Computers and Electronics in Agriculture 115 (2015) 118–128

Contents lists available at ScienceDirect

Computers and Electronics in Agriculture

journal homepage: www.elsevier.com/locate/compag

Localization and control of an autonomous orchard vehicle

Gokhan Bayar [a,b,*], Marcel Bergerman [b], A. Bugra Koku [a], E. ilhan Konukseven [a]

[a] Mechanical Engineering Department, Middle East Technical University, Dumlupinar Bulvari, Cankaya, 06800 Ankara, Turkey
[b] Field Robotics Center, Robotics Institute, Carnegie Mellon University, 5000 Forbes Avenue, 15213-3890 Pittsburgh, PA, USA

ARTICLE INFO

Article history:
Received 14 October 2014
Received in revised form 15 May 2015
Accepted 21 May 2015
Available online 10 June 2015

Keywords:
Autonomous orchard vehicle
Orchard automation
Tree fruit production
Agricultural robotics

ABSTRACT

In this paper we propose a novel model-based control method for an autonomous agricultural vehicle that operates in tree fruit orchards. The method improves path following performance by taking into account the vehicle's motion model, including the effects of wheel sideslip, to calculate speed and steering commands. It also generates turn paths that improve visibility of the orchard rows, thus increasing the probability of a successful turn from one row into another, while respecting maximum steering rate limits. The method does not depend on GPS signals for either state estimation or path following, relying instead only on data from a planar laser scanner and wheel and steering encoders. This makes it suitable for real agricultural applications where acquisition cost is key to a farmer's decision to invest in new technologies. We show the controller's stability using Lyapunov functions and demonstrate its feasibility in experiments conducted in an orchard-like nursery.

1. Introduction

Autonomous orchard vehicles have the potential to dramatically transform tree fruit production by automating key operations such as mowing and spraying and augmenting workers conducting pruning, thinning, tree maintenance, and harvesting. From a robotics perspective, all these applications can be enabled with a relatively simple yet challenging capability: safely driving up along one row, turning at the end of the row, and entering the next one. The challenges involved include reliably sensing the trees despite sloped terrain, branches sticking out of the canopy, tall grass, and missing trees; localizing the vehicle within the row; following in- and outside-the-row trajectories; and avoiding obstacles such as people and bins. Furthermore, all of these must be accomplished with as low-cost an infrastructure as possible, to make such vehicles commercially viable.

Over the past three years a family of four autonomous orchard vehicles (we call them Tuscarora, Cascade, Allegheny and Laurel) has been developed with a common sensing and computing infrastructure that allows them to perform the aforementioned drive-and-turn operation continuously for hours. Because of the low-cost requirement, the vehicles do not carry a high-accuracy GPS-assisted inertial navigation system; instead, the low-cost sensing suite infrastructure consists simply of a laser rangefinder mounted on the front of the vehicle and steering and wheel encoders (Fig. 1). The laser rangefinder is "free" because it is needed for row following, therefore using it for localization adds no extra cost to the system. Note that the price of the low-cost sensing suite infrastructure including a laser scanning range finder and encoders of wheel and steering systems is approximately $6.000. On the other hand the price of a high-accuracy GPS-assisted inertial navigation system costs $40–60 k.

To this date the vehicles drove a combined 350 km in experimental and commercial orchards, vineyards, and nurseries in the US states of Washington, Pennsylvania, New York, and Maryland. The longest run lasted five hours and covered 25 km (Singh et al., 2010; Hamner et al., 2012). Time trials conducted by The Pennsylvania and Washington State Universities Extension educators showed that workers onboard an autonomous orchard platform can be twice as efficient as workers using ladders or on foot when working on the top portion of apple and peach trees (Fig. 2) (Davis, 2012). The vehicles, three examples of which are shown in Figs. 1 and 2, are based on Toro Workman MDE electric platform which has been retrofitted with the mechanical, electrical, and software elements necessary for autonomous driving. The retrofitting of the vehicles was done at the Field Robotics Center of Carnegie Mellon University under the Comprehensive Automation of Specialty Crops (CASC) project.

Despite the encouraging results obtained thus far, significant advances are needed before these vehicles can become part of a

* Corresponding author at: BEU, Mechanical Eng. Dept., incivez Mah., 67100, Zonguldak, Turkey. Tel.: +90 3722574010/1223.
E-mail addresses: gbayar@gmail.com (G. Bayar), marcel@cmu.edu (M. Bergerman), kbugra@metu.edu.tr (A.B. Koku), konuk@metu.edu.tr (E. ilhan Konukseven).

http://dx.doi.org/10.1016/j.compag.2015.05.015
0168-1699/

Fig. 1. (Left) Autonomous orchard vehicle "Tuscarora." The vehicle is capable of following an orchard row, turning at the end of the row, and entering the next one using only the hood-mounted laser rangefinder and steering and wheel encoders. (Right) Tuscarora during tests at the Fruit Research and Extension Center in Biglerville, PA. Here workers on foot harvest apples and deposit them on bins towed by the vehicle, eliminating the need for and the cost of a tractor driver.

Fig. 2. Autonomous orchard vehicles "Cascade" (left) and "Allegheny" (right). In these versions the vehicles are equipped with a lift platform from where workers can perform operations such as pruning, thinning, tree tying, and pheromone dispenser placing. Work on the top part of the trees, when conducted on the platform, can be twice as efficient as worked performed on ladders.

farmer's toolbox. Among others, at the top of our list is a path generation and control methodology that compensates for unmodeled dynamics—chief among them is rear wheel sideslip—and provides better path tracking then the current pure pursuit controller (Hamner et al., 2009, 2011). This is particularly important during the first part of the turn, when the empty space ahead of the vehicle does not provide any clues for localization via range and the vehicle has to rely solely on dead reckoning for navigation. Errors accumulated during this part of the traversal may prevent the vehicle from properly aligning itself with the next row, thus making it harder to find and enter it reliably.

In this paper we extend the control methodology proposed by Fang et al. (2006) and Eaton et al. (2009) in three directions. First, we add a harmonic term to the system's model to account for the effects of wheel sideslip and other unmodeled dynamics; and demonstrate the system's stability via the Lyapunov method. Second, we design the turns with the triple purpose of increasing the vehicle's visibility of the next row, making the turns "natural-looking" from a user perspective, and limiting steering rate. The latter is important to limit the steering motor current and therefore increase its operating life. The path generator concatenates straight and circular primitives interpolated by a fourth-order polynomial; it combines mathematical simplicity (and therefore ease of implementation) with a natural-looking behavior in which the vehicle positions itself parallel to and along the center of the next row before entering it. Third, we use only the vehicle's laser rangefinder and steering and wheel encoders to generate all signals needed by the feedback controller. This independence of accurate sensing such as that provided by differential GPS—which is not only expensive but also somewhat unreliable under thick orchard canopies—makes our work more amenable to adoption by farmers.

This paper is organized as follows. In Section 2 we review the relevant literature, with focus on path control for orchard vehicles. In Section 3 we formalize the problem statement, including the vehicle model used in the controller design. In Section 4 we present the contributions of the paper: a path generation and control methodology for improved autonomous vehicle navigation in orchards. In Section 5 we present experimental results obtained at our orchard-like experimental nursery in Pittsburgh, PA. We conclude the paper in Section 6 with an analysis of results and a line-up of future work.

2. Literature review

Generally speaking, the problem of reliable autonomous driving in orchards can be decomposed in various sub-problems: detection of tree rows and extraction of an in-row path between them; generation of an outside-the-row turn path; in- and outside-the-row path tracking; speed and orientation control; and obstacle detection and (possibly) avoidance. We review here the relevant literature in all but the last topic, as it is not dealt with in this paper. For a discussion on obstacle detection for orchard vehicles, see (Freitas et al., 2012a).

