Robotics and Autonomous Systems 162 (2023) 104364

Contents lists available at ScienceDirect

Robotics and Autonomous Systems

journal homepage: www.elsevier.com/locate/robot

Smart agriculture: Development of a skid-steer autonomous robot with advanced model predictive controllers

Cesar Wen Zhu [a], Elyse Hill [b], Mohammad Biglarbegian [a,*], S. Andrew Gadsden [c], John A. Cline [d]

[a] College of Engineering, University of Guelph, Guelph, Ontario, Canada
[b] Glenn Research Center, NASA, Cleveland, OH, USA
[c] Department of Mechanical Engineering, McMaster University, Hamilton, Ontario, Canada
[d] Department of Plant Agriculture, University of Guelph, Simcoe Research Station, Simcoe, Ontario, Canada

ARTICLE INFO

Article history:
Received 13 July 2022
Received in revised form 20 December 2022
Accepted 6 January 2023
Available online 13 January 2023

Keywords:
Mobile robots
Skid-steer
MPSMC
SMC
MPC
Tube-based MPC
Lyapunov
Agricultural applications

ABSTRACT

The agricultural domain has been experiencing extensive automation interest over the past decade. The established process for measuring physiological and morphological traits (phenotypes) of crops is labour-intensive and error-prone. In this paper, a mobile robotic platform, namely The Autonomous Robot for Orchard Surveying (AROS), was developed to automate the process of collecting spatial and visual data autonomously. Furthermore, six different control frameworks are presented to evaluate the feasibility of using a kinematic model in agricultural environments. The kinematic model does not consider wheel slippage or any forces associated with dynamic motion. Thus, the following six controllers are evaluated: Proportional-Derivative (PD) controller, Sliding Mode Controller (SMC), Control-Lyapunov Function (CLF), Nonlinear Model Predictive Controller (NMPC), Tube-Based Nonlinear Model Predictive Controller (TBNMPC), and Model Predictive Sliding Mode Control (MPSMC). This paper provides insight into the degree of disturbance rejection that the mentioned control architectures can achieve in outdoor environments. Experimental results validate that all control architectures are capable of rejecting the present disturbances associated with unmodelled dynamics and wheel slip on soft ground conditions. Additionally, the optimal-based controllers managed to perform better than the non-optimal controllers. Performance improvements of the TBNMPC of up to 209.72% are realized when compared to non-optimal methods. Results also show that the non-optimal controllers had low performance due to the underactuated constraint present in the kinematic model.

1. Introduction

The high demands from the growing population and the environmental impact put pressure on agricultural productivity. Estimations show that by 2050, an additional 70% of food production would be necessary to meet the future population [1]. The use of autonomous mobile robots has shown to have promising results with operations that require heavy workloads, and repetitive processes and have shown to be more efficient than humans. Field scouting and data collection is an application in mobile robots that seeks to gather phenotypic properties of the plant to assess its genotype and environmental properties during its growth. The assessment allows farmers to understand the impacts on the breeding process of plants and make corrective decisions to optimize yield. However, for phenotypic evaluations to provide a significant impact, timeliness and accuracy in information are crucial [2]. Developments in technologies and imaging and ranging techniques have allowed the new era of high-throughput phenotyping [3]. Various phenotypic data can now be gathered in sensory systems at high speeds without disrupting the plant habitat. The automation of data collection using a robotic platform is now emerging due to these recent advances. Field-based phenotyping scouting robots are expected to be flexible, multipurpose, and affordable to be viable on a commercial-scale [4]. Some of the scouting techniques include the use of 3D point cloud generation of plants and trees or computer vision techniques to assess structural, morphological, or physiological traits [5]. For instance, the Ladybird is a robotic research platform equipped with data collection systems such as LiDAR, stereovision, and thermal cameras to gather properties of small crops [6]. Similarly, the Robotanist is a ground-based robotic system equipped with a 3DOF manipulator and a force gauge to determine the stalk strength of sorghum plants [7].

* Corresponding author.
E-mail addresses: cwenzhu@uoguelph.ca (C. Wen Zhu), elyse.d.hill@nasa.gov (E. Hill), mbiglarb@uoguelph.ca (M. Biglarbegian), gadsdesa@mcmaster.ca (S.A. Gadsden), jcline@uoguelph.ca (J.A. Cline).

https://doi.org/10.1016/j.robot.2023.104364
0921-8890/

Other high-throughput phenotyping systems include the BoniRob which is equipped with spectral imaging and 3D TOF cameras to measure plant height, stem thickness, biomass, and spectral reflection [8].

Skid-steer Mobile Robots (SSMR) are a common type of robotic platform found in agricultural research environments [7,9]. They inherit a simple mechanical composition and manoeuvrability and provide large traction forces, beneficial for rough terrains. Although the inherent advantages that the SSMR holds, it is challenging to control and predict the motion of the SSMR due to its intrinsic nature of skidding when performing a turn [10]. Control strategies of SSMR typically utilize different aspects in modelling the robot, such as kinematics or dynamics. Most of the models used in control design feature the use of kinematic models, which omit the modelling of dynamic parameters such as friction, mass and skidding off the robot [11]. This assumption essentially simplifies the model of the SSMR to a two-wheel differential drive. Such an assumption imposes an array of uncertainties from unmodelled dynamic parameters such as mass, skidding, or friction. Additionally, most methodologies that implement kinematic models of the SSMR are typically observed in controlled indoor environments where the surface remains planar and friction remains relatively constant [12,13]. Hence, for a kinematic model to be viable in outdoor environments, the controller must be designed to be robust against unmodelled dynamics of outdoor conditions, as typically observed in agricultural terrains. Kinematic models also have inherent performance limitations in which they may tend to overactuate the system and chatter often to account for unmodelled dynamics. To address such limitations, the use of dynamic models often alleviates the input strain in kinematic models requires.

Tracking control of SSMR has become increasingly popular in various sectors such as mining [14], explorations [15], and farming [16]. One of the popular control strategies includes using a Sliding Mode Controller (SMC). Applications of SMC in SSMR compositions can be seen in the implementation such as [17]. It uses the dynamic model of the vehicle to create the Fuzzy-SMC (FSMC) composition to achieve tracking. Two sliding surfaces are designed for the yaw control and longitudinal velocity control of the SSMR, and the fuzzy controllers regulate the gain of the sliding surfaces. Although the methodology achieved good results, the experiments are situated in a simulation environment. Similarly, [18] implements a path following SMC using the dynamic model of the skid-steer system. The research uses Caracciolo's derivation of the dynamic model and creates two sliding surfaces, which are then superimposed to match the dimension of the states. Results show that the SSMR can follow the path in a simulated environment with minor errors but does not consider more complex paths. Aside from the mentioned SMC implementations in SSMR, the majority of implementations of SMC are seen in differential drive compositions. Implementations such as the ones seen in [19,20], or [21] use the differential drive kinematic model to derive the SMC control regime. The design involves the use of newly defined error equations that considers the error of the SSMR relative to the path. Once formulated, the error dynamics may be differentiated to derive the control law. From the implementations of authors, the method proves to be performed adequately, but it lies mostly in simulated or real-life controlled environments. Furthermore, extensive research has been observed on Model Predictive Control strategies for SSMR.

Model Predictive Control (MPC) is another robust path-tracking algorithm that creates an optimal control action while addressing set constraints and states. The controller uses a discrete-time finite-horizon optimization problem to solve for each time-step of the current state [22]. The horizon estimates give MPC the ability to forecast future trajectories to the control problem and provide optimal actuation under ideal conditions. Hence, MPC can provide skid-steer systems with a smooth trajectory by predicting skid behaviour and optimizing for any foreseeable trajectories. However, control of skid-steer mobile vehicles may be achieved by approximating the non-linearities associated with the skid behaviour.

MPC-based linear models are not feasible as the process control must be operated at a set point such that it can be formulated as a convex problem [23]. Hence, Non-linear MPC (NMPC) is appropriate for stabilizing the non-linearities of the wheel. For instance, [24] incorporates a point stabilization method to overcome the non-linearities of the vehicle. The method proves that the point stabilization of the nonlinear system achieves tracking control of the path. However, since the model does not include the behaviours of skidding in turns, it struggles to track the trajectory during the start. Additionally, the model is only feasible in indoor environments where the tire-surface friction does not vary as much. Another application of NMPC is seen in [25]. Kayacan et al. use a centralized NMPC scheme for a tractor–trailer system that considers forward and side slip parameters and uses a Nonlinear Moving Horizon Estimation (NMHE) to estimate non-linearities. Although good tracking is achieved by the tractor–trailer vehicle, it is very computationally extensive, which may pose difficulties in the real-time coordination and optimization process. Similarly, [14] also incorporates an adaptive NMPC, which considers a more simplified slip parameter with NMHE. In comparison to [14,25] implements an adjustable model which adapts depending on terrain changes. This method proved to overcome the drawbacks of computational load and was tested in rough outdoor terrains.

Unlike conventional approaches to NMPC, other variations of NMPC may be proved superior. For instance, Tube-based NMPC is robust against disturbances that may be present in traction loss [26]. Tube-based NMPC uses an online optimization problem transformed into a sequential control search rather than control policies. Additionally, it holds an auxiliary controller that ensures that the control problem remains bounded in the presence of disturbance and uncertain model dynamics [27]. Prado et al. [26] implement a tube-based non-linear model for skid-steer vehicles. Results showed it could reduce errors up to 50% in different terrains such as grass and gravel over conventional NMPC. Another variation on NMPC is Robust Constrained NMPC (RC-NMPC) which guarantees constraint satisfaction when considering uncertain systems [28]. RC-NMPC ensures constraint satisfaction by considering a fixed estimate of the model uncertainty and evaluates inputs using the estimate such that all plausible predictions satisfy the constraint. The method allows for a more flexible constraint applied to the prediction horizon. Hence, the reduction of conservatism associated with the constraint-tightening approach allows for a more feasible optimization problem that is robust to model errors [29]. An example of RC-MPC can be seen in [30]. The research uses a learning-based model with the RC-MPC model to achieve trajectory tracking in a skid-steer vehicle. The use of a learning-based model proved to have reduced tracking errors since it relies on real-world experience rather than mathematical models. Combined with the RC-MPC, the vehicle could effectively provide robust constraint satisfaction over various terrains. However, the model may be sensitive to the data to which the model is exposed.

In this paper, we develop several nonlinear controllers for the experimental robotic platform to perform trajectory tracking in agricultural terrain conditions. The first three controllers consist of two conventional nonlinear controllers: the Proportional Derivative (PD), the Control Lyapunov Function (CLF), and the Sliding Mode Controller (SMC). The other two controllers consist of a Nonlinear Model Predictive Controller (NMPC) and

Fig. 1. Robotic platform used for orchard operations.

a Tube-Based Nonlinear Model Predictive Controller (TBNMPC), optimal-based controllers. The last controller consists of a hybrid approach between NMPC and SMC.

