Automated Guided Robot with Backstepping Sliding Mode Control and Its Path Planning in Strip Farming

Jen C. Chien, Chung L. Chang, *Member*, *RST*, Chang C. Yu

***Abstract*—In this paper, a combination of path planning and backstepping sliding mode control approach is applied to the four-wheeled agricultural robot. First, the anchor points in a strip farm is located by the real-time kinematic positioning in global navigation satellite system (GNSS-RTK). Second, a point-to-point trajectory is formed on a strip farm map using cubic spline interpolation by selecting initial, control, and target points between the anchor points. Finally, backstepping and sliding model technique is used to solve the path following problem. The stability of the proposed guidance system is proved by Lyapunov theory.**

Keyword—Wheeled mobile robot, path planning, dynamic model, sliding model control, backstepping

I. Introduction

In recent years, the number of people engaged in agriculture has been decreasing year by year. Some of the new forms of farming have been changed to automated management. A variety of automated agricultural robots have become one of the important tools in new agricultural management. Today, many task-oriented autonomous field robots have been created and used to perform field tasks such as weeding, seeding, spraying, and fruit picking. To do these tasks well, it is first necessary to rely on the field robot to be able to move autonomously and stably in the field, so automatic guidance becomes very important. The use of automatically guided robots to perform field work can not only greatly reduce the workload of farmers, but also improve the safety of farmers when they are working in the field.

Autonomous robots need to have high operating efficiency, and the detouring range and operating speed of the robot must be taken into account. In addition, the availability rate, stability, endurance and durability of the equipment must also meet the needs of field operations. Managers or operators do not necessarily operate and manage on-site, and their main job is to assist in improving the safety of robot use. Among them, the automatic guidance to achieve high-precision positioning is still a challenging task. In addition, the robot can perform various agricultural tasks smoothly, and fast automatic guidance is also required. The robot operating system must be able to quickly plan the route, accurately control the speed and compensate for the positioning error [1][2].

This work is financially supported in part by the Ministry of Science and Technology, Taiwan, R.O.C., under grant MOST 109-2321-B-020-004; MOST 110-2221-E-020-019.

J. C. Chien and C. C. Yu are with the Department of Vehicle Engineering, National Pingtung University of Science and Technology (chyu@mail.npust.edu.tw).

C. L. Chang are with the Department of Biomechanical Engineering, National Pingtung University of Science and Technology (chungliang@mail.npust.edu.tw).

When the robot moves, it must first plan the route and set the positioning point in the field in advance, which should consider the size of the farmland and the configuration of the field border. Selecting too many path anchors increases farming time. Conversely, too few position points will cause the robot to travel erratically. Therefore, a few key positioning points are marked in the field in advance, and the positioning points are converted into two-dimensional plane coordinates through spatial coordinate transformation. Then, the spline interpolation method is used, which provides automatic guidance and positioning of the robot by planning a new path between the original positioning points through a software method. The above approach has been verified to improve the stability of robot motion [3][4][5]. In terms of speed control, the method of backstepping can be used to replace the traditional feedback linearization control [6]–[10]. In addition, the early sliding fuzzy control methods also have the ability to resist external and internal parameter disturbances [11]–[16]. This method replaces the general fuzzy controller as well as adaptive control [17]. At present, there are many integrated linear/non-linear control methods for autonomous control of agricultural robots [18]–[21].

This study proposes a cubic spline interpolation method to correct the original path, so that the robot can move smoothly and quickly correct the positioning error, thereby maintaining the ability of the agricultural robot to drive stably at high speed. In addition, the sliding mode control method is used to correct the tracking error, which can make the error converge within a period of time, and the ability to make the actual value catch up to the target value can increase the robustness of the overall system. In addition, backstepping control is used to assist the robot system in trajectory tracking, ensuring that the robot guidance process is not affected by external disturbances.

The organization of this paper are as follows: The second chapter is the methodology, which introduces the dynamic model of robot, path planning method and controller design; the simulation results are presented in the third chapter, including the setting of motion parameters, scenarios simulation and its results; finally, the conclusion is presented.

II. Methodology

The autonomous system of agricultural robot mainly includes drive system, steering device, positioning and control system. The terrain in the farmland is not all flat, so it is necessary to provide a specific position for the heading correction when the robot moves. To save power and labor time, it is not necessary to measure too many target points. It just needs to measure enough turning points. First, a smooth route is planned by cubic spline interpolation and the coefficients of the

route function are generated. Next, estimate the angle between the desired heading angle of the robot at the next target point and the heading angle of the current position point. Since there is an error between the actual robot position and the current position, the back-step sliding mode controller in the dynamic control system will be used to quickly converge the error to achieve the purpose of automatic navigation.

A. Dynamic Model of Wheeled Robot

Among them, the robot head towards the X-axis direction, the distance between the front wheel and the center of gravity (CG) of the robot is denoted as l_f, and the distance between the rear wheel and the center of gravity of the robot is defined as l_r. Longitudinal force, lateral force, steering angle, and sideslip angle are denoted as F_x, F_y, δ_f, δ_r and α, respectively [22].

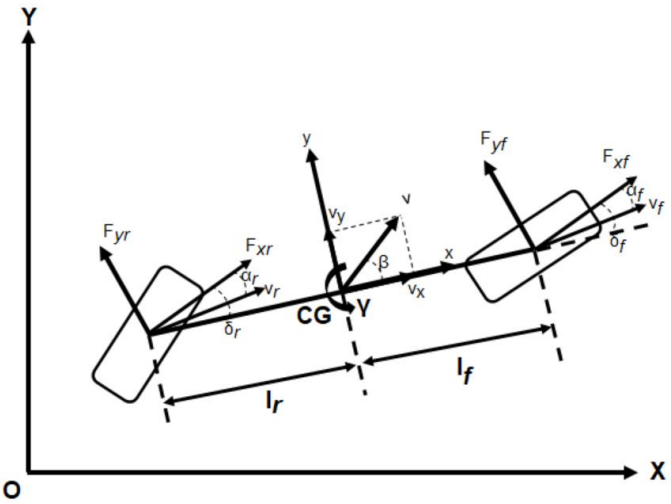


Fig. 1. Simplified dynamic model of robot.

