HEADLAND TURNING CONTROL METHOD SIMULATION OF AUTONOMOUS AGRICULTRUAL MACHINE BASED ON IMPROVED PURE PURSUIT MODEL

Peichen Huang [1], Xiwen Luo [2,*], Zhigang Zhang [3]

Key Laboratory of Key Technology on Agricultural Machine and Equipment of South China Agricultural University, Ministry of Education, Guangzhou 510640, P. R. China

[*] *Corresponding author, Address: Key Laboratory of Key Technology on Agricultural Machine and Equipment of South China Agricultural University, Ministry of Education, Guangzhou 510640, Guangdong Province, P. R. China, Tel: +86-020-38676975, Fax: +86-020-38676975, Email: xwluo@scau.edu.cn*

Abstract: According to the features of headland turning, new path planning and headland turning control algorithms for autonomous agricultural machine were presented in this paper. The turning path planning considered both the minimum turning radius and headland space was created by applying three straight lines. A path tracking algorithm based on the improved pure pursuit model was also proposed. This study used the BP neutral network to implement the dynamical look-ahead distance control for the improved pure pursuit model. Based on simplified bicycle kinematics model parameters, MATLAB/Simulink simulation results showed that the path planning algorithm were simple, occupied small headland space while still had a high tracking accuracy. The control method is feasible and practical.

Keywords: automatic guidance, agricultural machine, headland turning, path planning, pure pursuit algorithm, simulation

1. INTRODUCTION

With the development of computer and sensor technologies, automatic guidance of agricultural machine has received deep research over many developed countries and regions (Toru Torii, 2000). The automatic guidance of agricultural machine technology is not only the basic platform for precision farming but also one of the research hotspots in fields of agricultural engineering. The property of headland turning is remarkably different from traveling on straight line. A proper headland turning not only improve the tracking accuracy when a machine transfers from the current working row to the next one, but also minimize the time spent in the headlands so as to increase the efficiency of farming operation (Zhu Zhongxiang et al., 2007).

A headland turning algorithm for rice transplanter was designed by Y. Nagasaka etc to guide the rice transplanter to move forward and backward during a turn (Yoshisada Nagasaka et al., 2004). This method could minimize the headland space but makes steering control complicated due to its backward motion. Kise. M et al (2001) developed two types of turning paths, namely forward turning and switch-back turning, which were created by applying a third-order spline function based on the minimum turning radius and maximum steering speeds. Computer simulation showed that the maximum tracking error was less than 0.2m. As the curve turning path is created by utilizing the third-order spline function, the headland turning for machine has to be implemented by curve tracking. The control difficulty will increase correspondingly. ZHU Zhongxiang (2007) etc proposed an optimal control algorithm. A time-minimum suboptimal control method was used to generate the turning path. A path-tracking controller consisting of both feedforward and feedback component elements was also proposed. This method implemented the optimal path with respect to traveling in minimum time, but the design of tracking controller is very complicated.

According to the features of headland turning and pure pursuit algorithm, agricultural machine headland turning control algorithm was presented in this paper. It works as follows: 1) applying simplified bicycle kinematics model, the path planning was produced by assembling three straight lines; 2) using the dynamical look-ahead distance which was implemented by the BP neutral network, an improved pure pursuit algorithm was applied in the headland turning.

2. MATERIALS AND METHODS

2.1 Pure Pursuit Model

Pure Pursuit is a method for geometrically calculating the arc necessary for getting a vehicle onto a path (R.Craig Conlter, 1992; Vijay Subramanian et al., 2007). The method is simple, intuitive, easy to implement. The whole point of the algorithm is to choose a proper look-ahead distance. It is analogous to human driving in that humans look a certain distance ahead of the vehicle and steer such that the vehicle would reach a point at the look-ahead distance (Vijay Subramanian et al., 2007). The Pure Pursuit algorithm has been widely used in the field of path tracking. The algorithm is expressed as Fig.1 (All the parameters shown in Fig.1 are based on the machine's coordinate system without specification in the following part).

