

## Design and Simulation of Control Systems for a Field Survey Mobile Robot Platform

<sup>1</sup>Ramin Shamshiri and <sup>2</sup>Wan Ishak Wan Ismail

<sup>1</sup>Department of Agricultural and Biological Engineering, University of Florida, Gainesville, FL 32611, USA

<sup>2</sup>Department of Biological and Agricultural Engineering, Universiti Putra Malaysia, Serdang, Selangor, Malaysia

**Abstract:** The aim of this study was to design automatic and accurate control systems for wheel speed and steering of an agricultural mobile robot. Three controllers, including lead-lag compensator, Proportional-Integral-Derivative (PID) and fuzzy logic controller were designed and simulated in this study to control the angular rate of the shaft of a DC motor actuator for a field survey mobile robot that moves between plants rows to perform image acquisition task through a digital camera mounted on a two link arm attached on the robot base. The response of the actuator model for each controller were determined and compared for a sinusoidal and a step input that simulated robot speed and positioning references respectively. Performance analysis showed the effectiveness of the PID and lead-lag compensator response for the wheel steering task, while the fuzzy logic controller design had a better performance in wheel speed control. The output of this analysis was a proved satisfaction of the proposed design criteria which results enhanced mobility of the robot in terms of fast response, speed control accuracy and smooth steering at row-end turnings.

**Keywords:** Agricultural mobile robot, DC motor, fuzzy logic, PID, lead-lag compenstor

### INTRODUCTION

Agricultural mobile robots involve automatic and accurate control of different moving parts such as wheel speed and steering. The design challenges of a control system in this regard are the response overshoot, shorter settling time and smaller steady state error. Automatic guided devices have become an important part in different aspects of today's modern agriculture and precision farming. With the advances in controls theory, application of mobile robots in agriculture has shown growing interest towards automation. Such applications include chemical spraying for fungicide, crop monitoring, data collections, etc. An autonomous mobile robot for use in pest control and disease prevention application for a commercial greenhouse has been described by Sammons *et al.* (2005) where human health hazards are involved in spraying potentially toxic chemicals in a confined space. Another mobile robot for autonomous de-leafing process of cucumber plant has been studied by Van Henten *et al.* (2004). Various drive and guidance techniques have been implemented in farm robotic designs (Wada and Mori, 1996; West and Asada, 1997; Ferriere *et al.*, 2001). A simplified Direct Current (DC) motor equation has been implemented by Thomas and Hans (2004) for the driving actuator of an agricultural robot platform with four wheels steering for weed detection. Most of these past researches have focused on a particular, spatially non-varying task to be performed by the mobile robot

system in a farm environment, while a major difference between an industrial robot and its application with an agricultural robot is the environment impacts. A farm robot's platforms and implements are subjected to a dynamic environment, it may touch, sense or manipulate the crop and its surroundings in a precise manner which makes it necessary to have minimal amount of impact while increasing precision and efficiency (Kondo and Ting, 1998; De Baerdemaeker *et al.*, 2001). Hence, an important challenge in this area is the accurate control of the robot driving actuator due to the feedback errors and disturbances during the robot motion in stiff agricultural field which yields to the cumulative effects of the small error. As a result, a larger error in the robot's speed and position will be expected.

Today, more than 95% of control design applications utilize Proportional-Integral-Derivative (PID) controller due to its simplicity and applicability (Zhang *et al.*, 2004; Ang *et al.*, 2005). An important issue in controlling is the effect of nonlinearity in the actuators, for example, the nonlinear behavior of a DC motor actuator such as friction, saturation and external disturbance effects are ignored in the transfer function dynamic model. Although model based control methods such as variable structure control or reference adaptive control have been introduced to minimize these effects, the performance of such controllers still depends on the accuracy of the system dynamic model and its parameters. In addition, obtaining an accurate non-

linear model of an actual actuator like DC motor is difficult to find, or the parameter values from system identification may be approximate values. The concept of fuzzy logic techniques developed by Zadeh (1965) has been proved analytically to be equivalent to a nonlinear PID controller when a nonlinear defuzzification method is used. The operation of a Fuzzy Logic Controller (FLC) is based on system knowledge and linguistic description rather than crisp mathematical models.

The objective of this study was to design, simulate and compare a PID controller, a lead-lag compensator filter and a fuzzy logic controller for two driving Direct Current (DC) motor actuators of a field survey mobile robot platform. The control objective was the angular rate of the rotating shaft by varying the applied input voltage. The control design criteria was defined in such a way that for a step input ( $\dot{\theta}_{ref}$ ) of  $1 \frac{rad}{sec}$  that simulates the robot steering, the motor wheel speed satisfies a transient response with settling time  $\leq 0.1$  s, overshoot  $\leq 5\%$  and steady-state error (SSE)  $\leq 0.1\%$ . The criterion for the robot wheel speed controller was smooth following of a sinusoidal type input. Simulation was performed to show the response for each proposed design. Finally, the controllers were compared against each other based on their performance and control effort.

