IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received May 19, 2021, accepted July 15, 2021, date of publication July 20, 2021, date of current version July 26, 2021.

Digital Object Identifier 10.1109/ACCESS.2021.3098524

Nonlinear Model Predictive Path-Following Controller for a Small-Scale Autonomous Bulldozer for Accurate Placement of Materials and Debris of Masonry in Construction Contexts

SUBHAN KHAN AND JOSE GUIVANT, (Member, IEEE)
School of Mechanical Engineering, University of New South Wales (UNSW), Sydney, NSW 2052, Australia

Corresponding author: Subhan Khan (subhan.khan@unsw.edu.au)

This work was supported by the UNSW Sydney under Project PS38925/OP001, and in part by the University International Post-Graduate Award (UIPA) under Grant RSRE7061.

ABSTRACT This paper presents a nonlinear model predictive control (NMPC) scheme for the medium-level control of a small-scale autonomous bulldozer to accurately and safely displace crushed materials of masonry in construction contexts. For this purpose, the controller is required to minimize the error between the achieved and required paths; additionally, the control actions must be smooth for minimizing the mistreatment of equipment, which is intended to operate in the long term, as it is usual in industrial-grade solutions. The proposed NMPC based path-follower can adequately handle the platform's constraints and usual perturbations. In terms of state estimation, a map-based localizer is implemented via an extended Kalman filter (EKF) based on the platform nominal process model and light detecting and ranging (LiDAR) and inertial measurement unit (IMU) sensors measurements. The localizer provides estimates of the platform's pose, necessary for the MPC controller's state feedback. An actual experiment on a modified UGV (Clearpaths Husky A-200) is performed to validate the performance of the proposed control scheme. The UGV is retrofitted with a blade (for pushing material) and appropriate sensors for the necessary perception tasks. Experimental results indicate that the proposed control scheme is robust and suitable for safely pushing the crushed materials, presenting appropriately low deviation from the nominal path and requiring reasonably low processing time.

INDEX TERMS Extended Kalman filter, masonry construction, nonlinear model predictive control, path-follower, small-scale bulldozer, unmanned ground vehicle.

I. INTRODUCTION

The demand for unmanned ground vehicles (UGVs) is rapidly growing in the construction, transportation, civil and military applications due to the reliability, efficiency, and advancement in the technology [1]–[3]. Small-scale UGVs play a useful role in pushing the materials and debris [4]–[7] in construction areas. These materials could also be used for masonry construction. Some of the unused crushed bricks and stones remain on the ground during a unit or building construction. For this purpose, small-scale platforms may operate in a full or semi-autonomous fashion. It can depend on how the high-level planning is generated (depending on the degree of participation of the human operators).

Control of autonomous bulldozers is challenging due to multiple factors. First, bulldozer blades may require to be raised or lowered by an operator (often by remote or automatic control) [8]. Second, they are often operated in an uncertain and uneven environment [9]. In addition, construction bulldozers have to deal with complex sites and with measurement errors, which lead to hazardous situations in the operations [10]. Thus, the path-following control problem is relevant for achieving reliability, and safety [11], [12]. A platform needs to achieve it by applying sufficiently gentle control actions; that additional requirement usually results in saving energy and treating the equipment adequately, in the same way a well-trained human driver should do.

The associate editor coordinating the review of this manuscript and approving it for publication was Emanuele Crisostomi.

Diverse techniques have been applied in related problems; such as the Lyapunov-based linear quadratic regulator (LQR) applied to solve the guidance problem for specific non-holonomic systems [13], a robust path following controller for autonomous electric vehicle [6], implementations of fuzzy logic control [14], sliding mode control [15]. Additionally, some implementations in the context of articulated vehicles [16], [17], fault-tolerant control [18], robust tracking controller [19], and adaptive MPC [20]. These existing techniques can properly solve the tracking problem; however, the vehicles are usually treated as rigid points that ignore the boundaries while following the path. For practical applications, it is also important limiting the required processing cost. An onboard processor performs the control processes whose resources are shared by multiple expensive processes (terrain modeling, localization, and safety). All those processes operate in real-time, usually under limited resources, as industrial-grade computers are typically less powerful than state-of-the-art consumer electronics computers. There are pros and cons associated with each control technique regarding processing cost and accuracy in practice. Model predictive control (MPC) is a great contender to solve the path-following problem for the small-scale autonomous bulldozers due to its ability to handle constraints [21], [22].

In recent years, MPC has been used in construction-based applications. For instance, to optimize the energy usage of buildings [23], for experimental analysis of heating system [24]. For more generalized applications, MPC is now being used for mitigating fatal accidents in construction [25], for keeping balanced slurry pressure in shield tunneling [26], and for excavation [27].

Many researchers have proposed solutions to solve the tracking problem of UGVs using MPC controllers. For instance, a robust formulation of MPC to handle the constraints which consider additive disturbances by only focusing on the simulations [28], a robust MPC for the linearised error dynamics model used by [29] solves the tractor-trailer system's tracking problem with low tracking errors. A simulated trajectory tracking problem solved by [30] has reasonable tracking errors, a path-tracking controller based on MPC, and nonlinear filtering [31], and a tracking error-based MPC for mobile robots in real-time applications [32]. However, for practical applications of MPC, the aspect of its processing cost must be addressed, for guarantying sufficiently low processing time at each instance of sampling and control, thus allowing it to operate under real-time constraints. For this purpose, a generalized minimum residual (GMRES) reduces the computational cost [33], experimental validation of Linear MPC(LMPC) and Nonlinear MPC(NPMC) are compared with nonlinear moving horizon estimation in [34], experimental verification of cars subject to measurable disturbances are investigated in [12]. These techniques perform well in simulations and specific experimental contexts; however, improvements in accuracy and computational time are still required, mainly for specific industrial applications, such as those in construction contexts. In addition, further investigation is required for nonlinear MPC to solve construction applications providing robust performance.

In terms of existing work on autonomous bulldozers, high-level planning based on artificial intelligence is proposed in [35]. The authors compare their proposed autonomous planning and the manual operation typical of an experienced human operator. They showed that their proposed approach outperformed the typical human operator. Similarly, an efficient path to compute given tasks to autonomous bulldozers [36], [37] and automatic material distribution operation [38]. However, these studies focused on the path planning of material distribution of autonomous bulldozers, not including how to follow those generated trajectories neither solving them efficiently.

In this paper, we propose an efficient nonlinear model predictive path-tracking controller for the autonomous bulldozer used in [36], [37], [39] to push the materials of masonry in the safe destination zone. We have run all the processes under the industrial-grade system used by [40] and [41]. It is challenging to safely place materials and debris of masonry in a goal location in practical work. Therefore, we want to design a controller that can handle the constraints and predict the systems' response to the upcoming behavior. Thus, we opted for NMPC for this task. So, our main contributions based on the above challenges are as follows:

1) To design and implement a nonlinear model predictive control to solve the path-following problem for small-scale autonomous bulldozers.
2) To investigate the accuracy and processing time of the proposed control technique.
3) To safely push the materials and debris of masonry in a goal location using the proposed control scheme and localization process.

The remaining paper is organized as follows; in Section 2, the problem is formulated, and the platform's kinematics is discussed. Section 3 presents the design of a Nonlinear MPC scheme for the small-scale bulldozer. In Section 4, the localization process is covered. Section 5 discusses the results of the proposed control scheme. Finally, Section 6 concludes the paper.