The research contributions for this work include the creation of a robotic platform used for data collection in apple orchards and the implementation of several nonlinear controllers. The controllers use the kinematic model in their design to verify its robustness against unmodelled dynamics. Typically, kinematic-based controllers are implemented in two-wheel differential drive robots in indoor environments. This work shows that the presented six controllers can be applied to skid-steer compositions in outdoor environments. The performance of the controllers was validated through field experiments. The novel controllers implemented in this work include the MPSMC and SMC. The MPSMC uses a simplified variation of [31] which does not require the dynamic model. Furthermore, a modified SMC strategy is proposed using the kinematic model. The SMC improves on the works of [21] by adding tuning gain for faster tracking stability.

The rest of the paper is structured as follows. In Section 2, the experimental platform and field site is presented. In Section 3, the kinematic model is described. Section 4 presents the controllers to be tested. The results of the experiments are then presented in Section 5 and discussed in Section 6. Finally, the paper is concluded in Section 7.

2. Robot platform & experimental site

2.1. Platform

The skid-steer platform used in the experiment consists of a custom-built mobile equipped with individual motors on each wheel, as shown in Fig. 1. Each motor features a 400 W Falcon 500 DC motor with embedded 2048 Counts Per Revolution (CPR) encoders by Vex Robotics. The platform features a complete aluminium structure measuring 31 × 38 × 75 in. in its exterior dimensions with 13-in. agricultural tread wheels and 4 in. of ground clearance. The batteries feature two Lithium Iron Phosphate (LiFePO4) with a total capacity of 110 Ah. The sensors equipped in the SSMR include a Jetson AGX Xavier, EMLID Reach RS+ RTK-GNSS, Pololu's UM7 IMU, RPLiDAR's S1 LiDAR as a horizontal scanner, and Velodyne's VLP-16 LiDAR as a vertical LiDAR scanner. An overview of the sensor and communication networks embedded in the system can be shown in Fig. 2. Additionally, the robot's communication features a WiFi modem to communicate with the robot via SSH using Ubuntu Bionic. The software framework of the robot was developed using Robot Operating System (ROS). Each robot subsystem holds a node that holds its independent software that communicates with other components. The motors were programmed to be controlled using two inputs: the linear velocity (v_x), and angular velocity (ω), operating at 25 Hz.

2.2. Experimental site

Experimental testing of the mobile platform system takes place at the University of Guelph's Simcoe Research Facility located in Simcoe, Ontario, Canada (42°51′30.4562″N, 80°15′56.1024″W). The study area includes two tree rows spaced 2.7 m apart and roughly 37 m long, with 104 trees per row spaced at 1 m. The final results of the controller were collected in mid-March, 2022. During this time, the ground conditions were soft due to the melting snow. The experimental site has an incline of approximately 3 degrees in the middle of the rows and uneven ground variations of 1–3 cm deep.

2.3. Defined path

The path is designed to traverse inter-row routes between the orchard. The planned path is set to start from the middle of the two rows and finish at its starting position. T-shaped turns are introduced to increase the complexity of the controller's manoeuvring of turns. The map shown in Fig. 3 illustrates the recorded map from LiDAR scans and the generated path for the experimentation. The black dots represent individual trees; the white space represents the explored space, and the grey is unexplored.

3. Skid-steering mobile robot kinematic model

In this section, the kinematic model for the SSMR and the localization framework of the model are presented. The localization framework relies on the platform's sensors, namely, RTK-GNSS, encoders and IMU. The measurements are then fused to provide state estimates of the platform using an Extended Kalman Filter (EKF) [32].

3.1. Kinematic model

The kinematic model follows similar steps to the [33] approach. This model assumes that the vehicle experiences no lateral or longitudinal slippage of the wheels and imposes the nonholonomic constraint depicted in [34]. This assumption simplifies the kinematic model of the SSMR to represent that of a differential drive. Additionally, further assumptions are made, such as the robot's movement remains planar.

The development of the kinematic model considers that the vehicle's position is directed by the base link frame, which is placed at the Centre Of Mass (COM) of the robot (assumed to be the centroid of the SSMR). If motion is applied to the SSMR, the position and orientation of the robot can be defined as the vector $\mathbf{q} = [x, y, \theta]^T$, and $\dot{\mathbf{q}} = [\dot{x}, \dot{y}, \dot{\theta}]^T$ as the velocity vector. Using the relation of the base link frame and the map frame, shown in Fig. 4, the velocity of the SSMR can be defined by the following matrix

$$\begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{\theta} \end{bmatrix} = \begin{bmatrix} cos(\theta) & sin(\theta) & 0 \\ -sin(\theta) & cos(\theta) & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} v_x \\ v_y \\ \omega \end{bmatrix} \tag{1}$$

where v_x is the longitudinal velocity, v_y is the lateral velocity, ω the angular velocity of the SSMR, and θ the heading angle of the SSMR. Note that the equation does not impose any movement

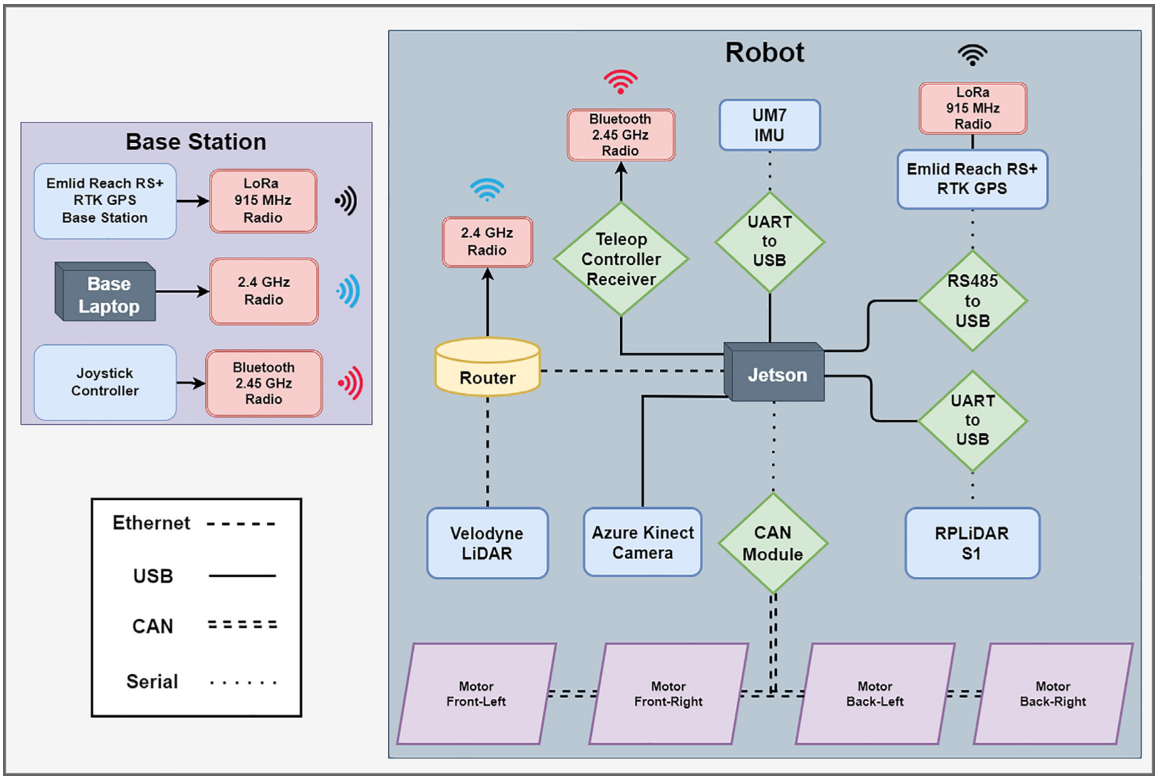


Fig. 2. Sensor components of the robotic platform.

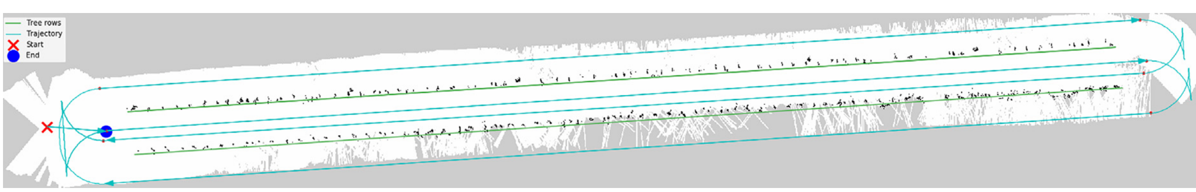


Fig. 3. Generated path of the experimental site.

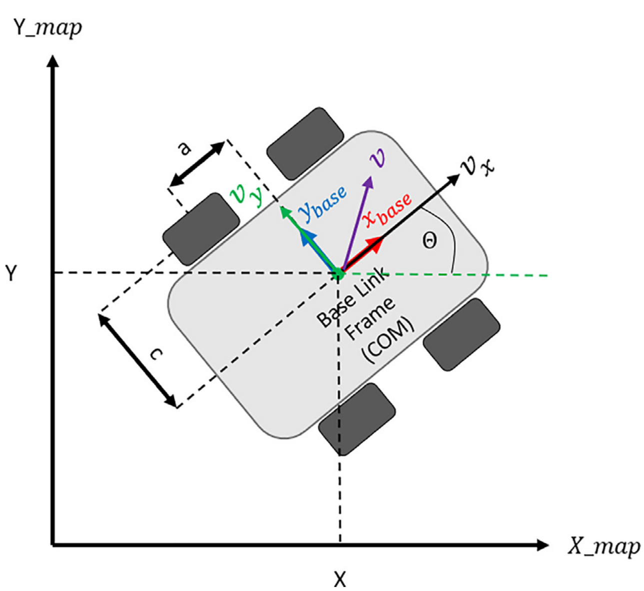


Fig. 4. Free body diagram.

constraints; hence, it is necessary to associate the nonholonomic constraint with the velocities. Therefore, the following velocity constraint must be satisfied [33]:

$$v_y + x_{ICR}\dot{\theta} = 0 \tag{2}$$

Since the model is simplified to assume that the COM is at the centroid of the vehicle, the nonholonomic constraint simplifies to $v_y = 0$. Therefore, using Eq. (2) with the nonholonomic constraint, the kinematic model becomes:

$$\begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{\theta} \end{bmatrix} = \begin{bmatrix} cos(\theta) & 0 \\ sin(\theta) & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} v_x \\ \omega \end{bmatrix} \tag{3}$$

where $\eta = [v_x, \omega]^T$ is the control input that considers the nonholonomic constraint.