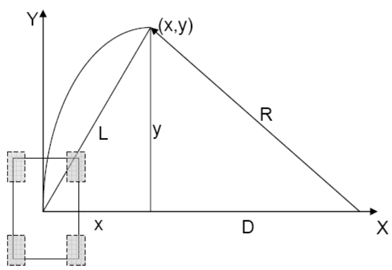


Fig.1: Geometry of the Pure Pursuit Model

The x and y axis construct the machine's coordinate system. The point *(x, y)* is a point some distance ahead of the machine. The L is the length of the cord of the arc connecting the origin to the point *(x, y)*. R is the radius of curvature of the arc. The relationship of x, L and R is as follows:

$$D + x = R \tag{1}$$

$$D^2 + y^2 = R^2 \tag{2}$$

$$x^2 + y^2 = L^2 \tag{3}$$

From Eq. (1), Eq. (2) and Eq. (3),

$$R^2 - 2Rx + x^2 + y^2 = R^2$$

$$R = \frac{L^2}{2x}$$

By choosing a look-ahead distance and calculating the path error x, the radius of the curvature required to get the machine on the required path can be calculated.

2.2 Headland Turning Control Method

2.2.1 Simplified Bicycle Kenimic Model

In this study, a simplified bicycle kenimic model is used to describe the machine motion. For the sake of simplified, it is assumed that the machine moves at a low constant speed over a flat surface with no wheel slippage and the wheels is considered as rigid wheels.

According to the kenimatic analysis (A.J.Kelly, 1994), the machine's motion equation is given as follows:

$$x'(t) = v(t)\cos\phi(t)$$
$$y'(t) = v(t)\sin\phi(t)$$
$$\phi'(t) = v(t)\tan\delta(t)/l$$

where l is the wheel base; δ is the front wheel steering angle; ϕ is the heading angle. As illustrated in Fig.2:

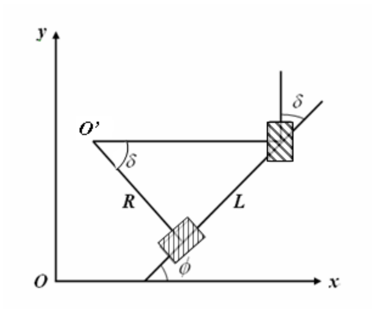


Fig.2 The Kenimec Model of Simplified Bicycle

$$R = l/\tan\delta$$

where R is turning radius of the machine, l is wheel base, and δ is the front wheel steering angle. Combined with the pure pursuit equation derived above, the steering angle is represented by $\delta = \arctan(2lx/L^2)$. It indicates the relationship between the pure pursuit algorithm parameters and steering angle. This will help to lay a theoretical foundation of building the tracking control system.

2.2.2 Headland Turning Path Planning

Fig.3 shows the algorithm of path planning for headland turning in case of left turning. It is assumed that the minimum turning radius of a machine is 0.9*m* and the width *M* denotes $2R_{\min}$,i.e. 1.8 *m*. The headland turning path shown in the Fig.3 is created by using three straight lines. In the established coordinate system, the three straight lines equations can be expressed as follows:

Path1: $x = 1.8 \quad (0 \le y \le 3.5)$

Path 2: $y = 3.5 \quad (0.9 \le x \le 1.8)$

Path 3: $x = 0$

As shown in Fig.3, the width *M*, i.e, the distance between path1 and path2,

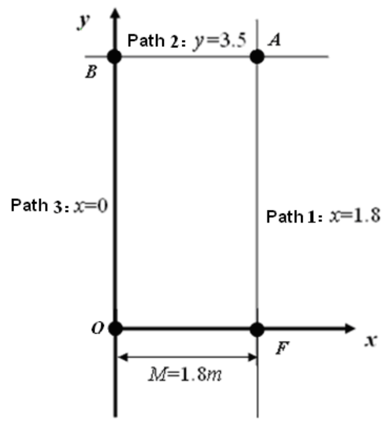


Fig.3 Path planning of headland using three straight lines

is two times equal to the machine's minimum turning radius, which not only fulfills the constrain of minimum turning radius, but also ensures the machine occupy the smallest width of headland. Path2, i.e $y = k$ is the important part in headland turning. The value *k* affects the turning space size and determines whether the machine is able to approach the next operation path precisely. The machine will have enough distance to adjust its position before approaching the next operation path while *k* is sufficient large, but meanwhile it occupies sizeable headland space. On the contrary, if *k* is smaller, it will reduce the space but the corresponding distance to adjust the machine’s position will also be shortened.

