A TURNING SCHEME IN THE HEADLAND OF AGRICULTURAL FIELDS FOR AUTONOMOUS ROBOT

N. M. Thamrin, N. H. M. Arshad, R. Adnan, R. Sam, N. A. Razak, M. F. Misnan and S. F. Mahmud
Faculty of Electrical Engineering, Universiti Teknologi MARA, Shah Alam, Selangor, Malaysia

ABSTRACT
In agriculture field, headland turning is one of the important aspects in controlling the unmanned vehicle to move from one row to another autonomously. Therefore, this paper presents the headland turning based on the pre-defined path using Bezier curve with optimum path and minimal navigation and orientation control schemes. Real-time experiments have been carried out in the laboratory environment with an experimental unmanned mobile robot. The deviation between the planned path and the real-time trajectory is recorded to analyze the efficiency of the proposed work. It is found that the average error is larger compared to other approach due to direct connection of the Bezier curve. Nevertheless, it is suitable to be implemented on a small-scaled unmanned vehicle in a narrow headland with a simple and optimum turning scheme.

Keywords: headland turning, bezier curve, optimum headland turning.

INTRODUCTION
In the agricultural field, the plant is coordinated a well-organized and multiple rows to facilitate the movement of the farmers and machines. At the same time, it leaves numbers of headlands at the end of the rows which is one of the important challenges that needs to be considered for autonomous implementation in the field. Therefore, the turning method in this autonomous system is responsible to guide the vehicle to move from one row to the next when it has arrived at the end of the row.

There are few important message need to be digested by the vehicle in order to ensure the turning process executed smoothly, such as the minimum turn radius of the vehicle, width and length of the headlands [1] as well as the whole dimension of the orchid. Experimental works between two types of turning processes, which are (1) turning between adjacent rows and (2) skipping the rows have been carried out [1]. In the former technique, they implement an “asymmetric thumb turn” that is named after its distinctive shape. When the vehicle moves away to the next row, it will turn away from the row that it will enter and then turns again toward it. After that, the vehicle will locate itself by adjusting its position to be aligned with the landmark. While on the second technique, a hardcoded plan prior to which row is decided to be travelled by the vehicle is used. A reverse turning technique has been introduced by group of researchers from France in 2009 to reduce the size of the normal headland turning [2]. The work is initiated by generating a trajectory strategy which is based on the connected primitives in order to generate the path for the reverse turning operation. The primitives are the line segment and the arc of circle of the headland. These lines are connected in order to ensure the curvature of the followed motions is continued. Then, a series of steering controller is considered for two and four wheels steering vehicles by implementing the kinematic model extended with the sliding parameters. A velocity controller is also designed to anticipate the speed variations of the vehicle during the turn. Another approach by utilizing view from a camera to realize the headland turning has been implemented on an agricultural robot in the corn fields [3].

In this technique, there are two cameras installed to change the view of the robot by rotating the attached DC motors separately in the vertical and horizontal planes. The vertical camera is facing the forward direction of the robot and the horizontal camera is facing the ground of the plantation. The headland turning operation of this approach consists of six steps, namely, (1) End of row detection, (2) Move forward when there is no other plant, (3) Turn 90 degree to the left (4) Position calculation to ensure that the robot is still in the range of the next inter-row length, (5) Adjust the position of the robot, and (6) Turn 90 degree to the left with a certain delay. As a result, a square-shaped of trajectory is produced. A headland turning approach for tractor-trailer combination or a tractor with a towed implement by considering the information of the vehicle’s mechanical and field’s geometrical constraints has been simulated [4]. In this method, a dynamic model is used to optimize the control element of the system. However, the parameters used in this model must be carefully chosen to avoid any control element problems as well as time consumption due to numerical analysis technique. Therefore, this technique demand high computational cost and difficult to implement in real-time. Unfortunately, the stated techniques require extensive computation programs as well as carry large electrical equipment and therefore most of their autonomous vehicles are large in dimension. This paper addresses a simplified turning operation based on Bezier curve implementation with less computation times and suitable for small-scaled unmanned vehicle for headland turning in the agricultural field.

