Preprints, 4th IFAC Conference on
Advances in Proportional-Integral-Derivative Control
Almería, Spain | June 12-14, 2024

A PID-based control architecture for mobile robot path planning in greenhouses

Fernando Cañadas-Aránega * **José C. Moreno** *
José L. Blanco-Claraco **

* *University of Almería, Department of Informatics, CIESOL, ceiA3, Almería, 04120, Spain (e-mail: fernando.ca@ual.es, jcmoreno@ual.es)*
** *University of Almería, Department of Engineering, ceiA3, Almería 04120, Spain (email: jlblanco@ual.es)*

Abstract: This article presents a control strategy for an Ackermann mobile robot designed for greenhouse operation, based on a cascade control mechanism. It controls both, (i) an inner loop, based on PID, for two motors, and (ii) a classic Pure Pursuit control for the outer loop. In addition, a feedforward control strategy is proposed to address the issue of the natural variable terrain in greenhouses in southern Spain. Path following will be evaluated using a 2D model of the experimental greenhouse to obtain coordinates that will be assessed in simulation. Open-loop tests were conducted on a real agricultural robot prototype, AGRICOBIOT II, designed and built at the University of Almeria.

Keywords: Greenhouse, mobile robot, PID Control, Cascade Control, Feedforward Control, Pure pursuit

1. INTRODUCTION

The control and autonomous navigation of mobile robots continue to be highly active areas of research today. In the literature, systems developed with classical control techniques to address such problems can be found (Jalving, 1994). Currently, mobile robots are becoming increasingly essential tools for humans. Given their popularity, their tasks become more challenging, often involving collaboration with humans or complex environments. To increase reliability and due to its simple and efficient implementation, the PID control strategy has gained popularity as one of the most widely used low-level control formulations in different types of applications, from robotic systems to the processing industry. However, the need for an accurate model to describe the system's dynamics, in order to use classical methods like (Ziegler and Nichols, 1942), is a significant drawback of these methods. Precisely, to control a mobile robot, this implies that any change in the payload or in the robot itself requires both a model update and a subsequent update of the controller parameters, a time-consuming process that increases costs.

Focusing on agriculture, planning, and controlling the different tasks that mobile robots can perform becomes a real challenge, given the complexity of the terrain. Moreover, this complexity is considerably increased in greenhouses as it is a changing environment that requires excellent precision. Much research has been carried out in greenhouses focusing on automation, from PID-based temperature control algorithms (Liu et al., 2023) to the elaboration of maps that could be used to test different control techniques implemented in autonomous robots (Cañadas-Aránega et al., 2024), among other applications for robots in greenhouses (Sánchez-Molina et al., 2024). For example, at the University of Almeria, a robot was developed for spraying tasks in greenhouses. This paper deals with the control problem for the spraying operation and for the navigation of the robot. The PID has also been used as a trajectory tracking navigation system due to its robustness and simplicity of implementation (Normey-Rico et al., 2001).

The robotized tasks require high precision and speed, often leaving little room for errors during operation. Therefore, it is important to have software tools that allow the simulation of robot behavior. Various robot simulation tools have emerged, allowing the prediction of the actual robot behavior. Additionally, these tools provide valuable insights into problems that could arise in the real world (Chen et al., 2009). The environment Simulink Matlab (2012) has been used in this work.

This work proposes a control strategy to manage the trajectory of an Ackermann mobile robot inside greenhouses. A cascade control mechanism is outlined with an inner loop dedicated to the robot actual motors, utilizing classical control techniques, and an outer loop for the robot position within the greenhouse coordinates. Additionally, a novel feedforward scheme is applied in the inner loop of a cascade control to address and reduce the control effort experienced by the robot when navigating the uneven terrain of a typical Mediterranean greenhouse.

⋆ This work has been partially financed by the 'CyberGreen' Project, PID2021-122560OB-I00, and the 'AgroConnect.es' infrastructure used to carry out this research, grant EQC2019-006658-P, both funded by MCIN/AEI/ 10.13039/501100011033 and by ERDF A way to make Europe.

