scientific reports

OPEN

Fast nonlinear model predictive planner and control for an unmanned ground vehicle in the presence of disturbances and dynamic obstacles

Subhan Khan✉ & Jose Guivant

This paper presents a solution for the tracking control problem, for an unmanned ground vehicle (UGV), under the presence of skid-slip and external disturbances in an environment with static and moving obstacles. To achieve the proposed task, we have used a path-planner which is based on fast nonlinear model predictive control (NMPC); the planner generates feasible trajectories for the kinematic and dynamic controllers to drive the vehicle safely to the goal location. Additionally, the NMPC deals with dynamic and static obstacles in the environment. A kinematic controller (KC) is designed using evolutionary programming (EP), which tunes the gains of the KC. The velocity commands, generated by KC, are then fed to a dynamic controller, which jointly operates with a nonlinear disturbance observer (NDO) to prevent the effects of perturbations. Furthermore, pseudo priority queues (PPQ) based Dijkstra algorithm is combined with NMPC to propose optimal path to perform map-based practical simulation. Finally, simulation based experiments are performed to verify the technique. Results suggest that the proposed method can accurately work, in real-time under limited processing resources.

List of symbols

a_r	Reference Acceleration
q	Actual trajectory of the UGV
q_r	Reference trajectory of the UGV
q_g	Initial states of the UGV
$S(q)$	Kinematics function of UGV
Ω	actual velocity vector
Ω_r	Proposed velocity vector
η	Perturbed angular velocities due to slipping of the two actuated wheels
ρ	Mismatched disturbances vector
μ	Lateral skidding velocity of the UGV
M	Symmetric and positive-definite inertia matrix
H	Centripetal and coriolis matrix
B	Input transformation matrix
η_1	Longitudinal slip velocity
η_2	Yaw rate perturbation due to slippage of the wheels
v	Longitudinal velocity of the UGV
ω	Yaw rate of the UGV
τ	Applied torque vector
τ_d	Torque disturbance vector
r	Radius of wheels
m	Mass of the UGV
d	Distance between the centre of mass of the UGV and wheel axis

School of Mechanical and Manufacturing Engineering, University of New South Wales (UNSW), Sydney 2052, Australia. ✉email: Subhan.khan@unsw.edu.au

θ	Heading of the UGV
x, y	Position of the UGV
J	Moment of inertia
N	Mechanical gear ratio
N_p	Prediction horizon
V_u	Applied DC voltage to the motors
L_a	Rotor inductance
R_a	Rotor resistance
K_t	Torque constant
i_a	Armature current
T_s	Sampling time
F_s	Sampling frequency
k	Discrete indexing
C_T	Terminal cost
C_S	Stage cost
Q	Semi-positive state-penalty gain matrix
R	Diagonal positive definite input-penalty gain matrix
$\bar{Q}$	Semi-positive terminal cost state-penalty gain matrix
q_e	State tracking error matrix
u_c	compensated velocity vector
u_p	Tuned velocity vector
F	Dynamic matrix
T	Error matrix
ζ	Velocity tracking error
$\Delta\tau$	Rate of change in torque vector
L	Observer gain matrix
$\hat{D}$	Auxiliary variable
K	Constant matrix to design observer
α	Diagonal positive definite matrix
β	Controller gain matrix
K_c	Input saturation compensation gain
r_c	Radius of the bounding circles of UGV
r_o	Radii of the obstacles
k_x	Kinematic compensation gain for x
k_y	Kinematic compensation gain for y
k_θ	Kinematic compensation gain for θ
$\tilde{k}_x$	sub-optimal scaling factor for x
$\tilde{k}_y$	sub-optimal scaling factor for y
$\tilde{k}_\theta$	sub-optimal scaling factor for θ
K_f	Electromotive force coefficient
K_c	Input saturation's compensation gain
W	Evolutionary weights
P	Population vector
$S(q)$	Kinematics function of UGV
C	Centripetal and coriolis matrix
C_{opt}	Optimal individuals of population P of evolutionary programming

In recent years, unmanned ground vehicles (UGVs) have been applied extensively in agriculture, construction, mining and military operations, etc. The UGVs have lightweight, fast speed, low energy consumption, high maneuverability, convenient drive, and simple mechanism[1–3]. Consequently, these advantages make UGVs suitable for indoor[4,5] and outdoor[6,7] applications. However, the ground vehicles can be perturbed due to uncertainties in models, and description of the context of operation[8], skidding and slipping[9], noisy sensor's measurements,[10], and failures in electromechanical components[11], making the navigation and control of the platform a challenging task[12,13].

Motion planning and control are the two crucial factors for a vehicle to complete an assigned task safely and effectively[14]. First, it should navigate in an environment with obstacles. Second, it must satisfy the requirement of reaching a specified goal, constantly dealing with perturbations[15]. Thus, the first requirement implied using a path-planner to plan the vehicle's motion, while the second requirement can be treated by a controller to control the kinematics and dynamics of the vehicle.

In the past few decades, research on motion planning and control of UGVs has witnessed remarkable development. In particular, trajectory tracking and target posture stabilization problems on kinematic models were investigated[16–19]. Additionally, many researchers developed motion control at dynamic levels (such as torque controllers[20,21] or voltage controllers[22,23]). Moreover, there are some methods developed on model predictive control (MPC), which considers the constraints (such as[24–26]). Similarly, many path-planners and path-followers were developed to reach the specified final location safely[27–30]. From the perspective of UGVs, some studies consider the presence of disturbances, and uncertainties[31–33]; some others assume pure rolling without skid-slip disturbances[34,35]. In addition, some autonomous system use safe switched tracking[36] and neural network based techniques[37]. Thus, for practical applications, it is important to consider those disturbances.

Overall, most of the existing techniques mentioned above assume that the UGVs satisfy pure rolling conditions (without skid-slip). However, pure rolling may not always hold due to the terrain, tire dynamics, voltage drive circuitry, etc. Thus, making the control problem challenging for the UGVs in the presence of external disturbances and skid-slip. Nevertheless, on the other hand, it allows solving these challenges with advanced planning and control strategies. In[38], kinematic models that are explicitly linked to skid-slip are presented, which considers the design from a controller's perspective. An analysis of the hybrid MPC for stabilization of robot under the presence of wheel slippage is discussed in[39]. In[40], a nonlinear disturbance observer is used for a self-balancing mobile wheeled mobile robot. Additionally, this work focuses on a robust tracking control problem with unknown disturbances. In[41], an NDO and extended Kalman filter (EKF) are combined to perform the trajectory tracking control for wheeled mobile robots. First, a kinematic model excluding perturbation and distorted dynamic model is discussed. Then, NDO is used to observe the external disturbances, and EKF is used to estimate the platform's states. In addition, robust constrained control is developed by observing the disturbances using disturbance observer in[42].

