DIFFERENTIAL FLATNESS−BASED FORMATION FOLLOWING OF A SIMULATED AUTONOMOUS SMALL GRAIN HARVESTING SYSTEM

Y. Hao, B. Laxton, E. R. Benson, S. K. Agrawal

ABSTRACT. *Researchers around the world have focused on autonomous agriculture with systems encompassing greenhouse, orchard, field, and other applications. Research has shown the potential and ability of the technology to allow a vehicle or selection of vehicles to follow a specified task. In this study, one aspect of the problem, that of operating a tractor−cart combination in conjunction with a small−grain combine harvester, was investigated. The tractor−cart combination and combine harvester application was selected because of the high fatigue and long duration aspects of the problem. Differential flatness−based formation following was tested in software and robotic simulation. The software simulation was based on an actual field track from a combine yield monitor and demonstrated the potential of the system. The robotic simulation used two iRobot Magellan Pro robots in an indoor environment and demonstrated that the methodology could be implemented in real time.*

Keywords. *Automatic control, Combine harvesters, Computer simulation, Farm machinery, Farm management, Model validation, Modeling.*

Over the last 100 years, agriculture has made significant advances. Revolutions in mechanization, infotronics, and genomics have increased yields. The combine harvester is a critical element in the harvesting of important field crops in North America. Combine size and capacity have increased to meet yield and management changes.

Research groups in the U.S. and internationally have begun to develop robotic systems for agriculture (Reid, 2000; Reid et al., 2000; Noguchi et al., 1997). The majority of the projects have dealt with the development of robotic control systems and navigation systems for individual agricultural vehicles, primarily tractors. Tractors are used for a wide variety of tasks, including tillage, planting, cultivation, and harvest support. Researchers have begun to investigate harvester guidance systems; however, the state of the art has not reached that of tractor guidance systems (Callahan et al., 1997; Fitzpatrick et al., 1997; Benson et al., 2001).

Article was submitted for review in July 2003; approved for publication by the Information & Electrical Technologies Division of ASAE in April 2004. Presented at the 2003 ASAE Annual Meeting as Paper No. 033109.

iRobot is a trademark of iRobot Corporation. Mention of trade name, proprietary product, or specific equipment does not constitute a guarantee or warranty by the University of Delaware and does not imply the approval of the named product to the exclusion of other products that may be suitable.

The authors are **Yongxing Hao,** Graduate Research Assistant, Department of Mechanical Engineering, **Benjamin Laxton,** Undergraduate Research Assistant, Department of Computer and Information Sciences, **Eric R. Benson, ASAE Member Engineer,** Assistant Professor, Department of Bioresources Engineering, and **Sunil K. Agrawal,** Professor, Department of Mechanical Engineering, University of Delaware, Newark Delaware. **Corresponding author:** Eric R. Benson, Department of Bioresources Engineering, University of Delaware, Newark DE 19716; phone: 302−831−0256; e−mail: ebenson@udel.edu.

In the field, harvesters operate in conjunction with one or more grain carts. The grain cart travels from one or more harvesters in the field to the road transport or grain storage areas. Portions of the cart movement are done independently, away from other vehicles. When transferring harvested grain, the cart must synchronize its movement with the harvester. The dimensions of the combine and cart require precision operation of both vehicles. The continuous operation and precision required for transfer are fatiguing.

The cart and harvester interactions are governed primarily by the harvester. During harvest, the primary objective is to harvest the maximum quantity at the highest quality with a minimum of inputs (fuel, time, labor, etc.). To achieve the maximum quantity in the minimum time, an overriding objective is to keep the harvesters operating at maximum effectiveness during the entire process. The grain cart has to sequence its transfer and movement operations to prevent any harvesters in the field from reaching capacity (forcing a stoppage) before the cart can arrive. The cart has to select the appropriate harvester, based on distance and time to fill, locate the harvester using a combination of local and/or global sensors, travel to the harvester, and travel in formation with the harvester as grain is transferred. After completion of the transfer, the cart is free to travel to other harvesters or return to an in−field storage station (typically a tractor trailer).

Current research has not concentrated on the cart and harvester interactions. Portions of the grain cart, tractor, and harvester interactions can be developed from formation control of mobile robots (Hao et al., 2003; Guo and Parker, 2002; Fredslund and Matarsc, 2001; Balch and Arkin, 1998). Algorithms for control and coordination of the harvester and cart must account for the dynamic nature of the environment in which they operate (Pledgie et al., 2002; Desai et al., 1998). The control of a single robot with a trailer has been

  ISSN 0001−2351

investigated extensively (Lamiraux and Laumond, 1998; Sekhavat et al., 1997). Combining formation planning and control of mobile robots with trailers is a challenging problem.

OBJECTIVE

The objective of our research was to develop a robotic simulation of agricultural vehicle formation and docking strategies. Within that overall objective, two sub−objectives were identified:

- Use differential flatness to plan the trajectory of a combine and tractor with trailer formation.
- Simulate the combine and tractor with trailer formation using both computational and robotic methods.

RESEARCH DESCRIPTION AND FRAMEWORK

The harvester and tractor combination can be thought of as a formation of autonomous robots that need to maintain a specified geometric relationship. Thus, we propose a practical framework for the on−line planning and control of multiple mobile robots with trailers moving in groups. The group is trailer−centered, must maintain some predetermined geometric shape while moving, and is allowed to change formation as necessary to negotiate through the environment. The combine path dictates the path of the trailer to ensure a collision−free path that will allow grain transfer. The path of the trailer, in turn, dictates the required path of the tractor to ensure that the trailer is in the optimal location at the correct time.

Path planning based on graph search is typically discussed in artificial intelligence literature, while trajectory optimization is addressed in control literature. In most cases, path planning and trajectory optimization are considered separate research problems. The graph search method is suitable for global path planning and can be implemented on−line. It will generate a discrete path sequence and can be changed to a smooth trajectory without considering system dynamics. The optimization method gives optimal trajectory but is hard to implement in real time when dimensions and constraints are high. For this reason, optimization is normally performed off−line. Differential flatness can be used to simplify the solution of dynamic optimization problems. Differential systems that exhibit the properties of differential flatness were first studied by Fliess et al. (1995) and later summarized by Sira−Ramírez and Agrawal (2004). The use of flatness−based planning of groups of autonomous vehicles was reported in scientific literature (Fossas et al., 2000; Ferreira and Agrawal, 1999; Pledgie et al., 2002). Differentially flat systems are well suited to problems requiring trajectory generation. Since the outputs of a flat system completely describe its behavior, the trajectory can be planned in output space, and the inputs that will cause the system to follow this trajectory can be calculated directly. The idea of differential flatness will be used to plan and optimize local trajectories for mobile robots with trailers. Figure 1 shows four−wheel and two−wheel mobile robots pulling hitch−mounted and kingpin−mounted trailers. Let us denote the coordinates of the robot and the trailers by (x, y, θ), (x_1, y_1, θ_1), ..., (x_n, y_n, θ_n), respectively. In each case, the system is differentially flat with the last trailer's x_n and y_n as the linearizing outputs.