According to the robot dynamic model, when the vertical force and horizontal torque torque of the robot heading are obtained, the angular acceleration , the vertical velocity and the horizontal velocity of the heading can be obtained, which are respectively as (1)–(3). When the robot corrects the heading angle, the change of angle difference is very small. Let $\cos\delta_f = \cos\delta_r \approx 1$, $\sin\delta_f = \sin\delta_r \approx 0$, then (1)–(3) are simplified to (4)–(6).

$$\dot{\gamma} = ((l_f - F_{xf}\sin\delta_f + l_r F_{xr}\sin\delta_r) + l_f F_{yf}\cos\delta_f - l_r F_{yr}\cos\delta_r)/I_z \quad (1)$$

$$\dot{v}_x = \left(\frac{F_{xf}\cos\delta_f + F_{xr}\cos\delta_r - F_{yf}\sin\delta_f - F_{yr}\sin\delta_r}{m}\right) + \gamma v_y \quad (2)$$

$$\dot{v}_y = \left(\frac{F_{yf}\sin\delta_f + F_{xr}\sin\delta_r + F_{yf}\cos\delta_f + F_{yr}\cos\delta_r}{m}\right) - \gamma v_x \quad (3)$$

$$\dot{\gamma} = \frac{l_f F_{yf} - l_r F_{yr}}{I_z} \quad (4)$$

$$\dot{v}_x = \frac{F_{xf} + F_{xr}}{m} + \gamma v_y \quad (5)$$

$$\dot{v}_y = \frac{F_{yf} + F_{yr}}{m} - \gamma v_x \quad (6)$$

B. Path Planning

1) Coordinate transformation

After the dynamic model of the robot is established, the user can pre-select several locations on the actual field and use the real-time kinemetric positioning in Global Navigation Satellite System (GNSS-RTK) to obtain the latitude/longitude position data [23]. These World Geodetic Coordinate System (WGS-84) position data will be converted to Taiwan-Geodetic Datum-97 (TWD97) data [24].

2) Cubic spline interpolation

Only the position data converted into the two-dimensional plane coordinate system of TWD97 can be used to generate a cubic curve, and a line segment of a cubic equation can be generated between every two target position points [5], as shown in Fig. 2. However, in the process of position estimation, we only get the relationship equation between the *x* coordinate and the time instant and the relationship equation between the *y* coordinate and the time instant. These two equations can obtain the position of the robot on the two-dimensional plane at each time instant. "Point *i*" can be the *x* or *y* component of the target point. "T_i" is the movement time required for the robot to move from the target point *i* to the next target point (*i*+1). Assume a_i to be the constant coefficient of the cubic equation i, and also the *x*-value or *y*-value of the starting point (target point) of each line segment. b_i is the coefficient of the first-order term of the cubic equation *i*, which is also the *x*-direction velocity component or the *y*-direction velocity component of each time instant in the line segment. c_i is the quadratic term coefficient of the cubic equation *i*, which can be obtained using (7). d_i is the coefficient of the cubic term of the cubic equation *i*, which can be calculated using the (8). Finally, for each line segment between target points, the cubic coefficients of *x* versus time and the cubic coefficients of *y* versus time can be obtained. Assuming that there are *N* target points in total, there will be *N*–1 line segments, resulting in two sets of coefficient matrices of size 4×(*N*–1).

$$c_i = \frac{1}{T_i}\left[\frac{3(p_{i+1} - p_i)}{T_i} - 2v_i - v_{i+1}\right] \quad (7)$$

$$d_i = \frac{1}{T_i^2}\left[\frac{2(p_i - p_{i+1})}{T_i} + v_i + v_{i+1}\right] \quad (8)$$

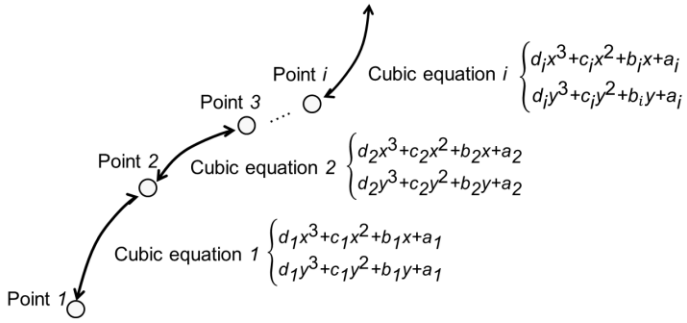


Fig. 2. The cubic equation of each line segment and its coefficients.

C. Speed Control

After knowing the cubic equation and its coefficients of each line segment, the desired path can be planned. At the same time, after giving the time to reach the target point, the target

speed at each moment can be calculated, v is the speed vector, $v = [v_1 \quad v_2 \quad \cdots \quad v_{N-2} \quad v_{N-1}]^T = A \cdot c$, which represents the velocity combination of each line segment, where A represents the time matrix of each starting point reaching the target point, and c is the acceleration vector of each line segment, as in (9) –(10).

$$A = \begin{bmatrix} 2(T_0+T_1) & T_0 & 0 & \cdots & \cdots & 0 \\ T_2 & 2(T_1+T_2) & T_1 & \cdots & \cdots & \vdots \\ 0 & \cdots & \cdots & \ddots & \cdots & 0 \\ \vdots & \cdots & \cdots & \cdots & 2(T_{N-3}+T_{N-2}) & T_{N-3} \\ 0 & \cdots & \cdots & 0 & T_{N-1} & 2(T_{N-2}+T_{N-1}) \end{bmatrix} \tag{9}$$

$$c = \begin{bmatrix} \frac{3}{T_0 T_1}\left[T_0^2(x_2-x_1)+T_1^2(x_1-x_0)\right]-T_1 v_0 \\ \frac{3}{T_1 T_2}\left[T_1^2(x_3-x_2)+T_2^2(x_2-x_1)\right] \\ \vdots \\ \frac{3}{T_{N-3} T_{N-2}}\left[T_{N-3}^2(x_{N-1}-x_{N-2})+T_{N-2}^2(x_{N-2}-x_{N-3})\right] \\ \frac{3}{T_{N-2} T_{N-1}}\left[T_{N-2}^2(x_N-x_{N-1})+T_{N-1}^2(x_{N-1}-x_{N-2})-T_{N-2} v_N\right] \end{bmatrix} \tag{10}$$

The above formula is executed by the backstepping-based sliding mode controller.