In terms of path-tracking control problem, an obstacle avoidance mobile robot is introduced in[26]. Firstly, a CasAdi-based[43] single shooting NMPC path-planner is designed. Secondly, a velocity-based virtual control and saturated torque control are designed to apply control actions to the mobile robotic platform. Finally, an extended state observer (ESO) is designed to estimate the disturbances. This technique only considers a single shooting (SS) optimization problem, which requires a larger prediction horizon to compute the NMPC-based path planner. A higher prediction horizon can be understandable in the presence of a complex control problem. However, in the simulations, only a simple point with one dynamic obstacle is avoided. The processing time of the onboard computer is not considered in the simulations, which can cause an issue in the real experiment with the emphasis on the processing time per iteration of the computer. One way to fix the problem is to introduce the multiple shooting(MS) optimization problems, which takes a small prediction horizon. Alternatively, the proximal averaged Newton-type method of optimal control (PANOC) can be used[24]. In terms of voltage control strategies, an adaptive perturbation rejection technique is proposed in[44]. Although they have not considered skid-slip explicitly, the proposed voltage control strategy can solve the trajectory tracking control problem.

This paper focuses on the tracking control problem of a UGV to safely reach a goal location in the presence of moving and static obstacles, skid-slip, and external disturbances. An NMPC-based path-planner, voltage driving control, kinematic and dynamic control, and NDO are used to stabilize the target posture, while PPQ-Dijkstra is used to generate optimal path for the map-based simulation, which is also used in[45]. The main contributions of this research are: (1) In contrast to[26], the proposed NMPC-based path planner is designed to propose a suitable trajectory to achieve the goal location and avoid the vehicle from colliding with multiple dynamic and static obstacles. Additionally, the MS optimization problem and voltage control strategy are proposed in this paper, which improves the processing time of a single or multicore CPU. (2) A novel EP tuning method is used to tune the gains of the KC to generate velocity commands to the dynamic controller. A dynamic controller and NDO are then proposed to provide voltage-based control actions to the UGV and compensate for the lumped disturbances, respectively. (3) An investigation is performed to measure the single-core and multi-core CPUs processing time per iteration using the proposed MS-based NMPC with the SS method in[26]. In this paper, the proposed control scheme is designed to cope with the dynamic environment. To verify the working of the proposed scheme, extensive simulations have been carried out to show the vehicle's behavior in the presence of disturbances in a realistic framework.

Problem formulation

In this paper we use dynamic and kinematic models which consider disturbances, including skidding and slippage; those models are the following ones[46,47]

$$\dot{q} = S(q) \cdot (\Omega - \eta) + \rho(q, \mu) \tag{1}$$

$$M \cdot (\dot{\Omega} - \dot{\eta}) + H_1(q, \dot{q}) \cdot (\Omega - \eta) + H_2(q) \cdot \dot{\rho}(q, \mu) + H_3(q) \cdot \rho(q, \mu) = B \cdot ((\tau - \tau_d) \tag{2}$$

with,

$$q = [x, y, \theta]^T \in \mathbb{R}^3, \Omega = [v, \omega]^T, \eta = [\eta_1, \eta_2]^T \in \mathbb{R}^2, \tau_d = [\tau_{d_1}, \tau_{d_2}]^T \in \mathbb{R}^2, \eta_1 = \frac{r}{2}(\eta_r + \eta_l), \eta_2 = \frac{r}{2b}(\eta_r - \eta_l),$$

$$m = m_c + m_w, H_1(X, \dot{q}) = \begin{bmatrix} 0 & -m_c d\dot{\theta} \\ m_c d\dot{\theta} & 0 \end{bmatrix}, H_2(q) = \begin{bmatrix} m + \frac{2J}{r^2}\cos\theta & (m + \frac{2J}{r^2})\sin\theta & 0 \\ -m_c d\sin\theta & m_c d\cos\theta & J + \frac{2b^2 J}{r^2} \end{bmatrix},$$

$$H_3(q) = \begin{bmatrix} \frac{-2J\dot{\theta}\sin\theta}{r^2} & \frac{2J\dot{\theta}\cos\theta}{r^2} & -m_c d\dot{\theta} \\ 0 & 0 & 0 \end{bmatrix}, S(q) = \begin{bmatrix} \cos\theta & \sin\theta & 0 \\ 0 & 0 & 1 \end{bmatrix}^T, \rho(q, \mu) = \begin{bmatrix} -\mu\sin\theta \\ \mu\cos\theta \\ 0 \end{bmatrix}^T,$$

$$M = \begin{bmatrix} m & 0 \\ 0 & J - md^2 \end{bmatrix}, \quad \tau = [\tau_1, \tau_2]^T, \in \mathbb{R}^2, \quad H = [H_1(q, \dot{q}), H_2(q), H_3(q)]^T$$

Consider the UGV represented in Fig. 1 is driven by two DC motors attached on each side with same mechanical properties with $N > 1$. An expression for the angular velocities of motor could have the following expression:

$$\omega_l = \frac{v + b \cdot \omega}{r}, \qquad \omega_r = \frac{v - b \cdot \omega}{r}$$
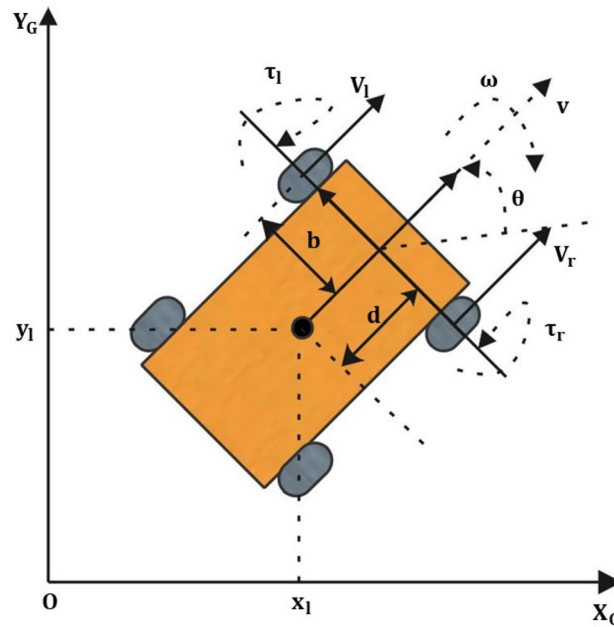
Figure 1. UGV configuration.
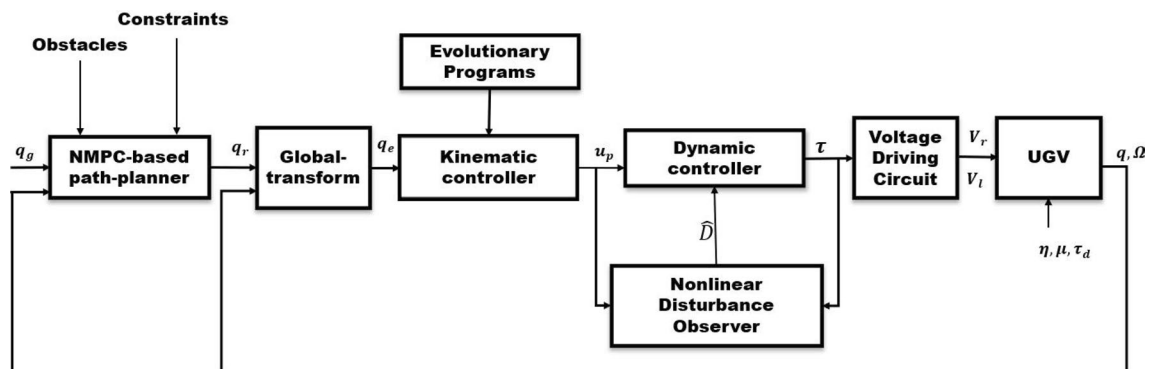
Figure 2. Proposed planner and control structure diagram for the UGV.