Bak and Jakobsen (2004) proposed a four-wheeled vehicle for weed detection, focusing on row finding and guidance and on weed detection models. Their vehicle was equipped with a magnetic compass, heading gyroscope and GPS. The experiments were conducted in an open field. Fang et al. (2005) proposed a robust adaptive controller for guidance of autonomous farm vehicles in the presence of slippage. A backstepping control procedure was used to design the adaptive controller, with the slippage effects learned and compensated for by parameter adaptation. The method was demonstrated in actual field experiments by using an autonomous farm tractor with a differential GPS (Fang et al., 2006). Ampatzidis et al. (2006) proposed a four-stage procedure to the problem of autonomous tractor navigation in orchard applications:

field coverage planning, motion and action generation, motion planning, and trajectory generation, including straight and turning motions. They created two turning strategies between rows called straight alternation and overlapping alternation patterns. The studies presented were conducted in simulation only. Barawid et al. (2007) developed an autonomous navigation system for an orchard tractor using a two-dimensional laser rangefinder. They extracted the supporting lines of tree rows from laser data using a Hough transform, and used them to correct the lateral and orientation errors with respect to the center line. In addition to laser scanner, a heading sensor and RTK-GPS were used for navigation. Eaton et al. (2008) proposed a wheel slippage compensation control method for agricultural vehicles, and validated it on a small weeding robot. Sliding mode control was used for robust trajectory tracking using a stereo camera pair, a laser scanner, and a GPS. Gonzalez et al. (2009) addressed trajectory tracking and localization for an autonomous mobile robot using feedback linearization techniques. Their study was conducted using a small differential mobile robot following a 14-m straight path followed by a 10-m wide U-shape turn. Johnson et al. (2009) developed a team of three autonomous tractors for peat moss harvesting. The tractors were capable of coordinated driving for efficient field coverage. Color and infrared cameras, laser scanner, and GPS composed the sensing suite. Huynh et al. (2010) addressed the path tracking problem for a tractor-trailer by defining an error model based on the kinematic model of the vehicle and desired path definition. The proposed methodology was tested in a simulation environment. Lenain et al. (2010) presented adaptive and predictive control of an off-road, four-wheel, car-like robot. At the core of their work was a wheel slippage estimator based on the robot's kinematic and dynamic models. Trajectory tracking was obtained via position feedback coming from differential GPS, with experiments conducted in an obstacle-free open field. Matveev et al. (2010) studied the tracking problem for an autonomous farming tractor. A mixed controller combining nonlinear and sliding mode strategies was proposed and adapted for a curved path following task. The methodology was tested in a simulation environment. Weiss and Biber (2011) proposed a model for detection and segmentation of plants and ground using a laser rangefinder, and implemented it on an autonomous agricultural vehicle to perform field mapping.

What distinguishes our work from the ones presented above is that we address autonomous orchard vehicle navigation in a systemic and comprehensive way. Systemic because we start from the perspective of the end user and look for solutions that use only low-cost sensors (compared to differential GPS). Comprehensive because our method combines system modeling—including wheel sideslip—, controller design—including regulation of the steering rate—, path planning, stability analysis, and successful field experimentation.

3. Problem statement

Consider a four-wheel autonomous orchard vehicle such as the one pictured in Fig. 1. Its goal is to follow the row between the trees, possibly with an offset to one side depending on the task; detect the end of the row; turn around so as to align itself parallel to and along the center of the next row; and enter it. By repeating this pattern continuously, the vehicle enables all production operations listed in Section 1, i.e., it allows for both production automation and human worker augmentation.

For controller design purposes, we model the vehicle as a kinematic platform, because at the speeds we operate most dynamic effects are negligible. (Wheel sideslip is a non-negligible dynamic effect that is considered in the controller design, see Section 4.) The model is given by:

$$\begin{aligned} \dot{x} &= \upsilon \cos\theta \\ \dot{y} &= \upsilon \sin\theta \\ \dot{\theta} &= \omega = \tfrac{\upsilon}{L} \tan\phi \end{aligned} \tag{1}$$

where (x,y) are the Cartesian coordinates of the center of the rear axle in an inertial reference frame, θ is the vehicle orientation with respect to that frame, ω is the vehicle angular speed, ϕ is the steering angle, υ is the forward speed, and L is the inter-axis distance (Fig. 3).

Assume the vehicle must follow a desired path $\Gamma = [x_d\ y_d\ \theta_d]^T$. As in (Fang, 2004; Fang et al., 2006; Eaton et al., 2009), we define the longitudinal error x_e, lateral error y_e, and orientation error θ_e as (Fig. 4):

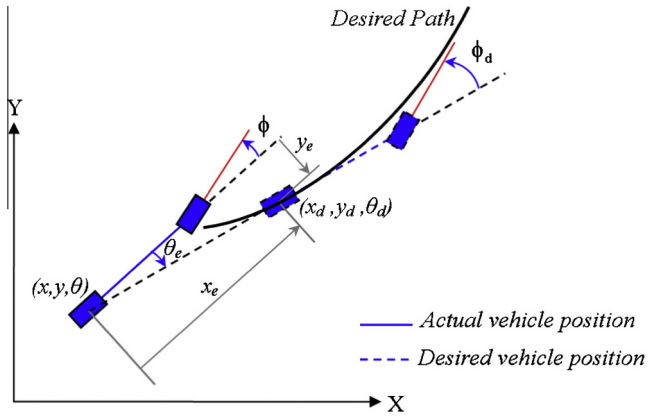

Fig. 4. The vehicle's longitudinal, lateral, and orientation errors.

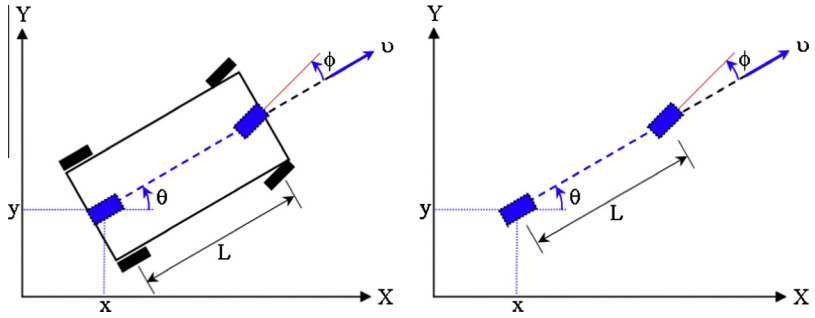

Fig. 3. (Left) Kinematic model of the autonomous orchard vehicle used for controller design. (Right) Because of the vehicle's symmetry it can be represented by a bicycle model. The pair (x, y) represents the coordinates of the center of the rear axle. See the text for a definition of the other variables.

$$\begin{bmatrix} x_e \\ y_e \\ \theta_e \end{bmatrix} = \begin{bmatrix} \cos\theta & \sin\theta & 0 \\ -\sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_d - x \\ y_d - y \\ \theta_d - \theta \end{bmatrix} \quad (2)$$

Taking the time derivative of Eq. (2) we obtain:

$$\begin{aligned} \dot{x}_e &= -\upsilon + \upsilon_d \cos\theta_e + y_e\omega \\ \dot{y}_e &= \upsilon_d \sin\theta_e - x_e\omega \\ \dot{\theta}_e &= \upsilon_d C(s) - \tfrac{\upsilon}{L}\tan\phi \end{aligned} \quad (3)$$

where $C(s)$ is the curvature along the path and υ_d is the desired speed.