3.2. State estimation

State estimation of the SSMR is one of the most fundamental aspects of motion control. All mobile systems that require tracking a certain trajectory must obtain its location with respect to the desired trajectory. There are several localization techniques with SSMR, however, in the present work, an EKF is used to obtain the location estimates. The EKF used in this work is adopted from a ROS package known as *robot_localization*, developed by [35]. The package uses an EKF to fuse three sensory components that

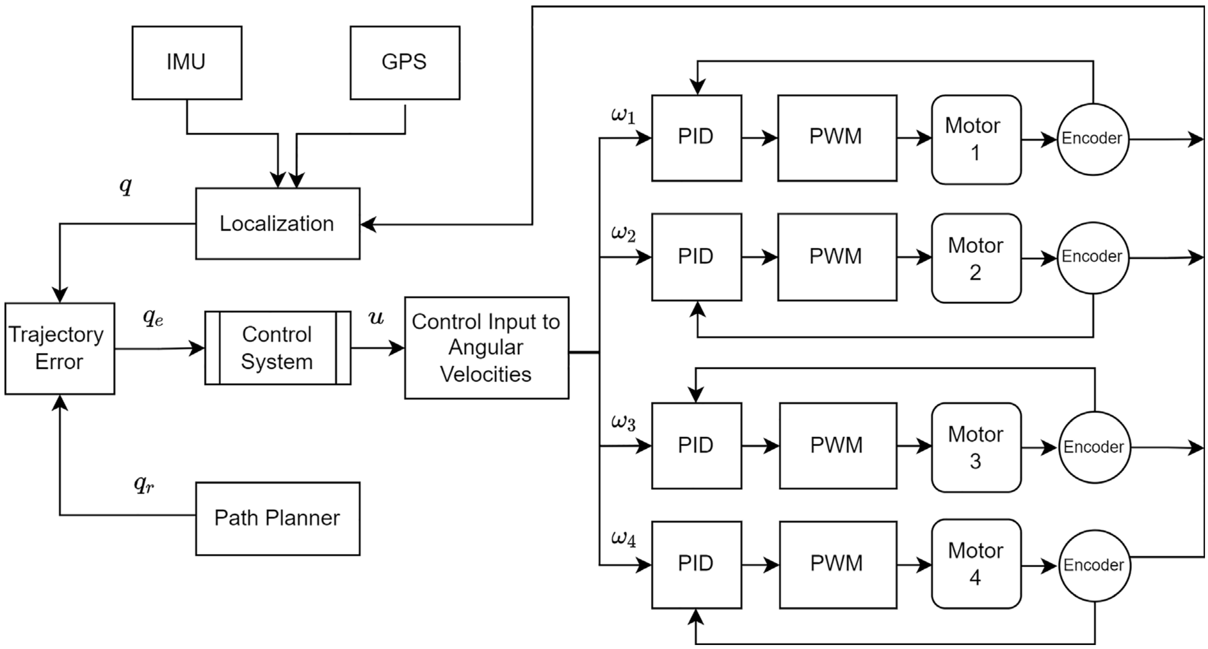


Fig. 5. Overall control architecture.

define the overall location estimate of the mobile robot. These sensors are defined as IMU, wheel encoders and RTK-GNSS. The IMU provides orientation and angular velocity estimates; the encoders provide linear velocity and angular velocity estimates; the GNSS provides globally accurate position estimates. The EKF runs at a frequency of 14 Hz to provide estimates of the global location with respect to its defined UTM zone. The accuracy of the localization method shows to be in the range of ± 5 cm.

4. Control design

This section presents six controllers for a trajectory tracking problem design for an under-actuated SSMR model. Some controllers feature the extension of a particular controller structure. The main control structures include the Proportional-Derivative (PD) controller, the Control Lyapunov Function (CLF), the Sliding Mode Control (SMC) and the Nonlinear Model Predictive Control (NMPC). Variations of the NMPC include the Tube-Based NMPC (TBNMPC) and Model Predictive Sliding Mode Control (MPSMC). In the design of the controllers, the trajectory tracking problem's objective is to control the nonholonomic SSMR to follow the reference path. Each trajectory point will hold spatiotemporal information that contains positional and velocity profiles. In the design of the control architectures, the kinematic model will be used for all subsequent controllers. The problem is defined as a nonlinear, time-invariant system defined in Eq. (3) which can be generalized in the following form:

$$\dot{q} = f(q, u) + d \tag{4}$$

where $q \in \mathbb{R}^n$ is the state variable, $u \in \mathbb{R}^m$ is the control input of the system and $d \in \mathbb{R}^n$ is the bounded white noise disturbance. The current state of the robot is defined as $q = [x, y, \theta]^T$ and the path planner in Section 2.3 generates the desired trajectory providing the states $q_r = [x_r, y_r, \theta_r, \dot{x_r}]^T$. The input to the kinematic model is dependent on the linear velocity v_x and the angular velocity ω, which can be summarized as $u = [v_x, \omega]^T$. Therefore, the control objective must be designed such that the inputs u guarantee asymptotically stability to the error states $q_e = [x_e, y_e, \theta_e]^T$. Note that the control problem presents an under-actuated system scenario. Two control inputs are needed to track the three independent states defined in the vector q_r. The controller must also be robust against unmodelled system dynamics, terramechanics, and uneven ground conditions. We make the assumption that θ_e is bounded for $|\theta_e| \le \frac{\pi}{2}$. The overall control structure is described in Fig. 5. Additionally, non-optimal controllers such as the PD, SMC, MPSMC and CLF require redefining the error of the trajectory to be relative to the orientation of the robot coordinate frame for the control system to use the kinematics model.

4.1. Trajectory tracking errors

The robot that is controlled to track the desired trajectory will exhibit an error which is expressed in terms of the robot's coordinate system, shown in Fig. 6. The error relative to the robot's orientation can be defined by the following:

$$q_e = \begin{bmatrix} x_e \\ y_e \\ \theta_e \end{bmatrix} = \begin{bmatrix} cos(\theta) & sin(\theta) & 0 \\ -sin(\theta) & cos(\theta) & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_r - x \\ y_r - y \\ \theta_r - \theta \end{bmatrix} \tag{5}$$

Thus, the velocity error can be determined by the derivative of Eq. (5) and using Eq. (3), giving the following:

$$\begin{bmatrix} \dot{x_e} \\ \dot{y_e} \\ \dot{\theta_e} \end{bmatrix} = \begin{bmatrix} y_e\omega - v + v_r cos(\theta_e) \\ -x_e\omega + v_r sin(\theta_e) \\ \omega_r - \omega \end{bmatrix} \tag{6}$$

Consequently, the acceleration error can also be determined by obtaining the velocity's error derivative as such:

$$\begin{bmatrix} \ddot{x_e} \\ \ddot{y_e} \\ \ddot{\theta_e} \end{bmatrix} = \begin{bmatrix} \dot{y}_e\omega + y_e\dot{\omega} - \dot{v} + \dot{v}_r cos(\theta_e) - v_r\dot{\theta}_e sin(\theta_e) \\ -\dot{x}_e\omega - x_e\dot{\omega} + \dot{v}_r sin(\theta_e) - v_r\dot{\theta}_e cos(\theta_e) \\ \dot{\omega}_r - \dot{\omega} \end{bmatrix} \tag{7}$$

4.2. Proportional-Derivative controller

The Proportional-Derivative (PD) controller is a popular control algorithm that determines the control input based on the trajectory errors of the system. By using only the system's errors, the system's kinematic model is not required to derive the control

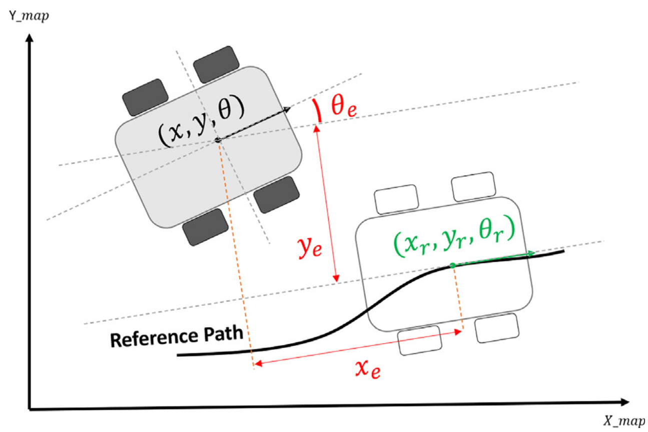


Fig. 6. Relative errors in the lateral, longitudinal and yaw directions.

structure. Mathematically, the PD control inputs are described as

$$u(t) = K_P e(t) + K_D \frac{de(t)}{dt} \tag{8}$$

where $e(t)$ is the error between the estimated state and the desired state, K_P & K_D are the tuning gains for the proportional, and derivative errors, respectively. However, the control input of the PD is generally designed for each state. In the case of the SSMR, the underactuated constraint hinder the usage of a conventional PD control. The two control inputs v_x and ω would need to control the tree states in the vector q. Two controllers are defined to overcome the underactuated constraint; the first controller stabilizes the longitudinal error (x_e) through the linear velocity input, and the second controller stabilizes both the lateral (y_e) and orientation (θ_e) error with the angular velocity input. The linear velocity controller uses a traditional PD controller, which tries to minimize the error and its derivative by the following equation.

$$v_x = K_{P_x} x_e + K_{D_x} \frac{dx_e}{dt} \tag{9}$$

The second controller follows the derivation of a Lyapunov candidate function to determine the necessary orientation error to stabilize the lateral error [21]. The following Lyapunov candidate function is defined as:

$$V_y(y_e) = \frac{1}{2} y_e^2 \tag{10}$$

with its derivative being

$$\dot{V}_y(y_e) = y_e \dot{y}_e = -y_e x_e \omega + y_e v_r sin(\theta_e) \tag{11}$$

The derivative of the Lyapunov function can be further simplified to $\dot{V}_y(y_e) = y_e v_r sin(\theta_e)$ since the controller ensures that $x_e \rightarrow 0$. Hence, asymptotical stability may only be achieved if $y_e v_r sin(\theta_e) \leq 0$. The following equation is then defined to satisfy the desired stability property [21]

$$\theta_e = -arctan(y_e) \tag{12}$$

The equation above re-defines the derivative of the Lyapunov candidate function to $-y_e v_r sin(arctan(y_e))$. Since it is known that $arctan(y_e) \in (\frac{-\pi}{2}, \frac{\pi}{2})$, the value of $sin(arctan(y_e)) \in (1, -1)$. Additionally, the sign of $\dot{V}_y(y_e)$ is not affected by the sign of y_e. If $y_e < 0$ or $y_e > 0$ the sign of $\dot{V}_y(y_e)$ still remains less than equal to zero. Therefore, it is proven that the derivative of the Lyapunov candidate function is bounded and non-increased which assures the asymptotic stability of the lateral error. With the definition of the orientation error in Eq. (12), the angular velocity controller may then be defined as

$$\omega = K_{P_\theta} \theta_e + K_{D_\theta} \frac{d\theta_e}{dt} \tag{13}$$

Eq. (13) is then extended to include the orientation error in Eq. (12) as such

$$\omega = K_{P_\theta} \theta_e + K_{P_y} arctan(y_e) + K_{D_\theta} \frac{d\theta_e}{dt} \tag{14}$$

The newly defined equation will ensure that both the lateral and orientation errors are driven to stability.