PLANNING AND CONTROL

A flowchart for formation planning and control is shown in figure 2. In this project, the combine operator was assumed

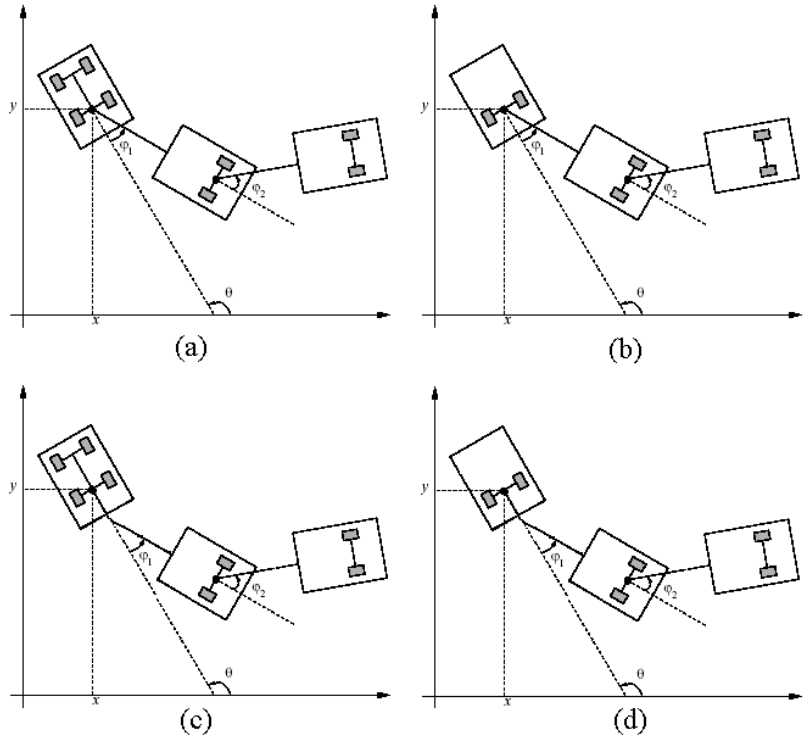

Figure 1. Four types of mobile robots with trailers: (a) four−wheel vehicle with a kingpin−mounted trailer, (b) two−wheel vehicle with a kingpin−mounted trailer, (c) four−wheel vehicle with a hitch−mounted trailer, and (d) two−wheel vehicle with a hitch−mounted trailer.

to provide the path, either from a GPS−based planting map or from on−board sensors. The combine path was reduced to a series of waypoints, and the trajectory generator produced a continuous time trajectory for it. Then, according to the trailers' positions in the formation with respect to the combine, the reference trajectories for trailers and real robots are generated. If there is a possible collision in the computed trajectories, the trailers' and robots' trajectories will be optimized given the reachable area for the trailers if a solution exists. Next, each robot tracks its own trajectory. Based on trailer stability analysis (Lamiraux and Laumond, 1998), if the robot tracks its reference trajectory well, the trailers attached to the robot will also converge to their own reference trajectories.

Robots obtain position feedback through onboard odometry readings. If the environment does not change and there are no collisions between robots after trajectory optimization, then the robots track their computed trajectories. Typical changes could include detection of an obstacle, soft ground, or other restrictions to movement. Operation in a changing environment is described by Hao et al. (2003).

The last trailer of each robot chain has a predetermined geometric relationship with respect to the leader. The relationship defines the reference trajectory for the last trailer based on the reference trajectory of the leader.

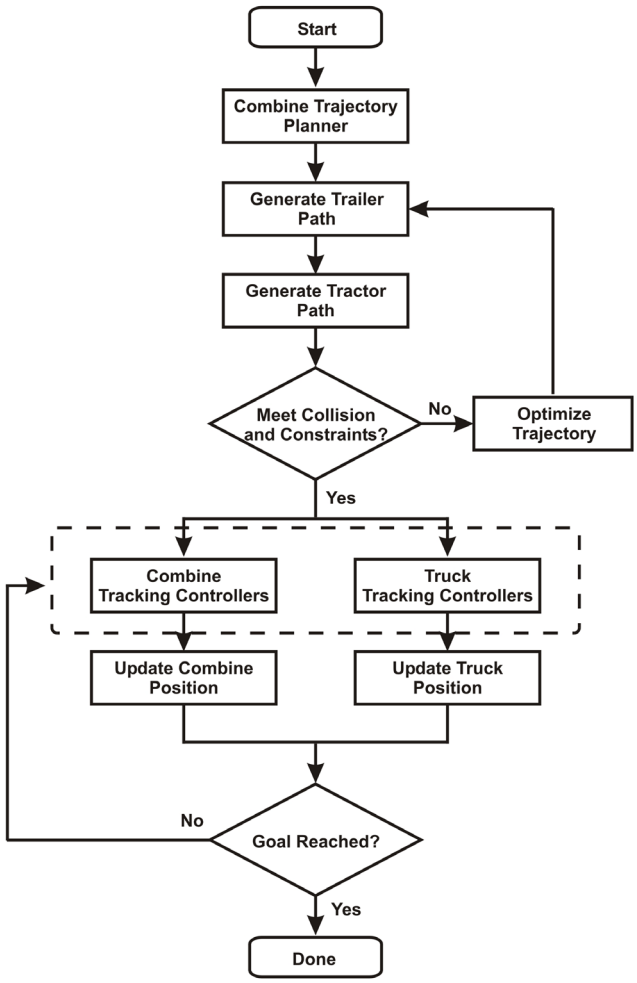

Figure 2. Flowchart for formation planning and control.

Trailer and follower trajectory generation uses the flatness property of the system. For the formation group, the flat output is the combine's trajectory (x_c, y_c). Given the leader's trajectory, the last trailer's nominal trajectory will be determined geometrically. Other trailers and the follower robot's trajectories will be determined recursively. Although collision−free trajectories for the combine and trailer can be dictated by the choice of offset, the possibility exists of collisions between the tractor and combine. These inter−formation collisions can be accounted for if there is some flexibility in the trailer's trajectories.