D. Controler Design

Since the speed and steering angle of the robot are the control parameters of the guidance system, they are related to the vertical acceleration, vertical force and lateral force of the robot. Write the target heading angle, the *x*-direction component of the target point, and the y-direction component of the target point as a matrix, $q=[\psi \quad x \quad y]^T$. Perform a differential operation on vector q, which is expressed as the target angular velocity $\dot{\psi}$, the *x*-direction velocity of the target point $\dot{x}$, and the y-direction velocity of the target point $\dot{y}$ which can be written in matrix form, $\dot{q}=[\dot{\psi} \quad \dot{x} \quad \dot{y}]^T=[\gamma \quad v_x \quad v_y]^T$. Differentiating vector $\dot{q}$ and the elements in the vector $\ddot{q}$ are expressed as target angular acceleration $\ddot{\psi}$, target point *x* direction acceleration $\ddot{x}$, target point *y* direction acceleration $\ddot{y}$, which can be written as a matrix $\ddot{q}=[\ddot{\psi} \quad \ddot{x} \quad \ddot{y}]^T=[\dot{\gamma} \quad \dot{v}_x \quad \dot{v}_y]^T$, such as (11). Rewrite (11) into the state equation, such as (12).

Equation (13) is the state variable matrix, and (14) is expressed as the robot parameters, including the distance from the front wheel to the center of gravity of the robot, l_f, the distance from the rear wheel to the center of gravity of the robot, l_r, the weight of robot m and the inertia torque I_z of the robot, (15) represents the theoretical control vector of the robot.

$$\ddot{q} = [\dot{\gamma} \quad \dot{v}_x \quad \dot{v}_y]^T \tag{11}$$

$$\ddot{q} = A(q,\dot{q}) + BU \tag{12}$$

$$A(q,\dot{q}) = [0 \quad 0 \quad -\gamma v_x]^T \tag{13}$$

$$B = \begin{bmatrix} 0 & \frac{l_f}{I_z} & \frac{l_r}{I_z} \\ 1 & 0 & 0 \\ 0 & \frac{1}{m} & \frac{1}{m} \end{bmatrix} \tag{14}$$

$$U = [A_{cc} \quad F_{yf} \quad F_{yr}]^T \tag{15}$$

Equations (16) to (17) represent the relationship between the horizontal force of the front and rear wheels and the sideslip angle. After obtaining the sideslip angle of the front and rear wheels, (18) to (19) can be used to obtain the steering angle of the front and rear wheels. The actual control parameter value can be obtained from the driving speed and the steering angle of the front and rear wheels.

$$\begin{cases} F_{yf} = -C_{\alpha f}\alpha_f, \text{ if } |F_{yf}| \le \frac{\mu N_f}{2} \\ F_{yf} = -\mu N_f \operatorname{sgn}(\alpha_f)\left[1-\frac{\mu N_f}{4C_{\alpha f}|\alpha_f|}\right], \text{ else} \end{cases} \tag{16}$$

$$\begin{cases} F_{yr} = -C_{\alpha r}\alpha_r, \text{ if } |F_{yr}| \le \frac{\mu N_f}{2} \\ F_{yr} = -\mu N_f \operatorname{sgn}(\alpha_r)\left[1-\frac{\mu N_f}{4C_{\alpha r}|\alpha_r|}\right], \text{ else} \end{cases} \tag{17}$$

$$\alpha_f = \frac{v+l_f r}{v_x} - \delta_f \tag{18}$$

$$\alpha_r = \frac{v-l_r r}{v_x} - \delta_r \tag{19}$$

1) Sliding mode with backstepping control

From the above derivation process, the theoretical speed and steering angle of the front and rear wheels of the robot can be obtained via (15), (18), and (19). However, there is still an error between the theoretical angle and the actual angle value. Since the accumulated error will cause the wheeled robot's moving trajectory to deviate from the desired path, the control system must be able to correct the error in real time. Therefore, the guidance control system adds backstepping sliding mode control to minimize errors.

Define the heading angle of the wheeled robot, the *x*-direction component of the position point, and the *y*-direction component of the point, which can be written as a matrix $x_r = [\psi_r \quad x_r \quad y_r]^T$. Sliding mode control is a knowledge of variable structure discipline, which mainly establishes a sliding mode surface function, which makes the system state approach a specific sliding mode surface and the control variables do not leave the sliding surface before approaching the target value. The sliding surface function is defined as: $S(\nabla_i) = k_1\nabla_1+k_2\nabla_2+....+k_{n-1}\nabla_{n-1}+\nabla_n = 0$, where "$\nabla$" is a user-defined observation variable, for example: The symbol "∇_1" is the error of the observation. The symbol "∇_2" is the error of the observation rate. k is a constant coefficient whose highest order coefficient is 1. The backstepping method can provide a recursive way to stabilize the system at the origin for strictly feedback type systems. Therefore, the differential to account for the error is defined as $\dot{\nabla}_1+\alpha$, where $\alpha = C_1\nabla_1$, and this term will maintain its stability in the subsequent derivation process. The error ∇_1 can be obtained by (20), then (21) and (22).

$$\nabla_1 = \begin{bmatrix} \nabla_{11} \\ \nabla_{12} \\ \nabla_{13} \end{bmatrix} = q - x_r \tag{20}$$

$$\nabla_2 = \begin{bmatrix} \nabla_{21} \\ \nabla_{22} \\ \nabla_{23} \end{bmatrix} = \dot{\nabla}_1+\alpha, \ \alpha = C_1\nabla_1 \tag{21}$$

$$\dot{\nabla}_2 = \ddot{\nabla}_1 + \dot{\alpha} = \ddot{q} - \ddot{x_r} + \dot{\alpha} \tag{22}$$

$$= \mathrm{A}(q,\ \dot{q}) + \mathrm{BU} + \rho - \ddot{x}_r + C_1\nabla_1$$

In order to make the error converge quickly, the Lyapunov function is defined, such as (23). Equation (24) is obtained after differentiating (23), and then the sliding mode plane is defined, as shown in (25). In order to make the guidance control system applicable to a wider controllable range, the backstepping control law is added. Therefore, the Lyapunov function of the second layer is defined, as in (26). Differentiating (26) yields (27). The control variables of the controller is selected, such as (28), and the sliding mode control must be satisfied when (28) is brought into (27), such as (29)–(34).