For simplicity, we have assumed that the torque τ generated by the motors are explicitly depended on the armature current

$$\tau = N \cdot K_t \cdot i_a, \tag{3}$$

where K_t is the torque constant. Now, we can obtain the armature voltage as follows

$$V_u = L_a \frac{di_a}{dt} + R_a i_a + N \cdot K_f \cdot \omega_a \tag{4}$$

where $V_u = [V_r, \ V_l]^T$. Now, we can convert our torque dynamics into the voltage driving circuit by using the above expression as follows:

$$\begin{bmatrix} V_l \\ V_r \end{bmatrix} = \frac{L_a}{NK_t} \begin{bmatrix} \dot{\tau}_l \\ \dot{\tau}_r \end{bmatrix} + \frac{R}{NK_t} \begin{bmatrix} \tau_l \\ \tau_r \end{bmatrix} - \frac{K_f}{r} \begin{bmatrix} -N & -Nb \\ -N & Nb \end{bmatrix} \begin{bmatrix} v \\ \omega \end{bmatrix} \tag{5}$$

Methods

In this Section, first, an NMPC-based path planner is to generate reference trajectory to reach the current destination pose, additionally satisfying the collision avoidance constrain. Second, an EP-based gain tuning method is designed to generate KC-based velocity commands. Third, a dynamic control is designed with the NDO to mitigate the effects of disturbances and treat the skid-slip effect. Finally, the stability analysis of the proposed controller is discussed. Figure 2 illustrates the overall close-loop structure of the proposed technique.

NMPC-based path planner.   we will consider NMPC-based path planner to generate a reference trajectory for the UGV from a starting pose $q(0)$. In particular, the the presences of static and dynamic objects are considered in the environment of the UGV. For the reference posture of the UGV, we consider that the platform operates free of skid-slip, which corresponds to the assumption of pure rolling. Thus, by using the pure rolling, the kinematics and dynamics of the UGV can be expressed as[48]

$$\dot{q}_r = S(q_r) \cdot \Omega_r, \qquad \dot{\Omega}_r = a_r \tag{6}$$

where $a_r = [\dot{v}, \dot{\omega}]^T$ and $\Omega_r = [v_r, \omega_r]^T$. Until now, we have taken we have modelled the platform by a continuous time model. as a continuous-time system. However, it is essential for a computer to generate discrete signals. Therefore, by using Euler's approximation with T_s, we can approximate our nonlinear continuous-time system by a discrete-time one, as follows

$$q_r(k+1) = q_r(k) + T_s \cdot \dot{q}_r(k), \qquad \Omega_r(k+1) = \Omega_r(k) + T_s \cdot a_r(k) \tag{7}$$

The discrete indexing $k \in \Lambda$, while Λ is a set of positive natural numbers.

In terms of collision avoidance, the UGV must move with some restrictions. Therefore, we have set constraints on the platform's physical position as follows:

$$g_o^i(q_r(k)) = \sqrt{(x(k) - x_o^i(k))^2 + (y(k) - y_o^i(k))^2} - (r_c + r_o^i) \tag{8}$$

where obstacles are $i = 1, 2, \ldots, I$; I represent number of currently known obstacles. Additionally, constraints on the velocities and accelerations are considered, as follows:

$$v_{r_{min}} \leq v_r(k) \leq v_{r_{max}}, \qquad \omega_{r_{min}} \leq \omega_r(k) \leq \omega_{r_{max}}, \qquad \dot{v}_{r_{min}} \leq \dot{v}_r(k) \leq \dot{v}_{r_{max}}, \qquad \dot{\omega}_{r_{min}} \leq \dot{\omega}_r(k) \leq \dot{\omega}_{r_{max}}. \tag{9}$$

Now, we can compute the cost-function for our NMPC-based path planner as follows:

$$\underset{a_r}{\arg \min} \quad J(a_r(k+j)) = C_T(q_{N_p}, \Omega_{N_p}) + \sum_{j=0}^{N_p-1} C_S(q_r(k+j), \Omega_r(k+j), a_r(k+j)) \tag{10}$$

subject to following constraints

$$q_r(k+j) = f_1(q_r(k+j-1), \Omega_r(k+j-1)), \qquad \Omega_r(k+i) = f_2(\Omega_r(k+i-1), a_r(k+j-1)),$$
$$a_r(k+j-1) \in U, j \in \mathbb{N}_{[0, N_p-1]}, \qquad g_o^i(q_r(k+j)) \leq 0, j \in \mathbb{N}. \tag{11}$$

where

$C_T(q_{N_p}, \Omega_{N_p}) = (q_{N_p} - q_g)^T \cdot Q_{N_p} \cdot (q_{N_p} - q_g) + \Omega_{N_p}^T \cdot \bar{Q}_{N_p} \cdot \Omega_{N_p} C_S(q_r, \Omega_r, a_r) = (q_r - q_g)^T \cdot Q \cdot (q_r - q_g)$
$+ \Omega_r^T \cdot \bar{Q} \cdot \Omega_r + a_r \cdot R \cdot a_r^T, j = 1, \ldots N$, The nonlinear problem proposed in (10) and (11) can be transformed into optimal control problem by using the multiple shooting numerical method proposed by[43]. Alternatively, the single shooting method, as proposed in[26], can be used. However, it would be computationally more complex.

