2016

Inter-row Robot Navigation using 1D Ranging Sensors

Tyler A. Troyer
University of Nebraska–Lincoln, tyler.troyer@huskers.unl.edu

Santosh Pitla
University of Nebraska-Lincoln, spitla2@unl.edu

Ethan Nutter
University of Nebraska–Lincoln, teg3.14@gmail.com

Published in *IFAC-PapersOnLine* 49-16 (2016), pp 463–468. doi:10.1016/j.ifacol.2016.10.084

Inter-row Robot Navigation using 1D Ranging Sensors

Tyler A. Troyer, Santosh Pitla, and Ethan Nutter

Department of Biological Systems Engineering, University of Nebraska–Lincoln, Lincoln, NE

Emails: tyler.troyer@huskers.unl.edu ; spitla2@unl.edu ; teg3.14@gmail.com

Abstract
In this paper a fuzzy logic navigation controller for an inter-row agricultural robot is developed and evaluated in laboratory settings. The controller receives input from one-dimensional (1D) ranging sensors on the robotic platform, and operated on ten fuzzy rules for basic row-following behavior. The control system was implemented on basic hardware for proof of concept and operated on a commonly available microcontroller development platform and open source software libraries. The robot platform used for experimentation was a small tracked vehicle with differential steering control. Fuzzy inferencing and defuzzification, step response and cross track error were obtained from the test conducted to characterize the transient and steady state response of the controller. Controller settling times were within 4 seconds. Steady state centering errors for smooth barrier navigation were found to be within 3.5% of center for 61 cm wide solid barrier tests, and within 38% for simulated 61 cm corn row tests.

Keywords: Fuzzy Control, Embedded systems, Control Systems, Feedback Control, Mobile robots

1. Introduction

The reach of typical remote sensing techniques have had history of image resolutions insufficient for plant-by-plant analysis (Moran et al., 1997). Adjustable product application rate techniques that rely on high resolution spatial data can have a positive effect on use efficiency of product such as fertilizer, water and herbicide (Sowers et al. 1994), and mechanisms for providing this data cannot solely rely on satellite image resolutions. Martin et al. (2012) presented a corn yield model which was based on plant spacing periodicity and plant height, but the model was produced with plant data which is not normally available from above-canopy sensing techniques.

Some of the field parameters most easily sampled locally include soil properties, weed presence, crop geometry and atmospheric variables. The systems developed in academia meant for these local sensing applications tend to be large machinery, or are human operated or powered. To fill a gap in agricultural automation at a time when precision agriculture takes root, such sensing systems as shown in this paper could be deployed on small, inter-row robots. Flexible and on-demand sensing platforms should become more marketable if they do not contribute to compaction, and are not liable to FAA licensing and regulation such as unmanned aerial vehicles. Reduced optical noise under the crop canopy from solar radiation also intuitively warrants simpler sensing techniques riding on inter-row mobile robotic platforms, as opposed to aerial platforms.

Several viable sensing devices for plant spacing and yield estimation have been developed (Shi et al., 2013). These systems have not yet been tested in an automated context. Researchers in sensing are interested in instrumentation development, and researchers of in-field automation produce research that drives and navigates machinery autonomously. The two fields of research need to be combined to provide optimal autonomous solutions for plant sensing. If robot navigation control systems are more accessible and universal, research in sensing may be able to continue more commonly under an automated context.

When sensing research is conducted on automated platforms, it is typically done with a robotic platform developed in-house. Many of these platforms depend on GPS with inertial measurement unit localization techniques to navigate what are essentially straight crop rows. With this knowledge, a simpler single-row robot appears as a feasible option for local sensing.

An inter-row, 2D laser scanning technique for plant spacing measurement relies on a clustering algorithm dependent on a thresholding process which assumes a constant distance from the scanner to the corn row (Shi et al., 2013). Constant spacing during measurement could be automated if the sensor was mounted on a crop row navigating robot that could maintain its distance from a crop row it navigates along.