II. PROBLEM FORMULATION

A. PLATFORM'S MODEL

This paper has used the unicycle configuration for the small-scale bulldozer in the global frame position coordinates $O : (X,\ Y)$ and body frame position coordinates $o : (x,\ y)$ as illustrated in Figure 1. The control actions produced by the wheels are linear velocity $v = (v_l + v_r)/2$ and angular velocity $\omega = (v_l - v_r)/L$. The distance between the rear wheels, rear and front wheels, and the wheels' radius is B, L, and r, respectively. The blade's length is b_1, which is attached to the UGV with a distance of b_2. The control actions produced by the system is $\mathbf{u} = [v,\ \omega]^T$. In addition, when operating in a 2D context, the platform has three degrees of freedom with translation motion in the form of positions (x, y) and orientation as a heading angle θ. Now, considering the

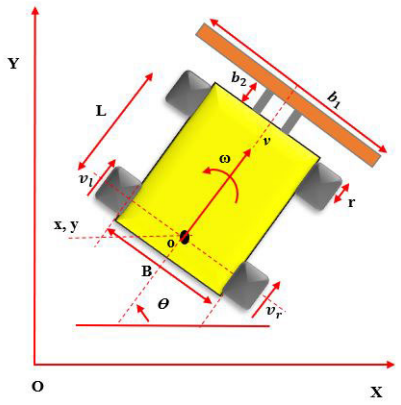

FIGURE 1. Configuration of the small-scale bulldozer.

following unicycle kinematics model for the UGV:

$$\dot{\mathbf{p}} = f(\mathbf{p}, \mathbf{u}) = \begin{bmatrix} \cos\theta & 0 \\ \sin\theta & 0 \\ 0 & 1 \end{bmatrix} \cdot \mathbf{u}, \tag{1}$$

where $\mathbf{p} = [x,\ y,\ \theta]^T$ consists of states of the vehicle. The above equation (1) can be approximated at every time-step by using Euler's approximation under adequate sampling time. Thus, making the resulting nominal process model as follows:

$$\mathbf{p_{k+1}} = \mathbf{p_k} + T_s \begin{bmatrix} \cos\theta_k & 0 \\ \sin\theta_k & 0 \\ 0 & 1 \end{bmatrix} \cdot \mathbf{u_k} \tag{2}$$

where T_s is the sampling time and $k = 0, 1, 2, \ldots$ are the number of samples.

B. PROBLEM STATEMENT

For a reference path of the vehicle $\mathbf{p_r} = [x_r,\ y_r,\ \theta_r]$, the UGV should track the path with sufficient accuracy while moving the crushed materials/stones to a desired goal. Figure 2 represents the 5-steps to be followed by the UGV to express the problem to achieve the material placement task visually. Initially, the UGV will approach the crushed stones/material with a proposed average speed. In step 2, as the platform is not allowed to turn when pushing material, a linear path is followed during the pushing action. The platform can turn and reverse in any other situation, e.g., positioning before a pushing action. In addition, pushing actions are performed at a low speed. In the example, a planner decides a plan that divides the task into two pushing actions because an obstacle impedes the task to be achieved in just one action. In step 3, the bulldozer should reverse since the goal location is perpendicular to the current pose. In step 4, the vehicle should move forward towards the crushed stones by correcting the actual pose $\mathbf{p}$. Finally, the bulldozer should reduce the speed before reaching the stones, pushing them at low speed, following a linear path to reach the temporary destination of the stones finally.

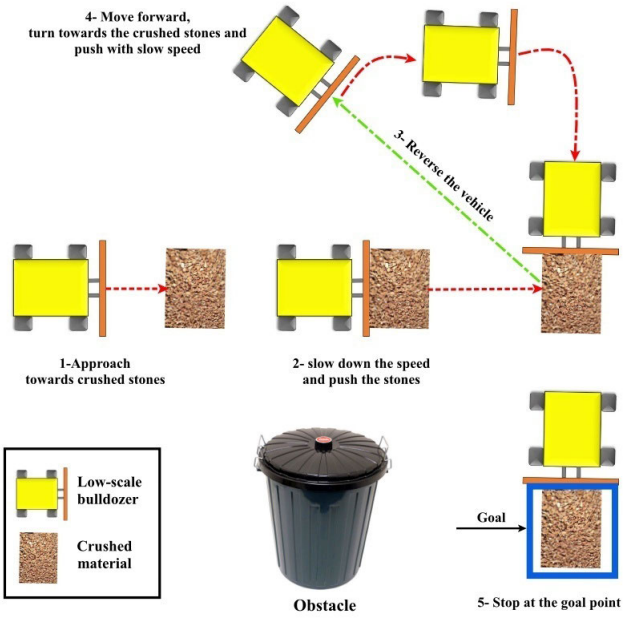

FIGURE 2. Proposed task for the small-scale bulldozer (it turn when pushing material).

III. DESIGN OF NMPC

To solve the path tracking problem, consider the following nonlinear model of the platform as:

$$\mathbf{p}_{k+1} = \mathbf{f}_a(\mathbf{p}_k, \mathbf{u}_k), \tag{3}$$

with states $\mathbf{p_k} \in \mathbb{R}^n$ and control inputs $\mathbf{u_k} \in \mathbb{R}^{\geq}$. The function f_a is not exactly known process model due to the system being perturbed by disturbances. Therefore, (3) can be expressed as

$$\mathbf{p}_{k+1} = \mathbf{f}(\mathbf{p}_k, \mathbf{u}_k) + \mathbf{d}(\mathbf{c}_k) \tag{4}$$

where the bounded disturbances have the dependency on $\mathbf{c}_k \in \mathbb{R}^p$, the nonlinear process model $\mathbf{f}(.)$ is the assumed process model (nominal), which represents our knowledge of $\mathbf{f_a}$. While $\mathbf{d}(.)$ is assumed to be a Gaussian process disturbance model as described [42].

NMPC's main goal is to find a set of a control sequence for predicting the system's future response over a given horizon. Therefore, the cost function to be minimised over the k samples is consists of stage cost $L(\mathbf{p}, \mathbf{u}) = (\mathbf{p} - \mathbf{p_r})^T \mathbf{Q}(\mathbf{p} - \mathbf{p_r}) + \mathbf{u}^T \mathbf{R}\mathbf{u}$ and terminal cost $L_N(\mathbf{p}, \mathbf{u}) = (\mathbf{p_N} - \mathbf{p_r})^T \hat{\mathbf{Q}}(\mathbf{p_N} - \mathbf{p_r}) + \mathbf{u}^T \hat{\mathbf{R}}\mathbf{u}$. The matrices $\mathbf{Q},\ \hat{\mathbf{Q}} \in \mathbb{R}^{Kn \times Kn}$ and $\mathbf{R},\ \hat{\mathbf{R}} \in \mathbb{R}^{Km \times Km}$ are the positive semi-definite and positive definite, the sequence of predicted states is $\mathbf{p} = (\mathbf{p}_{k+1} \ldots \mathbf{p}_{k+K})$, the reference states has a sequence of $\mathbf{p}_r = (\mathbf{p}_{r,k+1} \ldots \mathbf{p}_{r,k+K})$, and control actions have a sequence $\mathbf{u} = (\mathbf{u}_k \ldots \mathbf{u}_{k+K-1})$. The sequence of predicted states $\mathbf{p}$ is computed using the Sigma-Point Transform used by [42]. The net cost function can be defined as follows:

$$J(\mathbf{p}, \mathbf{u}) = L_N(\mathbf{p}, \mathbf{u}) + \textstyle\sum_{i=1}^{N-1} L(\mathbf{p}, \mathbf{u}) + \mathbf{d}(c_k + i) \tag{5}$$

subject to the following constraints:

$$\begin{cases} -\mathbf{u}_{min}(k+i) \leq \mathbf{u}(k+i) \leq \mathbf{u}_{max}(k+i) \\ -\mathbf{p}_{min}(k+i) \leq \mathbf{p}(k+i) \leq \mathbf{p}_{max}(k+i) \end{cases} \tag{6}$$

In addition, the above constraints can be defined as a complete set which compose both the state and input constraints, $\mathbf{a_i} \leq 0, i = 1 \ldots 2N(n+m)$, where N represents the prediction horizon. A constrained optimization problem can be defined to compute the optimal value of control action, as follows:

$$\mathbf{u}_{opt} = \arg\min_{u} J(\mathbf{p}, \mathbf{u}) \tag{7}$$

subject to following constraints:

$$\begin{cases} \mathbf{p}_{k+i+1} = \mathbf{f}(\mathbf{p}_{k+i}, \mathbf{u}_{k+i}) + \mathbf{d}(c_{k+i}), \\ i = 0 \ldots N-1, \\ \mathbf{a}_i(\mathbf{p}, \mathbf{u}) < 0, i = 1 \ldots N_c \end{cases} \tag{8}$$