Kinematic controller.   The evolutionary strategies narrow the degree of freedom in the instantiating of meta-heuristic algorithms. Thus, the main idea here is first to take the error between the reference and actual positions and orientation.

$$\tilde{q}_e = q_r - q = \begin{bmatrix} x_e \\ y_e \\ \theta_e \end{bmatrix} = \begin{bmatrix} x_r - x \\ y_r - y \\ \theta_r - \theta \end{bmatrix} \tag{12}$$

Now, by taking the global transformation mentioned in[49], we take the following posture error of the real UGV from (12):

$$q_e = \begin{bmatrix} e_1 \\ e_2 \\ e_3 \end{bmatrix} = \begin{bmatrix} \cos(\theta) & \sin(\theta) & 0 \\ -\sin(\theta) & \cos(\theta) & 0 \\ 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} x_e \\ y_e \\ \theta_e \end{bmatrix} \tag{13}$$

The time derivative of the tracking error of (13) is as follows

$$\begin{bmatrix} \dot{e_1} \\ \dot{e_2} \\ \dot{e_3} \end{bmatrix} = v \begin{bmatrix} -1 \\ 0 \\ 0 \end{bmatrix} + \omega \begin{bmatrix} e_2 \\ -e_1 \\ -1 \end{bmatrix} + \begin{bmatrix} v_r \cos(e_3) \\ v_r \sin(e_3) \\ \omega_r \end{bmatrix} \tag{14}$$

Now, a back-stepping method (as that presented in[49]) is used to get a control signals excluding skid-slip.

$$u_c = \begin{bmatrix} v_c \\ \omega_c \end{bmatrix} = \begin{bmatrix} v_r \cos(e_3) + k_x e_1 \\ \omega_r + k_y v_r e_2 + k_\theta v_r \sin(e_3) \end{bmatrix} \tag{15}$$

Algorithm 1 Evolutionary control algorithm

1: **procedure** EPKC(P_0, C_{opt}, N)
2:      $t \leftarrow 0$
3:      $i \leftarrow 0$
4:      **repeat**
5:          **while** $i \leq N$ **do**
6:              $q_r \leftarrow \sum_{i=1}^N (W_1 e_{1_i})^2 + (W_2 e_{2_i})^2 + (W_3 e_{3_i})^2$
7:              $X \leftarrow variation(q_1, \ldots, q_k)$
8:              $Q^t \leftarrow Q^t \cup q$
9:          **end while**
10:         $P^{t+1} = $ selection of best C_l individuals from $Q^t \cup (q \in P^t)$
11:         $t \leftarrow t + 1$
12:     **until** stopping criterion fulfilled
13: **end procedure**

In EP, we only require to know the fitness function of an individual population. Consider P as a population of all parameters involved in Eq. (1) and $k^* = [\hat{k}_x, \hat{k}_y, \hat{k}_\theta] \in \mathbb{R}^3$. Now, the main idea here is to find sub-optimal scaling factors $\hat{k}_x$, $\hat{k}_y$ and $\hat{k}_\theta$. These sub-optimal scaling factors can be achieved by applying the EP searching algorithm shown in Algorithm 1.

$$u_P = \begin{bmatrix} v_P \\ \omega_P \end{bmatrix} = \begin{bmatrix} v_r \cos(e_3) + \hat{k}_x e_1 \\ \omega_r + \hat{k}_y v_r e_2 + \hat{k}_\theta v_r \sin(e_3) \end{bmatrix} \tag{16}$$

The error function $u_e = [v_e, \omega_e]^T \in \mathbb{R}^2$ of velocities can also be computed as follows

$$v_e = -x_e \sin\theta + \omega y_e \cos\theta + v - v_r \cos\theta_e, \qquad \omega_e = \omega - \omega_r \tag{17}$$

Dynamic controller with NDO. The dynamics in Eq. (2) can be transferred into the following

$$\bar{M} \cdot \dot{u}_P + F = \bar{B} \cdot \tau \tag{18}$$

with

$$\bar{M} = T^T M T, \bar{B} = T^T B, \quad T = \begin{bmatrix} (x_e \sin\theta - y_e \cos\theta) & 1 \\ 1 & 0 \end{bmatrix}, \Pi = [v_r \cos\theta_e + \omega_r(x_e \sin\theta - y_e \cos\theta), \omega_r]^T.$$

$$F = T^T[M\dot{T}(u_e + \Pi) + MT\dot{\Pi} - M\dot{\eta} + C_1(\Omega - \eta)] + T^T[C_2\dot{\rho} + C_3\rho] + \bar{B} \cdot \tau_d,$$

The parameter M is unknown. However, the nominal M_0 is known, and taking the velocity tracking error $\zeta = u_e - u_P$, we have

$$\dot{\zeta} = \bar{M}_0^{-1}\bar{B} \cdot \tau + H \cdot \bar{B} \cdot \tau - \bar{M}^{-1} \cdot F - \dot{u}_P \tag{19}$$

where $H = \bar{M}^{-1} - \bar{M}_0^{-1}$. In addition, we can take $\Delta\tau = (\tau(k+1) - \tau(k))$ to handle the saturation constraints. This will also gives an intuition that $\dot{u}_P$ is related to unknown skid-slip. Therefore the above expression becomes as follows

$$\dot{\zeta} = \bar{M}_0^{-1}\bar{B}\tau + \Delta\tau + H \cdot \bar{B} \cdot \tau - \bar{M}^{-1}F - \dot{u}_P - (\bar{M}_0^{-1}\bar{B} - I)\Delta\tau \tag{20}$$

To accomplish the NDO, we first need to lumped the estimated disturbances as follows

$$\dot{\hat{\zeta}} = L(q) \cdot (\zeta - \hat{\zeta}) \tag{21}$$

where $L(q)$ is the observer gain. Now, we can define the auxiliary variable $\hat{D}$ to obtain an improved disturbance observer design as follows:

$$\dot{\hat{D}} = L(q) \cdot (\bar{M}_0^{-1}\bar{B} \cdot \tau + H \cdot \bar{B} \cdot \tau - \bar{M}^{-1} \cdot F - \dot{u}_P) - L(q) \cdot \hat{\zeta}, \qquad \hat{\zeta} = \hat{D} + p(u_P) \tag{22}$$

where the function $p(u_p)$ is selected from[50].

