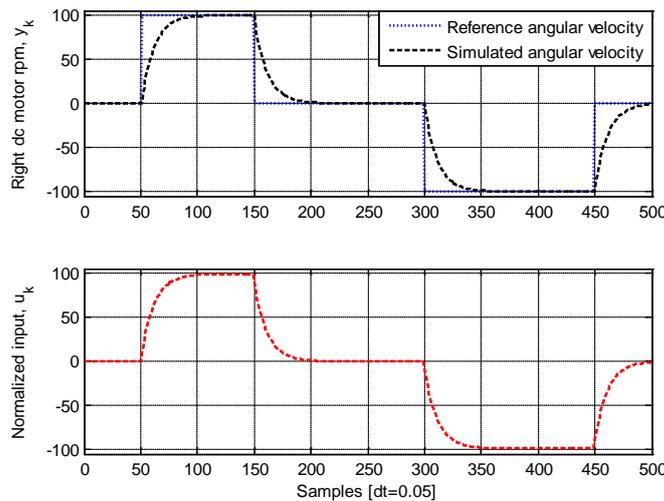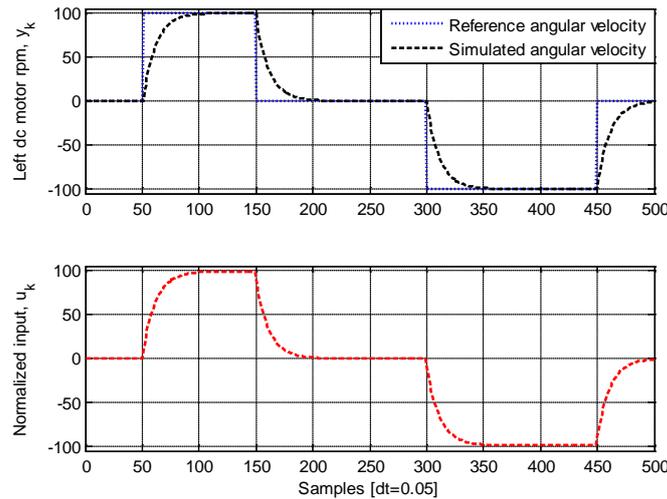
Figure 9. The simulation result of model (14) for both dc motors using the gain vector defined in (16).

8. Implementation

The delay caused by the computer system, during digital control, can lead to significant performance limitation. Therefore, the system I/O time constraints must be taken into consideration, [27]. For the sake of real-time implementation, a digital control program using Matlab® is created at equidistant sampling intervals, 20 Hz, with constant control delay from sampling to actuation. The flowchart showing the logic of the program created for trajectory following algorithm is depicted in Figure 5, for which the program instantaneously calculates the required vehicles' right and left wheel velocities according to the trajectory planned. The angular velocity feedback, for both right and left wheels, is always compared to the reference angular velocity. This error action together with the gradient of output angular velocity constitute the control action, which is performed in an incremental form as,