METHODOLOGY
Path planning at headland using Bezier curve
Generally in Bezier curve, the unique convex hull feature is used to ensure that the curve will never pass outside of the frame formed by the four control vertices which are consisted of start, intermediate and end points as shown in Figure-1 [5].
The basic functions of Bezier curve is shown in Equation (1) which four control points are used. Therefore, the third-order Bezier curve can be defined, where $B_i$ is the Bernstein function and $CP_n$ is the control vector.

$$f(u) = B_0^3 CP_0 + B_1^3 CP_1 + B_2^3 CP_2 + B_3^3 CP_3$$  \hspace{1cm} (1)

Which Bernstein function of $B_i(u)$ is,

$$B_i^3(u) = u^i (1-u)^{3-i}$$  \hspace{1cm} (2)

Therefore, by substituting Equation (2) into Equation (1), the third-order polynomial of Bezier curve in matrix form becomes,

$$f(u) = \begin{bmatrix} (1-u)^3 & 3u(1-u)^2 & 3u^2(1-u) & u^3 \end{bmatrix} \begin{bmatrix} CP_0 \\ CP_1 \\ CP_2 \\ CP_3 \end{bmatrix}$$  \hspace{1cm} (3)

However, in two-dimensional space, XY plane, four control points are given $(x_s,y_s)$, $(x_e,y_e)$,$(x_1,y_1)$ and $(x_2,y_2)$ with $(x_s,y_s)$ and $(x_e,y_e)$, are the start and end points. As a result, the Bezier function can be expressed as,

$$x(u) = \begin{bmatrix} (1-u)^3 & 3u(1-u)^2 & 3u^2(1-u) & u^3 \end{bmatrix} \begin{bmatrix} x_s \\ x_1 \\ x_2 \\ x_e \end{bmatrix}$$  \hspace{1cm} (4)

$$y(u) = \begin{bmatrix} (1-u)^3 & 3u(1-u)^2 & 3u^2(1-u) & u^3 \end{bmatrix} \begin{bmatrix} y_s \\ y_1 \\ y_2 \\ y_e \end{bmatrix}$$

L1 and L2 values need to be chosen wisely in order to generate a curvy or hemispherical shaped arch, which is not exceeded 50 cm from the end of row for headland turning or when the unmanned vehicle arrives at the end of certain row. This particular condition is set to assist the small space available in the laboratory as well as served for small agricultural area. In this context, after few curve synthesis and analysis, the value of L1 and L2 are set at 0.6m and 0 m, severally. Figure-2 illustrates the generated curve for the path planning during headland turning operation.

The problem of Equation (4) can be solved by using the calculated values of L1, L2 and intermediate points as shown clearly in above figure. These values are used to generate the desired Bezier curve for headland turning. Therefore, a complete third polynomial equation of the curve now becomes,

$$x'(u) = \begin{bmatrix} (1-u)^3 & 3u(1-u)^2 & 3u^2(1-u) & u^3 \end{bmatrix} \begin{bmatrix} 0 \\ 0.42 \\ 1 \\ 0 \end{bmatrix}$$  \hspace{1cm} (5)

$$y'(u) = \begin{bmatrix} (1-u)^3 & 3u(1-u)^2 & 3u^2(1-u) & u^3 \end{bmatrix} \begin{bmatrix} 0 \\ 0.42 \\ 1 \\ 0 \end{bmatrix}$$

**Optimized path and trajectory**

As discussed earlier, the planned Bezier curve from the home position towards its destination has generated approximately nine micro waypoints as illustrated in Figure-3 for headland turning. These waypoints are roughly 10 cm distance from one to another, which has made the trajectory planning for the unmanned vehicle tougher and has made the process of the velocities planning for each waypoint becomes harder. Furthermore, some of the orientation angle is way too small because of the shorter distance, resulting the turning motion of the unmanned vehicle becomes meaningless. If this happens, the tendency of the unmanned vehicle to deviate from its original path is greater and the risk of the collision with the available landmarks is higher. Despite that, the computed velocities, which are carried out by the unmanned vehicle to satisfy all these needs, at some point, have exceeded the maximum capacity of the DC motor. In order to solve this problem, a few micro waypoints are selected purposely to create another simplified route and at the same time maintaining the convex hull of the Bezier curve. Figure-3 describes illustratively the optimized path in this work.