2. GREENHOUSE AND AGRICOBIOT II MOBILE PLATFORM DESCRIPTION

On the one hand, the trials will occur at the facilities owned by the Andalusian Institute for Agricultural, Fisheries, and Food Research and Training (IFAPA) in the Municipal District of La Cañada de San Urbano, Almería. The location is situated at 36°50' N and 2°24' W, with an elevation of 3 meters above sea level (refer to Fig. 1) (Moreno Úbeda et al., 2022). The greenhouse is in one of the most common styles in the region (Almería's "raspa y amagado"), expands over 1850 square meters. The primary thoroughfare within this greenhouse is a central pathway measuring 2 meters in width. Radiating from this central aisle are narrower secondary pathways, each with a width of just one meter, facilitating the seamless movement of mobile robotic units. This paper uses a 2D model of this greenhouse to obtain accurate coordinates and assess the trajectories that the robot would follow in reality.

Fig. 1. Overview of the greenhouse used for the experimental validation

On the other hand, open-loop tests to carry out the control of the inner loop of the robot, upon which the present article is based, were performed on the prototype AGRICOBIOT II (Fig. 2), an Ackermann-type robot designed and manufactured at the University of Almería, equipped with the necessary technology to enable future autonomous navigation.

Fig. 2. AGRICOBIOT II robot

3. CASCADE CONTROL - MODELING AND CONTROL OF THE INNER LOOP

When the robot moves along a circular arc for a fixed steering wheel angle, the front wheels pivot in a plane perpendicular to the ground while the drive wheels move in parallel. This means the front wheels must follow a longer trajectory to reach a destination and turn faster than the rear wheel. Therefore, the steering motors, as well as the traction motors, are modeled and controlled independently. This is a behaviour of an Ackermann-type configuration, whose kinematic equations can be found in (Corke et al., 2011).

3.1 Motor Modeling

The inner loop is composed of two motors, one for traction and another for steering control. The traction motor has a mechanical transmission ratio of 1:0.00067, consisting of worm gear, spur gears, and chains, while the steering motor has a ratio o of 1:0.0075 through a direct worm gear.

After identifying the mechanical aspects, the next step is to determine the non-linearities in the motor behavior. First, open-loop tests are conducted on the actual prototype to determine saturation limits and dead zones. The saturation limits of the traction motor are between -23.7 and 23.5 V, with a dead zone between -1.7 and 1.5 V (see Fig. 3). The steering motor has similar saturation limits, ranging from -23.78 to -23.8 V, with a reduced dead zone between -0.89 and 1.1 V.

Then, three random tests were conducted with the robot on the ground, using an ascending/descending train of steps as input to each of the motors to characterize the system. The most significant test was used for modeling, while one of the remaining two was used to validate the results. In the case of steering, an integrator is added to the velocity model as desired position control (Zeng and Hemami, 1997).

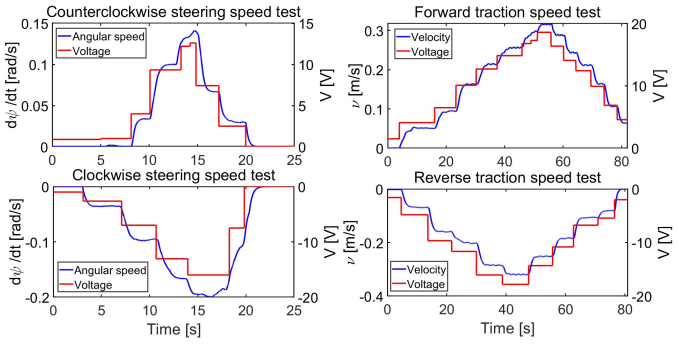

Fig. 3. Open-loop tests for the steering and traction motors

It is observed how the system can be approximated to a first order throughout the entire range. Finally, using the reaction curve method (Åström and Hägglund, 2006), gains and time constants are obtained at each step for both forward and backward movements. The gain and time constant of the approximate transfer function, tf, are obtained as the average of the gains (between 0.012 and 0.02 $[ms^{-1} \cdot V^{-1}]$ for traction and between 0.014 and 0.018 $[m \cdot s^{-1} \cdot V^{-1}]$ for steering) and time constants (between 1.2 and 1.45 [s] for traction and 0.25 and 0.45 [s] for steering), respectively, identified throughout the test. This method yields the traction model as $G_T(s)$ (Eq. (1)) and the steering model as $G_S(s)$ (Eq. (2)) of the robot:

$$G_T(s) = \frac{0.0187}{1.3375s + 1} \left[\frac{m/s}{V}\right], \quad (1)$$

$$G_S(s) = \frac{0.01515}{s(0.38s + 1)} \left[\frac{rad}{V}\right]. \quad (2)$$

Finally, to consider these models as adequate, validation is shown in one of the other conducted tests in Fig. 4. Since a position control will be carried out, the validation is performed on the model obtained with integrator Eq. (2).

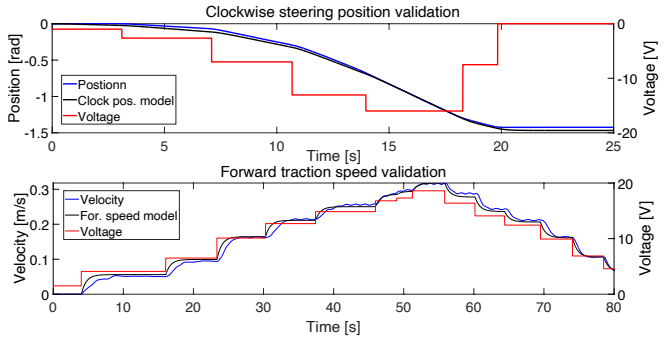

Fig. 4. Linear Dynamics validation

3.2 Motor control

In Eq. (1) and (2), it can be observed that the gains are practically equivalent. However, in the case of the time constant, the steering angular velocity is 72% faster than the forward linear velocity. This should be taken into account when designing the control law. Since motors have been independently modeled as SISO first-order systems, a PID control will be designed to seek a compromise between trajectory tracking and disturbance rejection (Åström and Hägglund, 2006).

For the traction motor, it will be established that the closed-loop motor behaves like a first-order system and reaches the reference without error in a steady state in half the time of the open-loop time constant. For the PID controller, the derivative action will be eliminated to avoid increased sensor noise and interference with other electronic components of the robot (Liao et al., 2019). Therefore, a PI controller will be used for reference tracking and disturbance rejection. Similarly, an Anti-Windup (AW) mechanism is added to the control scheme considering motor saturation. Parameters for PI with AW are computed using the pole-zero cancelation method (Åström and Hägglund, 2006). Values for the proportional action (K_{pT}) of 110.72 $[V{\cdot}m^{-1}{\cdot}s^{-1}]$, for the integral action (τ_{iT}) of 1.33 $[s]$, and for the tracking constant of 0.86 $[s]$ are obtained.

Regarding the steering motor, following the same approach as the traction motor, a PI controller is designed for reference tracking and disturbance rejection. The system must have no error in a steady state, as it could experience a slight permanent deviation in the trajectory, potentially leading to a collision. Considering that the time constant of this system is 72% faster than that of the traction motor, the specification should respond as an overdamped system with poles placed so that its average is half of the open-loop time constant ($\tau_1 = 0.55$ $[s]$ and $\tau_2 = 0.75$ $[s]$). In this way, both closed-loop time constants will respond at the same speed, improving coordination during navigation. In this case, in order to realize the PI controller (Yamamoto et al., 1996), the dynamics of the plant pole will be ignored in designing the controller. Through adjustment of pole assignment and the addition of AW adjustment, the values for the proportional action (K_{pS}) of 351.33 $[V \cdot rad^{-1}]$, for the integral action (τ_{iS}) of 0.5 $[s]$, and for the tracking constant of 0.86 $[s]$ are obtained. Finally, a filter must be applied to the reference to cancel the effect of the resulting zero in the final closed-loop function, reducing oscillations at the expense of resulting in a slower system (Åström and Hägglund, 2006).

Finally, the controllers are evaluated by providing set points of velocity and position for the motors. They are validated only in simulation, since the physical prototype is not yet fully operational. The control results in the simulation are shown in Fig. 5.