For $k = 1$, is the currently estimated UGV's pose; so that the MPC is synchronized to the last valid estimates. The term $N_c = 2N(n+m)$ and control action are subject to both the equality and inequality constraints. A slack variable $\mathbf{w} = (w_1 \ldots w_{N_c})$ and a logarithmic barrier function [43] can be introduced to approximate the inequality constraints as follows:

$$\mathbf{u}_{opt} = \arg\min_{u} J(\mathbf{p}, \mathbf{u}) - \mu \sum_{i=1}^{N_c} \log(w_i) \tag{9}$$

subject to following constraints:

$$\begin{cases} \mathbf{p}_{k+i+1} = \mathbf{f}(\mathbf{p}_{k+i}, \mathbf{u}_{k+i}) + \mathbf{d}(c_{k+i}), \\ i = 0 \ldots N-1, \\ \mathbf{a}_i(\mathbf{p}, \mathbf{u}) - w_i = 0, i = 1 \ldots N_c \end{cases} \tag{10}$$

The symbol μ has been opted to be close to zero throughout the optimization process. The optimization process begin with an initial guess $\bar{\mathbf{u}}$ for the applied control sequence with $\mathbf{u} = \bar{\mathbf{u}} + \delta\mathbf{u}$. The term $\delta\mathbf{u}$ is a small deviation of control action; this enables the input constraints to compensate for the delta function in the cost function. It is recommended to use the initial guess for $\bar{\mathbf{u}}$ same as the previous time-step. In our optimization case, we have used $\bar{\mathbf{u}} = 0$ for the first time-step. To finalize the optimization process, we have used Lagrange methods [43].

$$\begin{aligned} L(\mathbf{s}) = J(\mathbf{p}, \mathbf{u}) &- \mu \sum_{i=1}^{N_c} \log(w_i) \\ &- \sum_{i=0}^{N-1} \lambda^T (\mathbf{p}_{k+i+1} - \mathbf{f}(\mathbf{p}_{k+i}, \mathbf{u}_{k+i}) - \mathbf{d}(c_{k+i})) \\ &- \sum_{i=1}^{N_c} \zeta(c_i(\mathbf{p}, \mathbf{u} - w_i)) \end{aligned} \tag{11}$$

The function $L(s)$ is the Lagrangian, $\lambda_i \in \mathbb{R}^n$, all the remaining parameters are in $\mathbf{s} = (\mathbf{p}, \mathbf{u}, \mathbf{w}, \lambda, \zeta)$. For optimal condition, consider $\nabla L(s) = 0$. and $\mathbf{s} = \hat{\mathbf{s}} + \delta\mathbf{s}$ described in [42]. A threshold $-5cm \leq \eta_i \leq -5cm$ is set for the tracking of each way-point.

Algorithm 1: Nonlinear MPC for Path-Tracking

Data: $\mathbf{p_k}$, $\hat{\mathbf{p}}_k$ and $\mathbf{s}_i$
Result: $\mathbf{u}_{opt}$
initialization: $\hat{\mathbf{s}} = \mathbf{s}_i$;
while $\|\mathbf{p(k)} - \mathbf{p_r(k)}\| \geq \eta_i$ **do**
  Get the sequence of actual state $\mathbf{p}(k)$ at runtime
  estimated state $\hat{\mathbf{p}}(k)$, (4), and $\hat{s}$;
  if *Bulldozer has a contact with crushed stones* **then**
    slow down the speed of the bulldozer;
    update the constraints;
  else
    maintain the current speed;
    update the constraints;
  compute $\nabla L(\mathbf{s}) = 0$ and solve $\delta\mathbf{s}$;
  Update the value of $\hat{s}$

Figure 3 illustrates the block diagram of the complete system to address the control signals provided to the small-scale bulldozer. The generated path first goes to the comparator to compute tracking error fed to NMPC and the current state. The processing of NMPC produces an optimal solution for the control actions sent to the actual platform. Thus, the feedback control scheme drives the existing platform to the proposed path.

IV. LOCALIZATION

A localization process is used as the path follower continuously requires an estimate of the pose (2D position and heading). Those estimates are provided by the localization process based on sensor data fusion of measurements provided by a LiDAR and a gyroscope. The localization process used in this application is a well-known approach in the Robotics community, a map-based localizer, i.e., it uses a small number of landmarks, which are deployed in the area of operation (usually on the boundaries of the area of operation, for minimizing disturbing the regular operation of the platform and other agents.) Each time a LiDAR scan is acquired (for the LMS291, in the mode of operation being used, about 38HZ), the scan is pre-processed for performing feature extraction, particularly for detecting the poles. After the poles are detected, a data association process is performed, in which the detected objects are associated with the known map's landmarks. Those successfully associated landmarks allow us to perform updates in the EKF fusion process. The prediction steps of the EKF are based on a kinematic model, whose only input is the heading rate, which is provided by one of the IMU gyroscopes. Although the platform has wheel encoders that would allow measuring longitudinal velocity, those measurements are not used due to the low reliability when slippage is present, usually in certain types of terrain. Consequently, the state vector also includes the longitudinal velocity, estimated along with position and

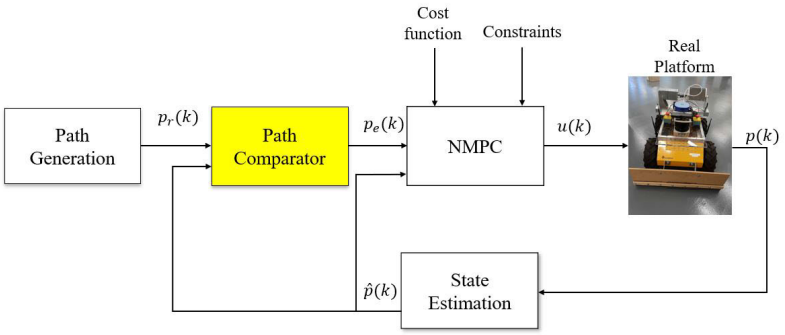

FIGURE 3. Block diagram representation of the path tracking controller for small-scale bulldozer.

the heading. The slow time-varying gyroscope's bias is also estimated in addition to those four states without compromising the system's observability. All the onboard processor measurements are fed to EKF, which estimates the pose of the Husky A200.

A. ESTIMATING POSE, SPEED, AND THE GYROSCOPE BIAS

Considering the case in which speed measurements are not permanently available (e.g. no such sensor is installed on the platform). This concept is also useful when the speed measurements are available but are of low quality or are sampled at an inadequate sampling rate. In such cases, the speed can be estimated in addition to the rest of the states of interest. We also estimate the bias of the vertical gyroscope; consequently, we propose the following states and input variables for the process model: $\boldsymbol{X} = [x\,, y\,, \theta\,, b\,, v]^T$ and $u(t) = \omega(t)$; in which the couple $(x,\ y)$ is the 2D position, θ the heading, v the longitudinal velocity, and the variable b represents the gyroscope's bias. The model's input ω is the biased angular rate which is measured from the gyroscope. The analog process model is

$$\frac{d}{dt}\mathbf{X}(t) = \frac{d}{dt}\begin{bmatrix} x(t) \\ y(t) \\ \phi(t) \\ b(t) \\ v(t) \end{bmatrix} = \begin{bmatrix} v(t)\cdot cos(\phi(t)) \\ v(t)\cdot sin(\phi(t)) \\ \omega(t) - b(t) \\ 0 \\ 0 \end{bmatrix} \tag{12}$$

with

$$\mathbf{X}(t) = \begin{bmatrix} x(t) \\ y(t) \\ \phi(t) \\ b(t) \\ v(t) \end{bmatrix} \in \mathbb{R}^5, \quad \mathbf{u}(t) = \omega(t) \in \mathbb{R}^1$$