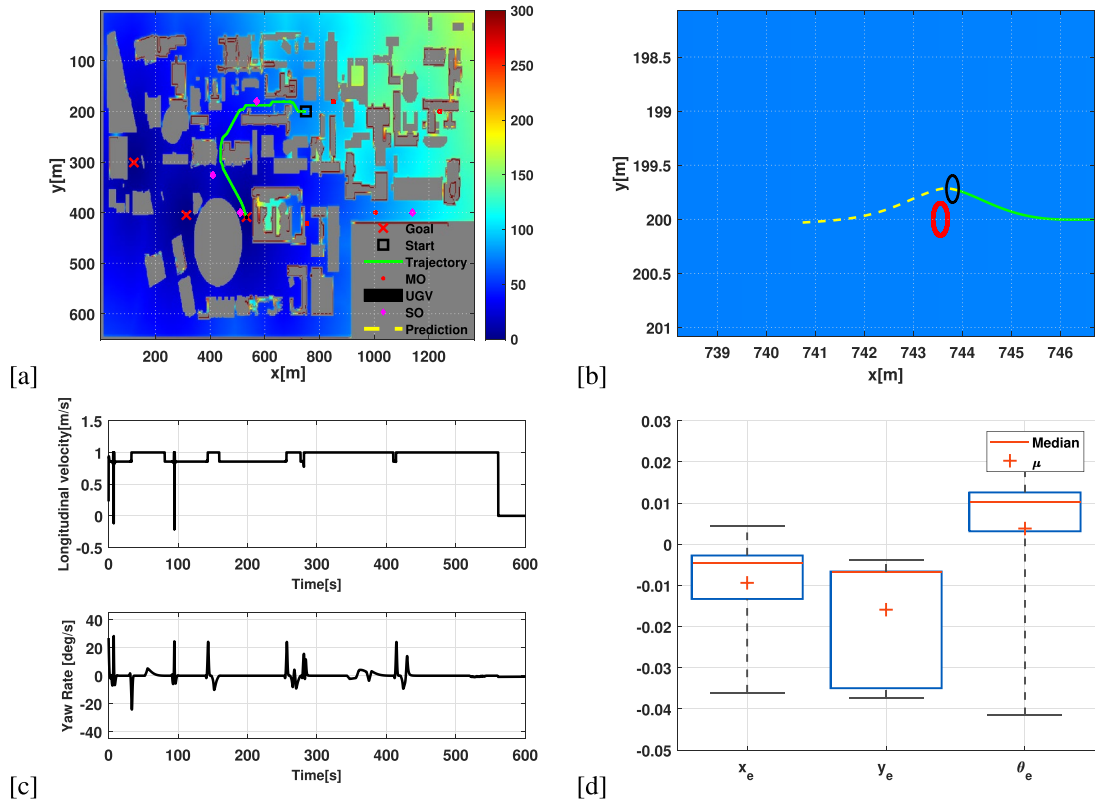$$p(u_P) = K \cdot u_P, \qquad \dot{p}(u_P) = K \tag{23}$$

where K is the constant matrix. Now, by taking the time derivative of (22)

$$\dot{\hat{\zeta}} = \dot{\hat{D}} + \dot{p}(u_P) \cdot \dot{u}_P \tag{24}$$

By combining the expressions Eqs. (21) and (22)

$$\dot{\hat{\zeta}} - \dot{\hat{D}} = L(q) \cdot \bar{M}\dot{u}_P, \qquad L(q) \cdot \bar{M} = \dot{p}(u_P) \tag{25}$$

Thus, we select the observer gain matrix as $L(q) = K \cdot \bar{M}^{-1}$. Now, using Eqs. (19) and (22) the expression can be re-written as follows:

Figure 3. (**a**) Path-tracking of the UGV with a birds eye-view of the UNSW map. (**b**) Collision avoidance of incoming MO as well as the prediction of future state. (**c**) Applied longitudinal and angular velocity of the UGV. (**d**) Tracking error representation of the vehicle's actual states.

$$\dot{\zeta} = \bar{M}_0^{-1} \cdot \bar{B} \cdot \tau + \Delta\tau + \hat{D} \tag{26}$$

For the system proposed in (26) the applied torque can be achieved as follows:

$$\tau = \bar{M}_0^{-1} \cdot \bar{B} \cdot \left[-\alpha \cdot \zeta + \beta \cdot K_c - \frac{\beta^2 \zeta}{2} - \hat{D} \right] \tag{27}$$

The stability analysis of the observer is performed as per[26]. Therefore, we will discuss the closed loop stability analysis of the controller scheme in the following subsection.

Stability analysis. To provide the stability analysis of the closed loop system in Fig. 3, we will first define an important assumption as follows:

Assumption 1[49] Consider a first order continuous differential variable $\alpha_1(t) \in [0, \infty)$, which has a limit with $t \to \infty$. Thus, the second derivative of $\alpha(t)$ does existed and bounded for all $t \in [0, \infty)$ such that first derivative holds the condition $\lim_{t \to \infty} \dot{\alpha}_1(t) = 0$.

Consider the model of UGV Eqs. (1) and (2), the posture control error Eqs. (17) and (20), the NDO Eqs. (21)–(26), and controllers Eqs. (5) and (27). Then a compensated error-inequality as follows

$$\|e_c{}^2\| \geq \frac{H_{max}}{K_{c_{min}} - \|\bar{C}_m\|} (K_{c_{min}} > \|\bar{C}_m\|) \tag{28}$$

can be refined and we can find error-based asymptotic stability.

while,

$$\bar{C}_m = C_1(q, \dot{q})(\Omega - \eta) + C_2(q)\dot{\rho}(q, \mu) + C_3(q)\rho(q, \mu) \tag{29}$$

.Consider the following Lyapunov function

$$V_1 = k_1(e_x + l(1 - \cos(e_\theta))^2 + k_1(e_y - l\sin(e_\theta))^2 + 2k_3 v_r(1 - \cos(e_\theta)) + V_2 \tag{30}$$

where $V_2 = \frac{e_c^T \bar{M} e_c}{2}$. In addition, the Lyapunov function $V_1 \geq 0$ and $V_1 = 0$ when $e = 0$ and $e_c = 0$. Now, by taking the following derivative

$$\dot{V}_1 = 2k_1(e_x + l(1 - \cos(e_\theta))(\dot{e}_x + l\sin(e_\theta)\dot{e}_\theta) + 2k_1(e_y - l\sin(e_\theta)(\dot{e}_\theta - l\cos(e_\theta) + 2k_e v_r \sin(e_\theta)\dot{e}_\theta + \dot{V}_2$$

(31)

Let the function of Eq. (20) has an error $e_F = F - \hat{F}$, and controller of Eq. (27), we can differentiate V2 and get

$$\dot{V}_2 = \frac{1}{2} e_c^T (\dot{\bar{M}} - 2 \cdot \bar{C}_m) \cdot e_c + e_c^T \cdot (-K_c \cdot e_c + e_F)$$

(32)

It is important to notice here that skew symmetry property makes the first term zero, which allows us to rewrite the $\dot{V}_2$ as

$$\dot{V}_2 = e_c^T (-K_c \cdot e_c + e_F) \leq (-K_{c_{min}} \|e_c{}^2\| + \|C_m\|\|e_c{}^2\| + \|e_c\|H_{max})$$