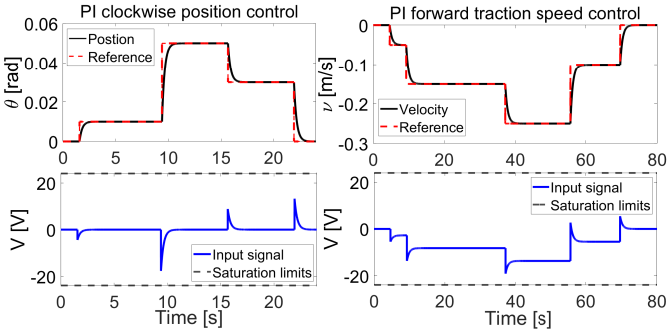

Fig. 5. Control of steering and traction motors

As can be observed, the control achieves reference tracking with zero error and a correct time constant for differences in changes in the position and velocity of the motors. It is emphasized that, although not visible in the control signal given the scale, the dead zone of the motors has been taken into account.

4. CASCADE CONTROL - EXTERNAL LOOP

With the internal loop controlled, the next step is to control the external loop focused on guiding the robot to coordinates (x, y) within a space. Under Jacobian assumptions, the kinematic equation for an Ackermann configuration is determined, obtaining the directional and positional vector in space to a coordinate origin (Baturone, 2005). Using the values of these parameters, a Pure Pursuit control and an Anti-Windup strategy will be designed.

4.1 Kinematic model

Focusing on a 4-wheel Ackermann-type robot, the complexity of analysis increases considerably, since it has various degrees of freedom. In this work, following common practices (Macenski et al., 2023; Blanco-Claraco et al., 2023), the robot is simplified as a single rigid rigid planar body. With the considerations mentioned earlier, the equivalent model will be reduced to that of a bicycle (Corke et al., 2011), as shown in Fig. 6.

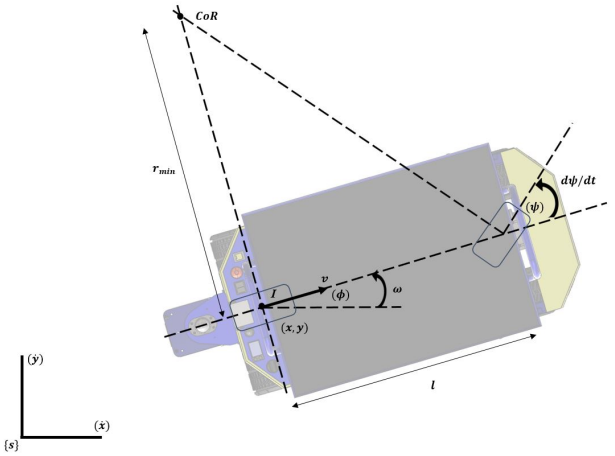

Fig. 6. Equivalent kinematic model of robot prototype AGRICOBIOT II

Therefore, the resulting kinematic equation (Eq. 3) is:

$$\begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{\phi} \end{bmatrix} = \begin{bmatrix} cos(\phi) & 0 \\ sin(\phi) & 0 \\ 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} v \\ w \end{bmatrix}, \tag{3}$$

where the linear velocity v, for the current prototype configuration, has a maximum and minimum active range of $[-0.45, 0.45]$ $m \cdot s^{-1}$, the angular speed of the robot ω is in $[-0.27, 0.27]$ $rad \cdot s^{-1}$, and the steering angle ψ has a range of $[-35, 35]$ degrees (physical turning limit). This steering angle is defined by the direction measurement, which is transformed to the origin as $\phi = v/L \cdot \tan(\psi)$.

4.2 Classical Pure Pursuit control

The pure pursuit algorithm serves as a geometric approach for tracking paths. This work refrains from delving into the internal dynamic model of the vehicle, instead directing our attention solely to its overall velocity and heading alterations. As a result, a kinematic model (Kong et al., 2015), rooted in the motion of the vehicle, is chosen for position updates. The pure pursuit algorithm generates the steering angle necessary to return the vehicle to the reference path. Fig. 7 illustrates the classical frame of the pure pursuit algorithm. Initially, since drivers typically look forward while driving, the pure pursuit algorithm calculates and defines the look-ahead point on the reference path using the look-ahead distance and the current actual position of the vehicle.