The above equation is of a nominally constant velocity kinematic model. The state $v(k)$ is the longitudinal velocity of the platform. The variable $b(k)$ is the gyroscope bias, which is usually slowly time-variant. In theory, we could use an accelerometer for measuring accelerations, and avoid this simplification. However, we do not try that approach in this case because accelerometers are not reliable enough for this purpose. An approximated discrete-time model (based on Euler's approximation), for a sample time T_s, is the following one:

$$\mathbf{X}(k+1) = X(k) + T_s \cdot \dot{X}(k) \tag{13}$$

B. UNCERTAINTY IN-PROCESS MODEL

Several sources of uncertainty pollute this nominal process model; one is related to the existence of translation in the transversal direction (perpendicular to heading), which means that the non-holonomic constraint is not strictly respected. Another discrepancy is that the real longitudinal velocity is not constant as there exists an acceleration which is unknown but still known to be bounded $\|a(t)| < a_{max} = 1.2m/s^2$, for this particular platform. Consequently, we can assume that

$$\begin{cases} \Delta v(k) = v(k) - v(k-1), \\ -T_s \cdot a_{max} < \Delta v(k) < T_s \cdot a_{max} \end{cases} \tag{14}$$

The speed variation, $\Delta v(k) = v(k) - v(k-1) = 0$ is unknown and thus considered to be noised, modelled as a random variable (RV). That RV is not white, however as it bounded and it fluctuates around 0, it is assumed to be GWN, in the estimation process. This approximation results in the following process model of the longitudinal velocity:

$$\begin{aligned} v(k+1) &= v(k) + T_s \cdot 0 + \zeta_{dv}(k), \\ \sigma_{dv} &= T_s \cdot a_{max} \end{aligned} \tag{15}$$

where the component $\zeta_{dv}(k)$ represents the uncertainty, which pollutes the process model of the speed state. The proposed value for σ_{dv} approximates the standard deviation of this uncertainty.

For the gyroscope's bias, an initial expected value $b(0) = 0$ is assumed, and initial standard deviation $\sigma_b = 2\ deg/s$. Its process model is that of random constant, $b(k+1) = b(k)$; however, it is known to be slowly time-varying, but having unknown variation, consequently certain GWN process noise is assumed, $b(k+1) = b(k) + T_s\sigma_{\Delta b}$. Provided that we have

sufficient available observations of range (r) or/and bearing (α), the system would be observable, i.e. the full state vector, including the longitudinal velocity, would be estimated.

C. OBSERVATION MODEL

The observations related to each observed landmark are range and bearing:

$$\mathbf{Y} = \begin{bmatrix} r \\ \alpha \end{bmatrix} \tag{16}$$

Their observation models are:

$$h(\mathbf{X}) = \begin{bmatrix} h_1(x, y, \theta, v) \\ h_2(x, y, \theta, v) \end{bmatrix} = \begin{bmatrix} r \\ \alpha \end{bmatrix} = \begin{bmatrix} \sqrt{(x_a - x)^2 + (y_a - y)^2} \\ \tan^{-1}(y_a - y, x_a - x) - \theta + \frac{\pi}{2} \end{bmatrix} \tag{17}$$

A couple of constants (x_a, y_a) is the center of geometry of the particular landmark involved in the observation. In the EKF pose estimator, the covariance matrices are based on the following assumption about the noises and uncertainties in process and observation models: the standard deviation of noise in gyroscope is $1\,deg/s$ (free of bias noise), the standard deviation of initial bias is $1/deg/s$, the standard deviation of bias drift is $0.01/deg/s)/s$, the standard deviation of noise in range measurement is 15 cm, and the standard deviation of noise in bearing measurement is $2.5\,deg$ (noise characteristics include LiDAR noise, errors due to pitch and roll effects on LiDAR measurements due to inclination respect to the nominally horizontal plane). Even in cases in which we do measure the speed, we may have a slow sensor (whose sample rate is not the one we need); additionally, the measurements from that sensor may be noisy; which means it is convenient to use an estimated version of it, i.e., we estimate it, in place of using it directly. In such a case, we can use the speed measurements for implementing observations (updates). Although speed measurements (from encoders) are not reliable due to possible slippage, those can be used at low frequency as observations and are assumed to have the noise of covariance higher than that of the sensor operating in good conditions. Those measurements are polluted with colored noise (far from being GWN) during slip episodes. However, it is still useful during the initialization stages, in which the covariance of the estimates is relevant, or in regions of operation in which the observability of the estimation process is weak due to the poor availability of landmarks. However, during the system's usual operation and the experiments, it is assumed that a rich deployment of landmarks is present. That assumption is realistic in the sense that the areas of operation, in industrial contexts, can be populated with the low-cost poles used as landmarks. Figure 6, some of those navigation landmarks (orange poles) can be seen.

V. RESULTS AND DISCUSSIONS

In this section, experimental results are presented. We have tested our proposed control scheme on a Clearpath Husky A200, which weights 50 Kg, to perform bulldozing tasks. To perform the real experiment, the controller was run in the onboard processor with an Intel Celeron 1.60 GHz dual-core processor with 2GB of RAM. The NMPC control actions were generated at 40 Hz. The LiDAR (LMS291) scans are pre-processed, feature extraction for detecting the poles for localization. The range and bearing observations are generated based on these detected poles. The LiDAR was read via RS422/USB by the onboard computer at about $38\,Hz$; prediction operates at $200\,Hz$, the onboard 3D camera is Primesense Carmine of which has depth image size VGA(640×480), and IMU via USB runs at around 200 Hz. The maximum vehicle speed is $1.0m/s$ with a rated payload of $20kg$. The platform and components used for the experiment are shown in Figure 4.

The gain tuning of Q and R are set to get a smooth and reasonable applied control action as per the experiment's requirement based on the masonry application. However, these values are not fixed, and variation could result in a discontinuous path and aggressive control actions. For circle shape trajectory, the values of Q and R are $diag[5\,,\,5\,,\,0.1]$ and $diag[0.1\,,\,0.1]$ respectively. The weights for the terminal cost are selected as $\hat{Q}$ and $\hat{R}$ are $diag[50\,,\,50\,,\,10]$ and $diag[1\,,\,1]$ respectively. For the saturation of $u(k)$, although the vehicle supports a maximum speed of 1 m/s in the masonry experiment, we have used 0.15 m/s maximum speed and 45 deg/s angular rate. In addition, for circular shape trajectory, we have used 0.25 m/s maximum speed and 45 deg/s angular rate. These control input constraints are set on the NMPC, making sure that the optimizer respects these constraints. The prediction horizon for both experiments are set to $N = 10$. The selection of these values is based on the Husky A-200 and the proposed masonry application's control specifications.

A. PATH GENERATION

The planer updates the path at a lower frequency, e.g., 1HZ. The planner always considers the approximated current platform's pose, the last estimated terrain description, and certain high-level logistic constraints; because the NMPC operates at a higher rate, it always uses the last planner's output. Because the planner always proposes a path synchronized with the last estimated platform's pose, the overall process is smooth.

The planner reduces the problem's dimensionality by generating sequences of piece-wise linear (PWL) path segments, trying to solve the problem of reaching the object of interests (OOIs) and pushing them to the destination dumpling area. Because the segments are long enough, it is assumed that the NMPC controller does always have enough time and space to align the platform with each segment properly. It is relevant because we need the machine to be in proper pose at the points of encounter and release of the OOIs, and also to follow well linear paths when the blade is pushing objects (this is usual in this class of machines: the machine does not turn when is pushing loads; even human drivers do respect this practical rule). The machine is allowed to turn when it is not pushing material. The planner considers that

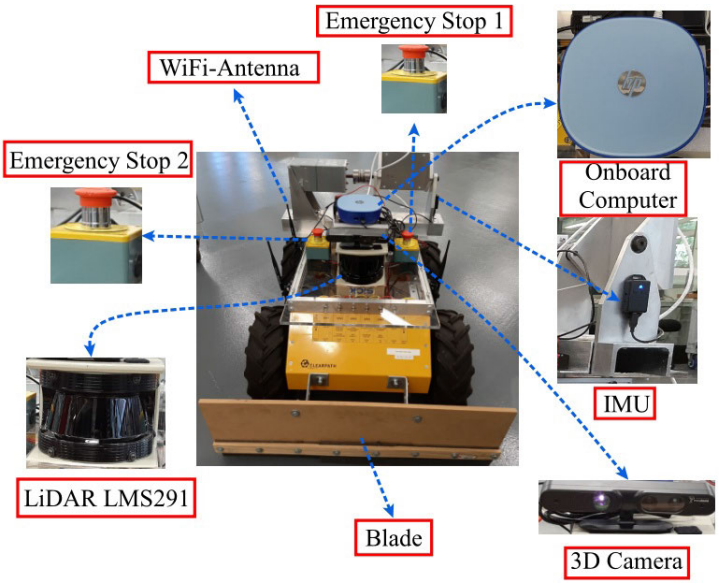

FIGURE 4. Small-scale bulldozer (Husky-A200) with components used for the experiment.