4.3. Control-Lyapunov Function

The Control-Lyapunov Function (CLF) is an extension of the Lyapunov stability theory to derive the control inputs necessary to drive the system to asymptotic stability. This requires proof that the Lyapunov candidate function ($V(x_e)$) will eventually drive the error states of the system to zero ($x_e \rightarrow 0$). Thus, the proof must satisfy two conditions: defining a positive-definite Lyapunov candidate function, and its derivative being strictly negative or negative semi-definite. If the proof is satisfied, the states are guaranteed to decrease the Lyapunov function, which would in return assure the reduction of the error states. The works by Sontag [36] show examples of how the candidate Lyapunov function may be designed to derive the control inputs to drive the error trajectories to zero. Furthermore, although the method provides a convenient approach to deriving the stability of the system, it becomes complex in finding the adequate Lyapunov function to satisfy the conditions. The CLF control design used in this work is adapted from the formulation presented in [37,38]. Consider the following positive definite quadratic Lyapunov function:

$$V(q_e) = \frac{1}{2} x_e^2 + \frac{1}{2} y_e^2 + (1 - cos(\theta_e))/K_y \tag{15}$$

where its derivative can be expressed as

$$\begin{aligned} \dot{V}(q_e) &= x_e y_e \omega - x_e v + x_e v_r cos(\theta_e) - y_e x_e \omega \\ &\quad + y_e v_r sin(\theta_e) - \omega_r sin(\theta_e)/K_y + \omega sin(\theta_e)/K_y \end{aligned} \tag{16}$$

To provide asymptotic stability to the system, the following controllers are defined:

$$v = v_r cos(\theta_e) + K_x x_e \tag{17}$$

$$\omega = \omega_r + K_y v_r y_e + K_\theta sin(\theta_e) \tag{18}$$

where $K_x, K_y, K_\theta > 0$. The new derivative of the Lyapunov function would then give the following function:

$$\dot{V}(q_e) = -K_x x_e^2 - v_r K_\theta sin^2(\theta_e)/K_y \leq 0 \tag{19}$$

The controllers shown above manage to provide asymptotic stability to the error states of the trajectory. From Eq. (15), the Lyapunov function is positive-definite bounded meaning that $V(q_e) > 0$ and $V(0) \neq 0$ for all $x \neq 0$. The derivative of the function also shows that the Lyapunov function is non-increasing or zero and $V(q_e) \rightarrow 0$. Hence, the states q_e is guaranteed to converge to zero as $t \rightarrow \infty$.

4.4. Sliding Mode Control

The Sliding Mode Controller (SMC) attempts to solve the trajectory tracking problem by controlling the linear and angular velocity of the SSMR based on the robot's relative errors in the system. The design of the SMC includes defining continuous functions that map the states into a control surface which is minimized by the surface control law. Typically, a sliding surface would be designed for each system state. However, since the SSMR presents an underactuated system problem, two sliding surfaces must be designed for each input. Thus, two surfaces are defined to map the error states q_e into the following:

$$\begin{bmatrix} s_1 \\ s_2 \end{bmatrix} = \begin{bmatrix} x_e \\ \theta_e + \frac{\lambda_y}{2} arctan(ye) + \frac{\lambda_\theta}{2}|\theta_e| sign(ye) \end{bmatrix} \tag{20}$$

The first surface controller minimizes the longitudinal error x_e by controlling the linear velocity v_x of the SSMR. The second controller minimizes both the lateral error y_e and yaw error θ_e through the control of the angular velocity of the model. In the second surface, two error variables are defined to satisfy the asymptotic Lyapunov stability theorem. The Lyapunov candidate function is first defined as [21]

$$V_y(y_e) = \frac{1}{2} y_e^2 \tag{21}$$

with its derivative being

$$\dot{V}_y(y_e) = y_e \dot{y}_e = -y_e x_e \omega + y_e v_r sin(\theta_e) \tag{22}$$

The derivative of the Lyapunov function can be further simplified to $\dot{V}_y(y_e) = y_e v_r sin(\theta_e)$ since the first sliding function ensures that $x_e \to 0$. Hence, asymptotical stability may only be achieved if $y_e v_r sin(\theta_e) \leq 0$. The following equation is then defined to satisfy the desired stability property

$$\theta_e = -\frac{\lambda_\theta}{2}|\theta_e| sign(ye) - \frac{\lambda_y}{2} arctan(ye) \tag{23}$$

where $\lambda_\theta \leq 1, \lambda_y \leq 1$. By defining θ_e as the equation above, it ensures that $y_e v_r sin(\theta_e)$ is bounded and that $V_y(y_e)$ is non increasing. Since it is assumed that $\theta_e \in (\frac{-\pi}{2}, \frac{\pi}{2})$ and it is known that $arctan(y_e) \in (\frac{-\pi}{2}, \frac{\pi}{2})$, it ensures that $|\theta_e| \leq \frac{\pi}{2}$. In return, $sin(\theta_e)$'s sign relies solely on the sign of y_e proving that $y_e v_r sin(\theta_e) \leq 0$. The derivation of the controllers can then be defined by obtaining the derivative of s as

$$\begin{bmatrix} \dot{s_1} \\ \dot{s_2} \end{bmatrix} = \begin{bmatrix} \dot{x}_e \\ \dot{\theta}_e + \frac{\lambda_\theta}{2}\dot{\theta}_e sign(\theta_e) + \frac{\lambda_y}{2}\frac{\dot{y}_e}{1+y_e^2} \end{bmatrix} \tag{24}$$

The control law of the SMC uses the method proposed by Gao and Huang [39] in which the reaching law includes both a constant and proportional reaching law to attract the trajectories into the switching manifold. The general form of the control law is

$$\begin{bmatrix} \dot{s_1} \\ \dot{s_2} \end{bmatrix} = \begin{bmatrix} -P_1 s_1 - Q_1 sign(s_1) \\ -P_2 s_2 - Q_2 sign(s_2) \end{bmatrix} \tag{25}$$

where the sign function is defined as

$$sign(s) = \begin{cases} 1, & \text{if } s > 0 \\ -1, & \text{if } s < 0 \end{cases} \tag{26}$$

Through the Lyapunov proof, it can be further proof that the control law would assure the asymptotic stability of the surface functions. The proof can be proved by defining the Lyapunov function of

$$V_s(s_1, s_2) = \frac{1}{2}s_1^2 + \frac{1}{2}s_2^2 \tag{27}$$

where its derivative is given as

$$\begin{aligned} \dot{V}_s(s_1, s_2) &= s_1\dot{s_1} + s_2\dot{s_2} = -s_1^2 Q_1 - s_1 P_1 sign(s_1) \\ &\quad -s_2^2 Q_2 - s_2 P_2 sign(s_2) \end{aligned} \tag{28}$$

It is then proved that $\dot{V}_s(s_1, s_2)$ is negative semi-definite only if the constants P_i and Q_i satisfy the bounds of $P_i, Q_i \geq 0$. Therefore, the sliding surfaces would reach asymptotic stability such that $s_i \to 0$. Trivially, this proves that from Eq. (20), $x_e \to 0$ and $\theta_e \to -\frac{\lambda_y}{2} arctan(ye) - \frac{\lambda_\theta}{2}|\theta_e| sign(ye)$. The convergence of θ_e also implies that $y_e \to 0$ which would lead to $\theta_e \to 0$.

The SMC is notorious for its chattering problem when implemented with the *sign* function as described in Eq. (26). Authors have proposed methods that overcome such drawbacks by implementing a lowpass filter structure to the surface variable s by introducing a boundary layer thickness Φ through a saturation function [40]. The saturation function can be described as:

$$sat\left(\frac{s}{\Phi}\right) = \begin{cases} \frac{s}{\Phi}, & \text{if } |s/\Phi| \leq 1 \\ sign\left(\frac{s}{\Phi}\right), & \text{if } |s/\Phi| > 1 \end{cases} \tag{29}$$

The saturation functions can then replace the *sign* functions defined in Eqs. (24) and (25). Consequently, using Eqs. (6), (24) and (25), the controllers of the kinematic SSMR model can be defined as such:

$$v_x = P_1 s_1 + Q_1 sat(s_1) + y_e \omega + v_r cos(\theta_e) \tag{30}$$

$$\omega = \frac{P_2 s_2 + Q_2 sat(s_2) + \omega_r (1 + \frac{\lambda_\theta}{2} sign(\theta_e)) + \frac{\lambda_y v_r sin(\theta_e)}{2(1+y_e^2)}}{(1 + \frac{\lambda_\theta}{2} sign(\theta_e)) + \frac{\lambda_y x_e}{2(1+y_e^2)}} \tag{31}$$

where $\lambda_\theta \leq 1, \lambda_y \leq 1$.

4.5. Nonlinear Model Predictive Control

The Nonlinear Model Predictive Control (NMPC) is used to minimize the optimal policy of the controller and to solve for the optimal input of the system. The optimal policy is set to minimize the state trajectory error q_e and input errors u_e. Additionally, the optimal solution is spanned across the prediction horizon to anticipate future events and tune the optimal control action accordingly. The first input among the time horizon sequence is executed, and the process re-iterates itself by providing the updated states to the controller. The design of the MPC is adopted from [24] which discusses the design of the optimal policy and stability analysis. The optimal control problem is formulated to minimize the cost function defined as

$$\begin{aligned} J_T(t_k, q_e(\tau), u_e(\tau)) &= \int_{t_k}^{t_k+T} \ell(q_e(\tau), u_e(\tau)) d\tau \\ &\quad + J_f(q(t_k + T)) \end{aligned} \tag{32}$$

where $\ell(\tau, q_e(\tau), u_e(\tau)) = \|q_e(\tau)\|^2_{Q_{MPC}} + \|u_e(\tau)\|^2_{R_{MPC}}$ is the running cost of the MPC; q_e is the error trajectory described in Eq. (5); $u_e = u - u_r$ is the input error between the input and reference input; $J_f = \|q_e(t_k + T)\|^2_{P_{MPC}}$ is the terminal cost; Q_{MPC}, P_{MPC} and R_{MPC} are positive definite symmetric weight matrices that tune the states, terminal cost and control action respectively; and T is the prediction time horizon. Terminal constraints are imposed on the problem to assure stability of the system [41,42]. The NLP problem can then be defined as

$$\begin{aligned} \min_u \quad & J_T(t_k, q, u) \\ \text{s.t.} \quad & \dot{q}(\tau) = f(q(\tau), u(\tau)), \\ & q(t_k) = \hat{q}(t_k), \\ & u(\tau) \in \mathbb{U}, \\ & q(\tau) \in \mathbb{Q}, \\ & q(t_k + T) \in \mathbb{Q}_f \\ & \tau \in [t_k, t_k + T] \end{aligned} \tag{33}$$

where $\mathbb{Q} \in \mathbb{R}^n$ is the state constraints, $\mathbb{U} \in \mathbb{R}^m$ is the input constraints, $\mathbb{Q}_f \in \mathbb{R}^n$ is the terminal constraints, and $\hat{q}(t_k)$ denotes the estimated state vector determined by the EKF after the optimal input from the previous instance ($u^*(t_{k-1})$) is given to the system. Note that $(\cdot)^*$ denotes the optimal solution.