The navigation control systems developed in past research are rarely portable to other robotic platforms, since they depend on robot geometry, actuator dynamics, and empirically determined lookup tables for system calibration and mapping (Darr, 2004). Others use RTK-GPS to augment position determination (Biber et al, 2012). Of the row navigation controllers that do not fall into this category, some systems do well by relying on the geometry and positioning of the crop row instead of the robot (Xue and Xu, 2010). This makes the navigation controller more portable to other systems, but rarely can the row navigation philosophy be divorced from the exact hardware used and the firmware used to drive it in each of these implementations. These issues are addressed and an alternative, more universal row navigation control scheme is described and implemented on very general purpose hardware and open source software libraries.

To the knowledge of the authors, limited research has been conducted on robotic, sub-canopy row navigation as a sensing platform. The goal of this project was to develop an inter-row navigation fuzzy controller architecture independent of specific hardware products or sensor implementations. The resulting controller was designed to be implemented on any ground-based mobile robotic platform, any microcontroller architecture, and any ranging sensor. The controller was implemented on an inexpensive robot to demonstrate controller portability, and ease of construction.

2. Materials and Methods

The robotic platform (Figure 2) selected for the fuzzy controller demonstration was the T-Rex tracked chassis manufactured by DAGU Hi-Tech Electronics (Model No. RS035, Zhongshan City, Guangdong Province, China, www.dagurobot.com). The chassis selected served as a simple and robust platform for differential steering experimentation. The chassis included a motor and a gearbox for each track which enabled simple control of the track velocity (Figure 4). The motors were driven with a Sabertooth motor driver made by LLC (2x25) (Dimension Engineering, 5171 Hudson Drive, Hudson, OH 44236), and powered with a 5 Amp-Hour Lithium cell (Model 5AH 2s, Turnigy Power Systems®). The size of the platform was sufficient for crop row navigation as required for plant spacing sensing devices. Position sensing was implemented with two LIDAR ranging sensors (LIDAR-Lite v2, PulsedLight, Inc, 700 NW Hill St. Suite 3, Bend, OR 97703) depicted in Figure 1.

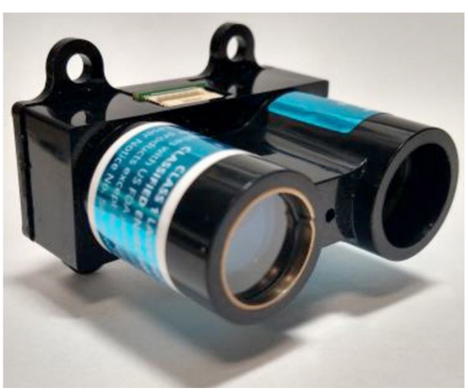

Figure 1: LIDAR ranging sensor

The sensors were positioned at the front of the chassis, and aimed 45 degrees ahead of the normal vector of the crop row. This look-ahead technique was demonstrated as valuable to the system's dynamic response as shown by (Stombaugh et al, 1998). The two sensors provided the input to the fuzzy controller. The constructed vehicle is shown in Figure 2.