$$V_1 = \frac{1}{2}\nabla_1^2 = \frac{1}{2}\nabla_1^{\mathrm{T}}\nabla_1 \tag{23}$$

$$\dot{V}_1 = (\frac{1}{2}\nabla_1^2)' = \nabla_1\dot{\nabla}_1 = \nabla_1^{\mathrm{T}}\nabla_2 - C_1\nabla_1^{\mathrm{T}}\nabla_1 \tag{24}$$

$$S = k\nabla_1 + \nabla_2 \tag{25}$$

$$V_2 = V_1 + \frac{1}{2}S^2 = V_1 + \frac{1}{2}S^{\mathrm{T}}S \tag{26}$$

$$\begin{aligned}\dot{V}_2 &= (V_1 + \frac{1}{2}S^2)' = \dot{V}_1 + S\dot{S} \\ &= \left(\nabla_1^{\mathrm{T}}\nabla_2 - C_1\nabla_1^{\mathrm{T}}\nabla_1\right) + S^{\mathrm{T}}\left(k\dot{\nabla}_1 + \dot{\nabla}_2\right) \\ &= \left(\nabla_1^{\mathrm{T}}\nabla_2 - C_1\cdot\nabla_1^{\mathrm{T}}\nabla_1\right) + S^{\mathrm{T}}(k(\nabla_2 - C_1\nabla_1) \\ &+ \mathrm{A}(q,\ \dot{q}) + \mathrm{BU} + \rho - \ddot{x}_r + C_1\cdot\dot{\nabla}_1)\end{aligned} \tag{27}$$

$$\mathrm{U} = \mathrm{B}^{-1}[-k(\nabla_2 - C_1\nabla_1) - \mathrm{A} + \bar{\rho}\,\mathrm{sign}(S) - \dot{\alpha} + \ddot{x}_r - hS] \tag{28}$$

$$\begin{aligned}\dot{V}_2 &= \nabla_1^{\mathrm{T}}\nabla_2 - C_1\nabla_1^{\mathrm{T}}\nabla_1 - hS^{\mathrm{T}}S + [S^{\mathrm{T}}\rho - |S^{\mathrm{T}}|\bar{\rho}] \\ &\le \nabla_1^{\mathrm{T}}\nabla_2 - C_1\nabla_1^{\mathrm{T}}\nabla_1 - hS^{\mathrm{T}}S + |S^{\mathrm{T}}|(|\rho| - \bar{\rho})\end{aligned} \tag{29}$$

$$\le \nabla_1^{\mathrm{T}}\nabla_2 - C_1\nabla_1^{\mathrm{T}}\nabla_1 - hS^{\mathrm{T}}S$$

$$\mathrm{z} = [\nabla_{11}, \nabla_{12}, \nabla_{13}, \nabla_{21}, \nabla_{22}, \nabla_{23}]^{\mathrm{T}} \tag{30}$$

$$\mathrm{Q} = \begin{bmatrix} C_1 + hk^2 & 0 & 0 & 2hk & 0 & 0 \\ 0 & C_1 + hk^2 & 0 & 0 & 2hk & 0 \\ 0 & 0 & C_1 + hk^2 & 0 & 0 & 2hk \\ -1 & 0 & 0 & h & 0 & 0 \\ 0 & -1 & 0 & 0 & h & 0 \\ 0 & 0 & -1 & 0 & 0 & h \end{bmatrix} \tag{31}$$

$$\mathrm{z}^{\mathrm{T}}\mathrm{Qz} = -\nabla_1^{\mathrm{T}}\nabla_2 + C_1\nabla_1^{\mathrm{T}}\nabla_1 + hS^{\mathrm{T}}S \tag{32}$$

$$\dot{V}_2 \le -\mathrm{z}^{\mathrm{T}}\mathrm{Qz} \tag{33}$$

$$\begin{aligned}|\mathrm{Q}| &= \left(h^2k^2 + 2hk + C_1h\right)\times \\ &\left(C_1^2h^2 + 2C_1h^3k^2 + 4C_1kh^2 + \right. \\ &\left. h^4k^4 + 4h^3k^3 + 4h^2k^2\right)\end{aligned} \tag{34}$$

When the parameters C_1, h, k are selected to be greater than zero, it can be known through (31) that the determinant Q value must be positive, then Q is a positive definite matrix. In (32), the matrix $\mathrm{z}^{\mathrm{T}}\mathrm{Qz}$ is positive, and $\dot{V}_2$ is guaranteed to be negative definite through (33). Then, take the parameter C_1 into (24) to obtain $\dot{V}_1<0$. In (23) and (27), the squared error is positive definite since the sign of the highest order term is positive, i.e. $V_1 > 0$ and $V_2 > 0$. Therefore, V_2 is positive definite, the error will gradually converge to zero, which proves the stability of the control system.

III. Simulation and Experiment Results

A. Parameter Setup

Before the simulation, some basic parameters of the robot and the conditions of the field must be set, including the basic size, weight, steering rigidity and friction force (see Table I).

TABLE I
PARAMETERS SETUP

Symbol	Value	Symbol	Value
m	440 Kg	C_{α_f}	765.79 (N/rad)
l	2.4 m	C_{α_r}	765.79 (N/rad)
w	1.06 m	μ	0.45
l_f	0.8 m	C_1	2
l_r	0.8 m	h	3
I_z	121 Kg · m^2	k	1

In order to make the control system stable, the parameters C_1, h, k are set to 2, 3, and 1, respectively.

Then measure the size of the field, and use the GNSS-RTK receiver to locate the start point (p_1, p_7) and end point (p_3, p_9) of each row of fields w_f, field width w_f, ditch width w_p, distance from the center of the field to the center of the wheel w_w, farmland boundary w_b and minimum turning radius r_m, etc., as shown in Fig. 3. The parameter values are shown in Table 3. Then a fast path planning method can be carried out. Calculate the minimum turning radius by calculating the steering angle of the front and rear wheels of the robot to obtain r_m, and indirectly calculate the three points (p_4, p_5, p_6) outside the field border. Finally, a cubic spline interpolation method is used to plan the desired path and observe whether the movement trajectories of the left and right wheels are within the expected movement range.