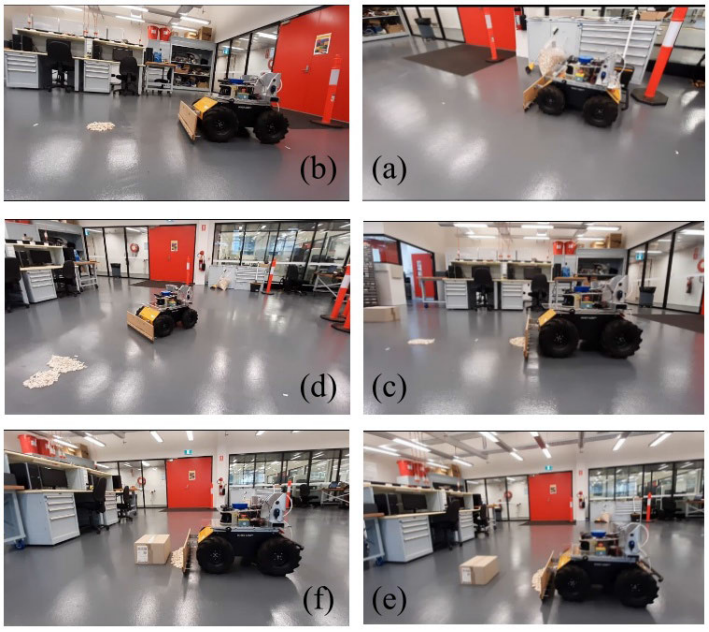

FIGURE 5. Small-scale bulldozer in the experimental environment for pushing the proposed construction application. Experiment video is available in [44].

restriction, so the paths are consistent with those requirements. For instance, if a load needs to be moved to a position and the platform needs to avoid obstacles, the path would be PWL because we cannot keep turning to follow a smooth curved path. The load is pushed linearly up to an intermediate point (following a linear segment), and then the UGV attacks the OOI again for following a second segment. Each time the UGV switches to a new segment, it needs to maneuver/reverse, take a proper position to attack the OOI properly, and push it linearly to the planned subsequent position. The paths generated by the planner usually contain a low number of PWL segments because each transition from one segment to the following implies a maneuver whose cost is considered in the planner's optimization. As a result of this overall process, the NMPC module focuses on following the currently proposed PWL paths, which are assumed to be feasible and well designed to avoid exposing the platform to unnecessary strict requirements, except if there are no

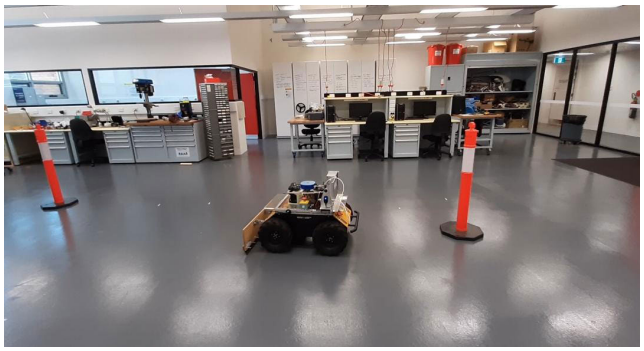

FIGURE 6. Set of 2 low-cost poles used as a landmarks.

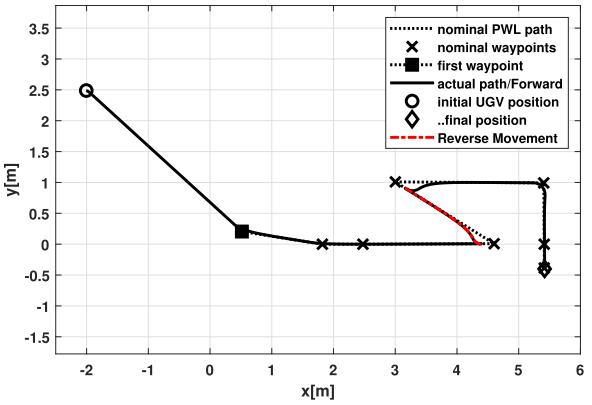

FIGURE 7. Actual path following of the UGV received from the planner.

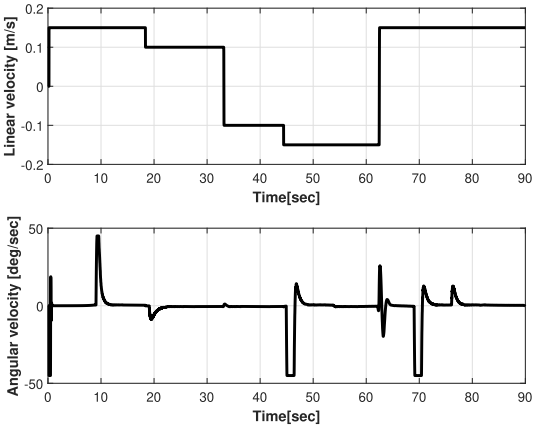

FIGURE 8. Control actions, required by the NMPC controller. These values are used as set-points of the low level controllers which actually operate the platform's electric motors for controlling the rotational speeds of the wheels. a) Longitudinal velocity. b) Angular velocity (heading rate).

alternatives. The planner is closing the loop at a high level, and the NMPC closes the loop at a medium level. The accuracy of the NMPC module is instrumental in guaranteeing that the overall process is feasible. The planner is designed to apply in the same way, and a human user would safely and efficiently perform the same complex sequence. This separation of the control process into levels allows high flexibility.

B. EXPERIMENTAL RESULTS

To test the system, we have used our lab to perform several experiments. In Figure 5, the experiment is summarised through six different steps: a) represents the vehicle's initial pose, b) the vehicle is approaching the first way-point and trying to reach the crushed stones, c) the vehicle pushes the stones with at low speed and moves forward d) after joining the crushed stones together the vehicle takes a reverse, e) the vehicle takes a forward turn and corrects its heading to push the stones again with at a low speed, f) finally, the vehicle approaches to the goal location and stops. We have split path-following tests into three categories: Balanced Tuning (BT), Less aggressive gain (LGA), and Varying Prediction Horizon(VPH). In each of our proposed gain tuning techniques, we have kept the control actions smooth, ensuring the used matrix R of the controller does not result in aggressive behavior. We call the setting a balanced tuning when the matrix R, Q, and prediction horizon N are neither large nor small values are in a normal range. A less aggressive gain (LGA) is where control input and states have less aggressive gain values. Finally, a varying prediction horizon (VPH) is where the prediction horizon can be large or small. These experiments enable us to get a feasible gain for our application (additional details in the supplementary document [44]). Figure 7 shows the path tracking of the given way-points in graphical representation. The idea is to follow the nominal PWL path and reduce the speed when pushing the materials. The reverse is performed at a higher speed.