### MATERIALS AND METHODS

An articulated type robot prototype shown in Fig. 1 to 3 with four DC motor actuators was designed and developed. The drive and steering subsystem is comprised of two independent wheels and motor units, also known as a differential drive (Lucas, 2001). The robot changes its direction and speed by varying the relative rate of rotation of its wheels motor. The physical parameters of the motors are: moment of inertia of the rotor  $J = 42.6e - 6 \text{ Kgm}^2$ , viscous friction coefficient  $b = 47.3e - 6 \text{ Nms}$ , torque constant  $k_t = 14.7e - 3 \text{ Nm/Amp}$ , back emf  $k_e = 14.7e - 3 \text{ V.s/rad}$ , terminal resistance  $R = 4.67 \text{ ohm}$  and electric inductance  $L = 170e - 3 \text{ H}$ . The behavior of the motor speed for a given voltage is derived from physics law described by the open-loop transfer function (1) in Laplace domain with voltage  $V(s)$  as input and shaft speed  $\omega = \dot{\theta}(s)$  as output (Dorf and Bishop, 2001). The relationship between the reference speed  $\dot{\theta}_{ref}$  and the output speed  $\dot{\theta}(t)$  with a constant gain of  $K$  is given by the closed loop transfer function in 2. The Routh-Hurwitz MATLAB© program (Shamshiri, 2009) was used to check the stability of the system. The state space representation of the plant dynamic as provided in (3) was used to confirm that the system is stable, controllable and observable.

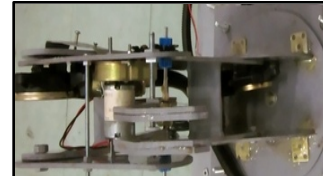


Fig. 1: DC motor actuator for the robot arm



Fig. 2: Field survey mobile robot platform

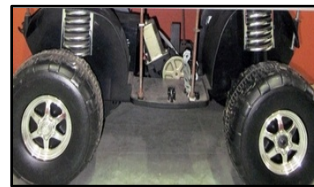


Fig. 3: DC motor driving actuators connected to the robot wheels

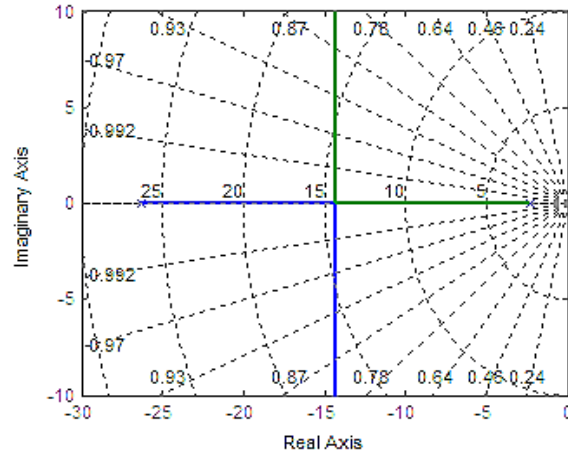


Fig. 4: Root locus of the DC motor system without any controller

$$\frac{\dot{\theta}(s)}{V(s)} = \frac{\frac{K_t}{JL}}{s^2 + \frac{(JR+bL)}{JL}s + \frac{bR+k_eK_t}{JL}} \quad (1)$$

$$\frac{\dot{\theta}_{out}}{\dot{\theta}_{ref}} = \frac{K.K_t/JL}{s^2 + \frac{(JR+bL)}{JL}s + \frac{K.K_t+bR+k_eK_t}{JL}} \quad (2)$$

$$\begin{bmatrix} \frac{di}{dt} \\ \ddot{\theta} \end{bmatrix} = \begin{bmatrix} -\frac{R}{L} & -\frac{k_e}{L} \\ \frac{K_t}{J} & -\frac{b}{J} \end{bmatrix} \begin{bmatrix} i \\ \dot{\theta} \end{bmatrix} + \begin{bmatrix} 1 \\ 0 \end{bmatrix} V(t) \text{ and } \dot{\theta} = \begin{bmatrix} 0 & 1 \end{bmatrix} \begin{bmatrix} i \\ \dot{\theta} \end{bmatrix} \quad (3)$$

**Lead-lag compensator filter design:** From the root locus of the open loop transfer function as shown in Fig. 4, the system has two real open loop poles at  $P_1 = -26.3$  and  $P_2 = -2.3$  which repel each other at  $-14.3$ , one going to the positive infinity and the other to the negative infinity. The desired damping ratio ( $\zeta = 0.6901$ ) and desired natural frequency ( $\omega_n = 57.962$ ) were calculated through the available equations for settling time and percent overshoot (Dorf and Bishop, 2001) and were then used in determining the desired characteristic equation as described in (4). Since the desired poles  $s_{1,2} = -40 \pm 41.94i$  do not satisfy the angle condition for the actual characteristic equation as shown in (5), the root locus will not go through the desired poles; hence a lead compensator filter was required to shift the root locus to the left half plane to meet the desired poles location. The closed loop poles were placed at the desired location (Fig. 5) by multiplying the lead compensator transfer function,  $G_1(s) = K_1(s + a_1)/(s + b_1)$ , ( $|b_1| > |a_1|$ ), to the open loop transfer function of the system and then performing coefficient matching which yields a system of three equations with four unknowns ( $a_1, b_1, c_1, K_1$ ), where  $c_1$  is the added pole to the desired characteristic equation. The one degree of design flexibility was satisfied by assuming  $a_1 = 16$ , which results in  $K_1=1.23$ ,  $b_1=64.49$  and  $c_1=13.0734$ . The performance of the designed lead compensator was evaluated for a step input of  $R(s) = 1/s$  through final value theorem. The limit showed convergence to 0.0888 percent which is larger than the desired SSE of 0.1%, hence a lag compensator with transfer function  $G_2(s) = (s + a_2)/(s + b_2)$ , ( $|b_2| < |a_2|$ ), was added by assuming  $a_2 = 5$  resulting  $b_2=0.0564$ . The final transfer function of the complete lead-lag compensator filter,  $G_{lead-lag}(s) = G_1(s)G_2(s)$  is provided in (6).