In general, the problem scenario consists of multiple vehicles moving as a group. Group G consists of N similar units, and the dynamics of the ith unit is given by:

$$\dot{\bar{x}}_i = f(\bar{x}_i, u_i),\ i = 1, ..., N \tag{1}$$

Here, $\bar{x}_i \in \Re^n$ denote the states, $u_i \in \Re^m$ are the inputs, and $f(\cdot)$ is a smooth mapping from its arguments. Trajectory planning for such a group consists of finding trajectories, which over a time horizon $[t_0, t_f]$ satisfy the dynamic equation (eq. 1) given the inequality constraints shown in equations 2 and 3 and minimizing a cost criterion shown in equation 4:

$$\bar{g}(x_1,..., x_N, t) \le 0, \bar{g} \in \Re^{n_g} \tag{2}$$

$$\bar{c}(\bar{x}_1,..., \bar{x}_N, u_1,..., u_N, t) \le 0, \bar{c} \in \Re^{n_c} \tag{3}$$

$$\min \bar{J} = \bar{\Phi}\left[\bar{x}_1(t_f)..., \bar{x}_N(t_f) t_f\right] + \int_{t_o}^{t_f} \bar{L}(\bar{x}_1,..., \bar{x}_N, u_1,..., u_N, t)dt \tag{4}$$

The inequality constraints involving configuration variables of the units in equation 2 have a well−defined structure that comes from the organization of the group and the geometry of the formation. For this reason, they are distinguished from other inequality constraints on states and/or inputs in equation 3 and are considered configuration constraints. For example, during grain transfer, the combine and trailer need to stay within a minimum (collision) and maximum (failure to transfer) distance. Such constraints fall within the category of equation 2.

This trajectory optimization problem involves finding $N(n + m)$ state and input trajectories in the presence of $n_g + n_c$ inequality constraints, while satisfying Nn state equations and given terminal constraints. The solution of such an optimization problem is known to be computationally demanding. In order to make this problem computationally more tractable and potentially solvable in close to real−time, the problem can be posed as multiple suboptimal problems that give some fixed forms and discretize the original continuous problem. If the resolution is small enough, the solution will be acceptable. Specifically, consider one tractor with a trailer, shown in figure 1a, that follows a combine time interval $[t_0, t_f]$. Let x_c and y_c denote the combine's x and y trajectories in global coordinates, respectively. The heading angle (θ_c) can be calculated as follows:

$$\theta_c = \tan^{-1}\left(\frac{\dot{y}_c}{\dot{x}_c}\right) \tag{5}$$

Next, the trailer's nominal trajectories (x_{tn}, y_{tn}) can be obtained by:

$$x_{tn} = x_c + dx \times \cos(\theta_c) - dy \times \sin(\theta_c) \quad (6)$$

$$y_{tn} = y_c + dx \times \sin(\theta_c) + dy \times \cos(\theta_c) \quad (7)$$

where *dx* and *dy* denote the trailer local position relative to the leader robot in the formation. Given flexibility in the trailer's motion, we can specify a circle around its nominal trajectory. A maximum permitted deviation distance (*R*) can be defined, in this case based on the dimensions of the grain cart. A suitable choice for the trajectory is polynomial such as $\Delta x = a_0 + a_1 t + a_2 t^2 + a_3 t^3 + a_4 t^4$ ... for the *x* trajectory and $\Delta y = b_0 + b_1 t + b_2 t^2 + b_3 t^3 + b_4 t^4$ ... for the *y* trajectory. Any number of polynomials can serve as a valid trajectory; however, the relative computation cost and performance need to be compared. The coefficients a_0, a_1, a_2, ... and b_0, b_1, b_2, ... are parameters, which in this project were calculated using the CFSQP software package. Thus, the trailer's trajectory (x_t, y_t) is given as:

$$x_t = x_{tn} + \Delta x \times \cos(\theta_c) - \Delta y \times \sin(\theta_c) \quad (8)$$

$$y_t = y_{tn} + \Delta x \times \sin(\theta_c) + \Delta y \times \cos(\theta_c) \quad (9)$$

The trailer's angle (θ_t) can be calculated as well:

$$\theta_t = \tan^{-1}\left(\frac{\dot{y}_t}{\dot{x}_t}\right) \quad (10)$$

The tractor pulls the trailer through the required trajectory. The tractor's trajectory (x_p, y_p) can be computed by:

$$x_p = x_t + l \times \cos(\theta_t) \quad (11)$$

$$y_p = y_t + l \times \sin(\theta_t) \quad (12)$$

where *l* is the distance between the midpoint of the tractor rear wheels and trailer wheels.

The cost function is aimed to minimize the trailer's deviation from the nominal trajectory while ensuring that the no−collision requirement is met. In order to optimize in real−time, a finite discrete set (*S*) must be used. For example, a uniform discretization is $t_o + i\frac{t_f - t_0}{n}$, where $i = 0 \ldots n$. Each vehicle in the formation was reduced to a series of circular constraint points (fig. 3a). For the tractor and combine, multiple constraint points were used to dictate the desired behavior and avoid collisions. Each constraint point included *x* and *y* coordinates and a radius of conformity. For collision avoidance, the radius was specified as a distance that other vehicles could not operate within. For grain transfer, the radius was specified as a distance that the combine discharge auger had to remain within. The radius constraints were dictated by vehicle geometry. For example, the combine was reduced to circular constraint points located at body center and at each end of the head. The circular constraint point that was centered on the combine body encircled the entire combine body. Circular constraint points were located at the outer edges of the head with a diameter equal to the width of the head. To avoid collision, the combine constraint points could not be located inside any other vehicle's constraint points. On satisfying the constraints at the discrete points, the optimization problem becomes:

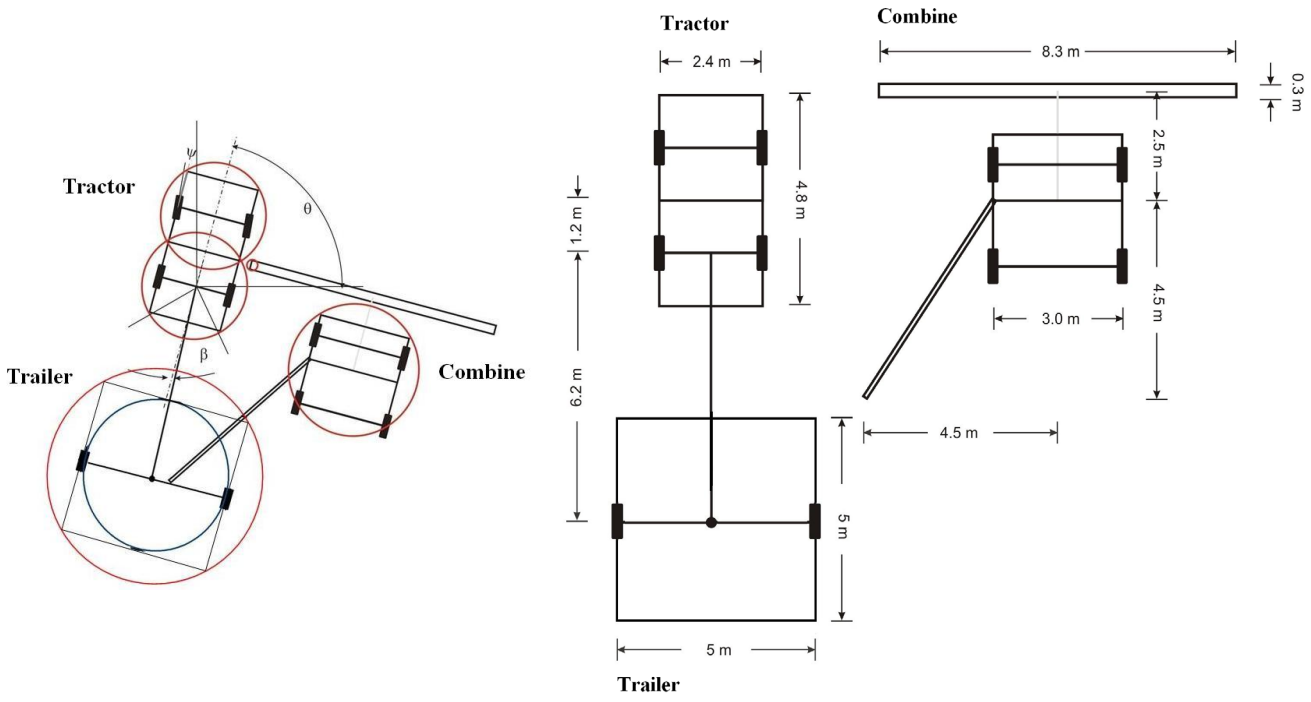

Figure 3. (a) Vehicle constraint points and (b) simulation dimensions.

$$\min J = \max_{i=1\ldots n}\left[\left(a_0 + a_1\tau + a_2\tau^2 + a_3\tau^3 + a_4\tau^4 + \ldots\right)^2 + \left(b_0 + b_1\tau + b_2\tau^2 + b_3\tau^3 + b_4\tau^4 + \ldots\right)^2\right] \quad (13)$$

and

$$|\psi| < \frac{\pi}{3}$$

$$|\beta| < \frac{\pi}{3}$$

$$(a_0 + a_1\tau + \ldots)^2 + (b_0 + b_1\tau + \ldots)^2 < R^2$$

$$(x_{pi} - x_{cj})^2 + (y_{pi} - y_{cj})^2 > d_{ij}^2$$

$$\forall \tau \in S, i \in M, j \in N$$

where ψ is the tractor front wheel angle, β is the tractor−trailer angle, d_{ij} is the safe distance between circle *i* and circle *j*, *M* and *N* are the approximation circle sets for the tractor and combine, respectively, and subscripts *pi* and *cj* denote the tractor and combine, respectively.

Given the desired optimized trajectory, a tracking controller was developed to ensure that the tractor correctly followed the correct trajectory. According to the proof given by Lamiraux and Laumond (1998), if the reference angle between the trailer and the tractor is inside [−π/2, π/2] and the tractor tracks its reference trajectory well during forward motion, then its trailer will also converge to its own reference trajectory. The lead vehicle (combine) dictates the path of the trailer; the path of the trailer in turn dictates the required path of the tractor to achieve the desired trailer trajectory. The computational process reverses the standard driving model in which the tractor dictates the trailer path.

The equations of motion for a rear wheel drive vehicle are governed as shown by equations 14 to 17:

$$\dot{x}_i = u_{1i}\cos(\theta_i) \quad (14)$$

$$\dot{y}_i = u_{1i}\sin(\theta_i) \quad (15)$$

$$\dot{\theta}_i = \frac{u_{1i}}{l}\tan(\psi_i) \quad (16)$$

$$\dot{\psi}_i = u_{2i} \quad (17)$$

where (x_i, y_i) is the Cartesian location of the center of its rear wheels, θ_i is the heading angle between the body axis and the horizontal axis, and ψ_i represents the steering angle with respect to the vehicle body. The distance between the location (x_i, y_i) and the midpoint of the steering wheels is denoted by l_i, u_{1i} corresponds to the translational velocity of the rear wheels of the vehicle, and u_{2i} corresponds to the velocity of the angle of the steering wheels.

The kinematic model presented in equations 14 to 17 of a rear wheel drive vehicle is differentially flat with the flat outputs given by (x_i, y_i), i.e., all system variables can be differentially parameterized solely in terms of x_i and y_i and a finite number of their time derivatives:

$$x_i = x_i \quad (18)$$

$$y_i = y_i \quad (19)$$

$$\theta_i = \tan^{-1}\left(\frac{\dot{y}_i}{\dot{x}_i}\right) \quad (20)$$

$$\psi_i = \tan^{-1}\left(\frac{\dot{x}_i\ddot{y}_i - \ddot{x}_i\dot{y}_i}{\dot{x}_i^2 + \dot{y}_i^2}\right)Sl_i \quad (21)$$

$$u_{1i} = S\sqrt{\dot{x}_i^2 + \dot{y}_i^2} \quad (22)$$

$$u_{2i} = \frac{\left[-3(\dot{x}_i\ddot{y}_i - \ddot{x}_i\dot{y}_i)(\dot{x}_i\dddot{y}_i + \ddot{x}_i\ddot{y}_i) + (\dot{x}_i^2 + \dot{y}_i^2)(\dot{x}_i\dddot{y}_i - \dddot{x}_i\dot{y}_i)\right]Sl_i\sqrt{\dot{x}_i^2 + \dot{y}_i^2}}{(\dot{x}_i^2 + \dot{y}_i^2)^3 + (\dot{x}_i\ddot{y}_i - \ddot{x}_i\dot{y}_i)^2 l_i^2} \quad (23)$$

where

$$S = \begin{cases} 1 & \textit{forward} \\ -1 & \textit{backward} \end{cases} \quad (24)$$

The condition shown in equation 25 should hold for the planned trajectory x_i and y_i, and then the control as well as state parameterizations are invertible:

$$\sqrt{\dot{x}_i^2 + \dot{y}_i^2} \neq 0 \quad (25)$$

In the combine−tractor−trailer system, the flat output is the combine's reference trajectory (x_c, y_c).