After repeated simulations, a qualitative conclusion obtained from results is presented in this paper: During the headland path planning, the value *k* which can both make the machine get exactly enough distance to adjust its position and occupy the smallest space is determined by the width *M* and the machine velocity. In this paper, the width is 1.8*m* and machine velocity is set at 0.3 *m/s*. When the value k is set at 3.5, it can obtain a favorable effect on both aspects. When the machine approaches at the end of the current operation path i.e. point *F*, it then starts to track Path1. When approaching the end point *A* of Path1, by using the path switcher, the machine switches to track the Path2. Likewise, the switch between Path2 and Path3 is to implement in the same way. Finally, the machine approaches at the start point *O* of the next operation path.

2.2.3 The Dynamical Look-ahead Distance Control Based On Neural Network

There is one key parameter in the pure pursuit algorithm, the look-ahead distance. Its value has great effect on the tracking accuracy. Large look-ahead distances result in a gradual and smooth approaching of the path, but one which may take a considerable amount of time. Short look-ahead distances approaches the path quicker, but may result in oscillation about the path (R.Craig Conlter, 1992). Based on the analysis above, the large and short look-ahead distances both go against tracking effect. Many researchers may prefer to get a relative better look-ahead distance among large and small through repeated experiments. This method may has a relative better tracking effect on a certain degree but it is not the optimum as the look-ahead distance is unchanged during the whole headland turning. In addition, the velocity of a machine also affects the path tracking. Small velocity result in high tracking accuracy but also high time consumption, and vice versa so it is also necessary to consider the selection of the velocity.

According to the headland path planning algorithm above, the major task during the path1 and path2 tracking is to implement the machine turning, so a fixed moderate look-ahead distance can be chosen. Krešimir Petrinec, Zdenko Kovacic etc pointed out that a good choice of fixed look-ahead distance is around one wheelbase (Krešimir Petrinec et al., 2003) (in our case 1*m*). The path3 tracking is related to the accuracy when entering the next operation path. It requires high accuracy, so the dynamical look-ahead distance control is used.

There will be a large offset at the initial stage when tracking path3 as the machine has to take some time to adjust the heading angle. Therefore a smaller look-ahead distance is necessary at this stage to make the machine regain the path quicker. When the offset becomes small enough (within assigning range), larger look-ahead is used to let the machine regain the path with less oscillation. Further considering the machine velocity, according to the Preview Follower Theory (Wang Jingqi et al., 2003), higher velocity requires larger look-ahead distance and vice versa. So different look-ahead distances should be used according to different velocities.

This paper utilized the neural network's seft-study and association memory ability to teach a neural network how to dynamically control the look-ahead distances. This network has two inputs (the machine velocity and the *x* coordinate) and one output (the desired look-ahead distances). After the comprehensively considering with the tracking accuracy and the consuming time, two velocities are selected in this paper. They are 0.3*m/s* and 0.4 *m/s*. The headland was divided into three areas by two straight lines ($x = -0.5$ and $x = 0.5$). Different look-ahead distances in every individual area are used. After using different combination of velocities and look-ahead distances in the repeated simulations, it is determined that when the velocity is 0.3 *m/s*, 1 *m* is used as large look-ahead distance and 0.01*m* as small look-ahead distance; when the velocity is 0.4 *m/s*, 1 *m* is as large and 0.5 *m* as small. Before establishing the neural network, it is necessary to determine the training set. At each training sample, the *x* coordinates and velocity values were used as the inputs and the desired look-ahead distance *L* for the output, e.g, $[x=-1, v=0.3, L=0.01]$ is one training data, thus there were a total of 148 examples (cases) in the training set. Table1 shows the training samples.