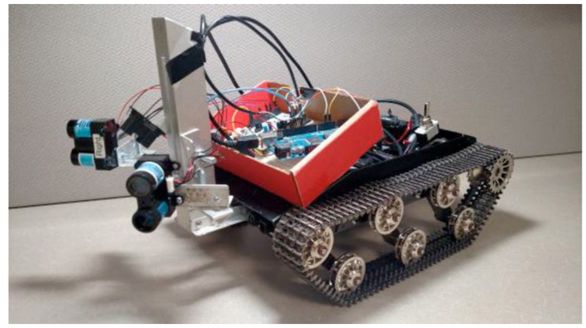

Figure 2: Fully constructed row-following platform

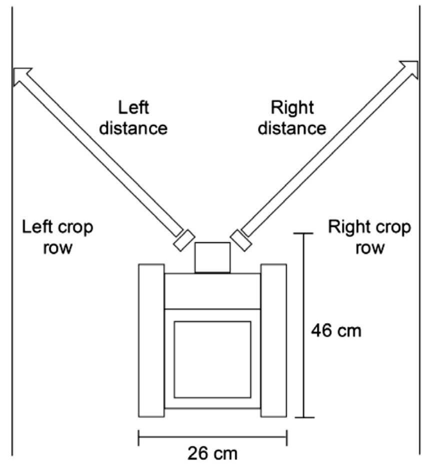

Figure 3: Vehicle control model

The control model (Figure 3) was based on the single goal of maintaining a centered position between the crop rows. Error amplitude was computed as the difference between the robot's distance from the right side row, and the distance from the left side row. Additional control input was provided by the sum of the distances, indicating the overall navigation width of the row. The amplitude was able to freely swing between positive and negative values in centimeter units.

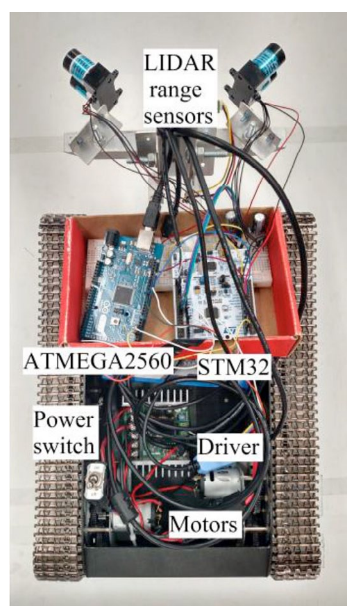

Figure 4: Vehicle construction

Mapping the error to a desired response required the design of a fuzzy membership function for the input and the output. The input membership function was contained in a universe of discourse with the unit of centimeters, which is the output of each of the ranging sensors. The output membership function was a signed 8 bit integer that described the difference between the left and right motor speeds, where the sign determined motor direction, and magnitude determined a pulse width modulated (PWM) duty cycle to the motor. To complete mapping, a Mamdani controller with 10 rules was arranged to describe the response of the robot through the legal range of possible controller inputs shown in Table 2 (Mamdani, 1974).

Figures 5 and 6 show the membership function for the inputs available to the controller, and Figure 7 shows the available outputs to the controller. Table 1 denotes each membership in the sets. The controller surface defined by the rule set is depicted in Figure 8, as plotted in MATLAB (Fuzzy Designer, The MathWorks, Inc, 3 Apple Hill Drive, Natick, Massachusetts 01760 USA)