In the experiment, the mobile robot shown in figure 1b has two coaxial powered wheels and a passive supporting caster wheel. The robots used in the experimental validation of the formation−following algorithm are differential drive and are governed as shown by equations 26 to 28. Both four−wheel and two−wheel vehicles with trailers are differentially flat. The required inputs to drive the pulling vehicle in a trailer set are calculated from the required states of the last trailer in the set for a differentially flat system:

$$\dot{x}_i = u_{1i}\cos(\theta_i) \quad (26)$$

$$\dot{y}_i = u_{1i}\sin(\theta_i) \quad (27)$$

$$\dot{\theta}_i = u_{2i} \quad (28)$$

where (x_i, y_i) denotes the position of the center of the axle with respect to the inertial frame, and θ_i denotes the orientation of the vehicle in the inertial frame. The inputs to the controller are u_{1i} and u_{2i}, which are the translational velocity and the rotation velocity of the robot, respectively. The actual front wheel angle (ψ) shown in figure 3a will be software−constrained, as shown in equation 29:

$$|\psi| = \left|\tan^{-1}\left(\frac{u_{2i}l}{u_{li}}\right)\right| = \left|\tan^{-1}\left(Sl_i\frac{\dot{x}_i\ddot{y}_i - \ddot{x}_i\dot{y}_i}{\left(\dot{x}_i^2 + \dot{y}_i^2\right)^{3/2}}\right)\right| < \frac{\pi}{3} \quad (29)$$

Corrective strategies are required to keep the vehicles on the trajectories. The tracking controller from Samson and Ait−Abderrahim (1991) was used here. If (x_r, y_r, θ_r) are the coordinates of the reference robot in the frame of the real robot, and if (u_{1i}^0, u_{2i}^0) are the inputs of the reference trajectory and (x_{ir}, y_{ir}) are the robot reference trajectory in global coordinates, then this control law has the following expressions:

$$u_{1i} = u_{1i}^0\cos(\theta_r) + k_1 x_r \quad (30)$$

$$u_{2i} = u_{2i}^0 + k_2\frac{\sin(\theta_r)}{\theta_r}y_r + k_3\theta_r \quad (31)$$

where k_1, k_2, and k_3 are positive numbers and

$$u_{1i}^0 = \sqrt{\dot{x}_{ir}^{\,2} + \dot{y}_{ir}^{\,2}} \quad (32)$$

$$u_{2i}^0 = \frac{d}{dt}\left[\tan^{-1}\left(\frac{\dot{y}_{ir}}{\dot{x}_{ir}}\right)\right] \quad (33)$$

Materials and Methods

The formation planning and control concept described in the preceding section was tested both in software and robotic simulation. The robotic simulation was performed using two differential−drive mobile robots (Magellan Pro, iRobot Corp., Burlington, Mass.) (fig. 4), one of which had a trailer attached to it. One robot was designated as the leader (combine), and the second robot was designated as a tractor. A scratch−built trailer was attached to the tractor robot, with the vertical axis of the hitch passing through the midpoint of the drive axle. A 10 kΩ potentiometer was installed on the kingpin of the trailer. A 12−bit data acquisition card (PCI−6024E, National Instruments, Austin, Texas) was used to process the signal from the potentiometer, providing the relative direction (β) of the trailer with respect to the direction of the tractor. Each robot has an on−board PC consisting of an EBX motherboard and a Pentium III processor. The robots operate under the Linux operating system and their software integrates sensor and communication data. The robots communicate through wireless Ethernet capable of transmitting data up to 3 Mb per second.

Translational and rotational velocity controllers are used to reposition each robot. MATLAB/C++/JAVA are used as the computational engine for decision making, control, and graphical display. A version of the CFSQP optimization program was used (Lawrence et al., 2002). In the experiment, the optimization computation time was 0.9 s (1.11 Hz). There were eight parameters to optimize, and the path was discretized into 100 points. The parameters are calculated first and if there are no changes in the environment, then the parameters are held constant. Control commands are sent out every 0.144 s (6.94 Hz).

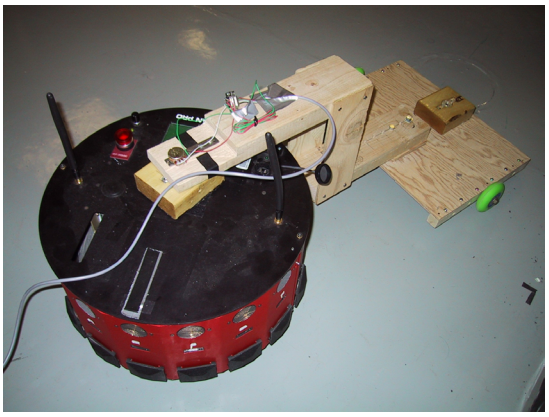

Figure 4. iRobot Magellan Pro robots were used to physically simulate the combine and tractor.

The purpose of the experiment was to show that these algorithms work in real−time for trajectory generation, optimization, and sensor updating in a dynamic environment. A block diagram of the computational procedure is shown in figure 5. Our experiment consisted of a leader robot (combine) and a follower robot−trailer combination (tractor and cart).

Results and Discussion

A sample agricultural setting was simulated both in software and using the experimental hardware. Simulation dimensions were established to mimic typical agricultural vehicles (fig. 3b). The vehicle dimensions were scaled to 10% for the simulations. Vehicle dimensions selected for the simulation modeled a John Deere 8120 MFD tractor pulling a J&M 1075 cart and a John Deere 9650 STS combine with a 30 ft grain platform. The dimensions used represent typical agricultural vehicles, not detailed models of any specific vehicle.

A yield monitor track from a similar combine operating in Kansas wheat under typical conditions was used to provide a pathway for the simulation (fig. 6a). Any yield monitor track could be used to provide a valid combine path for simulation. The 2−D simulation was performed on a subset of the pathway (fig. 6b). The subset selected represents one finishing pass of the combine, and grain transfer would typically occur after completion of the finishing pass. The path selected, however, includes greater turning motion than typical for grain transfer. A subset of the pathway was used to simplify simulation and ease evaluation; however, the entire pathway could also be simulated.

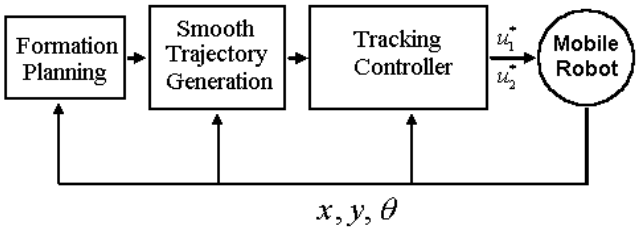

Figure 5. Formation planning and tracking control.