Table 1. Sample Data for Training Neural Network

x	*v*	*L*	*x*	*v*	*L*
-1	0.3	0.01	-1	0.4	0.5
-0.99	0.3	0.01	-0.99	0.4	0.5
-0.98	0.3	0.01	-0.98	0.4	0.5
…	…	…	…	…	…
-0.52	0.3	0.01	-0.52	0.4	0.5
-0.51	0.3	0.01	-0.51	0.4	0.5
-0.5	0.3	1	-0.5	0.4	1
-0.4	0.3	1	-0.4	0.4	1
-0.3	0.3	1	-0.3	0.4	1
…	…	…	…	…	…
1.7	0.3	1	1.7	0.4	1
1.8	0.3	1	1.8	0.4	1

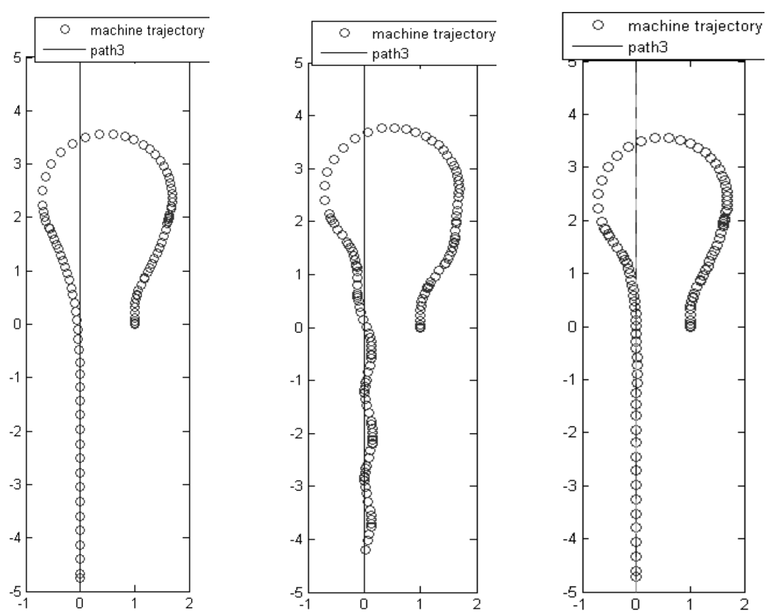
In Table 1, *x*, *v* and *L* represent the *x* coordinate, velocity (*m/s*) and look-ahead distance (*m*) respectively. After training with the training set, the network which memorized the look-ahead distances in different areas was established finally by using BP algorithm, which is commonly used in the engineering. The dynamical control of look-ahead distance for tracking can let the machine adapt to the changing of practical situation and occupy the small headland space as much as possible.

3. SIMULATION RESULTS AND DISCUSSION

3.1 The Headland Turning Simulation Based On Pure Pursuit Model

The headland turning simulation based on the pure pursuit model was performed on MATLAB/Simulink platform. In this case, the parameters were initialized as follows: V =0.3*m/s*; M=1.8*m*; R_{min} =0.9 *m*; the starting position was set at (1, 0); the initial heading was set at 90^0 . Two fixed look-ahead distances are simulated separately, the results are shown in Fig.4(a) and Fig.4(b). The simulation process and results of the case V=0.4 m/s is similar to the case V=0.3 m/s , so this paper only present the latter case.

In the whole headland turning simulation results, the tracking trajectory of path3 is the most important as it can show if the machine can entry the next operation path accurately, so the results only show the desired path3 and machine tracking trajectory. The straight line represents the path3 and circles represent the machine trajectory.



(a): Simulation result of headland turning control with fixed look-ahead 1*m*

(b): Simulation result of headland turning control with fixed look-ahead 0.01*m*

(c): Simulation result of headland turning control with dynamical look-ahead

Fig.4 Simulation result of headland turning control with different look-ahead distance

Fig.4(a) and Fig.4(b) illustrate the simulation results with fixed look-ahead distances 1 *m* and 0.01 *m*. Same initial conditions were set at both simulations.

It was found that when using the smaller fixed look-ahead distance 0.01*m*, the machine trajectory is more oscillations as shown in Fig.4(b). The larger look-ahead distance 1*m* resulted in a gradual and smooth trajectory, but took longer time for the machine approaching to the path as shown in Fig.4(a).