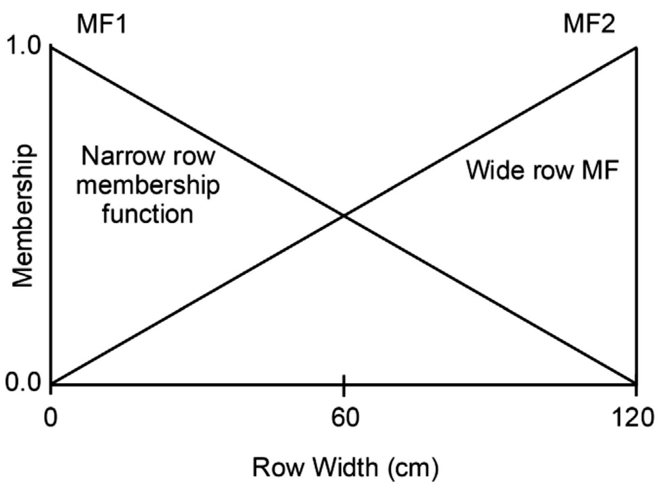

Figure 5: Membership function of the distance sums input

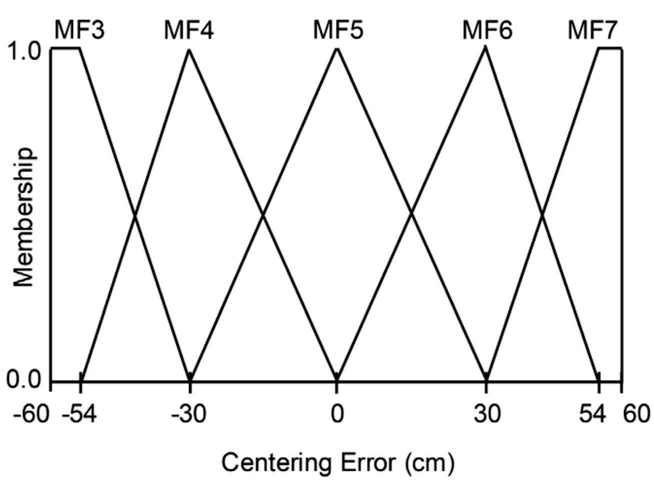

Figure 6: Membership function of the centering error input

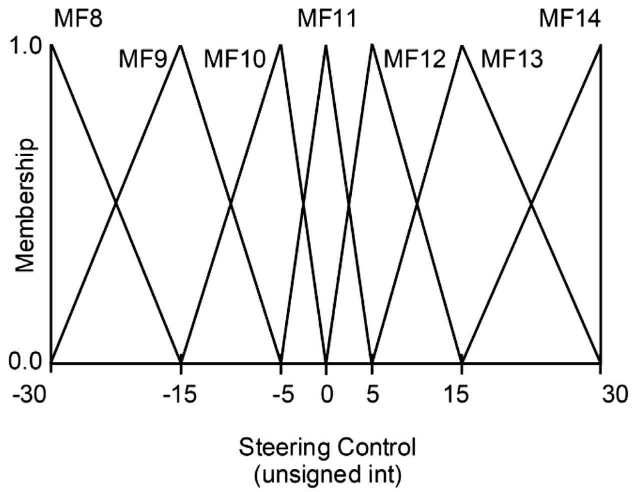

Figure 7: Membership function of the controller output set

Table 1: Fuzzy set descriptions of the fuzzy controller

Fuzzy Set	Description	Universe of Discourse
MF1	narrow row	sum input
MF2	wide row	sum input
MF3	far right	centering error input
MF4	right	centering error input
MF5	centered	centering error input
MF6	left	centering error input
MF7	far left	centering error input
MF8	hard right	Control integer output
MF9	nominal right	Control integer output
MF10	light right	Control integer output
MF11	go straight	Control integer output
MF12	light left	Control integer output
MF13	nominal left	Control integer output
MF14	hard left	Control integer output

The development of these fuzzy sets were the result of an iterative tuning process where controller response was evaluated on how well the robot performed while navigating between solid barriers. Desirable results were identified by the minimization of centering error, response period, and steady state oscillation magnitude. The rules determining the system behavior are described by Table 2.

Table 2: Rule set for fuzzy controller

Antecedent A	AND/OR	Antecedent B	Consequent
narrow row	AND	far right	hard left
narrow row	AND	right	nominal left
narrow row	AND	center	go straight
narrow row	AND	left	nominal right
narrow row	AND	far left	hard right
wide row	AND	far right	nominal left
wide row	AND	right	light left
wide row	AND	center	go straight
wide row	AND	left	light right
wide row	AND	far left	nominal right

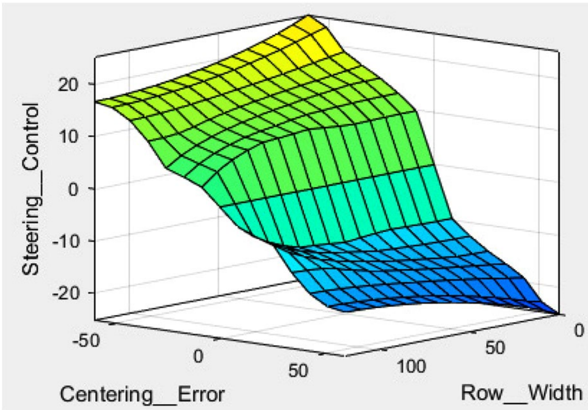

Figure 8: Rule surface as defined by 10 rules

Implementing the fuzzy controller was done with open-source development tools including the Arduino 1.6.6 development environment (http://www.arduino.cc), and the Embedded Fuzzy Logic Library (eFLL) which used max-min composition for inferencing, minimum Mamdani for composition, and center of area for defuzzification (http://github.com/zerokol/eFLL).