Consider the problem of moving toward a goal point (x', y') in the plane. On the one hand, it will control the robot velocity to be proportional to its distance from the goal, as observed in the Eq. (4):

$$\nu' = K_\nu \sqrt{(x' - x)^2 + (y' - y)^2}. \tag{4}$$

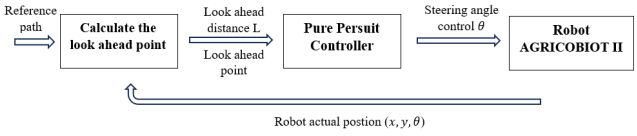

Fig. 7. Algorithm Pure Pursuit

In this paper the value of K_ν is calculated by seeking to minimize the error in the greenhouse coordinate. Taking into account motor dynamics, experiments are carried out with different values of K_ν (Wang et al., 2020), resulting in an optimal value of 0.045 $[s^{-1}]$.

On the other hand, Fig. 8 shows the schematic of the pure pursuit algorithm, with each variable marked. To obtain the output of the steering angle θ, the point of view and the position of the vehicle (rear wheel) are connected by a straight line. The angle between the line and the vehicle body is set as β, which is known as the angle of view of the distance.

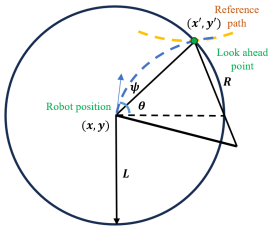

Fig. 8. Schematic of the pure pursuit algorithm

where L is the distance to the point, R is the turning radius, ψ is the turning angle, and θ is the orientation angle. To head towards the goal, it is necessary to calculate the angle relative to a local origin Eq. (5):

$$\theta' = \tan^{-1} \frac{y' - y}{x' - x}. \tag{5}$$

In (Snider et al., 2009), an analysis is presented demonstrating that pure pursuit, proportional to the steering angle, acts on a transverse error at a certain look-ahead distance in front of the vehicle Eq. (4) and (5), with a gain K_p of $2/L^2$. In this case, colocating the control point with the steered front wheels allows for an intuitive control law, where the first term K_p keeps the wheels aligned with the given path by adjusting the steering angle ψ according to the orientation error e_s given by Eq. (6):

$$e_s(t) = \beta - \theta. \tag{6}$$

So a proportional controller that reduces the error along a path, considering a more significant lateral displacement at a larger steering angle (Eq. 7), is achieved,

$$\psi(t) = K_p e_s(t) = \frac{2}{L^2} e_s(t). \tag{7}$$

After various experiments, the optimal value for K_p is 0.5 $[rad \cdot m^{-1}]$ (L equal to 2). Therefore, the pure pursuit algorithm can convert the lateral deviation between the current and target positions into lateral control.

4.3 Feedforward for the irregular terrain

The greenhouses in Almería typically have a sandy topography with various mounds, leading to unstable terrain. In (GREENHOUSES, 2013), it is mentioned that the location should be flat in the width direction, with a slope in the principal axes ranging between 0 and 0.5% and never exceeds 1 to 2%, as this would require terracing. This

slope affects the navigation of the robot, altering the front wheels (heading) and speed (traction). This can be interpreted as a disturbance that directly affects the motors output (speed and direction). In robotics, thanks to special sensors (e.g., Light Detection and Ranging - LiDAR -), these slopes are measurable, providing an opportunity to implement a feedforward design to cancel this disturbance (Åström and Hägglund, 2006).

The slope of a mound can be calculated as the tangent of the distances captured by the sensor along a greenhouse aisle. In this work, this value will be simulated by a random number between 0 and 1 percent, emulating varied terrain. An indirect feedforward is proposed (Guzmán and Hägglund, 2011) where the gains of the disturbance and the plant are divided. To smooth its interaction, the rule for optimizing the resulting gain from (Hoyo et al., 2023) will be followed, considering the criterion on output performance and control effort. In this paper, the signal will be multiplied by a parameter α to reduce the possible saturation. Finally, the value of α is 3.3 for the steering motor and 2.78 for the traction motor.