$$u_k = u_{k-1} + k_I (r_k - y_k) - k_p (y_k - y_{k-1}) \qquad (17)$$

The implementation of PIP controller is depicted in Figure 10, for which the right and left motors' angular velocity tracks their calculated reference values according to the path planned and its associated kinematics equations (3), (4), and (5)
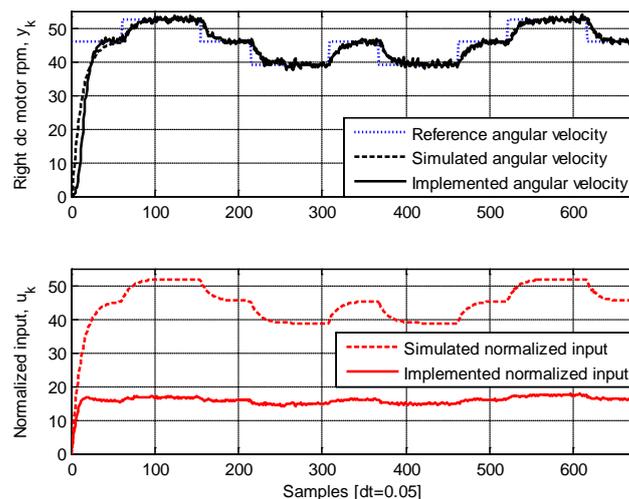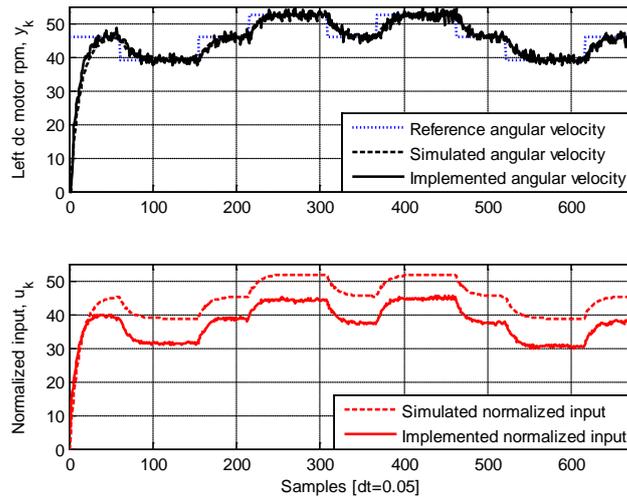
Figure 10. The implementation result of system. The practical responses of both dc motors using the gain vector defined in (16).

9. Results and Discussions

As shown in Figure 10, a satisfactory tracking performance application is achieved, for which the right and left motors' angular velocity tracks their set values at various levels within $0 \rightarrow 55$ rpm , with acceptable synchronized settling time response, about 40 samples (2 seconds), for both right and left wheels. Moreover, the simulated response and practical implementation show no considerable difference throughout the period of implementation. However, the implemented control actions for both motors are always less than the simulated control actions.

The forward kinematics equations (3, 4, and 5) can be used, with the aid of the measured angular velocities of the right and left wheels of the vehicle, to draw the trajectory achieved by the system using PIP controller, see Figure 11. The figure shows a comparison between the trajectory achieved by the system and the trajectory planned, see Figure 1. In this figure, a satisfactory tracking for the planned path is depicted, with a small deviation in the orientation of the vehicle, about 1.5^{o}, at the end of the path. This deviation takes place because of the relatively late settling time, about 3.5 sec, of the resolved x-component of the vehicle velocity, V_x , at the beginning of the path., see Figure 12. However, the other resolved y-component, V_y , tracks its reference values throughout the entire path, see Figure 12.
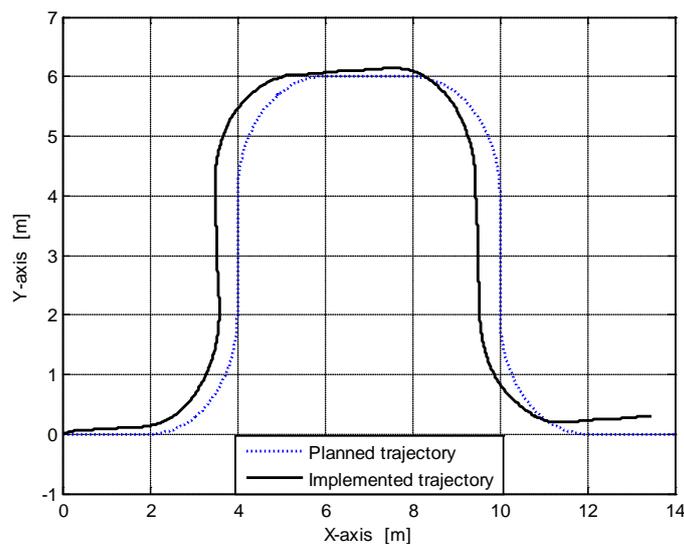