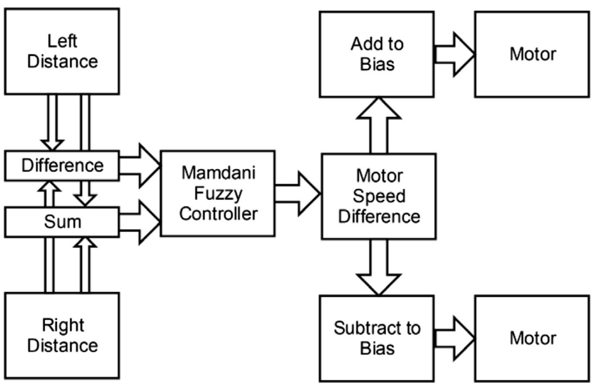

Figure 9: Input/output of the fuzzy controller

The Arduino MEGA development board was used as the hardware target, acting as the interface between the distance sensors, and the motor controller. The sensors produced a distance integer at a centimeter resolution at a rate of 20 Hz, and the motor controller was controlled and configured over the serial port on the microcontroller's board at 9600 kbps, 1 stop bit, no parity, no CTS or RTS. Device interfaces are shown in Figure 10.

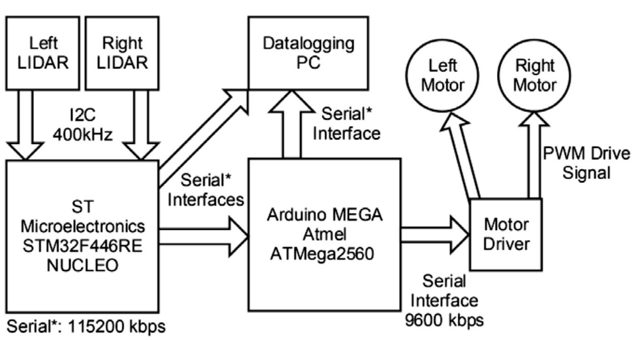

Figure 10: Physical layer communications interfaces

The C functions for motor control accepted a signed 8 bit integer to produce a PWM motor speed waveform with 7 bits of resolution in both directions. A forward bias of the motors defined in firmware provided a forward drive which required a nominal bias value of 32. This correlated to speed of 25% of the maximum for the motor in the forward direction. The output of the fuzzy controller was a motor speed difference. To apply this speed difference to the motors, the output of the fuzzy controller functions was added to the bias value of 32 and sent to the motor controller as the speed desired for one motor. The same fuzzy controller output was then subtracted from the bias value of 32, and then sent as the desired speed of the other motor to complete a differential steering effect.

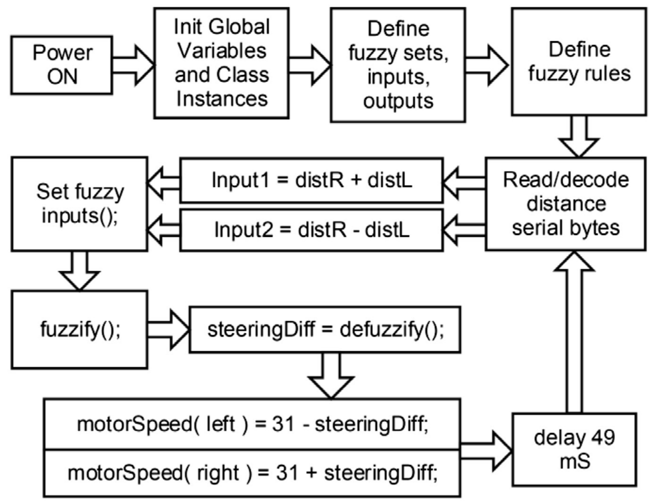

Figure 11: Firmware algorithm of the Arduino MEGA

The process shown in Figure 11 describes the control flow of the firmware and the use of the eFLL functions to achieve steering control in the ATMEGA2560. Firmware was written in C/C++ in the Arduino integrated development environment (IDE). The control loop was event driven which produced a controller process frequency of 20 Hz. This period was determined by LIDAR sampling loop defined in the STM32 (STM32F446RE, STMicroelectronics, Geneva, Switzerland) firmware as shown in Figure 12.