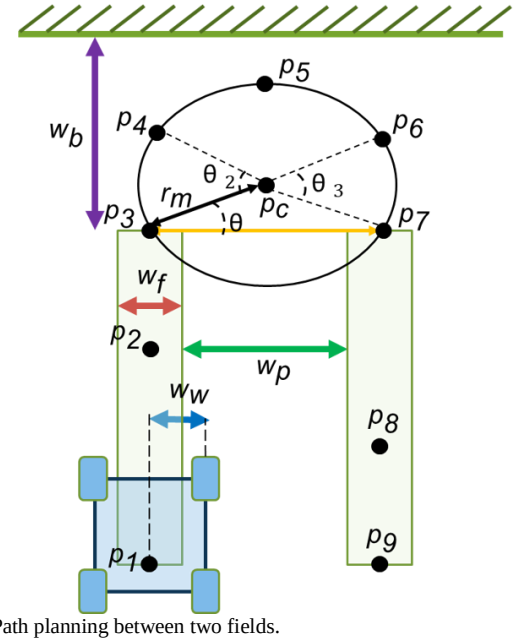


Fig. 3. Path planning between two fields.

TABLE II
SIZE OF FIELD SETTING

Symbol	Value
w_f	0.8 m

w_p	2.797 m
w_w	0.13 m
w_b	5 m
r_m	2.5 m

B. Simulation Process

The simulation process is shown in Fig. 4. First, measure the longitude and latitude of the anchor point of the field, and then perform coordinate transformation. These anchor point locations based on a 2D planar coordinate system can be used to generate cubic splines (point-to-point trajectories) and combine them into a path. Next, the dynamic model of the robot is established and the backstepping sliding mode controller is assisted to estimate the control amount and reduce the motion error of the robot. Finally, verify the control performance on the created farm map.

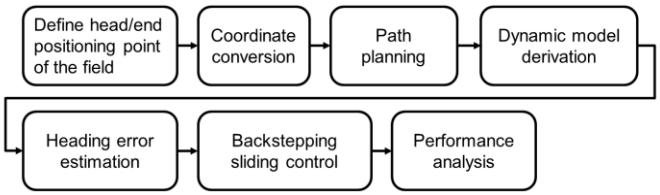


Fig. 4. Simulation process for robot navigation.

C. Simulation Results and Discussion

First define the start point p_1 and end point p_3 of the first row of fields, and the start point p_7 and end point p_9 of the second row of fields, and set the remaining target points according to Table III.

TABLE III
TARGET POINT SETTING FOR TURNING PATH

Description	Measurement
$\angle p_c p_3 p_7$	$\theta=\cos^{-1}(\frac{(w_f+w_p)/2}{r_m})$
The *x*-component of the center of the minimum turning radius	$p_{cx}=p_{3x}+r_m\cos\theta$
The *y*-component of the center of the minimum turning radius	$p_{cy}=p_{3y}+r_m\sin\theta$
$\angle p_3 p_c p_4$	$\theta_2=\frac{\theta+90^\circ}{2}+90^\circ$
$\angle p_6 p_c p_7$	$\theta_3=90^\circ-\frac{\theta+90^\circ}{2}$
x–component of interpolation point p_4	$p_{4x}=p_{cx}+r_m\cos\theta_2$
y–component of interpolation point p_4	$p_{4y}=p_{cy}+r_m sin\theta_2$
x–component of interpolation point p_5	$p_{5x}=p_{cx}$
y–component of interpolation point p_5	$p_{5y}=p_{cy}+r_m$
x–component of interpolation point p_6	$p_{6x}=p_{cx}+r_m\cos\theta_3$
y–component of interpolation point p_6	$p_{6y}=p_{cy}+r_m\sin\theta_3$

Then use cubic splines to plan the desired route, which enables the robot to move the shortest route without destroying the furrow border. The coefficients of the cubic equation for each line segment are shown in Table IV and Table V. The numbers “1”, “2”, “3”, ..., “8” represent the segment numbers on the path.

TABLE IV
COEFFICIENT COMBINATION FOR CUBIC SPLINE INTERPOLATION (x_i)

	1	2	3	4	5	6	7	8
a_i	0.1304	–0.3913	1.0237	–0.6717	–0.8552	1.1632	–0.4574	0.1525
b_i	–0.1304	0.2609	–0.9131	2.1581	0.1432	–2.4224	1.0673	–0.3049
c_i	0.0000	0.1304	–0.5218	0.7233	3.0246	0.7453	–0.6099	0.1525
d_i	4.2990	4.2990	4.2990	3.8879	6.0977	8.4102	7.8963	7.8963

TABLE V
COEFFICIENT COMBINATION FOR CUBIC SPLINE INTERPOLATION (y_i)

	1	2	3	4	5	6	7	8
a_i	–1.6415	0.4954	–0.3389	–0.2553	0.2593	0.3331	–0.4931	1.6407
b_i	3.8561	–1.0684	0.4177	–0.5989	–1.3647	–0.5868	0.4124	–1.0669
c_i	0.0000	2.7876	2.1369	1.9556	–0.0080	–1.9595	–2.1339	–2.7884
d_i	0.2982	2.5127	4.7273	6.9429	8.0444	6.9310	4.7178	2.5032

The black line in Fig. 5 represents the desired path. Since the actual movement of the robot will deviate from the original path due to the influence of the frictional force of the wheels, after using the backstepping sliding mode control, the corrected trajectory of the robot can be obtained, as shown in the blue dotted line in Fig. 5. The gray solid line depicts the movement trajectory of the left and right wheels. It can be seen that the trajectory does not exceed the set range, and the furrow border can be kept from being damaged by the wheels. Figure 6 shows the target heading angle and the actual heading angle of the robot.

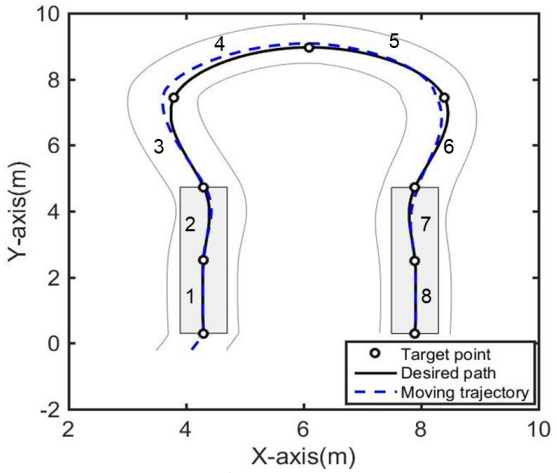


Fig. 5. The moving trajectory of the robot with back-stepping sliding mode control. The gray block represents the contour area of field.