Figure 11. The planned trajectory versus implemented trajectory for the ground vehicle system.
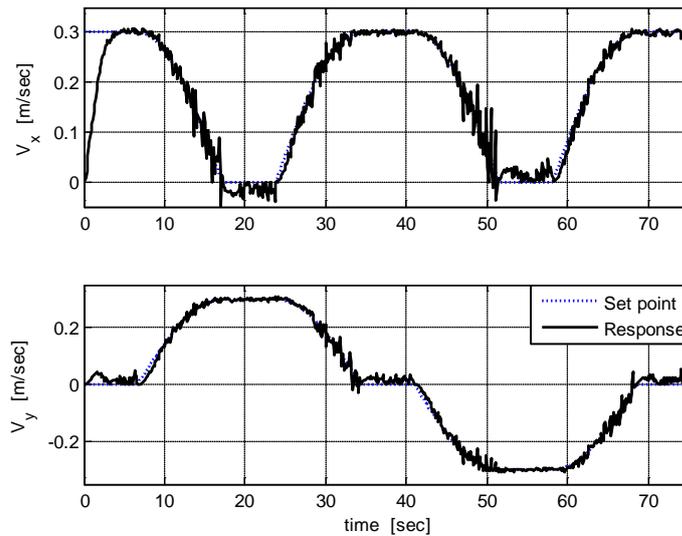
Figure 12. The resolved components of the vehicle velocity during the tracking of the designed trajectory.

10. Conclusion and Future Work

The present paper has developed a Proportional-Integral-Plus (PIP) control system for a ground vehicle during following a trajectory. Here, the linear model of the system is built and the PIP algorithm is performed using Linear Quadric (LQ) approach to get the State Variable Feedback (SVF) gain vector.

Implementation results suggest that the PIP algorithm introduces a feasible performance for the vehicle during avoiding an obstacle. An accurate tracking for the reference values for right and left wheels' angular velocities are achieved, as well as a smooth path tracking.

In terms of future work, a global positioning system (GPS) is required to correct the path of the vehicle and return the vehicle back to the straight line motion in x-direction, after avoiding the obstacle. Also, more sophisticated trajectory planning should be performed to enhance the performance of the obstacle avoidance.

References

[1] Kanayama Y., Miyake N., "Trajectory Generation for Mobile Robots", In: Proceedings of the International Symposium on Robotics Research, 1985, p. 6–23.

[2] Kanayama Y., Hartman B.I., "Smooth Local Path Planning for Autonomous Vehicles", In: Proceedings of the International Conference on Robotics and Automation, Vol. 3, 1989, p. 1265–1270.

[3] Wan TR., Chen H., Earnshaw RA., "A Motion Constrained Dynamic Path Planning Algorithm for Multi-Agent Simulations", In: Proceedings of WSCG'2005, 2005, p. 211–218.

[4] Nelson W., "Continuous–Curvature Paths for Autonomous Vehicles", In: Proceedings of the IEEE International Conference on Robotics and Automation, 1989, p. 1260–1264.

[5] Lamiraux F., Laumond J–P., "Smooth Motion Planning for Car–Like Vehicles", IEEE Transactions on Robotics and Automation, 2001; 17 (4): 498–502.

[6] Fraichard T., Sceuer A., "From Reeds and Shepp's to Continuous–Curvature Paths", IEEE Transactions on Robotics, 2004; 20 (6): 1025–1035.

[7] Reeds JA., Shepp LA., "Optimal Paths for a Car that Goes Both Forward and Backward", Pacific Journal of Mathematics, 1990; 145 (2): 367–393.

[8] Yamamoto M., Iwamura M., Mohri A., "Quasi–Time–Optimal Motion Planning of Mobile Platforms in the Presence of Obstacles. In: Proceedings of the IEEE International Conference on Robotics and Automation, 1999, p. 2958–2963.

[9] Mohammed Eghtesad, Dan S. Necsulescu, "Experimental Study of the Dynamic Based Feedback Linearization of an Autonomous Wheeled Ground Vehicle", Robotics and Autonomous Systems, 47 (2004), p. 47–63.

[10] A.M. Bloch, N.H. McClamroch, "Control of Mechanical Systems with Classical Nonholonomic Constraints", In: Proceedings of the 28th IEEE Conference on Decision and Control, Tampa, FL, 1989, pp. 201–205.