According to this figure, there is only one point, which the degree of turning orientation is obvious compared to other points. The point is at $W_3$. As a result, only this point is regarded as the turning point in the whole path, which then leaves the curvy path with only two routes in total, namely Route 4 and 5 to move from its initial point towards the end as described in Figure-4. With
this approach, the unmanned vehicle will only have two different orientations and the required velocities are more reasonable. In this work, the trigonometric computation is applied to compute the orientation angle of the unmanned vehicle should possess before heading to the next waypoint on a two-dimensional Cartesian plane. It is computed by using this equation,

$$
\theta = \tan^{-1} \frac{\delta y}{\delta x} = \tan^{-1} \frac{\delta x}{\delta y}
$$

(6)

Where $\delta y$ and $\delta x$ is the total distance in Y and X axis.

Once the orientation of the unmanned vehicle is determined, the velocities computation is followed. Without the assisting of GPS, rotary or encoder sensor, it is really hard to determine the exact distance from one waypoint to another. Moreover, the usable landmarks in this work, which are the trees, are not aligned to the location of the waypoints. Therefore, it is hard to determine the location of the next waypoint based on this information. As a result, to overcome this issue, a dead reckoning approach is utilized by providing necessary data such as Euclidean distance from the start towards destination and a specific timing for the unmanned vehicle to reach there. In this work, the distances should be provided to the velocities computation is denoted as $H_4$ and $H_5$. Eqn. 7 is used to determine the Euclidean distance for each route.

$$
H_4 = \sqrt{(\delta y)^2 + (\delta x)^2}
$$

$$
H_5 = \sqrt{(\delta y)^2 + (\delta z)^2}
$$

(7)

After the Euclidean distance computation is completed, these values can be used to calculate the desired velocities for each route. To compute these, additional information is needed, that is the total elapsed time for the unmanned vehicle to move from its home position towards its next point. To compute the necessary velocities, $V_1$, $V_2$, $V_3$, $V_4$, and $V_5$, below equations are applied.

$$
V_4 = \frac{H_4}{\delta t_2} \leq V_{MAX}
$$

$$
V_5 = \frac{H_5}{\delta t_2} \leq V_{MAX}
$$

(8)

In this implementation, the outcome velocities value must not exceed the maximum velocity, $V_{MAX}$, which is provided by the specification of the utilized motor. The allowed velocities values are less or equal with 0.634 ms-1. Once the velocities computation and determination is completed, the unmanned vehicle must be clearly directed to obey these values.

**EXPERIMENTAL SETUP**

This work is designed by deploying the inter-row tree detection and tracking techniques in an embedded microcontroller-based unmanned vehicle platform. Figure-5 presents the detail functions and elements participated in this work. The overall development phase can be divided into two categories, namely, software and hardware co-design systems. In the hardware co-design system, an internal timer, namely, Timer 2 is added to provide a delay that conceives the total navigation time to avoid a sudden velocity change effect. In the software co-design system, the tree detection system is designed to detect the presence of the landmarks or trees. This scheme has is implementing an enhanced two triangle diameter estimation [6] by using small discrete electrical and electronic components. While on the other hand, there are few functions that are designed to support the navigation scheme for the unmanned vehicle.