4. Vehicle control and path generation

In this section we present the contributions of this paper: a path tracking controller with proven stability combined with a path generation methodology that is simple to implement and yields natural-looking turns; and a methodology to estimate the vehicle's orientation in the absence of a GPS-assisted inertial navigation system.

4.1. Control system architecture

Fig. 5 presents the control system architecture that implements the model-based controller. The inputs to the controller are the desired path provided by the Path Generator, described in Section 4.3, the lateral and orientation errors provided by the Vehicle Localization module, described in Section 4.4, and the vehicle's current forward speed and steering angle.

4.2. Path tracking controller design

Assume for a moment that the vehicle's position (*x*,*y*), orientation θ, steering angle ϕ, and forward speed υ are known. Following (Fang, 2004; Fang et al., 2005; Fang et al., 2006; Eaton et al., 2009), define the candidate Lyapunov function:

$$V_1 = \frac{1}{2}x_e^2 + \frac{1}{2}y_e^2 \quad (4)$$

such that:

$$\dot{V}_1 = x_e\dot{x}_e + y_e\dot{y}_e \quad (5)$$

Using Eq. (3) this can be written as:

$$\dot{V}_1 = x_e(-\upsilon + \upsilon_d\cos\theta_e + y_e\omega + \varepsilon_1) + y_e(\upsilon_d\sin\theta_e - x_e\omega + \varepsilon_2) \quad (6)$$

where the terms ε_1 and ε_2 account for the errors introduced by the rear wheel sideslip and other unmodeled dynamics. Following (Eaton et al., 2009; Fang et al., 2006), these terms can be modeled as harmonic functions of the form:

$$\varepsilon_1 = \delta_1\sin(\hbar_1\theta) \text{ and } \varepsilon_2 = \delta_2\cos(\hbar_2\theta) \quad (7)$$

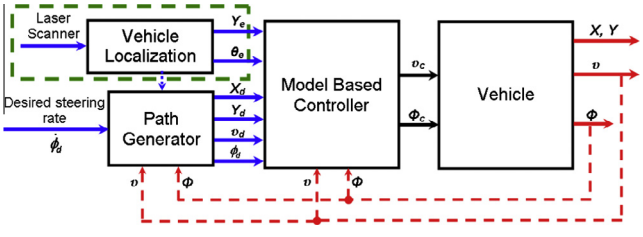

Fig. 5. Control system architecture of the autonomous orchard vehicle. The vehicle localization module is detailed in Fig. 9.

where we define the controller design parameters as $\delta_1, \delta_2, \hbar_1, \hbar_2$.

Select the vehicle forward speed υ and a time-varying function Ω as follows:

$$\begin{aligned} \upsilon &= \upsilon_d\cos\theta_e + k_x x_e + \tau_1 x_e\lambda_1^2(t) \\ \Omega &= \upsilon_d\sin\theta_e + k_y y_e + \tau_2 y_e\lambda_2^2(t) \end{aligned} \quad (8)$$

where $k_x > 0$, $k_y > 0$ are user-defined control gains, $\tau_1 > 0$, $\tau_2 > 0$ are tuning parameters, and $\lambda_1(t)$, $\lambda_2(t)$ are time-varying functions such that $\lambda_1(t) > |\varepsilon_1|$, $\lambda_2(t) > |\varepsilon_2|$. Using Eqs. (6) and (8) we obtain:

$$\dot{V}_1 = -k_x x_e^2 - k_y y_e^2 - \tau_1 x_e^2\lambda_1^2(t) - \tau_2 y_e^2\lambda_2^2(t) + x_e\varepsilon_1 + y_e(\Omega + \varepsilon_2) \quad (9)$$

which can also be written as:

$$\begin{aligned} \dot{V}_1 \leqslant & -k_x x_e^2 - k_y y_e^2 - \tau_1\left(|x_e|\lambda_1(t) - \frac{1}{2\tau_1}\right)^2 \\ & - \tau_2\left(|y_e|\lambda_2(t) - \frac{1}{2\tau_2}\right)^2 + \frac{1}{4\tau_1} + \frac{1}{4\tau_2} + y_e\Omega \end{aligned} \quad (10)$$

Define now the candidate Lyapunov function:

$$V_2 = V_1 + \frac{1}{2}\Omega^2 \quad (11)$$

such that:

$$\dot{V}_2 = \dot{V}_1 + \Omega\dot{\Omega} \quad (12)$$

The derivative of Ω is given by:

$$\dot{\Omega} = \upsilon_d\dot{\theta}_e\cos\theta_e + \dot{y}_e(k_y + \tau_2\lambda_2^2(t)) + 2\tau_2 y_e\lambda_2(t)\frac{\partial\lambda_2(t)}{\partial t} \quad (13)$$

Finally, substituting Eq. (13) in Eq. (12) gives:

$$\begin{aligned} \dot{V}_2 \leqslant & -k_x x_e^2 - k_y y_e^2 - \tau_1\left(|x_e|\lambda_1(t) - \frac{1}{2\tau_1}\right)^2 \\ & - \tau_2\left(|y_e|\lambda_2(t) - \frac{1}{2\tau_2}\right)^2 + \frac{1}{4\tau_1} + \frac{1}{4\tau_2} + y_e\Omega \\ & + \Omega\left\{\upsilon_d\dot{\theta}_e\cos\theta_e + \dot{y}_e(k_y + \tau_2\lambda_2^2(t)) + 2\tau_2 y_e\lambda_2(t)\frac{\partial\lambda_2(t)}{\partial t}\right\} \end{aligned} \quad (14)$$

For stability we need $\dot{V}_2 \leqslant 0$, which is satisfied when

$$\omega = \frac{k_\Omega\Omega + y_e + \upsilon_d^2\cos\theta_e C(s) + (k_y + \tau_2\lambda_2^2(t))(\upsilon_d\sin\theta_e + \varepsilon_2) + 2\tau_2 y_e\lambda_2(t)\frac{\partial\lambda_2(t)}{\partial t}}{\upsilon_d\cos\theta_e + x_e(k_y + \tau_2\lambda_2^2(t))} \quad (15)$$

where $k_\Omega > 0$ is a third user-defined control gain. With the choice of vehicle angular speed given in Eq. (15), the derivative of the candidate Lyapunov function becomes

$$\dot{V}_2 \leqslant -k_x x_e^2 - k_y y_e^2 - k_\Omega\Omega^2 + \frac{1}{4\tau_1} + \frac{1}{4\tau_2} \quad (16)$$

The first three terms of Eq. (16) are always negative; therefore, stability is guaranteed as long as τ_1, τ_2 are chosen "large" enough. The control signals applied to the vehicle therefore become:

$$\begin{aligned} \upsilon_c &= \upsilon_d\cos\theta_e + k_x x_e + \tau_1 x_e\lambda_1^2(t) \\ \phi_c &= arc\tan\left(\tfrac{L\omega}{\upsilon}\right) \end{aligned} \quad (17)$$

where ω is given by Eq. (15).

4.3. Turn path generation

The second element of our orchard vehicle navigation strategy, and a minor contribution of this paper, is a path generation methodology that guides the vehicle to an appropriate position at the end of the turn. In our experience, two factors determine where the vehicle should be before attempting to enter the next

row. First, from an autonomy perspective, the laser scanner should be as centered with respect to and pointed as parallel to the next row as possible. Clearly, this maximizes the detection of the trees in the target row and decreases the amount of noise introduced by trees in neighboring rows, which at times may confuse the autonomy system (Hamner et al., 2012). Second, from the user's perspective, the autonomous turn should be performed in a way similar to the way a person would follow when driving the vehicle manually. Qualitatively speaking, this means driving quickly away from the exit row, turning while at a considerable distance from the trees, and ending the turn with a predefined distance between the vehicle and the next row, so as to give the driver time and space for final adjustments. As luck would have it, the two factors roughly dictate the same turn behavior. We add here a third factor: limit the steering speed to a pre-defined value to prevent steering motor wear and tear.