(33)

where H_{max} is the maximum NDO estimation error. Now, we can substitute the inequality and the derivatives of the velocity error in Eq. (17), we have

$$\dot{V}_1 \leq 2 \cdot k_x \cdot e_x(v_r \cdot \cos(e_3) - v_c) + (2 \cdot k_y \cdot v_r \cdot e_2) + 2 \cdot k_\theta v_r \cdot (\omega_r - \omega_c) \cdot \sin(e_3) - K_{c_{min}}|e_c\|^2 + \|\bar{H}_m\|\|e_c{}^2\| + \|e_c C_{maxc}\|$$

(34)

Now, we will introduce EPKC from Eq. (22), defining $\hat{k}_y = \frac{k_x}{\hat{k}_\theta v_r}$ and substitute in Eq. (32).

$$\dot{V}_1 \leq -2\hat{k}_x^x - 2\hat{k}_\theta^2 \cdot v_r^2 \sin(e_3)^2 + \|e_c{}^2\|(-K_{c_{min}} + \|C_m\|) + \|e_c H_{max}\|$$

(35)

In which it can be seen that the first two terms of the above equation are negative, which also guarantees negative $\dot{V}_1$ as long as the following is satisfied

$$\|e_c\| \geq \frac{H_{max}}{K_{c_{min}} - \|C_m\|}$$

(36)

Therefore, we can write-down the velocity error between EPKC and the estimation velocity as

$$e_c = \begin{bmatrix} \bar{v}_e \\ \bar{\omega}_e \end{bmatrix} = \begin{bmatrix} (v_r \cos(e_3) + \hat{k}_x e_1) - \hat{v} \\ (\omega_r + \hat{k}_y v_r e_y + \hat{k}_\theta) - \hat{\omega} \end{bmatrix}$$

(37)

The error functions $e_s = [e, e_c]^T$ is bounded, therefore, both $\|e\|$ and $\|\dot{e}\|$ are bounded. In addition, $\lim_{t \to \infty} \dot{V}_1 = 0$ can be obtained from the Assumption 1. Thus, it can be concluded that posture and velocity errors of the UGV are asymptotically stable.

Results and discussions

In this section, we discuss the simulation-based experiments performed in MATLAB. The physical parameters of UGV are set as follows: $b = 0.67$ m, $r = 0.30$ m, $d = 0.6$ m, $m_c = 50$ kg, $m_w = 2$ kg, $J = 15$ kg m^2. The skid-slip are considered as $[\eta_1, \eta_2, \mu] = 0.02 \cdot [\sin(0.3t), 1, 1]$. The external torque-based disturbance $t_d = [0.01 \cos(0.5t) 0.01 \sin(0.5t)]^T$.

For NMPC gain tuning, we have selected the weighting matrices $Q = diag[5, 10, 0.01]$, $\bar{Q} = diag[10, 10, 10]$, and $R = diag[0.05, 0.05]$. The prediction horizon of $N_p = 20$ for the simulations related to path-tracking, and $N_p = 22$, the simulations related to tracking a circle. Additionally, the sample time of $T_s = 0.01$ s is selected. The input torque is saturated and assumed to be $\tau_{min} = -100$ Nm and $\tau_{max} = 100$ Nm. Finally, voltages on the right and left motors are $V_{r_{min}} = -12$ V, $V_{r_{max}} = 12$ V, $V_{l_{min}} = -12$ V and $V_{l_{max}} = 12$ V, respectively.

Simulation 1: Path-tracking in a large map using PPQ-Dijkstra. In this path-tracking experiment, the tracking process is combined with a 2 DoF optimal planner (PPQ-Dijkstra), and considered the University of New South Wale's evacuation map. The map has a size of (1350 m × 600 m) which consists of the buildings of the campus. The intuition behind the experiment is to avoid moving obstacles (MOs) and static obstacles (SOs) in a large map. For this purpose, three destinations are provided as shown in the Fig. 3a, while UGV able to select for selecting the shortest path from the PPQ-Dijkstra (which can deal with multiple destinations). The color lines in the Fig. 3a represents how expensive it is to reach the goal. For example, red has the highest value of 300 which refers to a higher cost, so it is necessary to follow the vehicle to follow blue color (0 value) in order to follow the lowest cost. The NMPC is looking after the non-holonomic constraints of the vehicle as well as the collision avoidance. Figure 3b represents the UGV avoiding the incoming MO, additionally, providing the prediction of the vehicle's state.

In terms of the performance of the proposed scheme, Fig. 3c represents the velocity profile applied to the real-platform. Clearly, the chattering in both longitudinal velocity and yaw rate represents the collision avoidance in the environment. As far as tracking errors are concerned, Fig. 3d shows that the actual states of the UGV has a good accuracy.