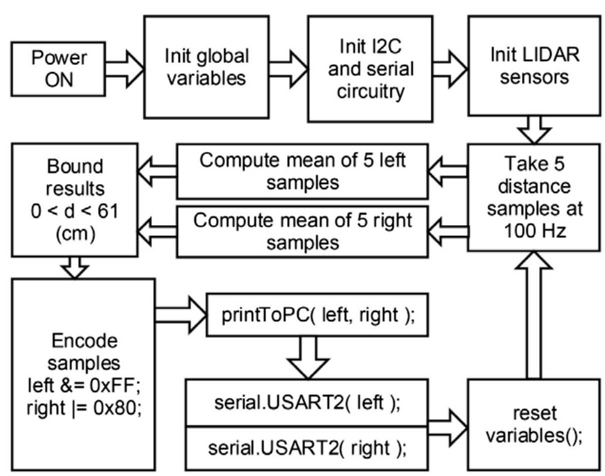

Figure 12: Sampling and streaming algorithm of the STM32

After implementation was complete, a test row was built to characterize the navigation controller. The test row made of solid barriers was built to impart a unit step error input to the fuzzy controller to allow for the evaluation of the controller response and steady state error. Both positive and negative step inputs were tested. The widths of the simulated row changed between 91 cm (36 in) and 61 cm (24 in), depending on the step function polarity of the test. The setup for these tests are depicted in Figures 13 and 14.

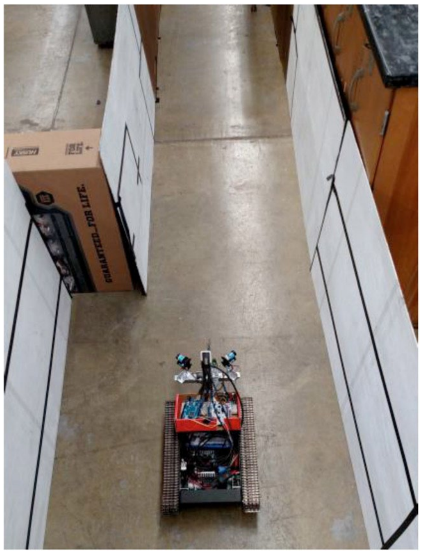

Figure 13: Unit step navigation testing

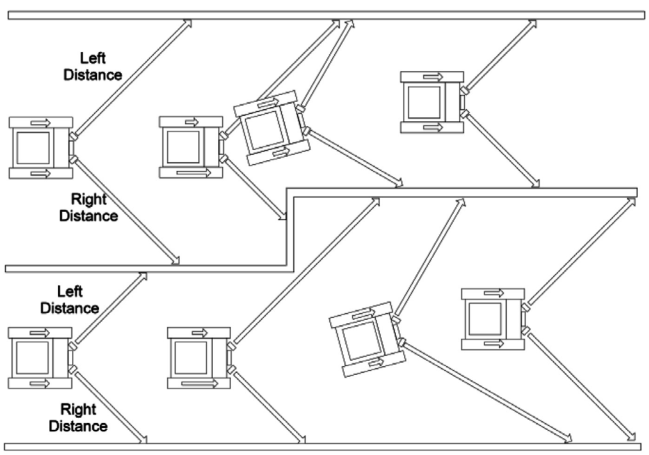

Figure 14: Positive and negative step function experiments

For simple performance measurement, the left and right distances were streamed from the Arduino MEGA to a PC for logging. This measurement method was selected because of a lack of inexpensive and simplified global localization references. All samples for the tests were collected at the 20 Hz control loop frequency.