5. RESULTS

With all the parameters obtained, the equivalent final control scheme is constructed (Fig. 11). To achieve a more realistic response, apart from considering the motors nonlinearities, the steering fork physical saturation limits are also implemented. Additionally, saturation blocks in the references, called hysteresis, are implemented, emulating the measurement uncertainty that sensors have for localization. Lastly, as the speed model is 72 % as slow as the steering model, the robot would start moving before turning, leading to unwanted movements. To address this, a delay of 0.5 seconds is added before calculating the distance to the point. For the simulation, this value will be adopted as a fixed parameter. However, in practice a control scheme will be designed to introduce a delay until steering motor reaches the reference, providing a variable delay. Fig. 9 shows the simulation results.

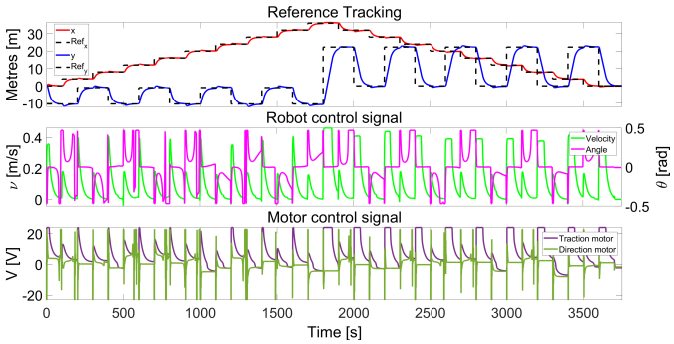

Fig. 9. Simulation results on the greenhouse coordinates

The reference tracking subfigure shows how the system reaches the reference within the proposed general specifications. It can also be noted that the motors tend to saturate (robot control signal sub-Fig.), causing them to steer and advance at maximum speed (motor control signal sub-Fig.), which is typical in robotics. Finally, in Figure 10, the control is evaluated on a real 2D plane of the greenhouse.

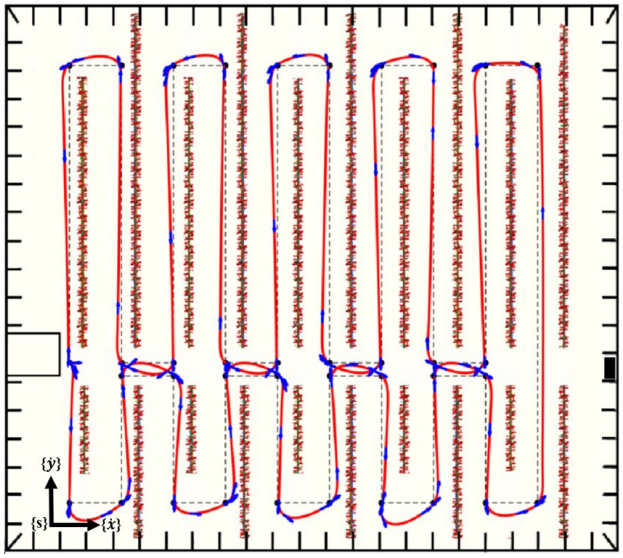

Fig. 10. Greenhouse simulation

As observed, the robot trajectory (red line) reaches the coordinates through the most optimal path with the direction indicated by the blue arrows, demonstrating natural navigation through the greenhouse. To evaluate the behavior, the mean of the IAE (Hoyo et al., 2023) and IADU indices is computed and compared for the strategy with and without feedforward, as they allow analyzing significant control efforts of the response. Applying disturbances in the motors input and without considering the feedforward, the mean value of the performance indices IAE and IADU (J) is 4.35. While implementing these adjustments, the new index is 3.61, improving the response by 18%.