Simulation 2: Circle trajectory tracking. For this simulation experiment, we have considered the following expression to form a circle:
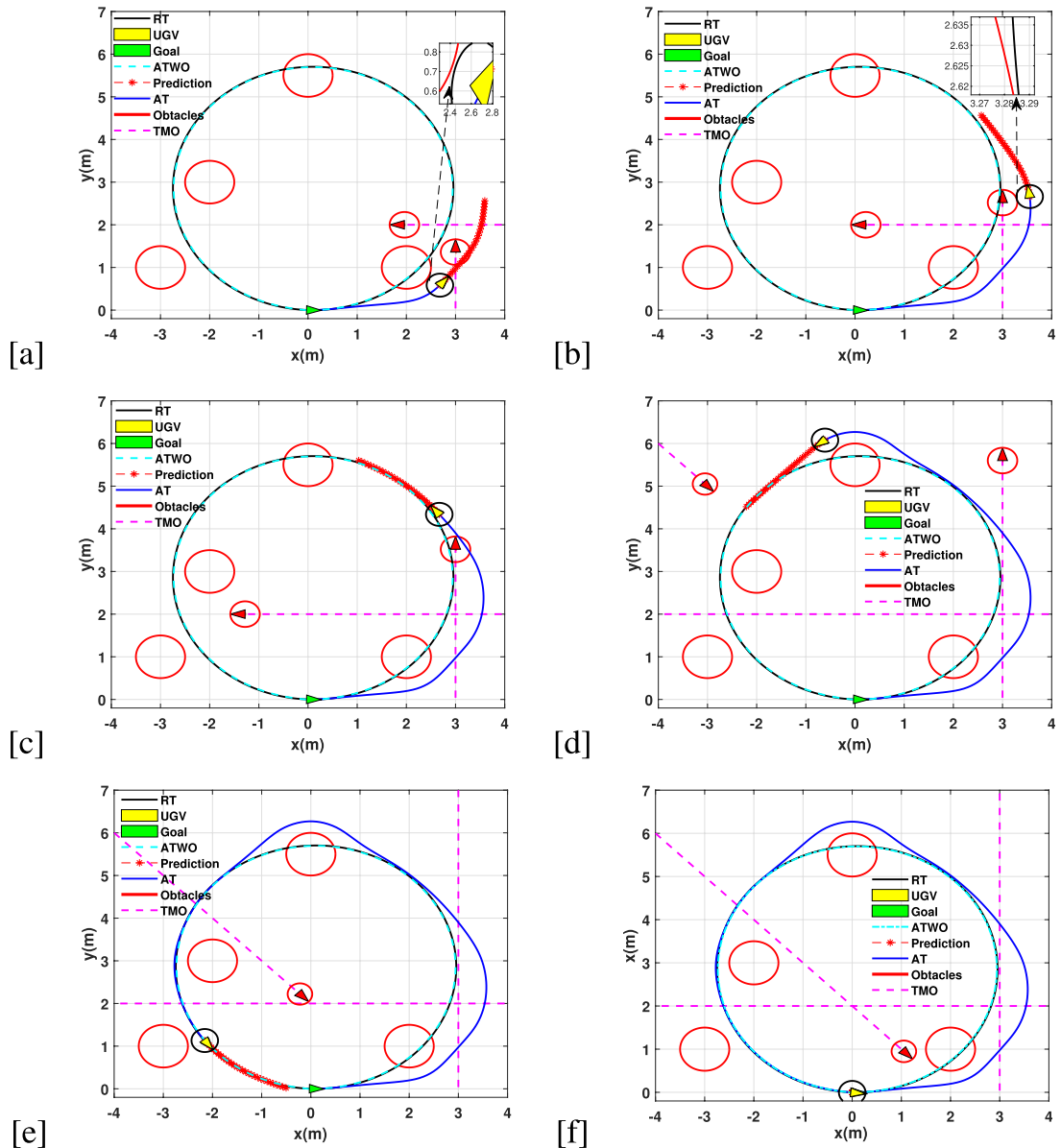
Figure 4. (**a**) Avoided first SO and MO by predicting them. (**b**) Speeds up to safely overtake the incoming vehicle. (**c**) Catching up the RT (**d**) Avoiding another SO and predicting incoming MO to slow down. (**e**) Approaching to the goal location. (**f**) Safely reached to the goal.

$$\begin{cases} x_r(t) = 5sin(0.2t), \qquad y_r(t) = 5cos(0.2t) \end{cases} \tag{38}$$

where $t \in [0, 25]$, for the above expression, the initial conditions are aligned with the reference trajectory as $[x, y, \theta] = [0, , 0, 0]$. The prediction horizon for this experiment is $N_p = 22$. The input constraints are selected differently for this experiment. We have the following input constraints:

$$-1m/s \le v \le 1m/s, \quad -45deg/s \le \omega \le 45deg/s, \quad -1m/s^2 \le a_{r1} \le 1m/s^2, \quad -45deg/s^2 \le a_{r2} \le 45deg/s^2.$$

We have considered four SOs and three MOs as shown in the Fig. 4. The SOs are located at $(x_1, y_1) = (2, 1)$, $(x_2, y_2) = (0, 5.5)$, $(x_3, y_3) = (-3, 1)$, and $(x_4, y_4) = (-2, 3)$. All the SOs are having a radius of $0.5m$. For MOs, the starting locations are $(x_5, y_5, \theta) = (-8, 10, pi/4), (x_6, y_6, \theta) = (3, 0, pi/2)$, and $(x_7, y_7) = (4, 2, -pi)$, while these MOs are subject to linear obstacles with linear velocities of $0.5\ m/s$, $0.4\ m/s$, and $0.6\ m/s$, respectively. The intention for this simulation is to avoid collisions and follow the reference trajectory. For first collision avoidance, the UGV predicts SO and MO to divert from the reference trajectory as shown in the Fig. 4a. Since, the longitudinal velocity of the vehicle is $1\ m/s$, it increases speed to avoid the MO safely as shown in the Fig. 4b. In particular, this is an example of a vehicle over taking another vehicle in the environment. The UGV tracks back on the
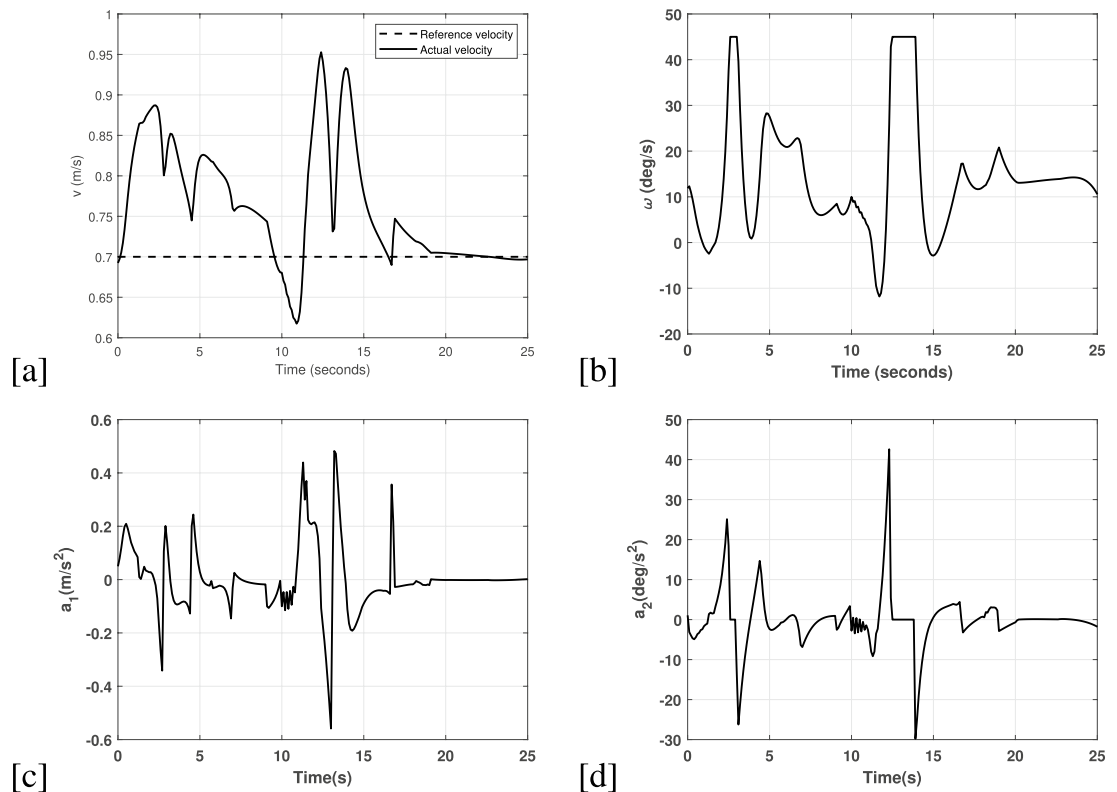
Figure 5. (**a**) Longitudinal velocity applied to the UGV in the simulation experiment 3. (**b**) Angular velocity applied to the UGV in the simulation experiment 3. (**c**) Rate of change in the longitudinal velocity applied to the UGV in the simulation experiment 3. (**d**) Rate of change in the angular velocity applied to the UGV in the in the simulation experiment 3.