$$\Delta_D = s^2 + 2\zeta\omega_n s + \omega_n^2$$

$$s^2 + 80s + 3359.6$$

$$\Delta_D = (s + 40 + 41.94i)(s + 40 - 41.94i) \quad (4)$$

$$\angle P(s) = \angle -1 = \pm 180(2q + 1), \forall q = 0, 1, 2, 3, \dots$$

$$(-\angle s + 26.3 - \angle s + 2.3)|_{-40 \pm 41.94i}$$

$$\therefore -tg^{-1}\left(\frac{41.94}{-13.7}\right) - tg^{-1}\left(\frac{41.94}{-37.7}\right) = 119.9 \neq \pm 180(2q + 1) \quad (5)$$

$$G_{lead-lag}(s) = \frac{1.233s^2 + 25.88s + 98.6}{s^2 + 64.55s + 3.64} \quad (6)$$

**PID controller design:** The general transfer function for a PID controller in Laplace domain can be written as shown in (7) where  $K_p$  the proportional gain is,  $K_D$  is the derivative gain and  $K_I$  is the integrator gain. Considering the effect of each term in the PID

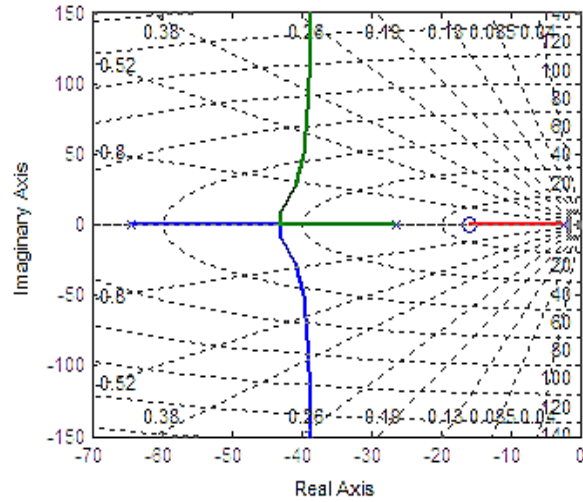


Fig. 5: Root locus plot of the DC motor with lead compensator

controller, the PID gains were selected through a trial and error approach and were then tuned by simulation with final values of:  $K_p = 0.6844$ ,  $K_I = 0.5975$ ,  $K_D = 0.0119$ . The open loop and closed loop transfer function of the system with PID controller is given by (8) and (9) respectively:

$$PID(s) = K_p + \frac{K_I}{s} + K_D s = \frac{K_D s^2 + K_p s + K_I}{s} \quad (7)$$

$$G_{open\_loop\_PID}(s) = \frac{24.15 s^2 + 1389s + 1213}{s^3 + 28.58s^2 + 60.34s} \quad (8)$$

$$G_{closed\_loop\_PID}(s) = \frac{24.15 s^2 + 1389s + 1213}{s^3 + 52.74 s^2 + 1450s + 1213} \quad (9)$$

**Fuzzy logic control design:** A typical structure of a fuzzy logic controller is shown in Fig. 6. Using a preprocessor, the inputs that were in the form of crisp values generated from feedback error (e) and change of error (de) were conditioned in terms of multiplying by constant gains before entering into the main control block. The fuzzification block converts input data to degrees of membership functions and matches data with conditions of rules. From the rule based commands, the Mamdani-type inference engine determined the capability of degree of employed rules and returned a fuzzy set for defuzzification block where the fuzzy output data were taken and crisp values were returned. The outputs of the fuzzy sets were converted to crisp values through centroid defuzzification method. The post processing block then converted these crisp values into standard control signals. In this project, experiential knowledge was borrowed from proportional integral control error and change of error to define fuzzy membership functions. The rule Table 1 was then designed and used with a triangular membership function inputs-output in the fuzzy logic controller and was implemented in the simulation.