On the contrary, when the robot system does not assist the back-stepping sliding mode control, there is an error between the actual movement trajectory of the robot and the pre-planned

path, which causes the robot to move beyond the expected range and causes the wheels to damage the edge of the field.

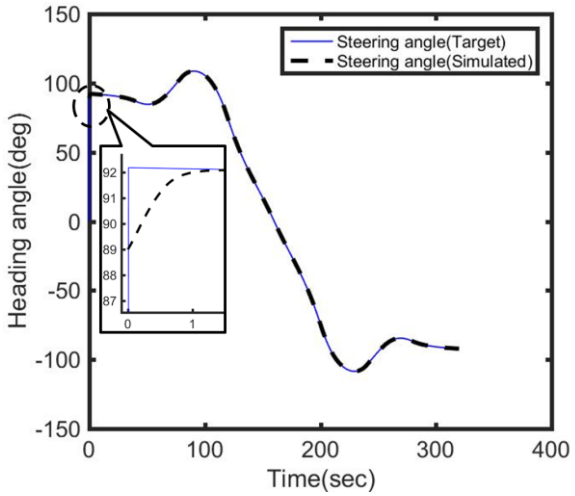


Fig. 6. The comparison result between the target heading angle and the actual heading angle of the robot.

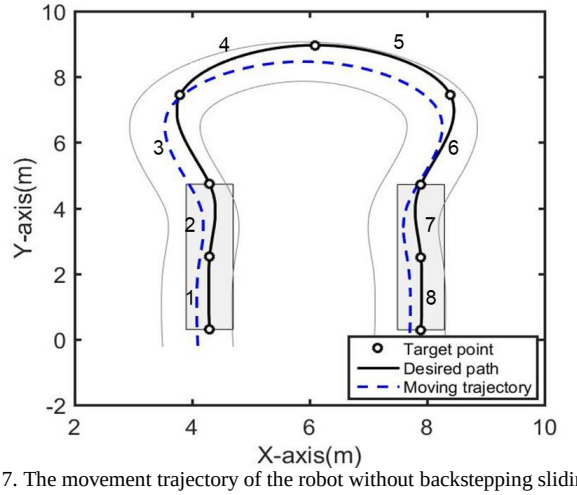


Fig. 7. The movement trajectory of the robot without backstepping sliding mode control.

It can be observed from Fig. 8 that the robot walks in the first row of fields ($p_1 \sim p_3$ in Fig. 3) from 0 to 80 seconds. At this time, the trajectories are kept outside the border of the field ($|w_{dl}| > w_f$ and $|w_{dr}| > w_f$). When the robot moves, its wheels will not damage the field. During 80–240 seconds, the robot U-turns from the first row of fields to the second row of fields ($p_3 \sim p_7$ in Fig. 3). From the results in Fig. 5, it can be observed that both $|w_{dl}|$ and $|w_{dr}|$ are greater than w_d. Between 240 and 320 seconds, the robot moves to the second row of fields ($p_7 \sim p_9$ in Fig. 3). At this time, it can also be seen that $|w_{dl}|$ and $|w_{dr}|$ are both greater than w_f .

In (28), k is a constant coefficient and the highest order coefficient of the controller is 1. Since C_1 is a weight coefficient, it determines the master-slave control relationship between the results of "speed and angular velocity error" and "position and heading angle error". Figures 9(a) and 9(b) show

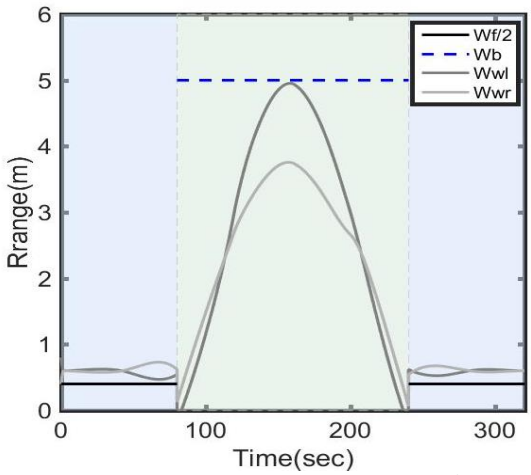
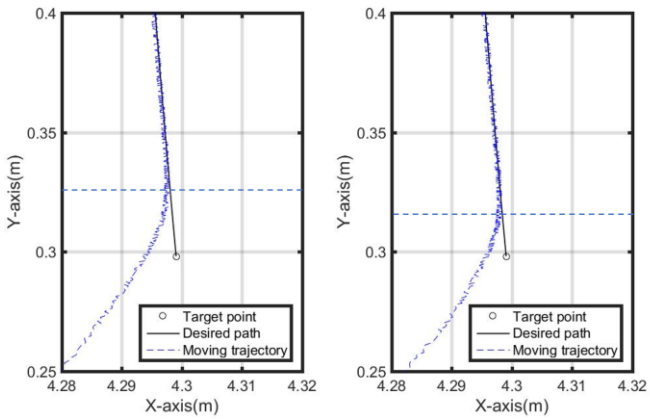


Fig. 8. The relationship between the movement trajectory of the robot and the range of the driving lane.

the tracking results of the desired target points where C_1 is 2 and 10, respectively. The figure shows that a higher value of C_1 means that the control system is dominated by position control, so the robot will be closer to the target position, however, the chattering phenomenon of the controller also increases. Therefore, the selection principle of the coefficient is obtained according to the experience after the actual test.



(a) (b)

Fig. 9. C_1 coefficient versus robot tracking results. (a) C_1 = 2, (b) C_1 = 10.

Since the lyapunov function can guarantee that when the time is infinite, S is 0. In practice, the system must reach a stable point in a finite time. Assuming $\dot{V} \le hV^{1/2}$, if h is large, the controller will reach a stable point faster. Figure 10 shows the movement trajectories generated by different h values. It can be seen from this figure that when the coefficient h is large (Fig. 10(b)), the trajectory is more similar to a smooth surface, so the control system can quickly correct to the desired position.