4.6. Tube-Based Nonlinear Model Predictive Control

Tube-Based Nonlinear Model Predictive Control (TBNMPC) utilizes two controllers to mitigate the unmodelled dynamics of the system and the disturbances. The first controller consists of the nominal open-loop NMPC, which drives the undisturbed system onto its desired trajectory. The second controller consists of an auxiliary NMPC feedback controller, which tries to mitigate the unmodelled dynamics and disturbances of the system. This is done by driving the discrepancy between the nominal NMPC output and the real system states to zero. The formulation of the auxiliary controller consists of a dual NMPC structure described by [43]. The nominal model is initially described using the kinematic model in Eq. (3) which can be expressed as

$$\dot{z} = f(z, \upsilon) \tag{34}$$

where z is the nominal state, and υ is the nominal control input. The optimal control input υ^* to drive the undisturbed system onto the desired trajectory is determined by minimizing the following cost function

$$V_{nom}(t_k, z_e(\tau), \upsilon_e(\tau)) = \int_{t_k}^{t_k+T} \ell(z_e(\tau), \upsilon_e(\tau))d\tau + V_{f,nom} \tag{35}$$

where $\ell(\tau, z_e(\tau), \upsilon_e(\tau)) = \|z_e(\tau)\|^2_{Q_{nom}} + \|\upsilon_e(\tau)\|^2_{R_{nom}}$ is the running cost; $V_{f,nom} = \|z_e(t_k+T)\|^2_{P_{nom}}$ is the terminal cost; z_e is the error between the nominal state and reference trajectories described in Eq. (5); υ_e is the error between the nominal input; Q_{nom}, P_{nom} and R_{nom} are positive definite symmetric weight matrices that tune the states; and T is the time horizon. The open-loop cost function for the nominal state and input trajectories is

$$\begin{aligned} \min_{\upsilon} \quad & V_N(t_k, z_e(\tau), \upsilon_e(\tau)) \\ \text{s.t.} \quad & \dot{q}(\tau) = f(z(\tau), \upsilon(\tau)), \\ & z(t_k) = z_0, \\ & \upsilon(\tau) \in \mathbb{V}, \\ & z(\tau) \in \mathbb{Z}, \\ & z(t_k + T) \in \mathbb{Z}_f \\ & \tau \in [t_k, t_k + T] \end{aligned} \tag{36}$$

where $\mathbb{Z} \in \mathbb{R}^n$ is the nominal state constraints, $\mathbb{V} \in \mathbb{R}^m$ is the nominal input constraints, $\mathbb{Z}_f \in \mathbb{R}^n$ is the nominal terminal constraints, and z_0 denotes the is the initial state selected from the solution of the previous sampling instant such that $z_0 = z^*(t_k, z(t_k-1), \upsilon^*(t_k), \upsilon^*(t_k-1))$. The nominal optimal control input is expressed as $\upsilon^*(t_k)$, where it is applied to the nominal kinematic model and determined at each sampling instance. The states and inputs obtained from the nominal controller are then passed to the auxiliary controller as the reference state and inputs. The design of the auxiliary NMPC follows a similar structure to the nominal controller. Instead of driving the nominal system to the trajectory, the cost function is designed such that the error between the nominal states and the real system's states is driven to stability. Note that the nominal controller has no interaction with the real system; it is an open-loop controller that provides the desired solution of an undisturbed system. The loss function of the auxiliary NMPC may then be defined as

$$V_{f,aux}(t_k, q_e(\tau), u_e(\tau)) = \int_{t_k}^{t_k+T} \ell(q_e(\tau), u_e(\tau))d\tau + V_{f,aux} \tag{37}$$

where $\ell(\tau, q_e(\tau), u_e(\tau)) = \|q_e(\tau)\|^2_{Q_{aux}} + \|u_e(\tau)\|^2_{R_{aux}}$ is the running cost; $V_{f,aux} = \|q_e(t_k+T)\|^2_{P_{aux}}$ is the terminal cost; q_e is the error between the estimated state and the nominal state; u_e is the error between the nominal input; Q_{aux}, P_{aux} and R_{aux} are positive definite symmetric weight matrices that tune the states; and T is the time horizon. The open-loop cost function for the nominal state and input trajectories is

$$\begin{aligned} \min_{u} \quad & V_N(t_k, q_e(\tau), u_e(\tau)) \\ \text{s.t.} \quad & \dot{q}(\tau) = f(q(\tau), u(\tau)), \\ & q(t_k) = \hat{q}(t_k), \\ & u(\tau) \in \mathbb{U}, \\ & q(\tau) \in \mathbb{X}, \\ & q(t_k + T) \in \mathbb{X}_f \\ & \tau \in [t_k, t_k + T] \end{aligned} \tag{38}$$

where $\mathbb{X} \in \mathbb{R}^n$ is the state constraints, $\mathbb{U} \in \mathbb{R}^m$ is the input constraints, $\mathbb{X}_f \in \mathbb{R}^n$ is the terminal constraints, and $\hat{q}(t_k)$ denotes the estimated state vector determined by the EKF after the optimal input from the previous instance ($u^*(t_{k-1})$) is given to the system. Additionally, to ensure robust constraint satisfaction, the tightening of the nominal constraints is set as a simple scaling factor of the initial constraint. This takes the form of $\mathbb{Z} = \alpha\mathbb{X}$ for the nominal state constraints and $\mathbb{V} = \beta\mathbb{U}$ for the nominal input constraints, where α and β are constants between the range of 0 and 1. Terminal constraints are also imposed in both the nominal and auxiliary controller for stability purposes, which can be further assessed in [43].

4.7. Model Predictive Sliding Mode Control

The Model Predictive Sliding Mode Control (MPSMC) follows a similar derivation to the TBNMPC. The first controller consists of the same nominal open-loop NMPC discussed in Section 4.6. The second controller uses a hybrid controller, which consists of an NMPC and SMC, which is known as an MPSMC, described by [31]. The MPSMC uses an NMPC to recast the SMC problem as an optimization problem. The auxiliary controller goal would be to maintain the sliding surface to zero. This is accomplished by defining the following cost function for the auxiliary controller:

$$V_{f,aux}(t_k, s(\tau), u(\tau)) = \int_{t_k}^{t_k+T} \ell(s(\tau), u(\tau))d\tau + V_{f,aux} \tag{39}$$

where $\ell(\tau, s(\tau), u(\tau)) = \|s(\tau)\|^2_{F_{aux}} + \|u(\tau)\|^2_{R_{aux}}$ is the running cost; $V_{f,aux} = \|s(t_k+T)\|^2_{P_{aux}}$ is the terminal cost; s is the sliding surface variable defined in Eq. (20); u is the system control input; F_{aux}, P_{aux} and R_{aux} are positive definite symmetric weight matrices that tune the controller; and T is the time horizon. If the optimal control input u^* drives the sliding surface vector to zero, by the Lyapunov proof in (21), one may assure that the states of the disturbed system also reach stability. Thus, the optimization problem for the auxiliary controller is

$$\begin{aligned} \min_{u} \quad & V_N(t_k, s(\tau), u_e(\tau)) \\ \text{s.t.} \quad & s = s_0 \\ & \dot{q}(\tau) = f(q(\tau), u(\tau)). \\ & q(t_k) = \hat{q}(t_k), \\ & u(\tau) \in \mathbb{U}, \\ & q(\tau) \in \mathbb{X}, \\ & q(t_k + T) \in \mathbb{X}_f \\ & \tau \in [t_k, t_k + T] \end{aligned} \tag{40}$$

where $\mathbb{X} \in \mathbb{R}^n$ is the state constraints, $\mathbb{U} \in \mathbb{R}^m$ is the input constraints, $\mathbb{X}_f \in \mathbb{R}^n$ is the terminal constraints, and $\hat{q}(t_k)$ denotes the

Table 1
Control parameters table.

Parameter	Value
PD parameters	
Parameter	Value
K_{Px}	0.8
K_{Py}	0.3
$K_{P\theta}$	0.6
K_{Dx}	0.1
$K_{D\theta}$	0.1
CLF parameters	
Parameter	Value
K_x	1
K_y	0.7
K_{theta}	0.9
SMC parameters	
Parameter	Value
λ_y	0.6
λ_θ	0.9
Q	diag[1e−1, 1e−1]
P	[5e−1, 1e−1]
NMPC parameters	
Parameter	Value
Q_{mpc}	diag[1, 1, 1.3]
R_{mpc}	diag[0.5, 0.5]
TBNMPC parameters	
Parameter	Value
$Q_{t,nom}$	diag[1, 1, 1.3]
$R_{t,nom}$	diag[1, 1]
$Q_{t,aux}$	diag[1, 1, 1.3]
$R_{t,aux}$	diag[1, 1]
MPSMC parameters	
Parameter	Value
$Q_{m,nom}$	diag[1, 1, 1.3]
$R_{m,nom}$	diag[1,1]
$F_{m,aux}$	diag[1, 1]
$R_{m,aux}$	diag[1,1]
λ_y	0.6
λ_θ	0.7

estimated state vector determined by the EKF after the optimal input from the previous instance ($u^*(t_{k-1})$) is given to the system. The stability of the controller may follow the same routine as the TBNMPC [43] and an MPSMC-specific stability analysis may also be explored in [31].

The tightening of the constraints revolves around the boundary layer thickness Φ designed for the control law in Eq. (25). The boundary layer is used to propagate the constraints of the sliding surface into the states and control inputs of the nominal controller. However, the boundary layer is made time-varying such that it proportionally tightens the sliding surface instead of it being a constant value. This is described as [40]:

$$\dot{\Phi} = -\lambda\Phi + \gamma \tag{41}$$

where λ is the rate of convergence of the boundary layer, and γ is the tuning parameter of the boundary layer. The boundary layer of the surface is then propagated into the states of the nominal controller by designing the tube size constraint. The tube size defines the bounds between the nominal and true system such that $x(t) - z(t) \in S, \forall t \in [t_k, t_k + T]$. If the constraint sets are rearranged for the nominal controller set using the Minkowski set subtraction, the following can be obtained:

$$\mathbb{Z} := \mathbb{X} \ominus \mathbb{S} \tag{42}$$

The tube size is designed to have a similar structure to the surface boundary layer. The structure assures that the size of the tube size has more lenient constraints at the beginning and reaches a tighter constraint over time. The equation can be represented as

Table 2
RMSE values.