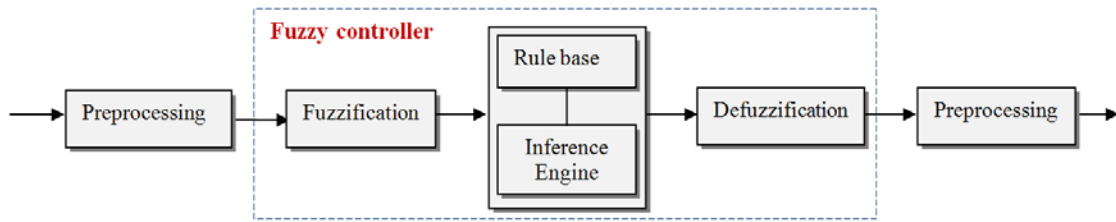


Fig. 6: Fuzzy logic controller block diagram

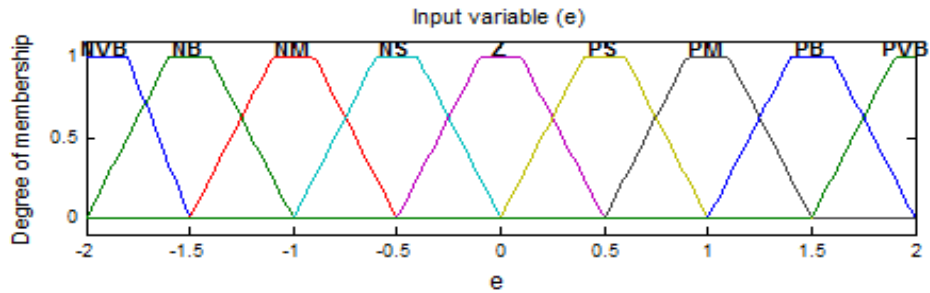


Fig. 7: Fuzzy logic first input variable, error

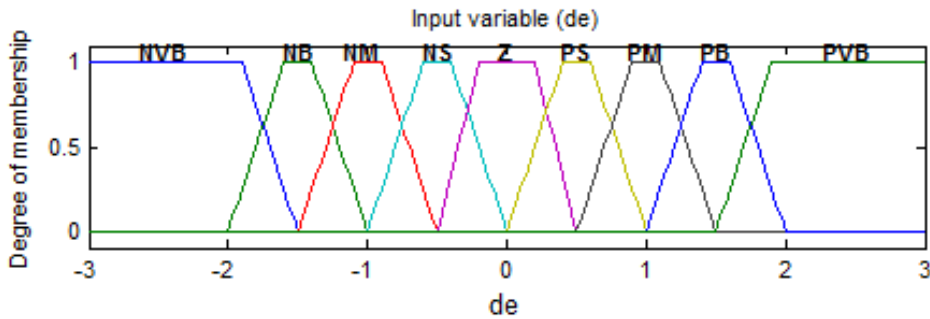


Fig. 8: Fuzzy logic second input variable, change of error

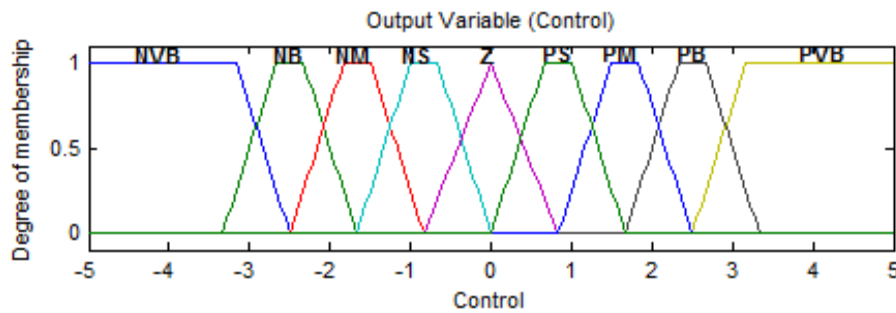


Fig. 9: Fuzzy logic output variable, control

Table 1: Fuzzy logic controller rule table

De/e	NVB	NB	NM	NS	Z	PS	PM	PB	PVB
NVB	PVB	PVB	PVB	PB	PM	PM	PS	Z	Z
NB	PVB	PVB	PB	PM	PS	PS	PS	Z	Z
NM	PVB	PB	PM	PS	PS	Z	Z	Z	NS
NS	PB	PM	PM	PS	PS	Z	Z	NS	NS
Z	PM	PM	PS	Z	Z	Z	NS	NS	NM
PS	PM	PS	PS	Z	NS	NS	NM	NM	NB
PM	PS	PS	Z	NS	NS	NM	NB	NB	NB
PB	PS	Z	Z	NS	NM	NM	NB	NVB	NVB
PVB	Z	Z	NS	NM	NM	NB	NB	NVB	NVB

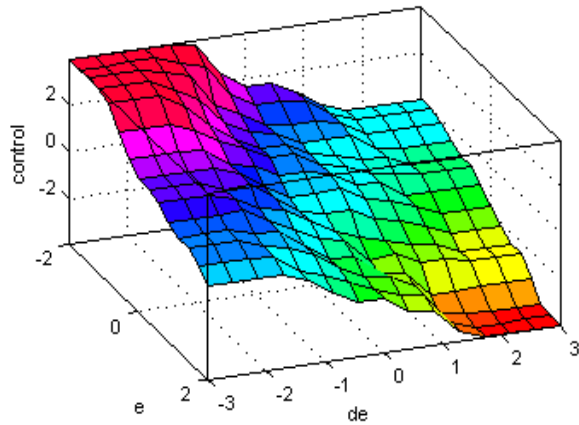


Fig. 10: Fuzzy logic rule surface

These rules make control efforts based on several if-then statements about (e) and (de), i.e., if the error is equal Negative Big (NB) and change of error is equal to negative medium (NM), then the change in control (c) is positive big (PB). The numbers of these if-then statements were determined based on experiment and tuning of the system. Plots of fuzzy logic membership function for the two inputs variables (e) and (de) and the output (c) are shown in Fig. 7 to 9. The rule surface corresponding to the rules of Table 1 is provided in

Fig. 10 which shows smoothness of the change in the control signal.

### SIMULATION AND RESULTS

The performances of the three designed controller were simulated in MATLAB© with block diagram shown in Fig. 11. A signal generator produces input references of step and sinusoidal function for each control blocks. The lead-lag and PID controller transfer function were implemented in the simulation followed by the DC motor dynamic model. The fuzzy logic controller block process the inputs and output of fuzzy inference engine and generate control signal. The corresponding defuzzification sub-block is shown in Fig. 12. The root locus plots of the system with final lead-lag and PID controller designs are also shown in Fig. 13 and 14 respectively which proves that the design criteria with the desired poles locations have been satisfied. According to the infinite gain and phase margin observed from the bode plots of the closed loop system in the presence of these controllers, as shown in Fig. 15 and 16, the system will not become unstable with increasing gain. The behavior of the open loop and closed loop response and the performance of the controllers were evaluated by input step functions with results plotted in Fig. 17 to 24.