In addition, the control actions can be observed from Figure 8a and Figure 8b, respectively. The linear velocity is $0.15m/s$ for the first 30 sec, but it reduces to $0.10m/s$ during the section of the path associated to a pushing action. The vehicle has linear velocity constraints of $-0.15m/s \leq v \leq 0.15m/s$. Therefore, for the reverse action of the vehicle, it will respect these constraints. The existence of abrupt changes in the linear velocity is because the critical path specifies it, not because the NMPC is deciding it, as those velocities are defined by the cruise speeds, which the high-level planner decides. The control action is smooth, which makes the bulldozing action more safe and reliable. For the angular velocity, the vehicle is always trying to catch up to 0 deg/s with the constraints of $-45\ deg/s \leq \omega \leq 45\ deg/s$. It is crucial to investigate the tracking errors associated with the individual states. For this purpose, we have investigated the state errors and Euclidean distance error, shown in Figure 9 between the actual and reference paths. Figure 10 states that the states' time convergence to be closed to zero starts at $20\,s$, which is the first way-point. The vehicle tends to catch up with the initial way-point. The moment the vehicle catches up with the first way-point, it stays on track without having any higher errors. Meanwhile, the Euclidean distance error is in the range of $[1\ cm - 1.5\ cm]$, which is acceptable as the vehicle has taken a reverse. It makes the proposed controller robust and efficient, and accurate.

The computational time of the controller is relevant for understanding the controller's feasibility for real applications.

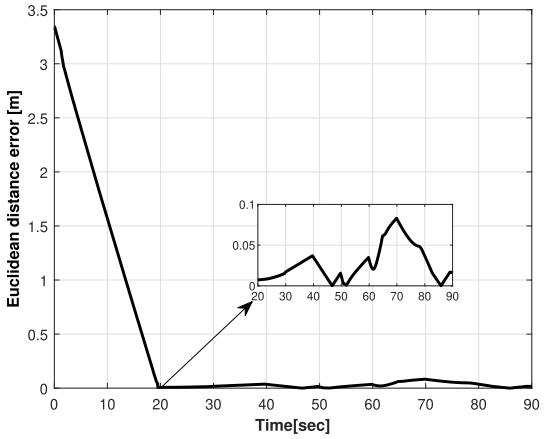

FIGURE 9. Euclidean distance error between actual and reference path.

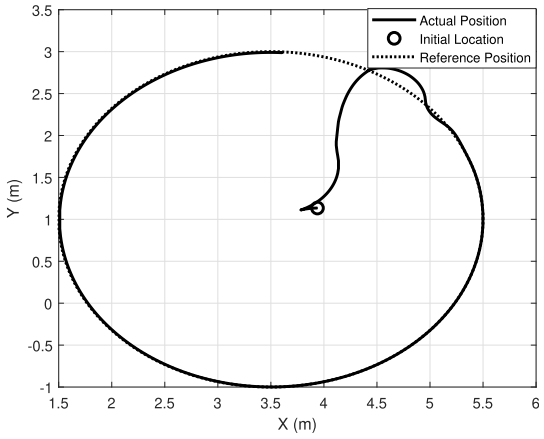

FIGURE 12. Proposed circle tracking by the real-platform.

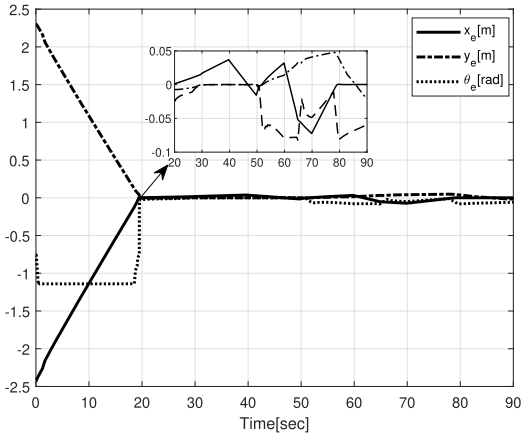

FIGURE 10. State errors of the real-platform.

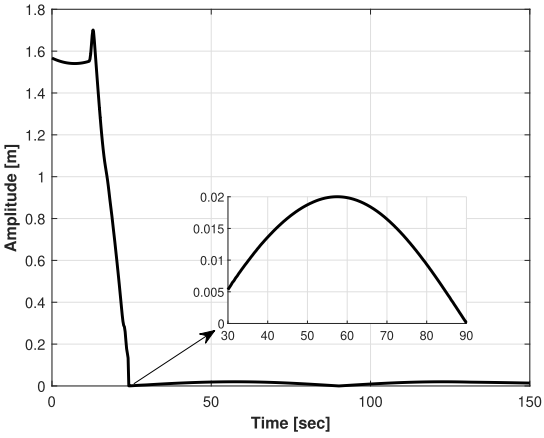

FIGURE 13. Euclidean distance errors between actual and reference path of the real-platform.

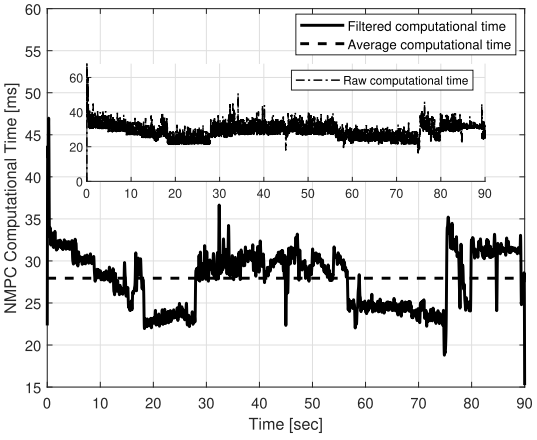

FIGURE 11. NMPC computational time.

Figure 11 shows the required processing time at each control instance. The average computation time for this experiment is around 27 *ms*, which is satisfactory for the practical application. In this experiment, as mentioned earlier, the onboard computer has 2-cores.

To further validate the robustness and effectiveness of the proposed control scheme, the following circular path to be tracked was considered;

$$\begin{cases} x_r(t) = 3.5 + 2 \cdot cos(0.2t) \\ y_r(t) = 1 + 2 \cdot sin(0.2t) \end{cases} \tag{18}$$

In addition, the platform's initial pose was far from the nominal path, $p_r = [0\ ,\ 0,\ 0]^T$; that fact can be appreciated from figure 12. The system is controlled at a rate of $f_s = 40\,Hz$ with a sampling time of $T_s = 25\,ms$. The response is bounded by constraints that allow the UGV to maintain local stability. However, the vehicle acts as a leader-follower formulation to attain stability for the reference trajectory.

In Figure 13, the robustness and effectiveness of the NMPC can be appreciated as the euclidean distance between the reference and desired path is always less than 1.5 *cm*. The discrepancy between nominal and real angular rate and longitudinal velocities can be appreciated in Figure 14. Those real inputs are not real but measured or estimated ones. A gyroscope provides the angular rate. This measurement is polluted by noise, whose statistical description can be assumed to be a GWN component and a slowly time-varying bias. The real longitudinal velocity is estimated by an EKF localizer (which estimates 2D position, heading,

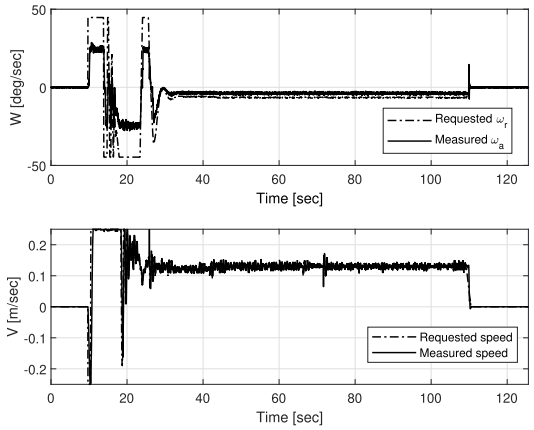

FIGURE 14. (a) Angular velocity required by the NMPC controller, and angular velocity actually measured, during the test in which the platform is required to follow a circular trajectory. The additional saturation of the actuators was not included in the nominal process model used by the NMPC. It can be appreciated that the low-level controllers do not achieve specific requested high angular rates. (b) Discrepancy between the speed required by the controller and the estimated actual speed for a circular trajectory. The discrepancies between the nominal velocities (required by the NMPC controller) and those achieved represent a relevant perturbation to the system.

and longitudinal velocity, and, in addition, the gyroscope's bias). We do not consider measurements from the wheel's encoders, as those are affected, at least sporadically, by slippage; consequently, the EKF estimates are more reliable.