Controller	x	y	θ	v	ω	Sum
PD	0.2484	0.2045	0.2144	0.0505	0.0499	0.7678
CLF	0.3181	0.1669	0.0669	0.0288	0.0263	0.6070
SMC	0.2447	0.2140	0.0636	0.02755	0.0337	0.5836
MPSMC	0.1750	0.1481	0.0766	0.0255	0.0327	0.4580
NMPC	0.1186	0.1087	0.01948	0.0320	0.0295	0.3083
TBNMPC	0.1015	0.0665	0.0230	0.0269	0.0300	0.2479

$$\dot{\mathbb{S}} = -\rho\mathbb{S} + \Phi \tag{43}$$

where ρ is the rate of convergence of the tube layer. Furthermore, the constraints of the nominal input are also designed based on the surface boundary layer.

The input constraints are designed based on the upper bound of the maximum control input introduced by the control law. The statement may be denoted by $u = \hat{u} + u_k$, where $\hat{u}$ is the equivalent control input necessary to keep the trajectory on the sliding surface $\dot{s} = 0$ and u_k is the switching control input determined by the control law such that the trajectory is driven to the sliding surface. Note that only the switching control of the SMC drives the uncertainties to stability. If the controllers of Eqs. (30) and (31) are written in the form of $u = \hat{u} + u_k$, the following can be obtained:

$$\begin{aligned} v = \hat{v} + v_k &= (y_e\omega + v_r cos(\theta_e)) \\ &+(P_1 s_1 + Q_1 sat(s_1)) \end{aligned} \tag{44}$$

$$\begin{aligned} \omega = \hat{\omega} + \omega_k = &\left(\frac{\omega_r(1 + \frac{\lambda_\theta}{2} sign(\theta_e)) + \frac{\lambda_y v_r sin(\theta_e)}{2(1+y_e^2)}}{(1 + \frac{\lambda_\theta}{2} sign(\theta_e)) + \frac{\lambda_y x_e}{2(1+y_e^2)}} \right) \\ &+ \left(\frac{P_2 s_2 + Q_2 sat(s_2)}{(1 + \frac{\lambda_\theta}{2} sign(\theta_e)) + \frac{\lambda_y x_e}{2(1+y_e^2)}} \right) \end{aligned} \tag{45}$$

Thus, the bounds of the switching control v_k & ω_k are then used to define the tube size of the control input. The bounds of the control input tube size may be expressed as:

$$\mathbb{V}_{vx} = P_1\Phi_1 + Q \tag{46}$$

$$\mathbb{V}_{\omega} = \frac{P_2 s_2 + Q_2\Phi_2}{(1 + \frac{\lambda_\theta}{2} sign(\theta_e)) + \frac{\lambda_y x_e}{2(1+y_e^2)}} \tag{47}$$

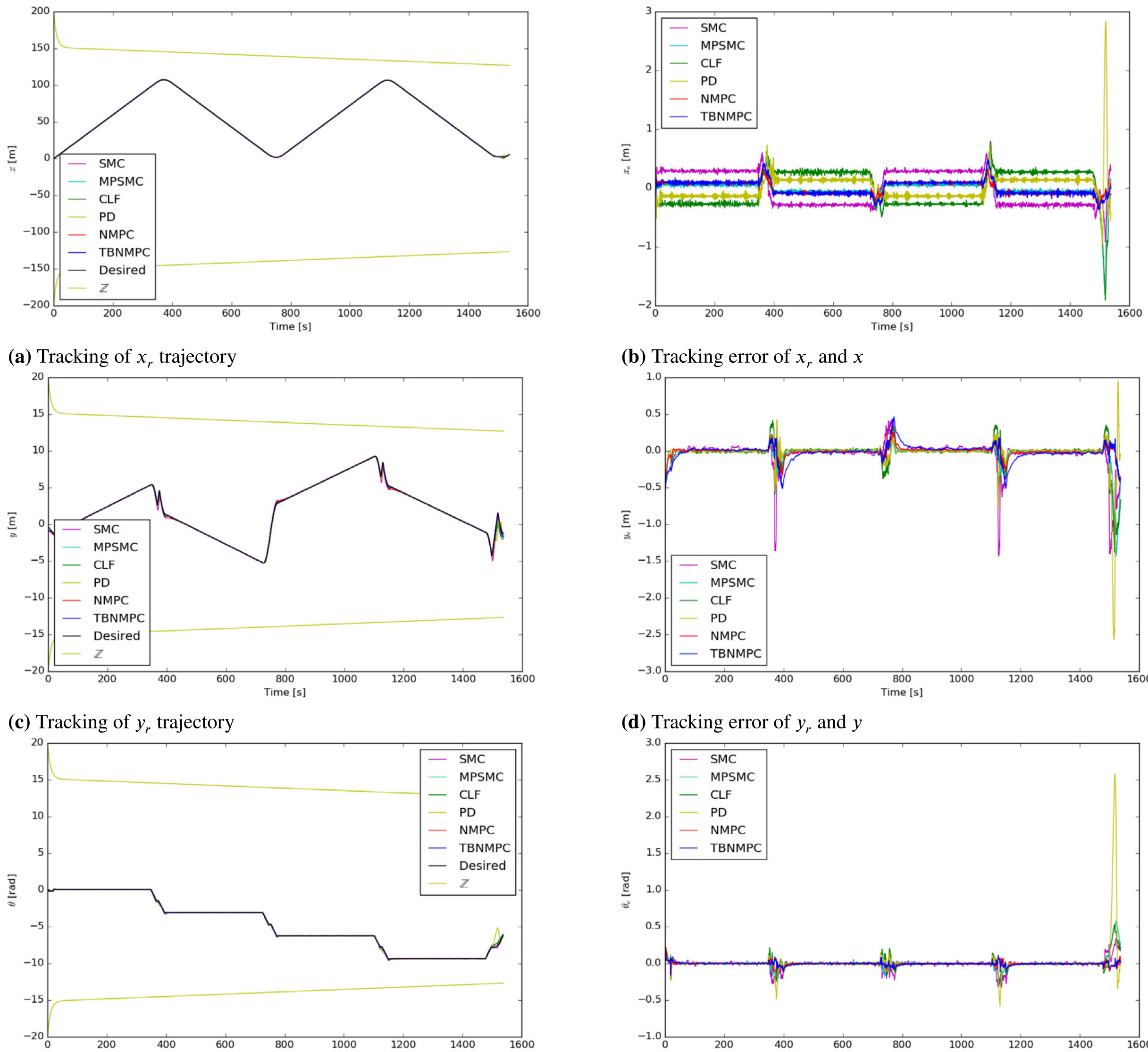
where $\mathbb{V}_{vx}$ is the constraint set for the nominal linear velocity input and $\mathbb{V}_\omega$ is the constraint set for the nominal angular velocity input.

5. Results

The experimental results of the controllers are consistent with the ones in the simulation, but only the field test experiments are presented due to limited space. The parameters for the controllers are provided in Table 1. Validation of the control structures are assessed using the generated path in Fig. 3. The path is set to formulate T-shaped turns to introduce more complex manoeuvres in which the controllers will be assessed. Additionally, the optimal-based controllers use the CasADi toolkit to solve the optimization problem. The optimal control problem is cast as a Nonlinear Programming (NLP) problem using multiple shooting. The sampling rates of the controllers were $T = 0.2$ s, and the prediction horizon

Table 3
IAC values for the field experiment results.

Controller	PD	CLF	SMC	MPSMC	NMPC	TBNMPC
v_x	1212.1937	1190.0362	1188.0317	1181.8350	1189.6999	1183.1618
ω	221.2788	189.6777	187.9554	177.1346	176.1430	176.9600
Sum	1433.4725	1379.7139	1375.9871	1358.9696	1365.8429	1360.1218



(a) Tracking of x_r trajectory (b) Tracking error of x_r and x

(c) Tracking of y_r trajectory (d) Tracking error of y_r and y

(e) Tracking of θ_r trajectory (f) Tracking error of θ_r and θ

Fig. 7. Tracking results.

for both the optimal controllers was set to $N = 10$. The total execution time of the trajectory lasted approximately 25 min or 1538 s. For the controllers without constraint tightening, the control inputs are saturated to $v_x = 1$ m/s for the linear velocity and $\omega = 1$ rad/s for the angular velocity. Starting position is also set to zero for all states since the states all start at zero once the EKF is initialized.

The overall tracking performance of each controller in the X-Y axis is shown in Fig. 8, and its respective control inputs in Fig. 9. The performance was evaluated using the Root-Mean-Square-Error deviation analysis for each state and control input as shown in Table 2. More specifically, the tracking performance of each state may be assessed in Fig. 7. As for the assessment in control effort, it is achieved by analyzing the Integral-Absolute-Control (IAC) as shown in Table 3.

Results of the RMSE also show that when using the PD as the benchmark, performance improvements of 26.49%, 31.56%, 67.64%, 149.04%, 209.72% are realized by the controllers CLF, SMC, MPSMC, NMPC, and TBNMPC, respectively. As for IAC performance, one can also notice the improvements from the PD controller by 3.90%, 4.18%, 5.48%, 4.95%, 5.39% for the CLF, SMC, MPSM, NMPC, and TBNMPC, respectively

The tuning parameters of the controllers in the field experiments were moderately close to the simulated ones. The controllers that required the least amount of tuning were the MPSMC and the TBNMPC. The other controllers such as the SMC, the PD and the CLF required tuning higher weights in the tracking of orientation errors. This was found to be the case as it tended to drive off the trajectory when performing T-shaped turns. As