The turn methodology is illustrated in Fig. 6. The safety parameters S_S, S_L, S_W, and S_F, collectively denoted the S-parameters, determine the form of the turn path and are selected in an ad-hoc way based on the orchard layout.

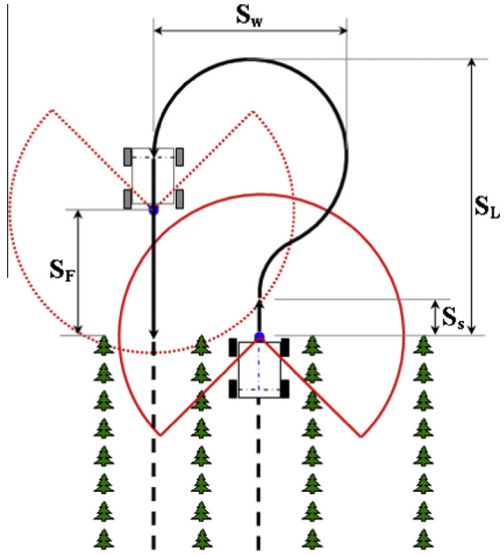

Fig. 6. From a robotics perspective the orchard vehicle's goal is to follow an orchard row, detect the end of the row, turn around, and enter the next. The S-parameters S_S, S_L, S_W, and S_F determine the form of the turn path. At the end of the turn, the vehicle has to be aligned parallel to and along the center of the next row.

To determine the turn path, we initially compared two diametrically-opposite solutions. The first is a static path composed by the concatenation of straight and circular primitives, as shown in Fig. 7 (left). The vehicle starts at the point (0,0), advances out of the row, and follows a symmetrical bulb-shaped path until it is parallel to and centered with the next row. Because the vehicle's frame of reference is at the rear axle, in practice the vehicle appears to be driving straight toward the trees on its left side, only to turn away from them and into the row at the last second. This path is easy to compute either in real time or as a look-up table, but last-minute avoidance of trees and sharp turns are not acceptable in autonomous drive.

The second solution is based on clothoids calculated via a real-time optimization algorithm (Shin and Singh, 1990; Kanayama and Hartman, 1989). An example is shown in Fig. 7 (center). In this case the vehicle is ready to enter the next row much earlier than in the previous case, making the turn very natural-looking; however, the real-time calculations necessary require computing power that drives up the cost of the vehicle.

The solution we adopted combines the bulb turn's simplicity and amenability to real time and look-up table calculation, with the clothoid turn's natural-looking behavior. We call it the "thumb" turn for its characteristic shape (Fig. 7, right). It consists of a concatenation of three circular primitives with radii R_1, R_2, R_3 and length S_1, S_2, S_3, and a final straight motion into the row (Fig. 8, left). The R_i are selected from a look-up table so as to obey the S-parameters specified in Fig. 6. Once the path shape is determined, it is transformed into a steering angle set-point, which in turn is interpolated by a fourth-order polynomial to provide a region of desirable steering angles and limit the steering speed (Fig. 8, right). In this transformation process, the fourth-order polynomial does not alter the perceived row width significantly, and therefore the vehicle can use the nominal width to plan the turning path.

At this point it is important to mention that when the vehicle is following the row between the trees, the desired path (Γ_{desired}) is simply the row centerline plus or minus a user-defined offset if the task requires the vehicle to be closer to one side than the other. The tree lines can be extracted from laser data using two methodologies; the first is a particle filter solution, yielding two line equations, details can be found in (Dellaert et al., 1999; Gustafsson et al., 2002; Hamner et al., 2012) and are omitted for brevity. The second method is extracting the supporting lines of the tree rows of which details are given in Section 4.4.

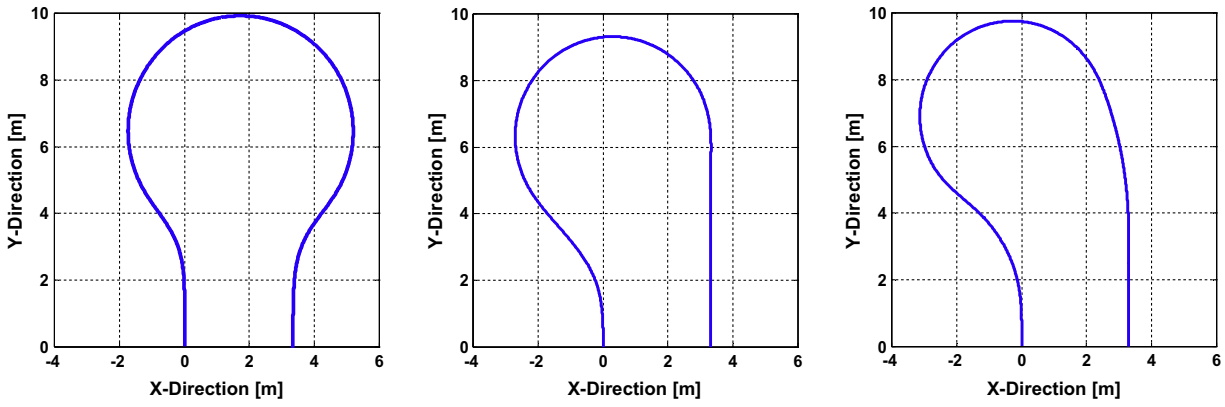

Fig. 7. Three possible solutions for the orchard vehicle turn path. Left to right: "bulb," "clothoid," and "thumb" turn. The vehicle starts at the point (0,0), which represents the end of the current row, and turns to the right. All graphs are for a typical orchard with row width of 3.3 m, S_S = 2 m, S_L = 9.5 m, and S_W = 6 m.

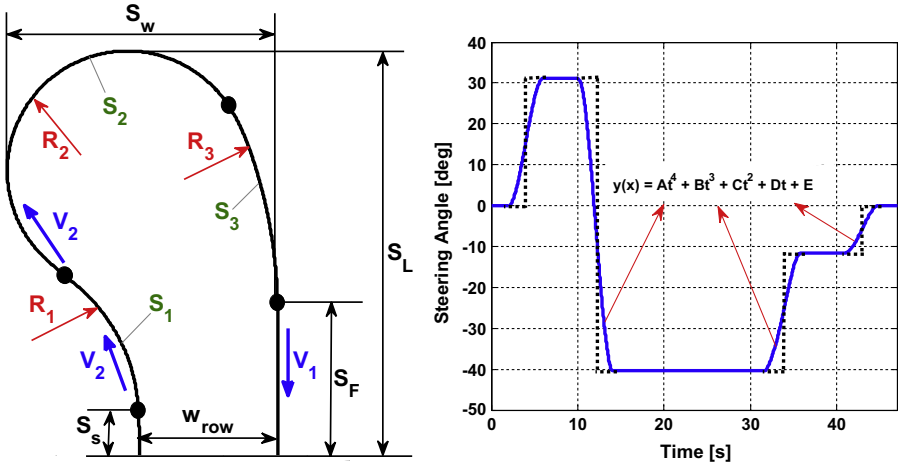

Fig. 8. (Left) Parameters of the thumb turn, calculated when the vehicle reaches the end of the row based on the S-parameters in Fig. 6. (Right) The corresponding steering angle set-point (dashed line) is interpolated by a fourth-order polynomial to eliminate high steering acceleration and limit the steering speed. The vehicle's forward velocities in straight and turning motions are V_1 and V_2, respectively.