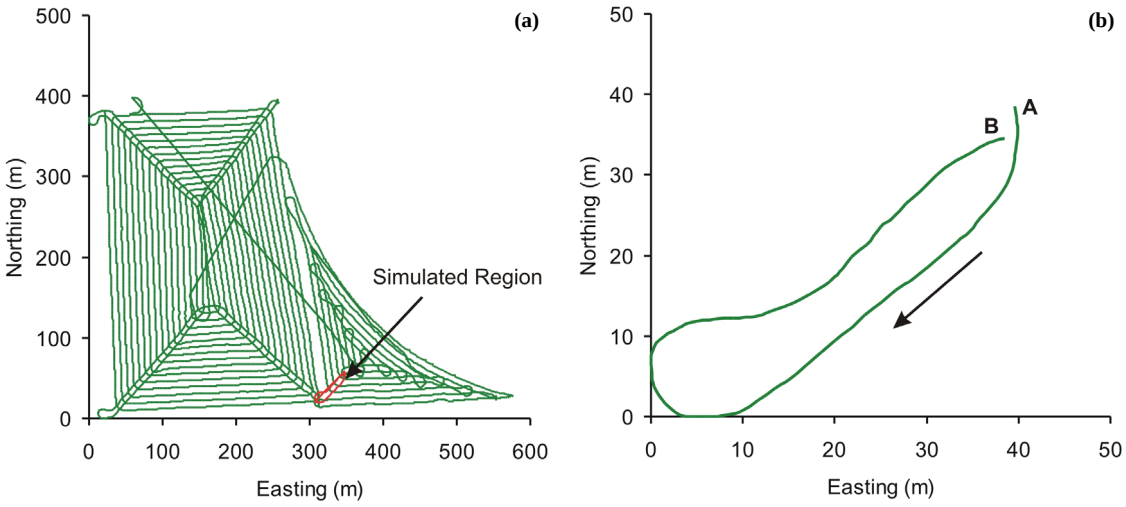

Figure 6. Combine and tractor/grain cart pathways were simulated on a portion of an actual combine pathway: (a) entire field pathway, as generated from the combine yield monitor, and (b) combine pathway for the simulated region.

The formation−following ability was modeled in C/C++, and the data were processed in Mathworks MATLAB. The simulation checked for collisions between all objects and angle constraints at each time step in the simulation. In the event of violation, the optimization routine developed a new, collision−free pathway. For the path and vehicle dimensions used in the simulation, the trailer's nominal trajectory would cause collisions between the tractor and the combine, as shown in figure 7a. After optimization, $\Delta x = 0$ and $\Delta y = 1$ ensured no collisions, as shown in figure 7b. The algorithm was able to develop an optimal path to ensure a collision−free trajectory, reasonable rotation angles, and that the combine discharge auger was within the cart for the entire pathway. Snapshots of the vehicle positions are shown in figures 7a and 7b.

The optimization considered multiple vehicle constraints. Vehicle constraints included: (1) ensuring no collisions, (2) maximum possible steering angles of $\pm\pi/3$, and (3) maximum trailer hitch angles of $\pm\pi/3$. As shown in figure 8a, the auger to trailer center distance was well within the 2.5 m radius specified by the trailer dimensions (5 × 5 m), indicating that the auger remained inside the trailer for the entire simulation. A time history of the trailer hitch angle is shown in figure 8b.

The simulation demonstrated that the formation−following algorithm was valid for typical combine paths. The kinematic software simulation did not, however, include vehicle dynamics in the model. Two−wheel and four−wheel vehicles with kingpin−mounted or hitch−mounted trailers (fig. 1) all exhibit the property of differential flatness. The specific derivations of the vehicle model vary based on hitch and vehicle type; however, all four cases remain differentially flat. The property of differential flatness allows the positions of the tow vehicle to be calculated from the position of the final trailer in the sequence. Because both the four−wheel and two−wheel case are differentially flat and exhibit parallel models, a robotic simulation using two−wheel vehicles is a valid method of testing the use of differential flatness in formation following.

Differential flatness is the key element that links the different trailer types. Whether hitch−mounted or kingpin−mounted, the entire system is differentially flat. This differential flatness property is used in the research presented to plan and control the motion of the vehicles. Changing

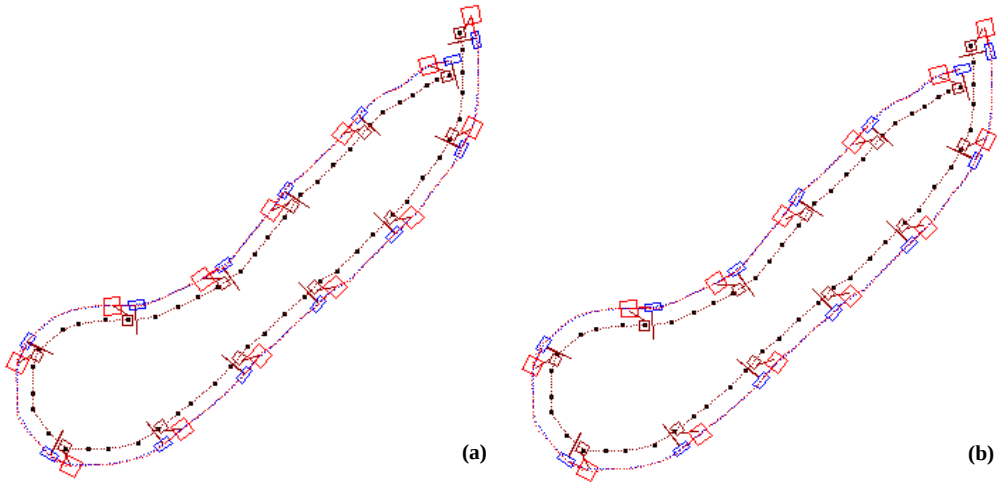

Figure 7. Snapshot sequences in the simulation, showing the positions of the combine, tractor, and trailer during the simulation: (a) pre−optimization and (b) post−optimization.