A second test was devised to entertain the performance possibilities of the controller in a straight corn row, where the distance measurements vary greatly due to gaps between the crops and overhanging leaves present in the LIDAR sensing space. The corn row used in the test was a simulated row made from painted PVC pipe stalks and aluminum foil leaves. The row was arranged to be 91 cm (36 in) wide, and 16 corn stalks long at 20 cm (8 in) spacing. The experiment layout for the second test is depicted in Figures 15 and 16.

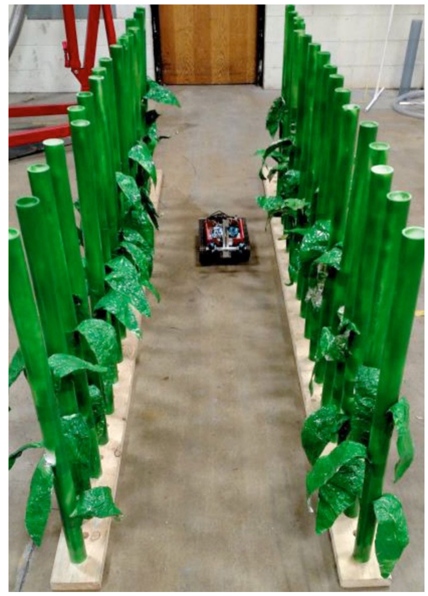

Figure 15: Centering tests in simulated row of corn

Figure 16: Navigation space was irregular in the second test

3. Results

The positive step response of the controller is shown in Figure 17. The rise time of the left distance shows the row becoming suddenly expanded, and settles to a steady state 3 seconds into the test when the robot steers to the left, allowing the right distance to converge with the left. This settling time was 1.5 seconds. The steady state error was evaluated within the 3 to 9 second period of the test. Maximum error in this steady state period was within 2.85 cm (3.14% for 91 cm row width), and had a mean value of 0.74 cm and a standard deviation of 0.88 cm.

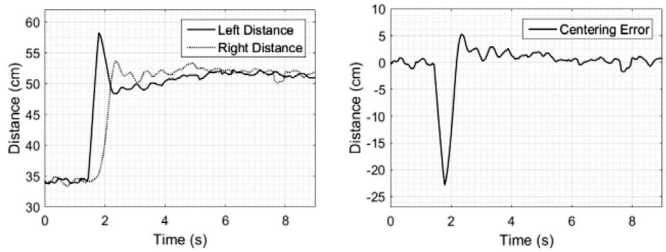

Figure 17: Step input response - row width expansion

The negative step response of the controller is shown in Figure 18. The fall time of the left distance shows the row becoming suddenly compressed, and settles to a steady state 6.5 seconds into the test when the robot steers to the right, allowing the right distance to converge with the left. This settling time was 3.5 seconds. The steady state error was evaluated within the 6 to 12.5 second period of the test. Maximum error in this steady state period was within 2.14 cm (3.5% for 61 cm row width), and had a mean value of 0.22 cm and a standard deviation of 0.69 cm.

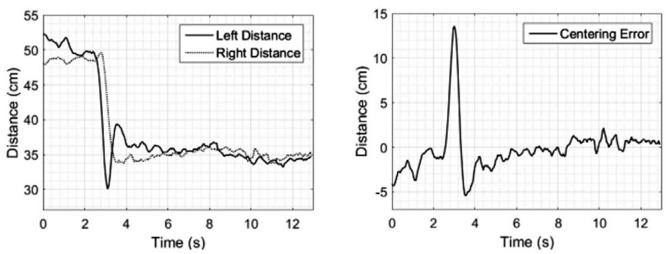

Figure 18: Step input response – row width compression

Response of the controller for simulated corn row navigation is shown in Figure 19. The inconsistency of the navigation references provided large and frequent error impulses to the controller. The entire simulated row was traversed in just over 7 seconds. The centering error was evaluated for the entire 7 second period of the test since the corn row was straight, without any intended step changes included. Maximum error for the test was within 34 cm (37.4% from 61 cm row width). For seconds 2 through 7 of the test, maximum error was within 14.4 cm (18% for 61 cm row width). Error throughout the test had a mean value of -3.55 cm, and a standard deviation of 8.43 cm.