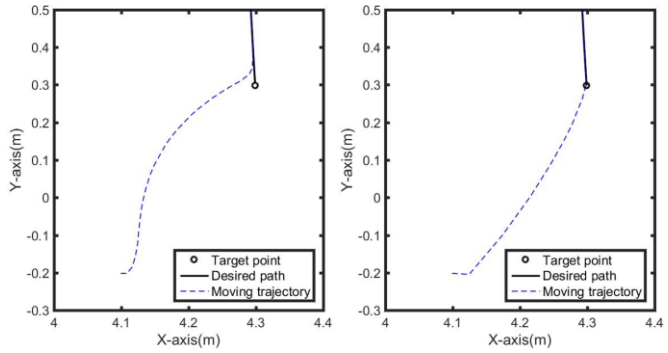


Fig. 10. *h* coefficient versus robot tracking results. (a) *h* = 1, (b) *h* = 7.

D. Experiment Results

The experimental site is located in front of the Agricultural Machinery Center of National Pingtung University of Science and Technology (Longitude: 120.60611° ; Latitude: 22.646490°). The climate is cloudy. The RTK-GNSS receiver is used to set four positions, such as the solid dots in Fig. 11, which are regarded as the endpoints of the two furrows. The four positions were expanded into nine positions according to Table 3. Eight-segment paths are obtained after cubic spline interpolation. The coefficient C_1 and the value of h will affect the stability of the robot's navigation process. As a rule of thumb, the coefficients C_1 and h are usually set to 2 and between 2 and 4, respectively. The position of the robot is provided by a Global Navigation Satellite System (GNSS-RTK) receiver. When the receiver obtains a fixed solution, its positioning error is very small. The results show that the control system can correct the position error under the path planned by cubic spline interpolation. The robot can move according to the planned path, but the correction process will be affected by the time delay and the chattering of the controller, resulting in the movement trajectory is not very smooth.

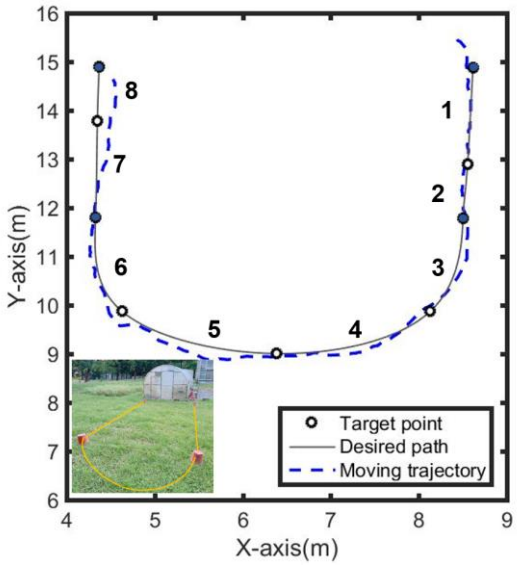


Fig. 11. Desired path versus moving trajectory of robot

IV. Conclusions

In this study, the cubic spline method has been successfully implemented on an autonomous robot for path planning. The user can select an appropriate number of interpolation points according to the size of the field to reduce the turning radius of the robot when changing borders. With the addition of back-step sliding mode control to the robot guidance system, it has the ability to correct the positioning error on the planned path. As long as the user obtains a small amount of positioning data in the farm, the positioning system can plan a suitable positioning point, so that the robot can move smoothly towards the positioning point and reduce the risk of unintentionally colliding with crops.

References

[1] C. H. Chiang, "Trajectory prediction and tracking with rear-wheel secondary compensation," M.S. thesis, Dept. ME., NYCU, Hsinchu city, Taiwan, Jun. 2009.

[2] C. F. Lin, A. G. Ulsoy, and D. J. LeBlanc, "Vehicle dynamics and external disturbance estimation for vehicle path prediction," *IEEE Trans. Control Systems Technology*, vol. 8, no. 3, May. 2000.

[3] J. Lian, W. Yu, K. Xiao, and W. Liu, "Cubic spline interpolation-based robot path planning using a chaotic adaptive particle swarm optimization algorithm," *Math. Probl. Eng.*, vol. 2020, no. 1849240, pp. 20–21, Feb. 2020. DOI: 10.1155/2020/1849240.

[4] U. Y. Huh and S. R. Chang, "AG2 continuous path-smoothing algorithm using modified quadratic polynomial interpolation", *Int. J. Adv. Robot.*, vol. 11, no. 195708668, pp. 1−11, Jan. 2014. DOI: 10.5772/57340.

[5] J. C. Chien, C. L. Chang, and C. C. Yu, "Fast path planning method for agricultural robot automatic guidance based on cubic spline interpolation in strip farming," presented at the *2022 International Conference on System Science and Engineering*, Taichung, Taiwan, May 26–29, 2022.

[6] S. Bouabdallah and R. Siegwart, "Backstepping and sliding-mode techniques applied to an indoor micro quadrotor," In *Proc. 2005 IEEE International Conference on Robotics and Automation Barcelona*, Spain, vol. 560, no. 1570447, pp. 2247–2252, Apr. 2005. DOI: 10.1109/ROBOT.2005.1570447.

[7] N. Chen, F. Song, G. Li, X. Sun, and C. Ai, "An adaptive sliding mode backstepping control for the mobile manipulator with nonholonomic constraints," *Commun. Nonlinear Sci. Numer. Simul.*, vol. 18, no. 10, pp. 2885–2899, Oct. 2013. DOI: 10.1016/j.cnsns.2013.02.002.

[8] Y. HAN, Y. Cheng, and G. Xu, "Trajectory tracking control of AGV based on sliding mode control with the improved reaching law," *IEEE Access*, vol. 17, no. 18484662, pp. 20748–20755, Feb. 2019. DOI: 10.1109/ACCESS.2019.2897985.

[9] P. T. H. Sen, N. Q. Minh, D. T. T. Anh, D. T. T. Anh, and P. X. Minh, "A new tracking control algorithm for a wheeled mobile robot based on backstepping and hierarchical sliding mode techniques," In *2019 First International Symposium on Instrumentation, Control, Artificial Intelligence, and Robotics (ICA-SYMP)*, Bangkok, Thailand, 2019, pp. 25–28.