4.4. Vehicle localization

Recall that the control methodology we propose depends on knowledge of the vehicle's position (x,y), orientation θ, steering angle ϕ, and forward speed υ. Recall also that the only sensors available are a laser rangefinder and steering and wheel encoders. For the purposes of this work, we rely on encoder-based odometry ("dead reckoning") to calculate the vehicle position along the row and during the turn. We make the reasonable assumption that the rows' length is known. Navigation is then simplified to calculating how far the vehicle has driven along the row and looking for empty space—signaling the end of the row—when the distance traveled is a fraction (say, 90%) of the row length. When the empty space is detected, the turn behavior is invoked. As the vehicle enters a new row, its position can be reset, thus eliminating any odometry errors accumulated thus far. (In a parallel work we have integrated a more accurate laser-based localization system; see (Freitas et al., 2012b) for details.) Speed is calculated by differentiating and filtering the encoder odometry, and the steering angle is provided directly by the steering encoder. For path tracking purposes, therefore, what is left to determine are the vehicle's lateral error y_e and orientation error θ_e. These are provided by the Vehicle Localization module in Fig. 5, shown in detail in Fig. 9.

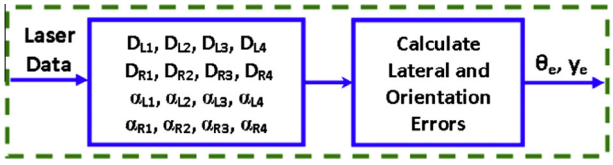

Fig. 9. Detail of the Vehicle Localization block in Fig. 5. Laser data is used to extract the supporting lines of the tree rows in the form of two linear equations, and the vehicle's distance to and orientation with respect with the trees ahead of it. The result is the lateral and orientation errors with respect to the desired path, which are inputted to the model-based controller. The physical meaning of the D and α parameters is shown in Fig. 10.

This determination is done by using the laser rangefinder data to locate the trees in the vicinity of the vehicle. For each tree to the left of the vehicle we record its distance D_{Li} from the laser, and analogously record the distance D_{Ri} to the trees on the right. Likewise, we find the angles α_{Li} and α_{Ri} (Fig. 10). Using trigonometry we can then calculate the coordinates of each tree with respect to the vehicle's coordinate frame, which we denote (x_{Li}, y_{Li}) for the trees on the left and (x_{Ri}, y_{Ri}) for those on the right. Next we

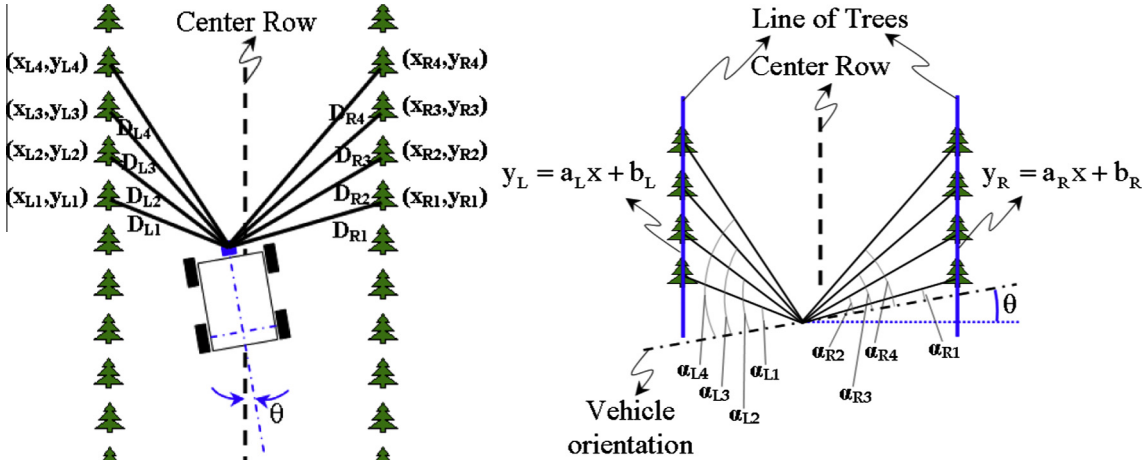

Fig. 10. Lacking a gyroscope or inertial measurement unit onboard the orchard vehicle, we calculate its orientation with respect to the tree lines via the distance to and orientation with respect to individual trees. The method is tolerant to missing trees, as long as there are at least two on each side within the laser's range.

calculate the supporting lines of the trees on the left, $y_L = a_L x_L + b_L$ and the right, $y_R = a_R x_R + b_R$:

$$\begin{bmatrix} b_L \\ a_L \end{bmatrix} = \begin{bmatrix} n & \sum_{i=1}^{n} x_{L,i} \\ \sum_{i=1}^{n} x_{L,i} & \sum_{i=1}^{n} (x_{L,i})^2 \end{bmatrix}^{-1} \begin{bmatrix} \sum_{i=1}^{n} y_{L,i} \\ \sum_{i=1}^{n} x_{L,i} y_{L,i} \end{bmatrix} \quad (18)$$

and equivalently for a_R, b_R. The orientation of the vehicle, θ, is given by:

$$\begin{aligned} \theta &= a\tan 2\{(\tfrac{\jmath}{2}), (d_k - d_{k-1})\} \text{ when } y_L < y_R \\ \jmath &= \begin{cases} (y_R - y_L)_k - (y_R - y_L)_{k-1} \text{ if } (y_L)_k < (y_L)_{k-1} \\ -(y_R - y_L)_k + (y_R - y_L)_{k-1} \text{ if } (y_L)_k > (y_L)_{k-1} \end{cases} \\ \theta &= a\tan 2\{(\tfrac{\jmath}{2}), (d_k - d_{k-1})\} \text{ when } y_L > y_R \\ \jmath &= \begin{cases} -(y_R - y_L)_k + (y_R - y_L)_{k-1} \text{ if } (y_L)_k < (y_L)_{k-1} \\ (y_R - y_L)_k - (y_R - y_L)_{k-1} \text{ if } (y_L)_k > (y_L)_{k-1} \end{cases} \end{aligned} \quad (19)$$

where the current and previous timestamps are denoted k and $k-1$, respectively, and the distance traveled between those instants, measured by the wheel encoder, is d_k. When $y_L < y_R$, the vehicle is located on the left side of the row center (Fig. 10); conversely, when $y_L > y_R$ it is on the right side of the row center. When $y_L = y_R$, the vehicle is located exactly at the center of the row.

Finally, the lateral and orientation errors inside the row are given by:

$$y_e = |y_R - y_L|, \theta_e = |\theta_d - \theta|, \quad (20)$$

Note that Eq. (20) assumes the vehicle's desired trajectory is along the center of the row. If the vehicle is to traverse closer to one side or the other, the corresponding off-set must be added to the calculation of the lateral error y_e.

5. Experimental results

To verify the validity of the model-based control, headland turning and row localization methods proposed in Section 4, we implemented and tested them in field conditions. The model-based controller was compared to the pure pursuit control strategy. Note that the latter uses the particle filtering for localization while the former uses the localization methods proposed here. Note also that the proposed path tracking algorithm is used both during row following and during turns.

The autonomous orchard vehicle used in the experiments reported here, shown in Fig. 1, is based on the Toro Workman MDE. The stock vehicle was augmented with steering and brake motors (Dynetic Systems, DC Servo Motor, 48V/9.2A), steering encoder (Hohner Encoder with 12 bits precision), wheel encoder (Toro Workman MDE encoder), laser rangefinder (Sick LMS111), and a custom electronic speed controller from Sensible Machines Inc. The maximum steering rate is 50°/s.