[11] R.W. Brockett, "Asymptotic Stability and Feedback Stabilization", In: R.W. Brockett, R.S. Millman, H.J. Sussmaan (Eds), Differential Geometric Control Theory, Birkhauser, 1983.

[12] G. Campion, B. d'Andrêa–Novel, G. Bastin, "Controllability and State Feedback Stabilizability of Non–holonomic Mechanical Systems", In: Proceedings of the International Workshop on Nonlinear and Adaptive Control: Issues in Robotics, Grenoble, France, November 21–23, 1990, pp. 106–124.

[13] A.P. Aguiar, A. Pascoal, "Stabilization of Autonomous Vehicles with Nonholonomic Constraints: Open Problems and Future Directions", In: Future Directions in Systems and Control Theory Wokshop, Cascias, Prtugal, June 1999.

[14] P.C. Young, M.A. Behzadi, C.L. Wang, A. Chotai, "Direct Digital and Adaptive Control by Input–Output, State Variable Feedback Pole Assignment", International Journal of Control, 46: 1867–1881, 1987.

[15] C.J. Taylor, P.C. Young, A. Chotai, "State Space Control System Design Based on Non–Minimal State–Variable Feedback: Further Generalisation and Unification Results", International Journal of Control, 73: 1329–1345, 2000.

[16] E.M. Shaban, M. Elsayed, "Design, Simulation, and Implementation of A Class of True Digital Control (TDC) Applied to Natural Gas Burner", Computer Engineering and Systems, 2009. Appears on ICCES 2009, International Conference: 14–16 December, 2009, p. 109–114.

[17] C.J. Talor, E.M. Shaban, M.A. Stables, S. Ako, "Proportional–Integral–Plus Control Applications of State Dependent Parameter Models", Journal of Systems and Control Engineering, 2007, 221, part I.

[18] E.M. Shaban, S. Ako, C.J. Taylor, and D.W. Seward, "Development of an automated verticality alignment system for a vibro-lance", Automation in Construction, 2007.

[19] E. Sidiropoulou, E.M. Shaban, C.J. Taylor, W. Tych, A. Chotai, "Linear, Nonlinear, and Classical Control of 1/5th Scale Automated Excavator", 18th International Conference on Systems Engineering, ICSE–06, September, 2006, Convey, UK.

[20] R. Dixon, C.J. Taylor, E.M. Shaban, "Comparison of Classical and Modern Control Applied to an Excavator Arm", International Federation of Automatic Control, 16th Triennial World Congress (IFAC–05), July, 2005, Prague, Czech Republic.

[21] K.J. Astrom, B. Wittenmark, "Computer Controlled Systems: Theory Design", Prentice–Hall Information and System Sciences Series, 1984.

[22] P.C. Young, "Simplified Refined Instrumental Variable (SRIV) Estimation and True Digital Control (TDC)", *a tutorial introduction, 1st Europeaan Control Conference*, pp.1295-1306 (Grenoble), 1991.

[23] P.C. Young, "Data-Based Mechanistic Modelling of Engineering Systems", *Journal of Vibration and Control*, Vol.4, pp.5-28, 1998.

[24] C.J. Taylor, A. Chotai, and P.C. Young, "State space control system design based on non-minimal state-variable feedback: further generalisation and unification results", *International Journal of Control*, Vol.73, pp.1329-1345, 2000.

[25] K.J. Astrom, and B. Wittenmark, "Computer Controlled Systems: Theory and Design", *Prentice-Hall Information and System Sciences Series*, 1984.

[26] A. Chotai, C.L. Wang, and P.C. Young, "Conditions for Arbitrary Pole Assignability by Input/Output State-Variable Feedback", *appears in N.K. Nichols and D.H. Owens (eds.)*, The Mathematics of Control Theory, Oxford University Press, Oxford, pp.87-96, 1992.

[27] Henriksson, Dan, Anton Cervin, and Karl-Erik Årzén. "TrueTime: Real-time control system simulation with MATLAB/Simulink." *Proceedings of the Nordic MATLAB Conference*, Copenhagen, Denmark. 2003.