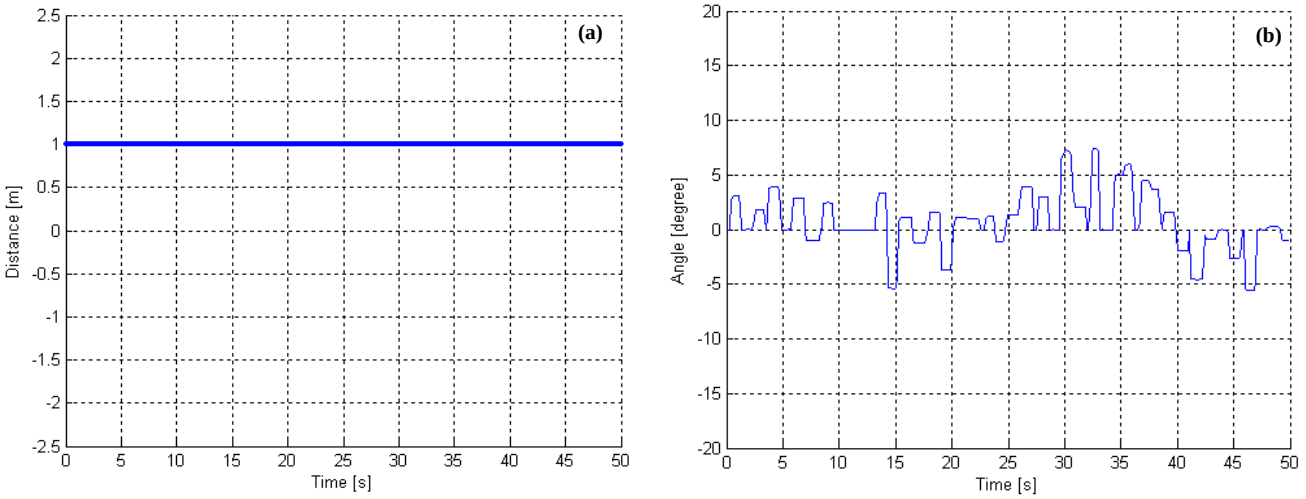

Figure 8. Vehicle constraint plots: (a) auger to trailer center distance and (b) trailer hitch angle.

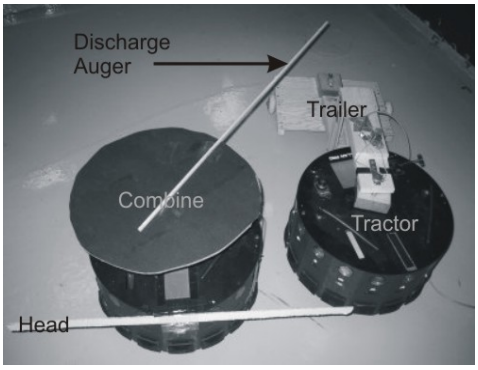

Figure 9. Sample image from the robotic simulation.

between hitch−mounted and kingpin−mounted trailers changes the specific derivation, but not the conclusions presented.

After simulating the algorithm in C/C++ and MATLAB, iRobot Magellan Pro robots were used to test the concept. The combine discharge auger, header, and trailer were added to the two Magellan Pro robots to better simulate the agricultural system. The dimensions of the physical simulation were approximately 10% of the real−world system. The combine path was arbitrary and sized to fit the available space within the lab. A sample image from the robotic simulation is shown in figure 9. Pre−optimization and post−optimization reference trajectories are shown in figure 10a. The optimized reference and actual robot trajectories are shown in figure 10b. Videos of the robotic simulation are available at: http://mechsys4.me.udel.edu.

The initial conditions at time $t = 0$ were chosen as $x_c = 0$, $y_c = 0$, $\theta_c = 1.06$, $x_t = -0.68$, $y_t = -0.19$, $\theta_t = 1.22$, $x_p = -0.48$, $y_p = 0.35$, $\theta_p = 1.12$. All positions are given in meters, and all angles are in radians. After optimization, the coefficients of the deviation distance (R) were generated: $a_0 = 0.000$, $a_1 = -3.075 \times 10^{-2}$, $a_2 = 2.777 \times 10^{-3}$, $a_3 = 8.329 \times 10^{-5}$, $a_4 = 7.460 \times 10^{-7}$, $b_0 = 0.000$, $b_1 = -8.401 \times 10^{-3}$, $b_2 = -6.164 \times 10^{-5}$, $b_3 = 1.427 \times 10^{-5}$, and $b_4 = -2.295 \times 10^{-7}$. The feedback gains determine the convergence and were experimentally determined as $k_1 = k_2 = k_3 = 0.5$. The mean error between the reference and actual trajectories is listed in table 1. As shown in figure 10b, the actual robot trajectories closely followed the reference trajectories.

From the results of this experiment, it is clear that the algorithms are feasible to implement real−time responsive behavior with currently available hardware.

Differences exist between the software simulation, robotic simulation, and physical implementation. The software simulation shows that differential flatness−based formation following is a valid approach for typical agricultural vehicles

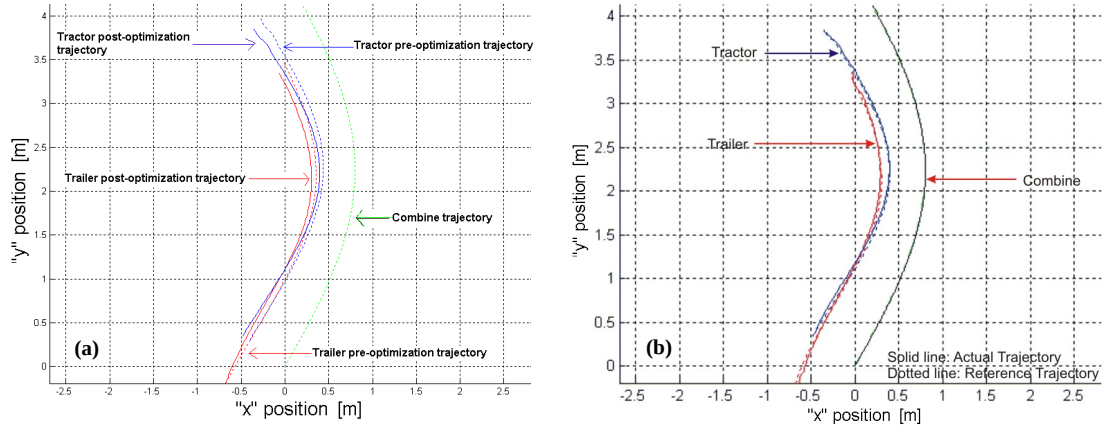

Figure 10. Combine, tractor, and trailer trajectories: (a) pre−optimization and post−optimization reference trajectories, and (b) reference and actual robot trajectories.

Table 1. Mean error in *x* and *y* coordinates.