6. CONCLUSION

This study presents a classical PID-based control scheme for the AGRICOBIOT II robot path tracking. Firstly, a successful cascade control has been implemented, identifying the nonlinearities commonly present in DC motors to design PID controllers in the inner loop and a classical Pure Pursuit control for the outer loop. Several PID tuning methods have been applied to establish a cascade control system. The inner loop characterizes the motor dynamics, while the outer loop ensures position control within a real greenhouse setting. In addition, a multitude of real-world constraints related to the dynamics of the real motors have been identified. The Ackermann configuration itself prohibits the implementation of a backlash mechanism due to the steering constraint. Additionally, the phenomenon of hysteresis has been simulated in the outer loop to model uncertainties in the position measurement, and a time delay has been introduced to reduce the error. Finally, a novel technique has been applied to improve the control effort by designing a simple feedforward implemented only in the inner loop for each motor. All these techniques will be implemented on the actual robot and analyzed in a natural greenhouse environment in the future.

REFERENCES

Åström, K.J. and Hägglund, T. (2006). *Advanced PID control*. ISA-The Instrumentation, Systems and Automation Society.

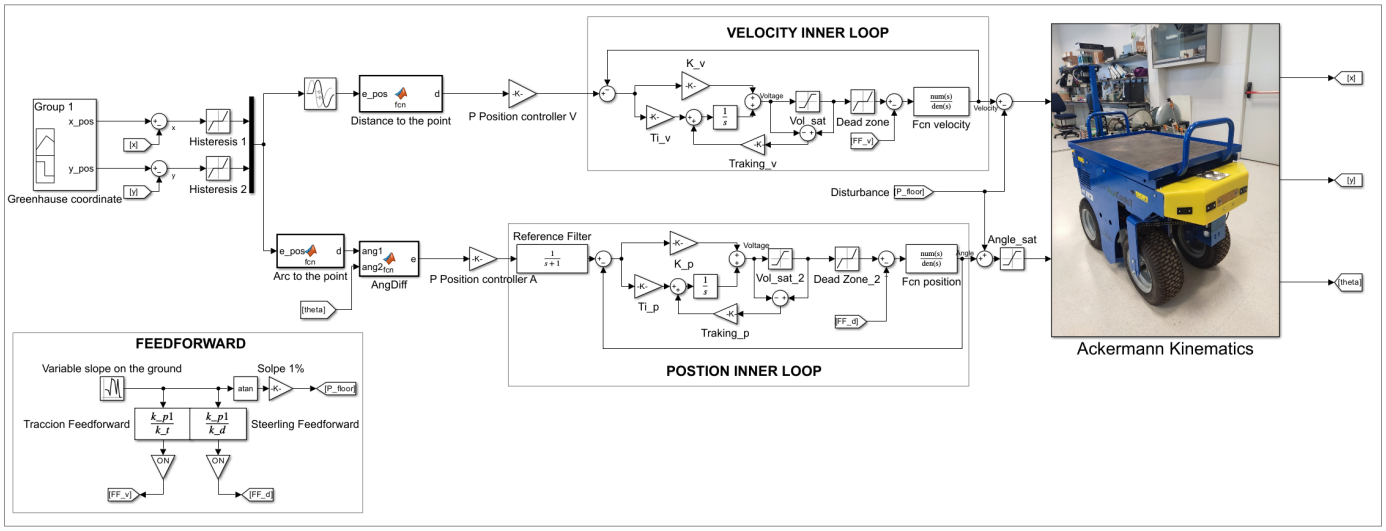

Fig. 11. Complete scheme of Pure Pursuit cascade control with Feedforward

Baturone, A.O. (2005). *Robótica: manipuladores y robots móviles*. Marcombo.

Blanco-Claraco, J.L., Tymchenko, B., Mañas-Alvarez, F.J., Cañadas-Aránega, F., López-Gázquez, Á., and Moreno, J.C. (2023). Multivehicle simulator (mvsim): Lightweight dynamics simulator for multiagents and mobile robotics research. *SoftwareX*, 23, 101443.