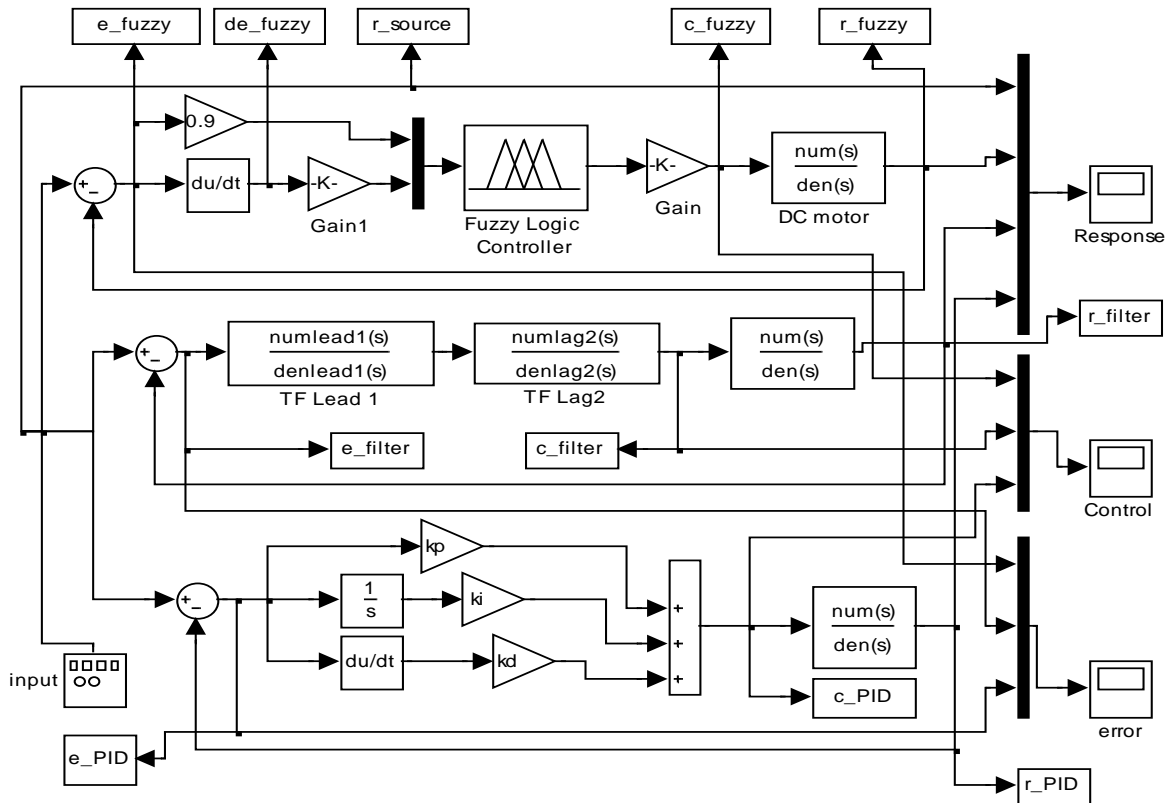


Fig. 11: Simulation diagram of controllers for DC motor speed

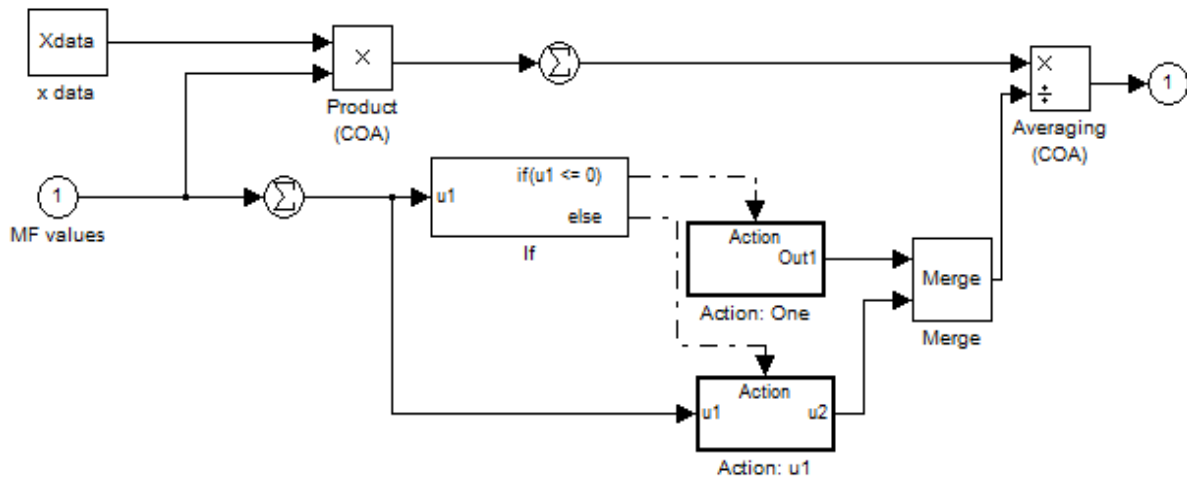


Fig. 12: Simulation diagram of defuzzification

Table 2: Performance of controllers to step input

Controller	Rise time (second)	Overshoot (%)	Settling time (second)	Peak amplitude	Final value
Lead compensator	0.084	0	0.194	0.911	0.911
Final lead-lag compensator	0.0798	3.27	0.434	1.03	0.999
PID control	0.0452	4.4	0.328	1.04	1
Fuzzy logic	0.0543	11.4	0.572	1.21	1