**Figure-3.** The generated micro (local) waypoints in one Bezier curve for 1 meter of path for headland turning.

**Figure-4.** Unmanned Vehicle trajectory planning based on the simplified waypoints for headland turning.

**Figure-5.** Hardware-Software co-design systems.
RESULT AND DISCUSSION

Another important aspect in providing automation feature in the agricultural field, which is organized in rows of trees, is to ensure that the automated vehicle able to self-navigate from one row to another without colliding with the trees and maintain its distance with them. Therefore, Figure-6 presents the real-time trajectory obtained for turning navigation of the automated vehicle based on the Bezier curve planned path. In this experiment, 10 sets of experiments are performed to collect and analyze the data for the proposed turning operation. From the figure above it can be seen that in Route 1, which is the first route experienced by the unmanned vehicle, results the least errors and the obtained real-time trajectory data are bound to comply closely with the planned path. However, when the unmanned vehicle is about to start its journey in the second route, Route 2, it is apparent that the coming errors are slightly expanding. Further analysis shows that, greater errors are resulting in Route 3. There are several possible explanations for this result. One of the obvious reasons that can be clearly seen in this figure is that the unmanned vehicle has been intended to deviate slightly from the planned path just after it started its journey from the starting point of Route 1, P1. At the end of Route 1, which is denoted as P2, the accumulated errors have become larger and this event contributes to the greater path deviation in Route 2 and Route 3 as well. Figure-7 illustrates the error contribution at every route for every attempt.

Figure-6. Real-time trajectory obtained from the turning navigation experiments.

Figure-8 describes the mean errors occur in all routes during this operation. According to this figure, Route 1 contributes the least error with the average of 0.0566 m, which leaves Route 2 and Route 3 with average errors of 0.1097 m and 0.1359 m, respectively. Thus, the total average error for all routes is 0.1007 m. Nevertheless, this error can be cancelled or ruled out throughout the entire navigation process by reposition the unmanned vehicle to its initial position, which is approximately in the middle of the row’s path when it has arrived at the end of the Route 3, P4. This procedure is so important so that the accumulated errors are not channeled to the next way point. Therefore, the unmanned vehicle is certainly remained approximately in the middle of the row’s path and collision with the available landmarks can be prevented. Despite the larger error in Route 3, the average error for all routes is still complied with the objective and scope of this research and hence, leaves a room for improvement in future. A comparison with the previous implementation in diameter measurement, straight and turning navigations are discussed in detail in the next section in order to demonstrate that the proposed framework is equate with others in this research context.

Figure-7. Error contribution in each step for Route 1, Route 2 and Route 3.

Figure-8. Average error contributed by each route.

In turning operation or headland navigation when it comes toward the end of a row, the result obtained for this work is compared with other related works in Table-1. In this case, the worst average error is given by this research, with 10.07 cm error due to improper Bezier
curve connections in providing the headland turning path. An image processing approach for end of row turning navigation in real-time has been implemented by [3] and resulted the least average error of 1.27 cm from its planned path. However, [4] and [2] has given a moderate mean error of 5 cm for both implementations. Even the former approach is using a continuous Bezier curve planned path, but both implementations are software simulation.

Table-1. Comparison with other previous related approach.

<table>
<thead>
<tr>
<th>Design</th>
<th>Method</th>
<th>Row Width (cm)</th>
<th>Average Error (cm)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Norashikin</td>
<td>Bezier Curve Path Planning with Real-time navigation scheme.</td>
<td>100</td>
<td>10.07</td>
</tr>
<tr>
<td>[3]</td>
<td>Using variable field of view of a camera.</td>
<td>75</td>
<td>1.27</td>
</tr>
</tbody>
</table>

CONCLUSIONS

In this research, Bezier curve is implemented in outlining the path for the unmanned vehicle to navigate in the headland of the agricultural field. Once the path is finally confirmed, velocities and orientation angles profile for the unmanned vehicle is built according to the capacity and ability of the dc motor and microcontroller features that are attached to the platform.

ACKNOWLEDGEMENTS

This research is sponsored by Research Incentive Faculty (RIF) Fund, Universiti Teknologi MARA (Contract number: 600-RMI/DANA 5/3/RIF (7/2012)).

REFERENCES