The experiments were conducted at our orchard-like nursery located at Robot City, a 178-acre site along the Monongahela River in Hazelwood, PA. Formerly the site of LTV Steel, it currently houses many CMU autonomous robotics projects that share its vast open spaces, roads, and buildings. Fig. 11 shows the nursery and the geographical disposition of the eight tree rows and seven driving rows. The real length and width values of each row are given in Table 1.

Each experiment consisted in starting the vehicle at the beginning of row 1 and commanding it to autonomously follow all rows until the end of row 7. The commanded speed both inside and outside the rows was 0.5 m/s (1.1 mph), which is a typical travel speed when workers are not riding onboard. (In operations such as the one seen in Fig. 2, speeds reach as low as 0.05 m/s.) The two variables of interest to measure controller performance are the lateral error, or deviation from the row center (and offset, if applicable), and the steering angle. To verify the path following properties of the model-based controller (MBC) + localization method proposed here, we repeated the experiments with the pure pursuit controller (PPC) + particle filter. Details about the PPC + particle filter can be found in (Hamner et al., 2009). In the experimental results presented, the comparison of the MBC and the PPC methods include (the MBC + localization method proposed) and (the PPC + particle filter), respectively.

Fig. 12 shows the results while the vehicle is in row 1. The graph on the left shows the lateral error; clearly, the MBC provides better path tracking than the PPC. This is in part explained by the steering angle commanded by each method (right graph). The lateral error histogram for row 1 is shown at the bottom of Fig. 12. The system including non-model-based PPC strategy and particle filter generates control signals of higher amplitude and frequency when compared to our model's (the MBC + localization method), sending the vehicle away from the centerline more often. This is because (the

Table 1
Length and width values of eight lines of trees and seven driving rows.

	Row 1	Row 2	Row 3	Row 4	Row 5	Row 6	Row 7	Row 8
Length (m)	52.9	53.1	53.1	53.1	53.2	53.2	52.9	53.1
Width (m)	4.4	3.9	3.5	3.2	3.1	3.3	4.7	–

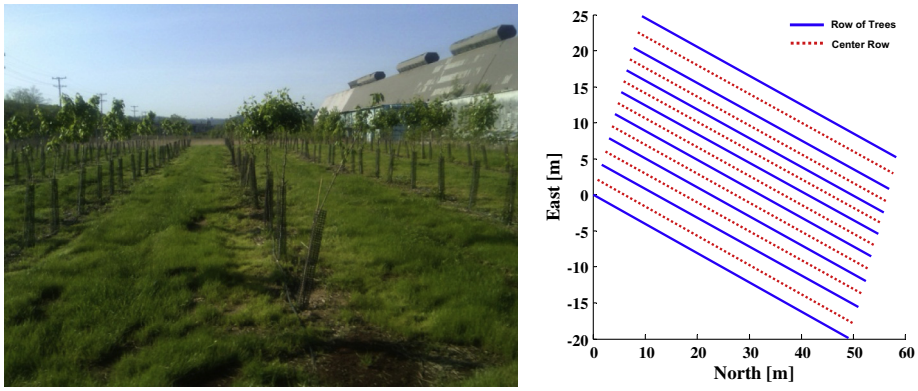

Fig. 11. (Left) Robot City nursery, a one-acre, eight-row orchard-like environment for autonomous vehicle testing. The nursery is located within the larger 178-acre site that formerly housed LTV Steel and is now shared by many CMU robotics projects. (Right) Geographical disposition of the eight lines of trees and seven driving rows in the Robot City nursery.

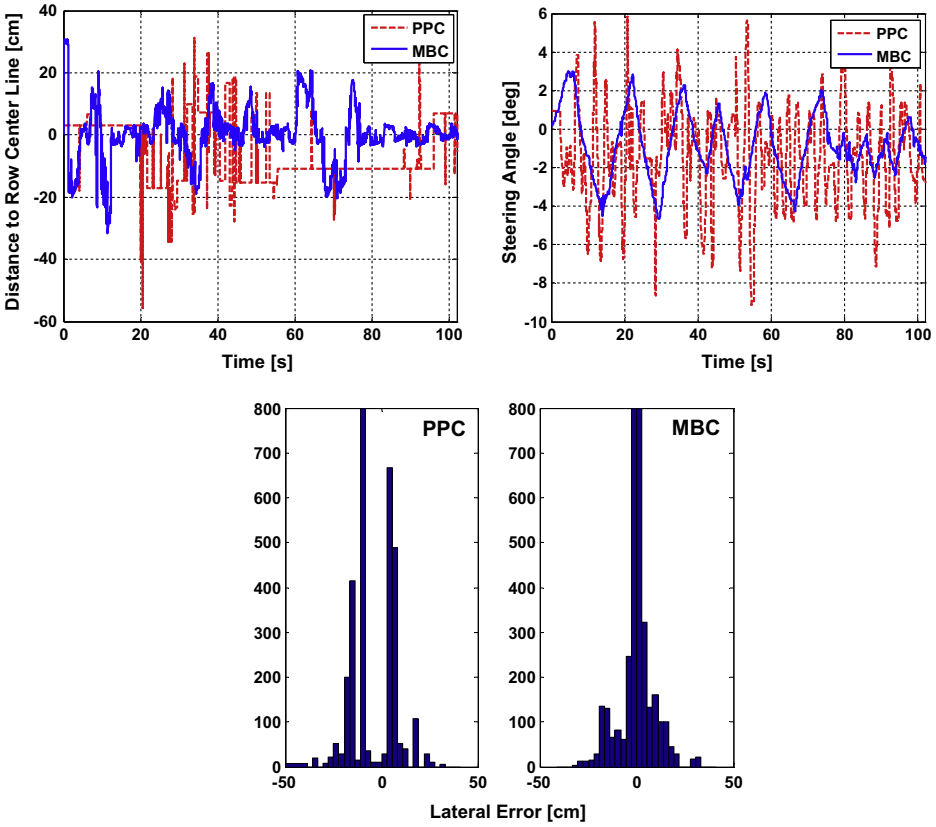

Fig. 12. Experimental comparison of the model-based controller (MBC) + localization method proposed and the pure pursuit controller (PPC) + particle filter in row 1 of the Robot City nursery. Top left: lateral errors; top right: steering angles; bottom: lateral error histograms for both methods (*Y* axis shows the number of lateral error values). The proposed method provides better path tracking and commands a smoother steering, thus extending the life of the steering motor.

PPC + particle filter) output is calculated based only on the lateral error with respect to the row center, and not on the vehicle's dynamic model or maximum steering rate. As a secondary effect, the much higher steering rate of the PPC taxes the steering motor and may cause it to overheat. One could conceivably decrease the PPC gain to reduce the amplitude of the oscillation, but that would just make the vehicle response more slowly. Additionally, the steering system is chain-based and therefore has some natural backlash; and the backlash affects both controllers equally; still, the MBC presents smaller control amplitude and better performance.

A natural question that arises when looking at the error plots in Fig. 12 is, how is it possible sudden and unexpected increase in lateral error, given that the vehicle is nonholonomic? The explanation is that, of course, the vehicle can never have such instantaneous lateral error changes; it is the estimation of the lateral error that does. Recall that, in the absence of a GPS-assisted inertial navigation system, we determine the vehicle's orientation and lateral error using measurements from the laser rangefinder. In particular, because the canopy shape varies significantly along the row, the estimation of the tree lines is not continuous (Fig. 13). This is reflected in the calculation of the lateral error.