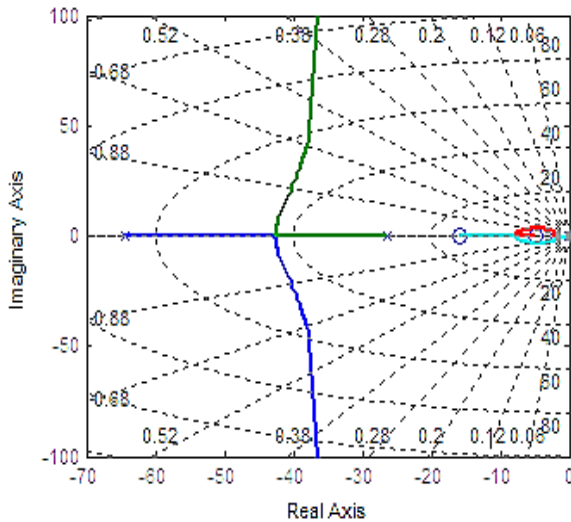


Fig. 13: Root locus of the DC motor with final lead-lag compensator

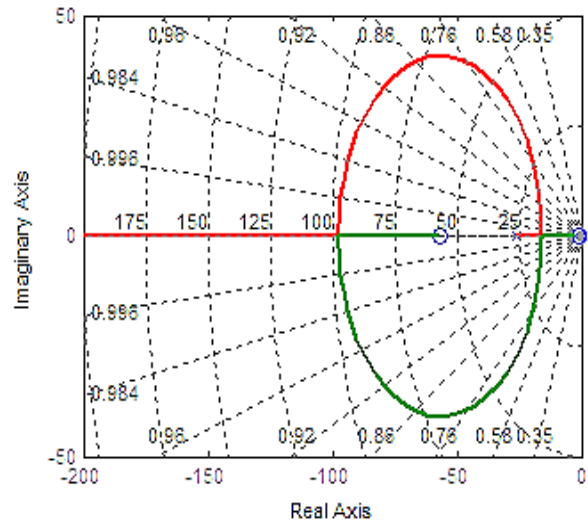


Fig. 14: Root locus of the DC motor with PID controller

Although the open loop system in Fig. 17 shows stability in nature, but the initial closed loop step response in Fig. 18 indicates a demand for a controller to improve rise time, settling time, overshoot and SSE. All these performances were improved in the final closed loop response with lead-lag compensator and PID controller as shown in Fig. 22 and 24 with the results summarized in Table 2.

Plots of tracking errors, change of errors, system responses and control signals are provided for the three

controller design in Fig. 25 to 32. For the step input which simulates the robot wheel steering task, the feedback errors of the three designed controllers are almost equal, while for the sinusoidal input corresponding to the robot wheel speed control task, the fuzzy logic controller produces significantly smaller feedback errors. It can also be observed from the response plots of the controllers in Fig. 28 and 29 that the fuzzy logic controller (green line) is perfectly following the sinus input; however the other two controllers showed better performances for the step

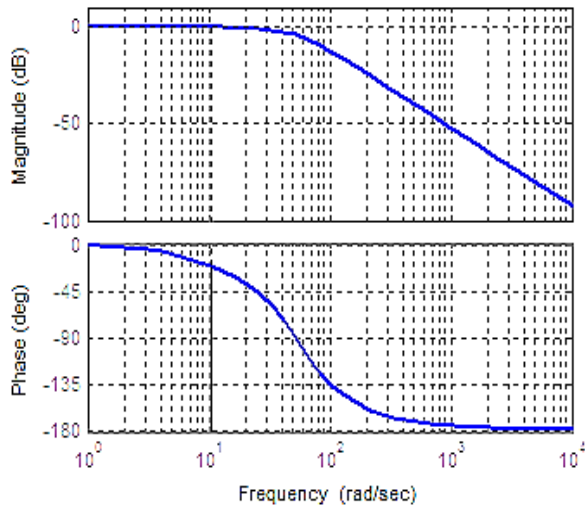


Fig. 15: Bode plot of closed loop TF with final lead-lag compensator,  $G_m = \infty$  dB (at  $\infty$  rad/sec),  $P_m = 159$  degree (at 10.4 rad/sec)

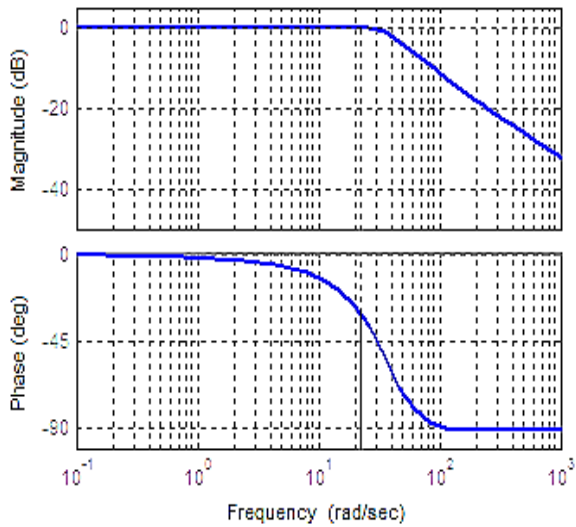


Fig. 16: Bode plot of closed loop TF with PID controller,  $G_m = \infty$  dB,  $P_m = 150$  degree (at 22.2 rad/sec)

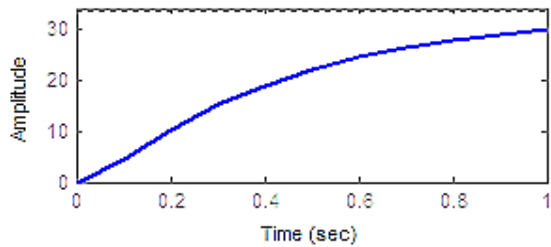


Fig. 17: Open loop step response without controller

input. The control efforts to the step input for lead-lag compensator and fuzzy logic controller are shown in