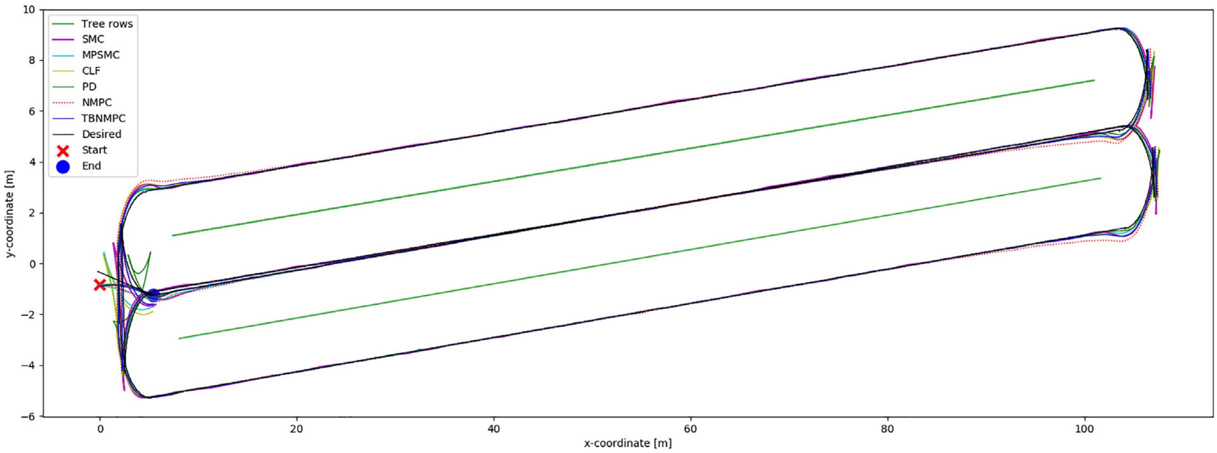
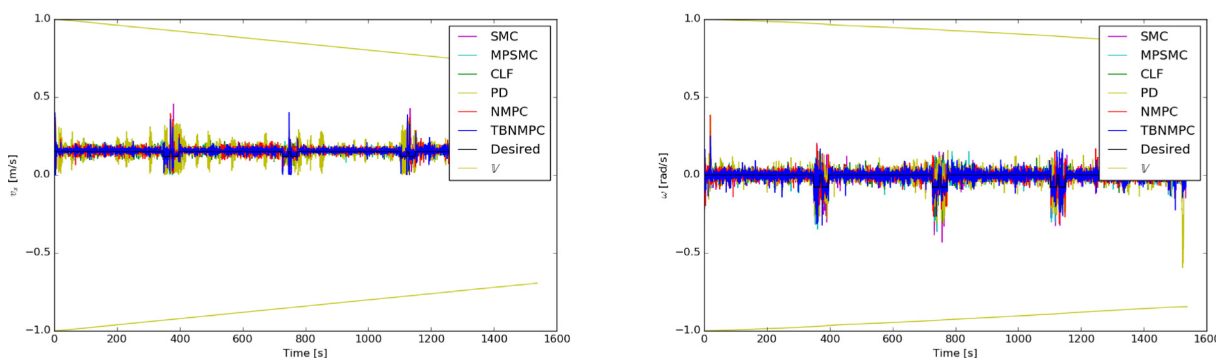


Fig. 8. Field experiment results.



(a) Linear Velocity

(b) Angular Velocity

Fig. 9. Control Inputs.

for the NMPC, the simulated parameters had significant actuation activity in the field experiments. It required lowering the tracking weights of all the states.

The T-turn analysis results of the controllers are determined by taking one section of the small T-shaped turn trajectory and comparing it against another section of a large T-shaped turn trajectory. The comparison is made through the RMSE value of each respective section. The analysis between large and small T-turns showcases the robustness of the controllers in changing turn radius. Results showed that most controllers performed better during small T-turns than large T-turns. However, the controllers that suffered the most when performing large turns were the NMPC, MPSMC, PD and CLF. The NMPC did 17.62% worse than small turns, MPSMC did 81.13% worse, the PD did 113.12% worse, and the CLF did 70.41% worse. Furthermore, the SMC and TBNPC did not show significant changes between the large and small T-turns in field experiments. Results showed that the SMC did 0.5% worse than large turns and TBNMPC did 4.76% worse.

6. Discussion

The evaluation of control performance shows that the best tracking is achieved by the TBNMPC. The relatively low performance of the PD, CLF, SMC and MPSMC may be associated with the design of the controllers. These controllers require the junction of both the y_e and θ_e tracking error to control the angular velocity of the SSMR. In the NMPC and TBNMPC, the controllers do not need to account for the underactuated constraints on the design of the controller architecture. In contrast, the other non-optimal architectures need to consider the underactuated constraint to design the linear and angular controller through the stability analyses. Essentially, the angular controller would need to balance between the lateral error and the orientation error of the SSMR, which may ultimately hinder the longitudinal tracking as well. One may also observe that the propagation of such limitations is seen in the MPSMC. Although the MPSMC uses an optimal control for minimizing the sliding surface, the design limitation limits the tracking capabilities during complex turns. For the non-optimal controllers and the MPSMC to achieve better tracking results, it would require the independent tracking of the lateral and orientation errors of the SSMR. This may be achieved by using polar coordinates to reduce the number of states to track. Further analysis was also made of the performance on large and small T-turns of the controllers.

The turn analysis results of the PD, CLF, NMPC and MPSMC showed to have poor performance when comparing small and large T-shaped turns. The results of the PD and CLF show that Lyapunov proof of angular velocity control is limited when the disturbance of the system changes. Hence, it may be realized that the method guarantees convergence to a certain degree but does not guarantee robustness against changing disturbances. On the other hand, the TBNMPC highlights the robustness of the Tube-Based approach. When comparing the NMPC with the TBNMPC, the NMPC performance suffers more than the TBNMPC during large turns. The approach validates that the robustness of the TBNMPC increases the turn performance over the NMPC by 270.2%. Furthermore, when doing a similar comparison between the SMC and the MPSMC, the same results are not perceived. The robustness of the SMC proves to be superior to the MPSMC. It is important to note that although the SMC remained more consistent during turns, the MPSMC had better RMSE performance in both small and large turns. The lack of consistency of the MPSMC in small and large turns may be associated with the design of the MPSMC. The MPSMC is derived from an open loop NMPC (nominal controller) followed by a reformulated NMPC which seeks to minimize the sliding surface of the SMC through an optimization problem (auxiliary controller). The auxiliary controller rejects the additional disturbances present through the sliding surfaces. However, the sliding functions hold the underactuated constraint in the angular velocity control. Hence, depending on the orientation and lateral weights, the optimization problem has limitations in determining the best angular velocity.

The overall Integral-Absolute-Control (IAC) actuation effort between the controllers shows that the MPSMC had the lowest control effort in comparison to the other controllers. The subsequent controllers that did comparably well after the MPSMC were the NMPC and TBNMPC. The relatively low IAC values may be associated with the optimization of the control input. The model predictive controls hold the penalization of both input and states. Whereas non-optimal controllers such as the PD, CLF and SMC, do not penalize the actuation input; instead, it is fully driven by the control law. Furthermore, the overall performance of the MPSMC shows that the particular composition of the auxiliary controller may minimize the overall actuation effort. The optimization problem of the sliding variable decreases the linear velocity actuation by 2% in comparison to the NMPC and the TBNMPC. On the other hand, the angular velocity had increased activity of about 3% in comparison to the NMPC and the TBNMPC. The increased control activity in the angular velocity of the MPSMC may be related to the underactuated constraint. Therefore, if the underactuated constraint is resolved through the use of polar coordinates, it is likely that the angular velocity actuation effort is minimized as well.

7. Conclusion

This paper presented the experimentation and validation of multiple control architectures of the Skid-Steer Mobile Robot (SSMR) in agricultural environments. The controllers consist of a Proportional-Derivative (PD) controller, a Sliding Mode Controller (SMC), a Control Lyapunov Function (CFL), a Nonlinear Model Predictive Control (NMPC), a Tube-Based Nonlinear Model Predictive Control (TBNMPC), a Model Predictive Sliding Mode Control (MPSMC). The PD controller uses a Lyapunov proof to determine the angular velocity control and is set as the benchmark by which the other controllers will be accessed. The SMC features a new design of the sliding surface functions which provides better tuning of angular velocity control. The CLF uses an existing methodology proposed by [37]. The TBNMPC uses the approach from [43]. Finally, the MPSMC features a new method of implementing TBNMPC with SMC using the kinematic model. Results show that all controllers were able to achieve trajectory tracking successfully by mitigating the unmodelled dynamics of the vehicle and tires in agricultural environments. Furthermore, the results show that the TBNMPC was the best-performing controller overall. The low performance of the non-optimal controllers may have been attributed to the underactuated system constraint present in the design of the controllers. This conclusion is further observed in the MPSMC performance which also inherits the underactuated constraint. Thus, implementing strategies such as trajectory tracking in the polar coordinate space may reduce the number of states, mitigating such problems. Additionally, the robustness of the controllers was accessed by determining their performance in large and small T-shaped turns. The experiments show that the SMC and TBNMPC were more robust to changing disturbances in large and small turns. Furthermore, the IAC results show that the MPSMC had the overall lowest actuation effort. The results are attributed to the optimization problem that provides the optimal input based on state and input penalization. Future works on the research include comparative experiments of the controllers using the dynamic model of the SSMR. The prospective results will allow for further validation of the degree of disturbance rejection that is achieved by using such methodologies.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

The data that has been used is confidential.

Acknowledgments

This research was funded by the Weston Seeding Food Innovation (#SFI19-0349) and Mitacs, Canada (#IT18697) grants.

References

[1] Z. Zhai, J.F. Martínez, V. Beltran, N.L. Martínez, Decision support systems for agriculture 4.0: Survey and challenges, Comput. Electron. Agric. 170 (2020) 105256.

[2] S.N. Young, E. Kayacan, J.M. Peschel, Design and field evaluation of a ground robot for high-throughput phenotyping of energy sorghum 20 (4) 697–722 http://dx.doi.org/10.1007/s11119-018-9601-6.

[3] R.R. Mir, M. Reynolds, F. Pinto, M.A. Khan, M.A. Bhat, High-throughput phenotyping for crop improvement in the genomics era 282 60–72 http://dx.doi.org/10.1016/j.plantsci.2019.01.007 URL http://www.sciencedirect.com/science/article/pii/S0168945218305752.

[4] R. R. Shamshiri, C. Weltzien, I.A. Hameed, I. J. Yule, T. E. Grift, S.K. Balasundram, L. Pitonakova, D. Ahmad, G. Chowdhary, Research and development in agricultural robotics: A perspective of digital farming http://dx.doi.org/10.25165/j.ijabe.20181104.4278 URL https://ntnuopen.ntnu.no/ntnu-xmlui/handle/11250/2595468.

[5] F.Y. Narvaez, G. Reina, M. Torres-Torriti, G. Kantor, F.A. Cheein, A Survey of Ranging and Imaging Techniques for Precision Agriculture Phenotyping 22 (6) 2428–2439 http://dx.doi.org/10.1109/TMECH.2017.2760866 Conference Name: IEEE/ASME Transactions on Mechatronics.

[6] A. Bender, B. Whelan, S. Sukkarieh, A high-resolution, multimodal data set for agricultural robotics: A Ladybird's-eye view of Brassica 37 (1) 73–96 http://dx.doi.org/10.1002/rob.21877 URL http://onlinelibrary.wiley.com/doi/abs/10.1002/rob.21877, _eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1002/rob.21877.

[7] T. Mueller-Sim, M. Jenkins, J. Abel, G. Kantor, The Robotanist: A ground-based agricultural robot for high-throughput crop phenotyping, in: 2017 IEEE International Conference on Robotics and Automation (ICRA), IEEE, 2017, pp. 3634–3639.

[8] P. Biber, U. Weiss, M. Dorna, A. Albert, Navigation system of the autonomous agricultural robot bonirob, in: Workshop on Agricultural Robotics: Enabling Safe, Efficient, and Affordable Robots for Food Production (Collocated with IROS 2012), Vilamoura, Portugal, 2012.