A different situation would have resulted from the same experiment performed in simulation instead of using a real platform. In the simulated case, we would have compared the simulated path with the proposed one because the outputs of the simulator could have been considered the ground truth without uncertainty. However, the achieved path is not perfectly known for the real experiment because the platform's pose is known through estimates. Consequently, the discrepancy between the desired and actual paths is due to imperfect control and uncertainty in the pose estimates. That uncertainty affects the validation of the performance due to perturbation in the control process. To mitigate this issue, in the validation stages, a better localizer can be used, for instance, a localized based on 3D scanning; however, at the moment, that type of localizer is not usual in industrial applications. We also know that the noisy estimates also affect the NMPC control because we use those as feedback, which is another perturbation to our control (in addition to those affecting the nominal process model). However, we could still appreciate the excellent performance of the NMPC implementation. Table 1 compares the accuracy of a circular path with some of the existing techniques in the literature. The accuracy of tracking errors in our experiment is acceptable among these promising approaches. In addition, in some techniques, they have not performed experiments using a real platform, being those validations focused on simulations. In our experimental case, the proposed control technique is feasible to use for industrial-grade applications.

In terms of chattering in Figures 8b and 14, one reason for the appreciated chattering is that the experiment is not a simulation and that we are showing real data in the presence of noise. The shown curves are accurate but still showing estimated values provided by the EKF. The EKF is an optimal estimator; however, the noise in measurements still produces chattering that seems unnatural. However, it is according to the estimated covariance. A similar situation would occur if a high accurate RTK GPS were used outdoor. A different situation would have resulted from the same experiment performed in simulation instead of using a real platform. In a simulated case, we would have compared the simulated path with the proposed one because the outputs of the simulator could have been considered the ground truth without uncertainty. However, the achieved path is not perfectly known for the real experiment because the platform's pose is known through estimates.

TABLE 1. Performance comparison with existing literature.

Control Scheme	Accuracy (m)	Validation Type
This paper	$0.005 \sim 0.015$	Experiment
MPC [30]	$-0.01 \sim 0.01$	Simulation
Robust Controller [45]	$-0.005 \sim 0.025$	Simulation
Robust Controller [19]	$-0.01 \sim 0.02$	Simulation
MPC [46]	$-0.01 \sim 0.02$	Experiment
MPC [47]	$-0.05 \sim 0.05$	Simulation

The proposed algorithm would be feasible for the full-size bulldozer as well. However, to compute the Nonlinear model predictive control, the onboard computer is also an essential factor. An excellent onboard processor could provide a better response. Additionally, localization is an essential factor for this experiment. In our case, we have provided an extended Kalman filter. Therefore, with all these proposed approaches, it would be feasible to work with full-size bulldozers.

VI. CONCLUSION

This paper has implemented an NMPC-based path follower for a small-scale autonomous bulldozer to operate as part of a construction application to displace materials and debris of masonry. The proposed controller is robust and efficient enough to solve the problem for proper industrial-grade applications. The overall system also involves a localization process based on sensor data fusion of LiDAR and gyroscopes to provide the pose estimates needed for the controller feedback and the analysis of the controller's performance.

The experiments on the platform show that the proposed approach is computationally feasible, accurate, and robust. The control signals applied to the platform are smooth for proper treatment of the equipment. In addition, the accuracy of the approach is compared with existing techniques. Thus, the proposed NMPC resulted in a feasible solution for industrial-grade applications.

ACKNOWLEDGMENT

The authors would like thank UNSW Sydney for providing the resources to perform these experiments in the Mechatronics Lab.

REFERENCES

[1] J.-S. Hu, J.-J. Wang, and D. M. Ho, "Design of sensing system and anticipative behavior for human following of mobile robots," *IEEE Trans. Ind. Electron.*, vol. 61, no. 4, pp. 1916–1927, Apr. 2014.

[2] T. P. Nascimento, C. E. T. Dórea, and L. M. G. Gonçalves, "Nonholonomic mobile robots' trajectory tracking model predictive control: A survey," *Robotica*, vol. 36, no. 5, pp. 676–696, May 2018.

[3] H. Yang, X. Fan, P. Shi, and C. Hua, "Nonlinear control for tracking and obstacle avoidance of a wheeled mobile robot with nonholonomic constraint," *IEEE Trans. Control Syst. Technol.*, vol. 24, no. 2, pp. 741–746, Mar. 2016.

[4] S.-H. Kim, Y.-S. Lee, D.-I. Sun, S.-K. Lee, B.-H. Yu, S.-H. Jang, W. Kim, and C.-S. Han, "Development of bulldozer sensor system for estimating the position of blade cutting edge," *Autom. Construct.*, vol. 106, Oct. 2019, Art. no. 102890.

[5] Q. P. Ha, L. Yen, and C. Balaguer, "Robotic autonomous systems for earthmoving in military applications," *Autom. Construct.*, vol. 107, Nov. 2019, Art. no. 102934.

[6] J. Ni, J. Hu, and C. Xiang, "Robust path following control at driving/handling limits of an autonomous electric racecar," *IEEE Trans. Veh. Technol.*, vol. 68, no. 6, pp. 5518–5526, Jun. 2019.

[7] M. B. Quemelli and A. S. Brandão, "Handling and pushing objects using unmanned guided vehicles," *Robot. Comput.-Integr. Manuf.*, vol. 63, Jun. 2020, Art. no. 101913.

[8] S. Yamamoto, S. Matsushita, S. H. Zhang, S. Nishita, and K. Nakata, "Automatic drive control system for a bulldozer," U.S. Patent 5 462 122, Oct. 31, 1995.

[9] S. G. Olsen and G. M. Bone, "Modelling of robotic bulldozing operations for autonomous control," in *Proc. 24th Can. Conf. Electr. Comput. Eng. (CCECE)*, May 2011, pp. 1188–1193.

[10] Y.-S. Lee, S.-H. Kim, J. Seo, J. Han, and C.-S. Han, "Blade control in Cartesian space for leveling work by bulldozer," *Autom. Construct.*, vol. 118, Oct. 2020, Art. no. 103264.

[11] F. M. Barbosa, L. B. Marcos, M. M. da Silva, M. H. Terra, and V. Grassi, "Robust path-following control for articulated heavy-duty vehicles," *Control Eng. Pract.*, vol. 85, pp. 246–256, Apr. 2019.

[12] H. Guo, D. Cao, H. Chen, Z. Sun, and Y. Hu, "Model predictive path following control for autonomous cars considering a measurable disturbance: Implementation, testing, and verification," *Mech. Syst. Signal Process.*, vol. 118, pp. 41–60, Mar. 2019.

[13] E. Alcala, V. Puig, J. Quevedo, T. Escobet, and R. Comasolivas, "Autonomous vehicle control using a kinematic Lyapunov-based technique with LQR-LMI tuning," *Control Eng. Pract.*, vol. 73, pp. 1–12, Apr. 2018.

[14] X. Wu, P. Jin, T. Zou, Z. Qi, H. Xiao, and P. Lou, "Backstepping trajectory tracking based on fuzzy sliding mode control for differential mobile robots," *J. Intell. Robot. Syst.*, vol. 96, no. 1, pp. 109–121, Oct. 2019.

[15] C. Hu, R. Wang, and F. Yan, "Integral sliding mode-based composite nonlinear feedback control for path following of four-wheel independently actuated autonomous vehicles," *IEEE Trans. Transport. Electrific.*, vol. 2, no. 2, pp. 221–230, Jun. 2016.

[16] H. Yuan and H. Zhu, "Anti-jackknife reverse tracking control of articulated vehicles in the presence of actuator saturation," *Vehicle Syst. Dyn.*, vol. 54, no. 10, pp. 1428–1447, Oct. 2016.

[17] F. Dou, Y. Huang, L. Liu, H. Wang, Y. Meng, and L. Zhao, "Path planning and tracking for autonomous mining articulated vehicles," *Int. J. Heavy Vehicle Syst.*, vol. 26, nos. 3–4, pp. 315–333, 2019.

[18] L. Zhang, Z. Wang, X. Ding, S. Li, and Z. Wang, "Fault-tolerant control for intelligent electrified vehicles against front wheel steering angle sensor faults during trajectory tracking," *IEEE Access*, vol. 9, pp. 65174–65186, 2021.