[10] X. Wu, P. Jin, T. Zou, Z. Qi, H. Xiao, and P. Lou1, "Backstepping trajectory tracking based on fuzzy Sliding mode control for differential mobile robots," *Journal of Intelligent & Robotic Sys.*, vol. 96, no. 4, pp. 109–121, Jan. 2019. DOI: 10.1007/s10846-019-00980-9.

[11] B. S. Park, S. J. Yoo, and J. B. Park, "Adaptive neural sliding mode control of nonholonomic wheeled mobile robots with model uncertainty" *IEEE Trans. Control Syst. Technol*, vol. 17, no. 1, pp. 207–214, Jan. 2009. DOI: 10.1109/TCST.2008.922584.

[12] H. Mehrjerdi, and M. Saad, "Chattering reduction on the dynamic tracking control of a nonholonomic mobile robot using exponential sliding mode," In *Proc. IMechE*, vol. 225, no. 17, pp. 875–886, 2011. DOI: 10.1177/2041304110394566.

[13] B. B. Mevo, M. R. Saad, and R. Fareh, "Adaptive sliding mode control of wheeled mobile robot with nonlinear model and uncertainties," *IEEE, CCECE*, vol. 6, no. 18060667, pp. 1–4, May. 2018. DOI: 10.1109/CCECE.2018.8447570.

[14] Y. Koubaa, M. Boukattaya, and T. Dammak, "Adaptive sliding-mode dynamic control for path tracking of nonholonomic wheeled mobile

robot," *J. Automation & Systems Engineering*, vol. 9, no. 2, pp. 119–131, Feb. 2015.
[15] B. Moudoud, H. Aissaoui, and D. Mohammed, "Robust trajectory tracking control based on sliding mode of differential driving four-wheeled mobile robot," In *2020 IEEE 6th International Conference on Optimization and Applications (ICOA)*, Beni Mellal, Morocco, 2020.
[16] B. Moudoud, H. Aissaoui and M. Diany, "Robust adaptive trajectory tracking control based on sliding mode of electrical wheeled mobile robot", *Int. J. Mech. Eng. Robot. Res.*, vol. 10, no. 9, pp. 505–509, Sep. 2021.
[17] W. Dixon, M. D. Queiroz, D. Dawson, and T. Flynn, "Adaptive tracking and regulation of a wheeled mobile robot with controller/update law modularity" *IEEE Tran. Control Systems Technology*, vol. 12, no. 1, pp. 138–147, 2004.
[18] N. Chen, F. Song, G. Li, X. Sun, and C. Ai, "An adaptive sliding mode backstepping control for the mobile manipulator with nonholonomic constraints," *Commun. Nonlinear Sci. Numer. Simul.*, vol. 18, no. 10, pp. 2885–2899, 2013.
[19] Z. Shuai, H. Zhang, J. Wang, J. Li, and M. Ouyang, "Lateral motion control for four-wheel-independent-drive electric vehicles using optimal torque allocation and dynamic message priority scheduling," *Control Eng. Pract.*, vol. 24, pp. 55–66, 2014.
[20] J. Zhang, D. Chen, S. Wang, X. Hu, and D. Wang, "Design and experiment of four-wheel independent steering driving and control system for agricultural wheeled robot," *Trans. Chin. Soc. Agric. Eng.*, vol. 31, no. 18, pp. 63–70, 2015.
[21] C. L. Chang and G. B. Song, "Vision-based guidance control scheme for autonomous navigation of 4WDS agricultural vehicle in a narrow space environment", In *Proc. ASABE Annual International Meeting*, no.170121, Spokane, WA, USA, Jul. 2017. DOI: 10.13031/aim.201701121.
[22] R. Dhaouadi and A. Abu Hatab, "Dynamic modelling of differential-drive mobile robots using Lagrange and Newton-Euler methodologies: A unified framework," *Adv. Robot. Autom.*, vol. 2, no. 2, pp. 1–7, 2013. DOI: 10.4172/2168-9695.1000107.
[23] P. Bennett, "The NMEA FAQ (Fragen und Antworten zu NMEA)," Version 6.1, Sep. 1997. Archived from the original on 2014–02-15. Retrieved 2022–01–21.
[24] P. T. Y. Shih, "TWD97 and WGS84, datum or map projection?," *J. Cadastral Survey*, vol. 39, pp. 1–12, Sep. 2020.

Appendix

l_f	Distance from center of gravity to front wheel
l_r	Distance from center of gravity to rear wheel
v_x	Longitudinal speed
v_y	Lateral speed
γ	Yaw rate
F_{xf}	Longitudinal force of front wheel
F_{xr}	Longitudinal force of rear wheel
F_{yf}	Lateral force of front wheel
F_{yr}	Lateral force of rear wheel
v_f	speed of front wheel
v_r	speed of rear wheel
δ_f	Steering angle of front wheels
δ_r	Steering angle of rear wheels
α_f	Side slip angle of the front wheel
α_r	Side slip angle of the rear wheel
Iz	Moment of inertia
m	Robot quality
A_{cc}	Robot acceleration
β	Robot side slip angle

Jen C. Chien receives the B.S. and M.S degree in Vehicle Engineering from the National Pingtung University of Science and Technology, Pingtung county, Taiwan, in 2020 and 2022, respectively. His current research interests are in the areas of agricultural robot, automatic guidance, and optimal control.

Chung L. Chang received the B.S. degree in Industrial Education from National Taiwan Normal University, Taipei, Taiwan, in 1998. He has M.S. and the Ph.D. degrees in Electrical Engineering from the National Cheng Kung University, Tainan, Taiwan, in 2004 and 2008, respectively. Since 2009, he has been with the Biomechatronics Engineering at National Pingtung University of Science and Technology and is currently a Distinguished Professor. His main research interests include farm robotics and bionics, protected agriculture, and artificial intelligent of things.

Chang C. Yu received the Ph.D. degree from National Cheng Kung University, Taiwan, in 2008. Currently, he is an Associate Professor of Vehicle Engineering at National Pingtung University of Science and Technology. His research interests include servo motor control, power electronics, and vehicle control system.