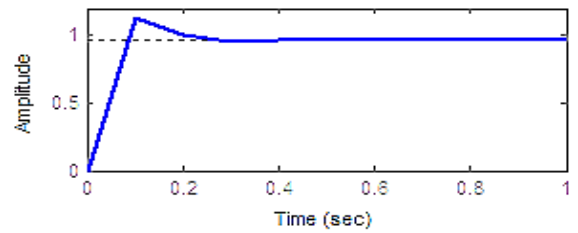


Fig. 18: Closed loop step response of DC motor without controller

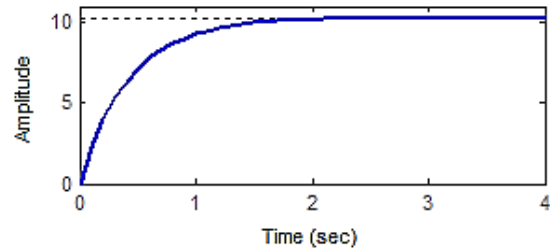


Fig. 19: Open loop step response of DC motor with first lead compensator

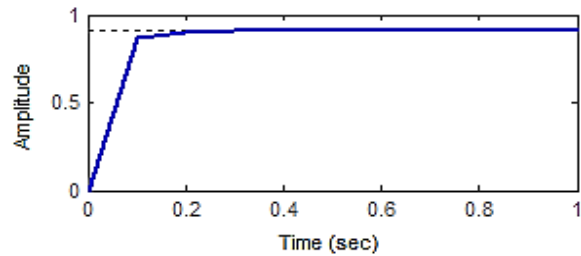


Fig. 20: Closed loop step response of DC motor with first lead compensator

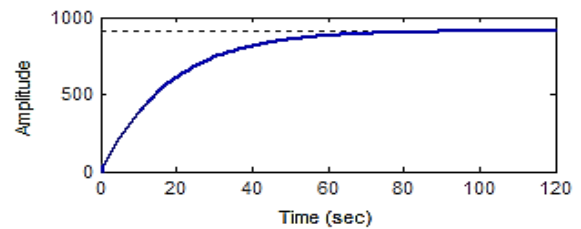


Fig. 21: Open loop step response of DC motor with final lead-lag compensator

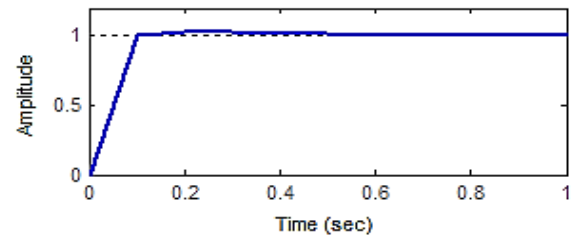


Fig. 22: Closed loop response of DC motor with final lead-lag compensator

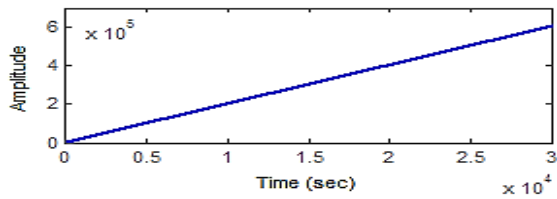


Fig. 23: Open loop step response of DC motor with PID controller

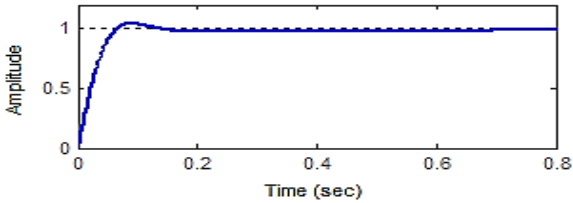


Fig. 24: Closed loop step response of DC motor with PID controller

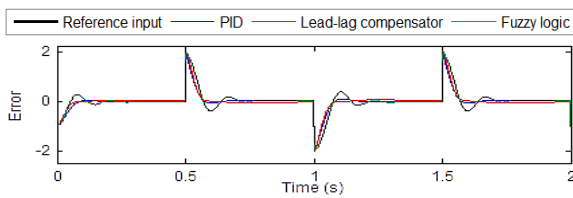


Fig. 25: Plot of errors to the controller responses for step input

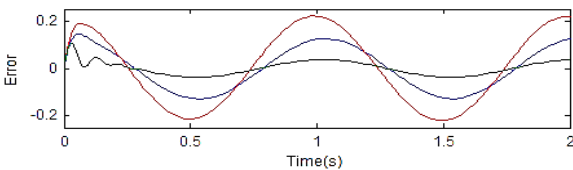


Fig. 26: Plot of errors to the controller responses for sinusoidal input

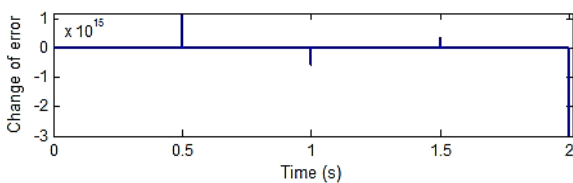


Fig. 27: Plot of change of error in time for the fuzzy logic controller with step input

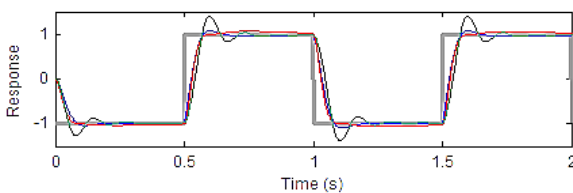


Fig. 28: Plot of step response and performance comparison for the designed controllers

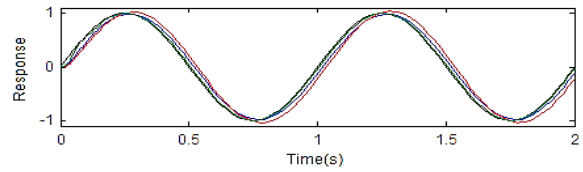


Fig. 29: Plot of sinusoidal response and performance comparison for the designed controllers