[19] H.-S. Kang, Y.-T. Kim, C.-H. Hyun, and M. Park, "Generalized extended state observer approach to robust tracking control for wheeled mobile robot with skidding and slipping," *Int. J. Adv. Robot. Syst.*, vol. 10, no. 3, p. 155, Mar. 2013.

[20] J. Wu, Z. Wang, and L. Zhang, "Unbiased-estimation-based and computation-efficient adaptive MPC for four-wheel-independently-actuated electric vehicles," *Mechanism Mach. Theory*, vol. 154, Dec. 2020, Art. no. 104100.

[21] S. Lucia, J. A. E. Andersson, H. Brandt, M. Diehl, and S. Engell, "Handling uncertainty in economic nonlinear model predictive control: A comparative case study," *J. Process Control*, vol. 24, no. 8, pp. 1247–1259, Aug. 2014.

[22] H. Li and S. Wang, "Model-based multi-objective predictive scheduling and real-time optimal control of energy systems in zero/low energy buildings using a game theory approach," *Autom. Construct.*, vol. 113, May 2020, Art. no. 103139.

[23] M. Killian and M. Kozek, "Ten questions concerning model predictive control for energy efficient buildings," *Building Environ.*, vol. 105, pp. 403–412, Aug. 2016.

[24] J. Širokỳ, F. Oldewurtel, J. Cigler, and S. Prívara, "Experimental analysis of model predictive control for an energy efficient building heating system," *Appl. Energy*, vol. 88, no. 9, pp. 3079–3087, 2011.

[25] J. Choi, B. Gu, S. Chin, and J.-S. Lee, "Machine learning predictive model based on national data for fatal accidents of construction workers," *Autom. Construct.*, vol. 110, Feb. 2020, Art. no. 102974.

[26] X. Li and G. Gong, "Predictive control of slurry pressure balance in shield tunneling using diagonal recurrent neural network and evolved particle swarm optimization," *Autom. Construct.*, vol. 107, Nov. 2019, Art. no. 102928.

[27] H. Zhou, P.-Y. Zhao, Y.-L. Chen, and H.-Y. Yang, "Prediction-based stochastic dynamic programming control for excavator," *Autom. Construct.*, vol. 83, pp. 68–77, Nov. 2017.

[28] Z. Q. Sun, L. Dai, K. Liu, Y. Q. Xia, and K. H. Johansson, "Robust MPC for tracking constrained unicycle robots with additive disturbances," *Automatica*, vol. 90, pp. 172–184, Apr. 2018.

[29] E. Kayacan, H. Ramon, and W. Saeys, "Robust trajectory tracking error model-based predictive control for unmanned ground vehicles," *IEEE/ASME Trans. Mechatronics*, vol. 21, no. 2, pp. 806–814, Apr. 2016.

[30] H. Yang, M. Guo, Y. Xia, and L. Cheng, "Trajectory tracking for wheeled mobile robots via model predictive control with softening constraints," *IET Control Theory Appl.*, vol. 12, no. 2, pp. 206–214, Jan. 2018.

[31] Q. Cui, R. Ding, B. Zhou, and X. Wu, "Path-tracking of an autonomous vehicle via model predictive control and nonlinear filtering," *Proc. Inst. Mech. Eng., D, J. Automobile Eng.*, vol. 232, no. 9, pp. 1237–1252, 2018.

[32] G. Klančar and I. Škrjanc, "Tracking-error model-based predictive control for mobile robots in real time," *Robot. Auton. Syst.*, vol. 55, no. 6, pp. 460–469, 2007.

[33] T. Ohtsuka, "A continuation/GMRES method for fast computation of nonlinear receding horizon control," *Automatica*, vol. 40, no. 4, pp. 563–574, 2004.

[34] E. Kayacan, W. Saeys, H. Ramon, C. Belta, and J. M. Peschel, "Experimental validation of linear and nonlinear MPC on an articulated unmanned ground vehicle," *IEEE/ASME Trans. Mechatronics*, vol. 23, no. 5, pp. 2023–2030, Oct. 2018.

[35] M. Hirayama, J. Guivant, J. Katupitiya, and M. Whitty, "Artificial intelligence in path planning for autonomous bulldozers: Comparison with manual operation," *Int. J. Innov. Comput., Inf. Control*, vol. 15, no. 3, pp. 825–844, 2019.

[36] M. Hirayama, J. Guivant, J. Katupitiya, and M. Whitty, "System and method for planning travel path for work machine, and work machine," AU Patent 2018 334 390, Jan. 30, 2020.

[37] M. Hirayama, M. Whitty, J. Katupitiya, and J. Guivant, "Control system for work vehicle, control method, and work vehicle," U.S. Patent 16 335 392, Jan. 16, 2020.

[38] M. Hirayama, M. Whitty, J. Katupitiya, and J. Guivant, "An optimized approach for automatic material distribution operations of bulldozers," *Int. J. Adv. Robot. Syst.*, vol. 15, no. 2, pp. 1–10, 2018.

[39] M. Hirayama, J. Guivant, J. Katupitiya, and M. Whitty, "Path planning for autonomous bulldozers," *Mechatronics*, vol. 58, pp. 20–38, Apr. 2019.

[40] J. Guivant, S. Cossell, M. Whitty, and J. Katupitiya, "Internet-based operation of autonomous robots: The role of data replication, compression, bandwidth allocation and visualization," *J. Field Robot.*, vol. 29, no. 5, pp. 793–818, Sep. 2012.

[41] E. Nebot, J. Guivant, and S. Worrall, "Haul truck alignment monitoring and operator warning system," *J. Field Robot.*, vol. 23, no. 2, pp. 141–161, 2006.

[42] C. J. Ostafew, A. P. Schoellig, and T. D. Barfoot, "Robust constrained learning-based NMPC enabling reliable mobile robot path tracking," *Int. J. Robot. Res.*, vol. 35, no. 13, pp. 1547–1563, 2016.

[43] S. Boyd, S. P. Boyd, and L. Vandenberghe, *Convex Optimization*. Cambridge, U.K.: Cambridge Univ. Press, 2004.

[44] S. Khan and J. Guivant. (May 5, 2021). *Supplementary Document*. [Online]. Available: http://www.possumrobot.com/NLMPC_bd01/NLMPC_SK_JG_2020_01.htm

[45] H. Yang, X. Fan, Y. Xia, and C. Hua, "Robust tracking control for wheeled mobile robot based on extended state observer," *Adv. Robot.*, vol. 30, no. 1, pp. 68–78, Jan. 2016.
[46] Z. Li, J. Deng, R. Lu, Y. Xu, J. Bai, and C.-Y. Su, "Trajectory-tracking control of mobile robot systems incorporating neural-dynamic optimized model predictive approach," *IEEE Trans. Syst., Man, Cybern., Syst.*, vol. 46, no. 6, pp. 740–749, Jun. 2016.
[47] C. Lian, X. Xu, H. Chen, and H. He, "Near-optimal tracking control of mobile robots via receding-horizon dual heuristic programming," *IEEE Trans. Cybern.*, vol. 46, no. 11, pp. 2484–2496, Nov. 2016.

SUBHAN KHAN received the B.Sc. degree in computer engineering from the COMSATS Institute of Information Technology (CIIT), Lahore, Pakistan, in 2013, and the M.Sc. degree in electronic communications and computer engineering from the University of Nottingham, U.K., in 2015. He is currently pursuing the Ph.D. degree in mechatronic engineering from the University of New South Wales (UNSW), Sydney, Australia. From 2013 to 2018, he worked as a Lecturer with the Department of Electrical Engineering, CIIT Lahore. His research interests include model predictive control, unmanned ground vehicle's, and satellite navigation technology.

JOSE GUIVANT (Member, IEEE) received the B.E. degree in electrical engineering from the Universidad Nacional del Sur (UNS), Bahia Blanca, Argentina, and the Ph.D. degree in robotics from The University of Sydney (ACFR/USYD), Australia. He is currently a Senior Lecturer in mechatronics with the School of Mechanical Engineering, UNSW, Australia.