Coordinate	Combine	Trailer	Tractor
x	0.24%	1.30%	1.65%
y	0.50%	0.75%	0.77%

and vehicle motions. The robotic simulation illustrates that differential flatness−based formation following can be implemented in real−time in a laboratory setting. Key differences exist between four−wheel agricultural vehicles and two−wheel laboratory robots. Agricultural vehicles operate in an outdoor, changing environment, while the simulation was demonstrated in a controlled laboratory situation. Simulations, however, can be performed at a fraction of the cost, during any season, and with little chance of injury to the participants. Simulations have been documented as acceptable means of algorithm development for most fields. A key goal for future research will be to implement the system developed with outdoor vehicles under more typical agricultural conditions.

CONCLUSION

Researchers around the world have developed individual robotic vehicles for agriculture. The interface or coordination of fleets of agricultural robotic vehicles has not been extensively investigated. In this project, the movement of a combine and tractor−cart combination was modeled. The tractor−cart combination and combine harvester application was selected because of the high fatigue and long duration aspects of the problem. Two−wheel and four−wheel vehicles with hitch−mounted or kingpin−mounted trailers are differentially flat, which allows the required inputs for the pulling vehicle to be calculated from the desired positions of the last trailer in the sequence. An optimization procedure was developed to create an appropriate path to satisfy performance objectives and constraints. The algorithm was validated through computer simulations and iRobot Magellan Pro robots. In both cases, an agricultural combine served as the lead vehicle, while a tractor pulling a grain cart maintained an optimal distance for grain transfer. The software simulation was based on the yield monitor track from an actual field and demonstrated the validity of using differential flatness−based formation planning for agricultural vehicles. The robotic simulation demonstrated that formation following could be performed in real−time with an arbitrary path.

ACKNOWLEDGEMENTS

We acknowledge the research support of the University of Delaware College of Agriculture and Natural Resources, the National Science Foundation (Award No. IIS−9912447), and the National Institute of Standards and Technology (Award No. 60NANB2D0137). We acknowledge the use of the CFSQP program provided by the University of Maryland.

REFERENCES

Balch, T., and R. C. Arkin. 1998. Behavior−based formation control for multi−robot teams. *IEEE Trans. Robotics and Automation* 14(6): 926−939.

Benson, E. R., J. F. Reid, and Q. Zhang. 2001. Machine vision−based steering system for agricultural combines. ASAE Paper No. 011159. St. Joseph, Mich.: ASAE.

Callahan, V., P. Chernett, M. Colley, T. Lawson, J. Standeven, M. Carr−West, and M. Ragget. 1997. Automating agricultural vehicles. *Industrial Robot.* 24(5): 364−369.

Desai, J. P., J. Ostrowski, and V. Kumar. 1998. Controlling formations of multiple mobile robots. In *Proc.* 1998 IEEE International Conference on Robotics and Automation, 4: 2864−2869. Piscataway, N.J.: IEEE.

Ferreira, A. M., and S. K. Agrawal. 1999. Planning and optimization of dynamic systems via decomposition and partial feedback linearization. In Proc. 1999 Conference on Decision and Control, 1: 740−745. Piscataway, N.J.: IEEE.

Fitzpatrick, K., D. Pahnos, and W. V. Pype. 1997. Robot windrower is first unmanned harvester. *Industrial Robot.* 24(5): 342−348.

Fliess, M., J. Levine, P. Martin, and P. Rouchon. 1995. Flatness and defect of non−linear systems: Introductory theory and examples. *International J. Control.* 61(6): 1327−1361.

Fossas, E., J. Franch, and S. K. Agrawal. 2000. Linearization by prolongations of two−input driftless systems. In Proc. 39th IEEE Conference on Decision and Control, 4 3381−3385. Piscataway, N.J.: IEEE.

Fredslund, J., and M. J. Matarsc. 2001. Robot formations using only local sensing and control. In *Proc. IEEE International Symposium on Computational Intelligence in Robotics and Automation*, 308−313. Piscataway, N.J.: IEEE.

Guo, Y., and L. E. Parker. 2002. A distributed and optimal motion planning approach for multiple mobile robots. In *Proc. IEEE International Conference on Robotics and Automation*, 3: 2612−2619. Washington, D.C.

Hao, Y., B. Laxton, S. K. Agrawal, E. Lee, E. Benson. 2003. Planning and control of UGV formations in a dynamic environment: A practical framework with experiments. In *Proc. IEEE International Conference on Robotics and Automation*, 1209−1214. Piscataway, N.J.: IEEE.

Lamiraux, F. and J.P. Laumond. 1998. A practical approach to feedback control for a mobile robot with trailer. In *Proc. IEEE International Conference on Robotics and Automation*, 4: 3291−3296. Piscataway, N.J.: IEEE.

Lawrence, C., J. L. Zhou, and A. Tits. 2002. User's Guide for CFSQP Version 2.5: A C code for solving (large scale) constrained nonlinear (minimax) optimization problems, generating iterates satisfying all inequality constraints. College Park, Md.: University of Maryland.

Noguchi, N., K. Ishii, and H. Terrao. 1997. Development of an agricultural mobile robot using a geomagnetic direction sensor and image sensors. *J. Agric. Eng. Res.* 67(1): 1−15.

Pledgie, S. T., Y. Hao, A. M. Ferreira, S. K. Agrawal, and R. Murphey. 2002. Groups of unmanned vehicles: Differential flatness, trajectory planning, and tracking control. In *Proc. IEEE International Conference on Robotics and Automation*, 4: 3461−3466. Piscataway, N.J.: IEEE.

Reid, J. F. 2000. Establishing automated vehicle navigation as a reality for production agriculture. In *Proc. 2nd IFAC/CIGR International Workshop on Bio−Robotics, Information Technology, and Intelligent Control for Bioproduction Systems*, 31−38. Amsterdam, The Netherlands: Elsevier.

Reid, J. F., Q. Zhang, N. Noguchi, and M. Dickson. 2000. Agricultural automatic guidance in North America. *Computers and Electronics in Agric.* 25(1/2): 154−168.

Samson, C., and K. Ait−Abderrahim. 1991. Feedback control of a nonholonomic wheeled cart in Cartesian space. In *Proc. IEEE International Conference on Robotics and Automation*, 3: 1136−1141. Piscataway, N.J.: IEEE.

Sekhavat, S., F. Lamiraux, J. P. Laumond, G. Bauzil, and A. Ferrand. 1997. Motion planning and control for Hilare pulling a trailer: Experimental issues. In *Proc. IEEE International Conference on Robotics and Automation*, 4: 3461−3466. Piscataway, N.J.: IEEE.

Sira−Ramírez, H., and S. K. Agrawal. 2004. *Differential Flatness*. New York, N.Y.: Marcel Dekker.