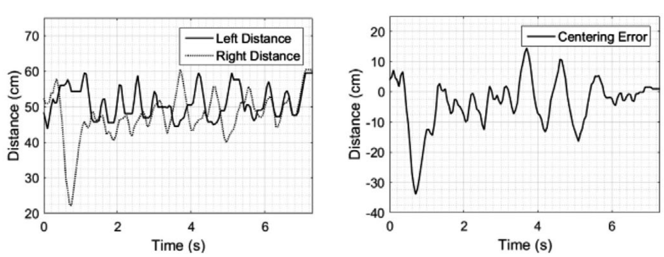

Figure 19: Navigation response for simulated corn row test

4. Discussion

The fuzzy controller as described was able to successfully navigate converging, diverging, and step changes in barrier arrangements made to simulate worst-case corn rows. The robot was able to navigate a simulated corn row with larger and more frequent error than observed in smooth barrier step response tests. Further testing will take place in the field to facilitate the development of crop row measurement techniques that navigate rows despite varying gaps between individual crops and overhanging leaves. This will enable the design of the fuzzy controller to be used in a large variety of row-following systems. As the number of sub-canopy system designs increases, the viability of large-scale data collection of plant-by-plant characterization will increase. These analyses will offer the benefits found in the field of precision agriculture by treating subplots of a field as individually controlled crop systems. As the row-following fuzzy controller is matured and further developed, inexpensive sub-canopy sensing systems can become adapted for automated deployment in both academia and industry, driving the collection of plant-by-plant field diagnostics as a wide-spread sensing technique.

References

Biber, P., Weiss, U., Dorna, M., & Albert, A. (2012). Navigation system of the autonomous agricultural robot Bonirob. In *Workshop on Agricultural Robotics: Enabling Safe, Efficient, and Affordable Robots for Food Production (Collocated with IROS 2012), Vilamoura, Portugal*. Retrieved from http://www.cs.cmu.edu/~mbergerm/agrobotics2012/01Biber.pdf

Darr, M. J. (2004, March). *Development and Evaluation of a Controller Area Network Based Autonomous Vehicle* (Masters). University of Kentucky. Retrieved from http://uknowledge.uky.edu/gradschool_theses/192

Mamdani E.H. (1974). Application of fuzzy algorithms for control of simple dynamic plant. *Electrical Engineers, Proceedings of the Institution of*, *121*(12), 1585–1588. http://doi.org/10.1049/piee.1974.0328

Martin, K., Raun, W., & Solie, J. (2012). By-plant prediction of corn grain yield using optical sensor readings and measured plant height. *Journal of Plant Nutrition*, *35*(9), 1429–1439. http://doi.org/10.1080/01904167.2012.684133

Moran, M. S., Inoue, Y., & Barnes, E. M. (1997). Opportunities and limitations for image-based remote sensing in precision crop management. *Remote Sensing of Environment*, *61*(3), 319–346. http://doi.org/10.1016/S0034-4257(97)00045-X

Shi, Y., Wang, N., Taylor, R. K., Raun, W. R., & Hardin, J. A. (2013). Automatic corn plant location and spacing measurement using laser line-scan technique. *Precision Agriculture*, *14*(5), 478–494. http://doi.org/10.1007/s11119-013-9311-z

Sowers, K. E., Pan, W. L., Miller, B. C., & Smith, J. L. (1994). Nitrogen Use Efficiency of Split Nitrogen Applications in Soft White Winter Wheat. *Agronomy Journal*, *86*(6), 942–948.

Stombaugh, T. S., Benson, E. R., & Hummel, J. W. (1998). Automatic guidance of agricultural vehicles at high field speeds. *ASAE Paper*, *983110*. Retrieved from http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.25.4167&rep=rep1&type=pdf