3.2 The Headland Turning Simulation Based On The Improved Pure Pursuit Model —— BP network Implement The Dynamical Look-ahead Distance Control.

The training of the network was performed with BP network function package provided by the MATLAB Neural Network Toolbox. First, function newff( ) was utilized to create a three layers' network (one input, three hidden, one output layer). Second, function train( ) was utilized to perform training. The training epochs were set at 100 while the mean square error (MSE) was 10^{-6} (reasonably low) after training. Finally, the network Simulink module generated by function gensim( ) was combined with the normalized and anti-normalized modules to create the dynamical look-ahead distance control module.

Fig.4(c) shows the simulation result after using the dynamical look-ahead distance control during the last desired path $x = 0$. When the *x* coordinate is large, small look-ahead distance (in our case 0.01*m*) tends to converge to the path more quickly. When the *x* coordinate is small, large look-ahead distance (in our case1 *m*) tends to regain the path with less oscillation.

It was indicated in Fig.4 that the dynamical look-ahead distance control could significantly improve the tracking effects. Not only the machine could converge the path more quicker, but also with less oscillation. This was favorable for the machine to entry the next operation accurately.

4. CONCLUSION

According to the pure pursuit algorithm which was originally devised as a method for mobile robot tracking the path, combined with the characteristics of agricultural machine, a new control method of headland turning based on the improved pure pursuit model for agricultural machine was presented in this paper. The MATLAB simulation results showed that this control method was feasible and effective.

The path planning algorithm, which is implemented by combining three straight lines, makes the path planning simple and easy to implement while the machine's tracking effect is good.

The dynamical look-ahead distance control was also proposed. This could let the machine adapt to the changing of practical situation. Compared to two fixed look-ahead distance simulation, the dynamical look-ahead distance control results show that it has the advantages of reducing time and space, more reliable and high accuracy during the headland turning.

In future work, attention must be paid to providing an accurate formula to quantitatively calculate the path2 position by utilizing the path1 and path3 positions so that the machine can obtain the best on both adjusted distance and tracking accuracy.

ACKNOWLEDGEMENTS

Research conducted at South China Agricultural University, funding from National Fund 863. Special thanks are extended to Professor Luo Xiwen and Mr.Zhang Zhigang for their technical support.

REFERENCES

A. J. Kelly. A Feedforward Control Approach to Local Navigation Problem for Autonoumous Vehicles,CMU Robotics Institute Technical Report, 1994, CMU-RI-TR-94-1

Kise. M., Noguchi. N., Ishii. K., Terao. H. Development of the agricultural autonomous tractor with an RTK-GPS and a FOG: Proceedings of the 4th IFAC symposium on Intelligent autonomous vehicles. IFAC, 2001: 103-106

Krešimir Petrinec, Zdenko Kovacic, Alessandro Marozin. Simulator of Multi-AGV Robotic Industrial Environments. IEEE, 2003

R.Craig Conlter. Implementation of the Pure Pursuit Path checking Algorithm. Camegie Mellon University, 1992

Toru Torii. Research In Autonomous Agriculture Vehicles In Japan, Computers and Electronics in Agriculture, 2000, 25:133-153

Vijay Subramanian, Thomas F Burks. Autonomous Vehicle Turning In the Headlands of Citrus Groves, An ASABE Meeting Presentation, ASABE, 2007, 1015

Wang Jingqi, Chen Huiyan, Zheng Pei. Application of Fuzzy Adaptive PID and Previewing Algorithm in Steering Control of Autonomous Land Vehicle, Automotive Engineering, 2003, 25(4): 367－371(in Chinese)

Yoshisada Nagasaka, Naonobu Umeda, Yutaka Kanetai, Ken Taniwaki, Yasuhiro Sasaki. Autonomous guidance for rice transplanting using global positioning and gyroscopes, Computer and Electronics in Agriculture, 2004, (43): 223-234

Zhu Zhongxiang, Chen Jun, YOSHIDA Toyofumi, TORISU Ryo, SONG Zheng-he, MAO En-rong. Path Tracking Control of Autonomous Agricultural Mobile Robots, Journal of Zhejiang University SCIENCE A, 2007, 8(10):159-1603 (in Chinese)