[9] G. Reina, A. Milella, R. Galati, Terrain assessment for precision agriculture using vehicle dynamic modelling, Biosyst. Eng. 162 (2017) 124–139.

[10] S. Rabiee, J. Biswas, A friction-based kinematic model for skid-steer wheeled mobile robots, in: 2019 International Conference on Robotics and Automation (ICRA), IEEE, 2019, pp. 8563–8569.

[11] N.H. Amer, H. Zamzuri, K. Hudha, Z.A. Kadir, Modelling and control strategies in path tracking control for autonomous ground vehicles: a review of state of the art and challenges, J. Intell. Robot. Syst. 86 (2) (2017) 225–254.

[12] R. Xiong, L. Li, C. Zhang, K. Ma, X. Yi, H. Zeng, Path tracking of a four-wheel independently driven skid steer robotic vehicle through a cascaded NTSM-PID control method, IEEE Trans. Instrum. Meas. (2022).

[13] J. Pentzer, S. Brennan, K. Reichard, The use of unicycle robot control strategies for skid-steer robots through the ICR kinematic mapping, in: 2014 IEEE/RSJ International Conference on Intelligent Robots and Systems, IEEE, 2014, pp. 3201–3206.

[14] A.J. Prado, D. Chávez, O. Camacho, M. Torres-Torriti, F.A. Cheein, Adaptive nonlinear MPC for efficient trajectory tracking applied to autonomous mining skid-steer mobile robots, in: 2020 IEEE ANDESCON, IEEE, 2020, pp. 1–6.

[15] J.M. Gregory, G. Warnell, J. Fink, S.K. Gupta, Improving trajectory tracking accuracy for faster and safer autonomous navigation of ground vehicles in off-road settings, in: 2021 IEEE International Symposium on Safety, Security, and Rescue Robotics (SSRR), IEEE, 2021, pp. 204–209.
[16] B. Sebastian, P. Ben-Tzvi, Active disturbance rejection control for handling slip in tracked vehicle locomotion, J. Mech. Robot. 11 (2) (2019) 021003.
[17] J. Aslam, S.-Y. Qin, M.A. Alvi, Fuzzy sliding mode control algorithm for a four-wheel skid steer vehicle, J. Mech. Sci. Technol. 28 (8) (2014) 3301–3310.
[18] Z. Jian, W. Shuang-shuang, L. Hua, L. Bin, The sliding mode control based on extended state observer for skid steering of 4-wheel-drive electric vehicle, in: 2012 2nd International Conference on Consumer Electronics, Communications and Networks (CECNet), IEEE, 2012, pp. 2195–2200.
[19] H. Aithal, S. Janardhanan, Trajectory tracking of two wheeled mobile robot using higher order sliding mode control, in: 2013 International Conference on Control, Computing, Communication and Materials (ICCCCM), IEEE, 2013, pp. 1–4.
[20] A. Filipescu, V. Minzu, B. Dumitrascu, A. Filipescu, E. Minca, Trajectory-tracking and discrete-time sliding-mode control of wheeled mobile robots, in: 2011 IEEE International Conference on Information and Automation, IEEE, 2011, pp. 27–32.
[21] J.H. Lee, C. Lin, H. Lim, J.M. Lee, Sliding mode control for trajectory tracking of mobile robot in the RFID sensor space, Int. J. Control Autom. Syst. 7 (3) (2009) 429–435.
[22] B.R. JBJ, D. Mayne, Model predictive control theory and design, 1999.
[23] T.P. Nascimento, C.E.T. Dórea, L.M.G. Gonçalves, Nonholonomic mobile robots' trajectory tracking model predictive control: a survey, Robotica 36 (5) (2018) 676–696, http://dx.doi.org/10.1017/S0263574717000637.
[24] M.W. Mehrez, G.K. Mann, R.G. Gosine, Stabilizing NMPC of wheeled mobile robots using open-source real-time software, in: 2013 16th International Conference on Advanced Robotics (ICAR), IEEE, 2013, pp. 1–6.
[25] E. Kayacan, E. Kayacan, H. Ramon, W. Saeys, Learning in centralized nonlinear model predictive control: Application to an autonomous tractor-trailer system, IEEE Trans. Control Syst. Technol. 23 (1) (2014) 197–205.
[26] Á.J. Prado, M. Torres-Torriti, J. Yuz, F.A. Cheein, Tube-based nonlinear model predictive control for autonomous skid-steer mobile robots with tire–terrain interactions, Control Eng. Pract. 101 (2020) 104451.
[27] S.S.M.P. Jean-Jacques, E. Slotine, Tube-Based MPC: a Contraction Theory Approach.
[28] D.Q. Mayne, Model predictive control: Recent developments and future promise, Automatica 50 (12) (2014) 2967–2986.
[29] A.G. Richards, Robust constrained model predictive control (Ph.D. thesis), Massachusetts Institute of Technology, 2005.
[30] C.J. Ostafew, A.P. Schoellig, T.D. Barfoot, Robust constrained learning-based NMPC enabling reliable mobile robot path tracking, Int. J. Robot. Res. 35 (13) (2016) 1547–1563.
[31] E. Hill, S.A. Gadsden, M. Biglarbegian, Robust nonlinear model predictive control with model predictive sliding mode for continuous-time systems, J. Dyn. Syst. Meas. Control 144 (3) (2022).
[32] B.D. Anderson, J.B. Moore, Optimal Filtering, Courier Corporation, 2012.
[33] K. Kozłowski, D. Pazderski, Modeling and control of a 4-wheel skid-steering mobile robot, Int. J. Appl. Math. Comput. Sci. 14 (4) (2004) 477–496.
[34] L. Caracciolo, A. De Luca, S. Iannitti, Trajectory tracking control of a four-wheel differentially driven mobile robot, in: Proceedings 1999 IEEE International Conference on Robotics and Automation (Cat. No. 99CH36288C), Vol. 4, IEEE, 1999, pp. 2632–2638.
[35] T. Moore, D. Stouch, A generalized extended kalman filter implementation for the robot operating system, in: Intelligent Autonomous Systems 13, Springer, 2016, pp. 335–348.
[36] E.D. Sontag, A Lyapunov-like characterization of asymptotic controllability, SIAM J. Control Optim. 21 (3) (1983) 462–471.
[37] Y. Kanayama, Y. Kimura, F. Miyazaki, T. Noguchi, A stable tracking control method for an autonomous mobile robot, in: Proceedings., IEEE International Conference on Robotics and Automation, IEEE, 1990, pp. 384–389.
[38] S. Kelouwani, C. Ouellette, P. Cohen, Adaptive nonlinear controller design for differential-drive mobile platforms, in: 2010 IEEE/ASME International Conference on Advanced Intelligent Mechatronics, IEEE, 2010, pp. 1238–1244.
[39] W. Gao, J.C. Hung, Variable structure control of nonlinear systems: A new approach, IEEE Trans. Ind. Electron. 40 (1) (1993) 45–55.
[40] J.-J.E. Slotine, W. Li, et al., Applied Nonlinear Control, Vol. 199, Prentice hall Englewood Cliffs, NJ, 1991.
[41] J.B. Rawlings, K.R. Muske, The stability of constrained receding horizon control, IEEE Trans. Automat. Control 38 (10) (1993) 1512–1516.
[42] L. Magni, D.M. Raimondo, F. Allgöwer, Nonlinear model predictive control, Lecture Notes in Control and Inform. Sci. 384 (2009).
[43] D.Q. Mayne, E.C. Kerrigan, E. Van Wyk, P. Falugi, Tube-based robust nonlinear model predictive control, Internat. J. Robust Nonlinear Control 21 (11) (2011) 1341–1353.

Cesar Wen Zhu received his B.S degree in Mechanical Engineering and M.Sc. degree in Engineering System & Computing from The University of Guelph, Guelph, ON, Canada, in 2020 and 2022, respectively. His current research interest includes the application of path planning nonlinear control theory robotics and deep neural networks.

Elyse Hill received a Bachelors of Science in Mechanical Engineering from the University of Maryland, Baltimore County, Baltimore, MD in 2017. She then completed her Ph.D. degree in Mechanical Engineering at the University of Guelph in Guelph, Ontario in the area of controls in 2021. Dr. Hill was awarded a NASA postdoctoral fellowship position to pursue research towards control of electrified aircraft systems. Her current research interests include model predictive control, fault tolerant control, and nonlinear systems.

Mohammad Biglarbegian received the B.Sc. degree (Hons.) from the University of Tehran, Iran, the M.A.Sc. degree from the University of Toronto Canada respectively and the Ph.D. degree from the University of Waterloo Canada. He is currently an Associate Professor with the School of Engineering, Guelph, Canada and the Director of the Autonomous & Intelligent Control or Vehicles (AICV) Research Laboratory. His research interests include advanced control estimation and optimization for different mechatronics systems particularly advanced autonomous robotics and vehicles. Dr. Biglarbegian received the University of Guelph Research Excellence Award in 2018. He also serves as associate editor or two international journals.

Dr. S. Andrew Gadsden is an Associate Professor in the Department of Mechanical Engineering at McMaster University and is Director of the Intelligent and Cognitive Engineering (ICE) Laboratory. His research area includes control and estimation theory, artificial intelligence and machine learning and cognitive systems. Dr. Gadsden completed his Bachelors in Mechanical Engineering and Management (Business) and then earned his Ph.D. in Mechanical Engineering at McMaster in the area of estimation theory with applications to mechatronics and aerospace systems. He worked as a postdoctoral researcher or nearly three years at the Centre or Mechatronics and Hybrid Technology (Hamilton, Ontario). He also worked concurrently as a Project Manager in the pharmaceutical industry (Apotex Inc., Toronto, Ontario). Before joining McMaster University, Dr. Gadsden was an Associate/Assistant Professor at the University of Guelph and an Assistant Professor in the Department of Mechanical Engineering at the University of Maryland, Baltmore County (USA). He worked and continues to work with a number o colleagues in NASA the US Army Research Laboratory (ARL), US Department of Agriculture (USDA), National Institute of Standards and Technology (NIST), and the Maryland Department of the Environment (MDE). Dr. Gadsden is an elected Fellow of ASME, is a Senior Member of IEEE, and is a Professional Engineer of Ontario. He is also a certified Project Management Professional (PMP). Dr. Gadsden is a reviewer for a number of ASME and IEEE journals and international conferences.

John A. Cline is a professor in the Department of Plant Agriculture, University of Guelph where he teaches undergraduate and graduate students and conducts research in tree fruit pomology. He also serves provides outreach in various capacities to the tree fruit industry. The primary objectives of Prof. Cline's research is to investigate factors that determine yield and fruit quality of tree fruit crops. More recent research has focused on orchard automation to reduce labour input costs. John holds a B.Sc. in soil science (Univ. of Guelph), M.Sc. in horticulture (Michigan State University), and a Ph.D. in Horticulture (University of London, UK). Further details about his program can be found at http://www.plant.uoguelph.ca/jcline.