Cañadas-Aránega, F., Blanco-Claraco, J.L., Moreno, J.C., and Rodriguez-Diaz, F. (2024). Multimodal mobile robotic dataset for a typical mediterranean greenhouse: The greenbot dataset. *Sensors*, 24(6). doi:10.3390/s24061874. URL `https://www.mdpi.com/1424-8220/24/6/1874`.

Chen, I.Y.H., MacDonald, B., and Wunsche, B. (2009). Mixed reality simulation for mobile robots. In *2009 IEEE International Conference on Robotics and Automation*, 232–237. IEEE.

Corke, P.I., Jachimczyk, W., and Pillat, R. (2011). *Robotics, vision and control: fundamental algorithms in MATLAB*, volume 73. Springer.

GREENHOUSES, S.E.U.I.M. (2013). 4. greenhouse climate control and energy use. *Good Agricultural Practices for greenhouse vegetable crops*, 63.

Guzmán, J.L. and Hägglund, T. (2011). Simple tuning rules for feedforward compensators. *Journal of Process control*, 21(1), 92–102.

Hoyo, Á., Hägglund, T., Guzmán, J.L., and Moreno, J.C. (2023). A practical solution to the saturation problem in feedforward control for measurable disturbances. *Control Engineering Practice*, 139, 105636.

Jalving, B. (1994). The ndre-auv flight control system. *IEEE journal of Oceanic Engineering*, 19(4), 497–501.

Kong, J., Pfeiffer, M., Schildbach, G., and Borrelli, F. (2015). Kinematic and dynamic vehicle models for autonomous driving control design. In *2015 IEEE intelligent vehicles symposium (IV)*, 1094–1099. IEEE.

Liao, W., Nagai, K., and Wang, J. (2019). An evaluation method of electromagnetic interference on bio-sensor used for wearable robot control. *IEEE Transactions on Electromagnetic Compatibility*, 62(1), 36–42.

Liu, R., Guzman, J.L., Garcia-Manas, F., and Li, M. (2023). Selective temperature and humidity control strategy for a chinese solar greenhouse with an event-based approach. *REVISTA IBEROAMERICANA DE AUTOMATICA E INFORMATICA INDUSTRIAL*, 20(2), 150–161.

Macenski, S., Singh, S., Martín, F., and Ginés, J. (2023). Regulated pure pursuit for robot path tracking. *Autonomous Robots*, 1–10.

Matlab, S. (2012). Matlab. *The MathWorks, Natick, MA*.

Moreno Úbeda, J.C., Cañadas-Aránega, F., Rodríguez, F., Sánchez-Hermosilla, J., and Giménez, A. (2022). Modelado 3d y diseño de un robot colaborativo para tareas de transporte en invernaderos. In *XLIII Jornadas de Automática*, 785–791. Universidade da Coruña. Servizo de Publicacións.

Normey-Rico, J.E., Alcalá, I., Gómez-Ortega, J., and Camacho, E.F. (2001). Mobile robot path tracking using a robust pid controller. *Control Engineering Practice*, 9(11), 1209–1214.

Sánchez-Molina, J., Rodríguez, F., Moreno, J., Sánchez-Hermosilla, J., and Giménez, A. (2024). Robotics in greenhouses. scoping review. *Computers and Electronics in Agriculture*, 219, 108750.

Snider, J.M. et al. (2009). Automatic steering methods for autonomous automobile path tracking. *Robotics Institute, Pittsburgh, PA, Tech. Rep. CMU-RITR0908*.

Wang, R., Li, Y., Fan, J., Wang, T., and Chen, X. (2020). A novel pure pursuit algorithm for autonomous vehicles based on salp swarm algorithm and velocity controller. *IEEE Access*, 8, 166525–166540.

Yamamoto, S., Ara, T., Sugiura, M., Sawaki, J., and Matsuse, K. (1996). A design method of two-degree-of-freedom pid position controller for linear servo motor drives. *IEEJ Transactions on Industry Applications*, 116(8), 868–873.

Zeng, G. and Hemami, A. (1997). An overview of robot force control. *Robotica*, 15(5), 473–482.

Ziegler, J.G. and Nichols, N.B. (1942). Optimum settings for automatic controllers. *Transactions of the American society of mechanical engineers*, 64(8), 759–765.