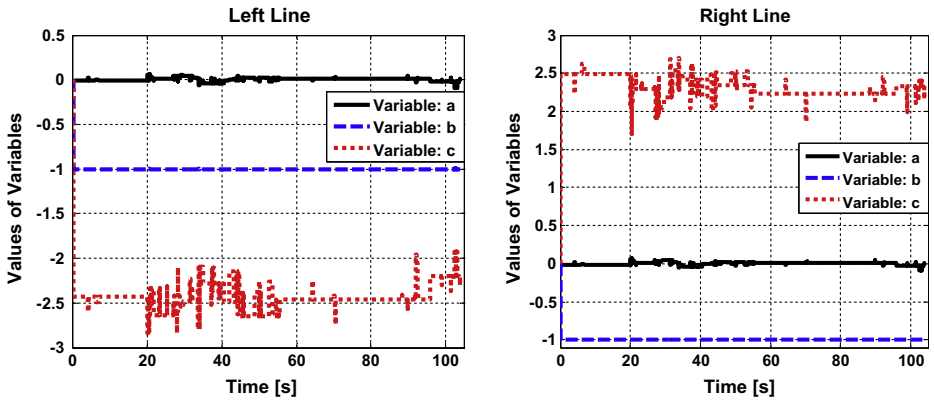

Fig. 13. Values of the tree line parameters *a*, *b*, *c* provided by the particle filter. Note that the larger variations are in parameter *c*, which indicates that the lines detected are essentially parallel to each other. The jumps in this parameter lead to apparent "jumps" in the vehicle's lateral error, as seen in Fig. 12.

Fig. 14. Autonomous orchard vehicle turning under the command of the proposed model-based controller + localization method. The vehicle aligns itself with the upcoming row following the natural-looking "thumb" path.

When the vehicle is turning from one row to another, we cannot calculate the lateral error because there are no trees to feed the localization process, and therefore we cannot show the MBC performance in a quantitative way. In this case, the best we can do is show a sequence of snapshots of the vehicle turning (Fig. 14), and a graph showing the steering angle commanded by the controller (Fig. 15). Note how the MBC provides a smooth "ride" by avoiding high-amplitude, high-frequency steering.

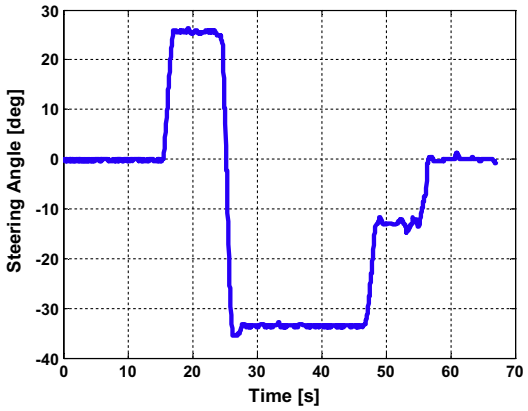

Fig. 15. Steering angles commanded by the model-based controller during the turn from row 1 to row 2. A typical fourth-order polynomial in this case has parameters $A = 1.465e{-}14$, $B = -0.9375$, $C = 28.125$, $D = -270.0$ and $E = 840.0$.

As seen in Fig. 14, the proposed method was tested on a flat terrain orchard with dry and muddy soil. The steering system uses a chain mechanism that introduces backlash in the system. The backlash disturbs the proposed model-based and pure pursuit controllers equally, with the former presenting smaller tracking errors than the latter.

A closer view of experimental data for presenting the localization and backlash contribution and the effects of controllers is shown in Fig. 16. The lateral errors recorded during autonomous drive in the seventh row of the experimental area are given in this figure with a zoomed view. The zoomed view indicates that the big ratio of lateral errors caused by some reasons, i.e. slippage, sensors accuracy, backlash, system model, etc. cannot be avoided when the PPC is in use. On the other hand, smoother motion is obtained if the MBC is used even if the reasons mentioned above are still available. In order to make comparison and analysis about the performance of the proposed control strategies, a trajectory following result is shown in Fig. 17. As seen in this figure, the row detected using the row-detection system has nearly length of 50 m and width of 4.5 m. Desired trajectory, which should be followed, is indicated as the row center and it is tracked using the MBC and the PPC methods.

The entire seven-row trip takes about 370 m. Lateral errors recorded during this entire trip are shown in Fig. 18 as a histogram plot. Noteworthy is the fact that most of the MBC's errors are centered around zero, while the errors for the PPC occupy a much larger band. As mentioned earlier, this is mainly due to the PPC's frequent steering, which not only makes the vehicle's steering

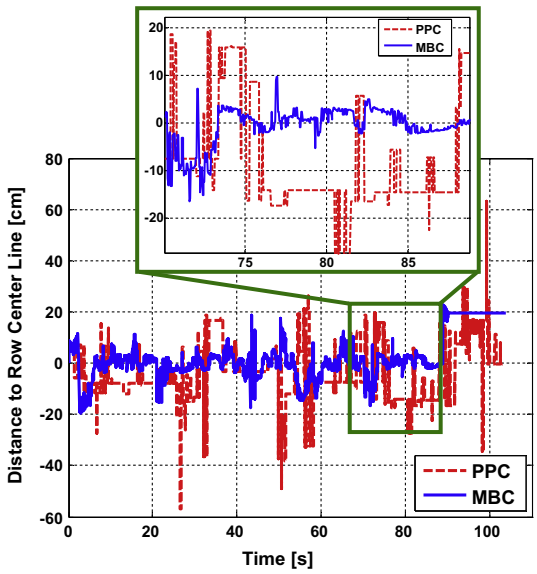

Fig. 16. A closer view of experimental data to show the localization and backlash contribution and the effects of two control strategies. (See Fig. 17 for the row localization of this experiment.)

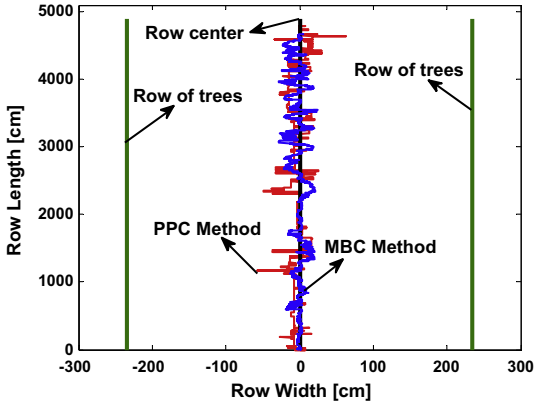

Fig. 17. A localization information plot prepared using the experimental data obtained in the seventh row of the field. Trajectory tracking (center row following) is achieved using two control strategies: (the PPC + particle filter) and (the MBC + localization method). (See Fig. 16 for the lateral error values.)

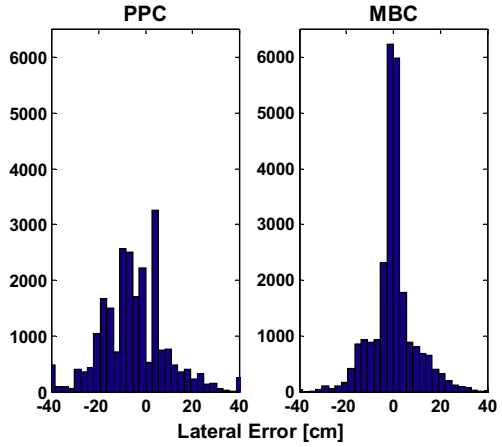

Fig. 18. Lateral error histogram for a seven-row trip (370 m). (The PPC + particle filter) errors occupy a large band around zero, while (the MBC + localization method) errors are mostly concentrated around zero.

wheel oscillates around the centerline, but also taxes the steering motor.