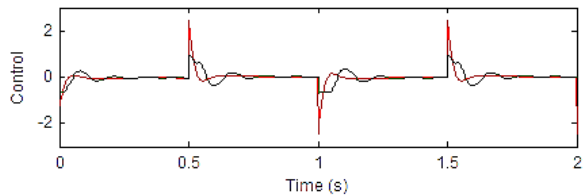


Fig. 30: Plot of control signal for lead-lag compensator and fuzzy logic controller for step input

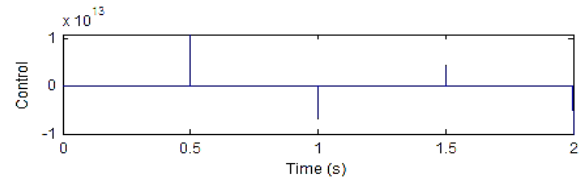


Fig. 31: Plot of control signal for the PID controller for step input

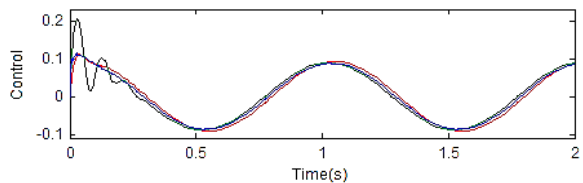


Fig. 32: Plot of control signal for the three controllers for sinusoidal input

Fig. 30 separated from step input PID control effort. It can be seen that the PID controller takes a huge numerical value of control effort in the step changes, which the other two controllers remain in the range of reasonable values of plus and minus 2. For the sinusoidal input, all the three controllers almost have equal control efforts, except at the very beginning of the process that fuzzy logic controller starts oscillating due to its design nature. The final results show that the PID and lead-lag compensator are best choices for the step inputs, while the fuzzy logic controller is a better candidate for sinusoidal changes.

## CONCLUSION

Mechanization and automation of farm operation requires precise autonomous devices to perform labor intensive task such as data collections and image acquisition. This study discussed about simulation and analysis of three controller design for speed control of a



DC motor actuator that was used in a field survey agricultural robot platform which moves between crop rows to collect image data. A linear differential equation describing the electromechanical properties of a DC motor to model the relation between input ( $V$ ) and output ( $\dot{\theta}$ ) was first developed using basic laws of physic. This transfer function was then used to analyze the performance of the system and to design proper controllers (lead-lag compensator, PID and fuzzy logic controller) to meet the design criteria. For the compensator design, the locations of the desired poles were found from the proposed values for settling time and percent overshoot. Using root locus, it was found that a lead compensator is required to place poles in the desired locations. A lag compensator was also designed and added to meet the steady state requirement of the problem. A PID controller was also designed and tuned based on the conventional methods. To achieve smoother control, a fuzzy logic controller with two inputs and one output including 81 rules was also designed. All the three controllers were implemented in the simulation. The results showed that for sinusoidal changes of the robot speed, the fuzzy logic controller has a better performance over conventional PID and lead-lag compensator design in terms of rise time, settling time and producing desired response.

#### REFERENCES

- Ang, K.H., G. Chong and Y. Li, 2005. PID control system analysis: Design and technology. *IEEE T. Contr. Syst. T.*, 13(4): 559-576.
- De Baerdemaeker, J., A. Munack, H. Ramon and H. Speckmann, 2001. Mechatronic systems: Communication and control in precision agriculture. *IEEE Contr. Syst. Mag.*, 21(5): 48-70.
- Dorf, R.C. and R.H. Bishop, 2001. *Modern Control Systems*. 9th Ed., Prentice-Hall, Inc., Upper Saddle River, NJ.
- Ferriere, L., G. Campion and R.B. Raucant, 2001. Rollmobs, a new universal wheel concept. *Robotica*, 19: 1-9.
- Kondo, N. and K.C. Ting, 1998. *Robotics for Bioproduction Systems*. American Society of Agricultural Engineering, Saint Joseph, Mich, pp: 325.
- Lucas, G.W., 2001. A Tutorial and Elementary Trajectory Model for the Differential Steering System of Robot Wheel Actuators. Retrieved from: <http://rossum.sourceforge.net/papers/DiffSteer/>.
- Sammons, P.J., T. Furukawa and A. Bulgin, 2005. Autonomous pesticide spraying robot for use in a greenhouse. *Proceedings of the Australasian Conference on Robotics and Automation*, Australia.
- Shamshiri, R., 2009. Routh-Hurwitz Stability Test. Retrieved from: <http://www.mathworks.com/matlabcentral/fileexchange/25956-routh-hurwitz-stability-test>.
- Thomas, B. and J. Hans, 2004. Agricultural robotic platform with four wheel steering for weed detection. *Biosyst. Eng.*, 87(2): 125-136.
- Van Henten, E.J., B.A.J. Van Tuijl, G.J. Hoogakker, M.J. Van Der Weerd, J. Hemming, J.G. Kornet and J. Bontsema, 2004. An autonomous robot for de-leafing cucumber plants grown in a high-wire cultivation system. *Proceeding of ISHS International Conference on Sustainable Greenhouse Systems*. Belgium, 691: 877-884.
- Wada, M. and S. Mori, 1996. Holonomic and omnidirectional vehicle with conventional tires. *Proceeding of the IEEE Conference on Robotics and Automation*. Minneapolis, MN, pp: 3671-3676.
- West, M. and H. Asada, 1997. Design of ball wheel mechanisms for omnidirectional vehicles with full mobility and invariant kinematics. *J. Mech. Design*, 119: 153-161.
- Zadeh, L.A., 1965. Fuzzy sets. *Inform. Cont.*, 8(3): 338-353.
- Zhang, J., N. Wang and S. Wang, 2004. A developed method of tuning PID controllers with fuzzy rules for integrating process. *Proceeding of the American Control Conference*. Boston, pp: 1109-1114.