The methodology proposed in this study would work in most orchards, not only in our orchard-like nursery. The test site is relatively benign but not too different from modern, high-density apple orchards, where tree trunks are also somewhat visible. Therefore, we would expect our methods to work on those orchards as well. For orchards where tree trunks are less visible or not visible at all, an alternative localization method such as the one in (Zhang, 2014) could be used.

6. Analysis and conclusion

The model-based control method proposed in this paper builds on the literature on path tracking controllers for field vehicles by integrating wheel sideslip effects and path planning and by enabling operation with a low-cost, GPS-free sensing suite. It also improves significantly the vehicle's tracking performance compared to the original pure pursuit controller. In particular, the new method allows us to regulate the maximum steering rate, thus extending the life of the steering servomotor.

Despite the progress reported here, there is still work to be done to bring autonomous orchard vehicles closer to commercialization. Some that we are investigating include:

- Laser- and odometry-based localization to provide the vehicle a more accurate position estimate within the block. This will improve path tracking and enable precision agriculture applications at a very low cost compared to the current solutions that use differential GPS.
- Obstacle detection to ensure the vehicle stops for people, bins, farm equipment, etc., but not for tall grass or weeds.
- Safety and operation standards that promote a legal framework for farm equipment manufacturers when deploying autonomous vehicles in the field.

Acknowledgment

This work was partially supported by the U.S. Department of Agriculture under Grant No. 2008-51180-04876. The first author, Gokhan Bayar, would like to thank to the Scientific and Technological Research Council of Turkey and Middle East Technical University for their financial supports.

References

Ampatzidis, Y., Vougioukas, S., and Bochtis, D., 2006. A decomposition framework for the autonomous navigation of agricultural vehicles. In: International Conference on Information Systems in Sustainable Agriculture, Agroenvironment and Food Technology, Greece.

Bak, T., Jakobsen, H., 2004. Agricultural robotic platform with four wheel steering for weed detection. Biosyst. Eng. 87 (2), 125–136.

Barawid, O., Mizushima, A., Ishii, K., Noguchi, N., 2007. Development of an autonomous navigation system using a two-dimensional laser scanner in an orchard application. Biosyst. Eng. 96 (2), 139–149.

Davis, B., 2012. CMU-led automation program puts robots in the field. AUVSI's Unmanned Systems: Mission Critical 2, 38–40.

Dellaert, F., Fox, D., Burgard, W., Thrun, S., 1999. Monte Carlo localization for mobile robots. In: IEEE International Conference on Robotics and Automation, Detroit, MI, USA.

Eaton, R., Katupitiya, J., Siew, K.W., Howarth, B., 2008. Autonomous farming: modeling and control of agricultural machinery in a unified framework. In: 15th International Conference on Mechatronics and Machine Vision in Practice, Auckland, New Zealand.

Eaton, R., Katupitiya, J., Pota, H., Siew, K.W., 2009. Robust sliding mode control of an agricultural tractor under the influence of slip. In: IEEE/ASME International Conference on Advanced Intelligent Mechatronics, Singapore.

Fang, H., 2004. Automatic guidance of farm vehicles in presence of sliding effects, A Scientific Report for the Post-Doc Research Project of All Terrain Autonomous Vehicle Control.

Fang, H., Lenain, R., Thuilot, B., Martinet, P., 2005. Robust adaptive control of automatic guidance of farm vehicles in the presence of sliding. In: IEEE International Conference on Robotics and Automation, Barcelona, Spain.

Fang, H., Fan, R., Thuilot, B., Martinetb, P., 2006. Trajectory tracking control of farm vehicles in presence of sliding. Robot. Auton. Syst. 54 (10), 828–839.

Freitas, G., Hamner, B., Bergerman, M., Singh, S., 2012a. A practical obstacle detection system for autonomous orchard vehicles. In: IEEE/RSJ International Conference on Intelligent Robots and Systems, Algarve, Portugal.

Freitas, G., Zhang, J., Hamner, B., Bergerman, M., Kantor, G., 2012b. A Low-Cost, Practical Localization System for Agricultural Vehicles. In: International Conference on Intelligent Robotics and Applications, Montreal, Canada.

Gonzalez, R., Rodriguez, F., Guzman, J., Berenguel, M., 2009. Localization and control of tracked mobile robots under slip conditions. In: IEEE International Conference on Mechatronics, Malaga, Spain.

Gustafsson, F., Gunnarsson, F., Bergman, N., Forssell, U., Jansson, J., Karlsson, R., Nordlund, P.J., 2002. Particle filters for positioning, navigation, and tracking. IEEE Trans. Signal Process. 50 (2), 425–437.

Hamner, B., Koterba, S., Shi, J., Simmons, R., Singh, S., 2009. Mobile robotic dynamic tracking for assembly tasks. In: IEEE/RSJ International Conference on Intelligent Robots and Systems, St. Louis, MO, USA.

Hamner, B., Bergerman, M., Singh, S., 2011. Autonomous Orchard Vehicles for Specialty Crops Production. ASABE Annual International Meeting, Louisville, KY, USA.

Hamner, B., Bergerman, M., Singh, S., 2012. Results with Autonomous Vehicles Operating in Specialty Crops. In: IEEE International Conference on Robotics and Automation, St. Paul, MN.

Huynh, V.T., Katupitiya, J., Kwok, N.M., Eaton, R.P., 2010. Derivation of an error model for tractor-trailer path tracking. In: International Conference on Intelligent Systems and Knowledge Engineering (ISKE), Hangzhou, China.

Johnson, D.A., Naffin, D.J., Puhalla, J.S., Sanchez, J., Wellington, C.K., 2009. Development and implementation of a team of robotic tractors for autonomous peat moss harvesting. J. Field Robot. 26 (6–7), 549–571.

Kanayama, Y., Hartman, B.I., 1989. Smooth Local Path Planning for Autonomous Vehicles, IEEE International Conference on Robotics and Automation, Scottsdale, AZ, USA.

Lenain, R., Lucet, E., Grand, C., Thuilot, B., Amar, F.B., 2010. Accurate and stable mobile robot path tracking: An integrated solution for off-road and high speed context. In: IEEE/RSJ International Conference on Intelligent Robots and Systems, Taipei, Taiwan.

Matveev, A.S., Hoy, M., Savkin, A.V., 2010. Mixed nonlinear sliding mode control of an unmanned farm tractor in the presence of sliding. In: 11th International Conference on Control, Automation, Robotics and Vision, Singapore.

Shin, D.H., Singh, S., 1990. Path Generation for Robot Vehicles Using Composite Clothoid Segments, Tech. Report CMU-RI-TR-90-31, Robotics Institute, Carnegie Mellon University.

Singh, S., Bergerman, M., Cannons, J., Grocholsky, B., Hamner, B., Holguin, G., Hull, L., Jones, V., Kantor, G., Koselka, H., Li, G., Owen, J., Park, J., Shi, W., Teza, J., 2010. Comprehensive automation for specialty crops: Year 1 results and lessons learned. J. Intell. Serv. Robot. Agr. Robot. 3 (4), 245–262.

Weiss, U., Biber, P., 2011. Plant detection and mapping for agricultural robots using a 3D LIDAR sensor. IEEE Trans. Robot. Automat. 59, 265–273.

Zhang, J., Maeta, S., Bergerman, M., Singh, S., 2014. Mapping orchards for autonomous navigation. In: ASABE and CSBE/SCGAB Annual International Meeting, Montreal, Quebec Canada.