proposed nominal trajectory, as shown in the Fig. 4c, in which the achieved platform's trajectory does converge to the nominal trajectory, after a short transient during the collision avoidance event. Soon after synchronizing with the nominal trajectory, the platforms detects the presence of an unexpected SO, which requires another collision avoidance manoeuvre, as shown in Fig. 4d. In Fig. 4e the platform properly continues the trajectory tracking process, for finally dealing with a second incoming DO and an additional SO, both requiring an avoidance treatment, which is successfully performed, for finally completing the required trajectory as shown in Fig. 4f.

In terms of meeting the input constraints imposed by the NMPC, Fig. 5 illustrates the applied velocities and accelerations to the UGV. The reference velocity is considered as $0.7m/s$, which the vehicles tries to match. However, since the environment is filled with obstacles, it can only reach the reference velocity once the obstacles are avoided as shown in the Fig. 5a. Additionally, the longitudinal velocity increases in the time instances $10(sec) - 15(sec)$, which indicates avoidance of obstacles safely and then converging to the reference velocity. For the angular velocity, a similar trend can be observed in the Fig. 5b. For the rate of change in the longitudinal and angular velocities in Fig. 5c,d, chattering are visible in the signal. However, it always converges to zero once it meets the reference trajectories. The reason for having chattering are due to the continuous occurrence of multiple obstacles, which diverts the UGV from it's nominal trajectory.

In terms of trajectory tracking error, we have computed a case with no obstacles and compared it with the one, which we have simulated in this experiment as shown in the Fig. 6a. The mean error for both situations are respectable. However, it is appreciable that without obstacles the tracking error is very small, while the error with obstacles are still having reasonable values. Additionally, for control action applied to the UGV, the applied voltages are converging to a steady-state as shown in the Fig. 6b, which proves that motor would work adequately despite these challenging conditions of obstacles and skid-slip.

Performance evaluation of the proposed method. To validate the robustness of the proposed method, we have evaluated our method with[26]. The comparison is performed as per the error accuracy, capability of the NMPC to run on a single core and multiple cores CPU, and average processing time of the NMPC to compute a single optimization iteration. As mentioned earlier in the paper, our numerical method is based on multiple shooting problem, which is tested in different prediction horizon of time.

The ability to perform any optimization method on an on-board processor must be feasible. For instance, a single-core CPU with limited resources must be able to solve the optimization problem effectively. For this comparison, Table 1 discusses the proposed method utilizes the average processing time per iteration (*PT/I*) for this paper (using MS) and compared with SS. In terms of computing NMPC on single or multi-core CPUs,
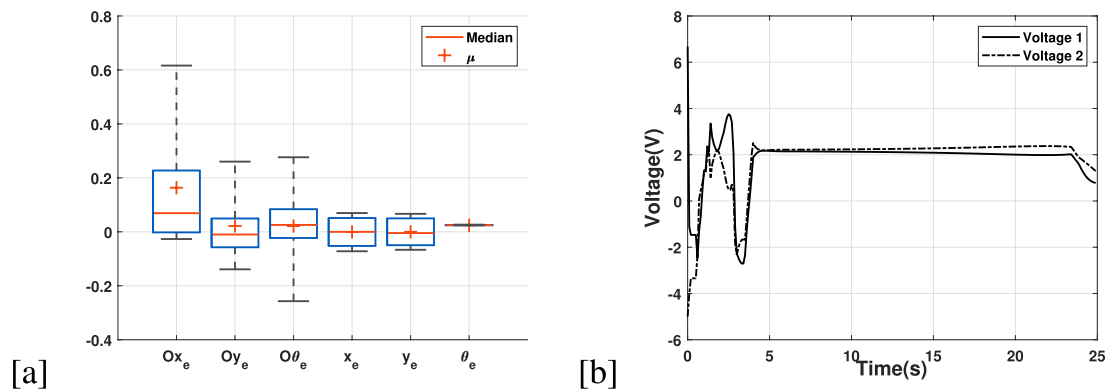
Figure 6. (**a**) Box plot to distinguish the actual state tracking errors; where pose (x,y) are in *c m* and heading is in *rad*. Additionally, Ox_e, Oy_e and $O\theta_e$ represents obstacles case. (**b**) Applied voltages to the UGV in the simulation experiment 3.

N_p	Single core this paper (ms)	Single core[26] (ms)	Multi-core this paper (ms)	Multi-core[26] (ms)
15	1.2	10.3	1	10
50	3.32	21.47	3	21
100	10.6	38.61	10	38
150	17.8	51.82	17	51

Table 1. NMPC average *PT/I* for single and multi-core CPUs.

N_p	This paper (accuracy (cm))	Accuracy (cm)[26]
15	− 3.0 to 3.0	− 10.0 to 10.0
50	− 0.5 to 0.5	− 2.5 to 2.5
100	− 0.05 to 0.05	− 0.25 to 0.25
150	− 0.01 to 0.01	− 0.01 to 0.01

Table 2. Comparison of Euclidean distance accuracy among MS and SS numerical methods.

the average *PT/I* in this paper is relatively better than SS method. In addition, even if we increase the prediction horizon to a larger number $N_p = 150$, the *PT/I* always feasible to run this on the CPU with limited resources. This also enables to perform a real experiment with an on board processor with limited processing capabilities.

In contrast to[26], a remarkable finding is that accuracy in the MS method is optimal for a small and a large prediction horizon. Although, an increase in the prediction horizons may cause a slower processing but a lower error as shown in the Table 2. However, the issue with increasing a too large prediction horizon isn't feasible for some cases. For instance, in[26], a prediction horizon of $N_p = 100$ was used to run the path planner, which has an accuracy of 2*cm*. However, this can lead to a larger processing time. Thus, based on these findings, the proposed method is reasonable for real-time applications.

Skid-slip analysis. In terms of including the skid-slip applied to simulations, we have also used the PWL-based function proposed in Fig. 7a for the circle trajectory simulation. Furthermore, the side-slip angle of front wheels is within a reasonable range (slip angle) $\beta \leq |10deg|$ as shown in Fig. 7b,c.

Conclusions

In this paper, we propose a tracking control and planner for a UGV in the presence of a skid-slip and dynamic and static objects. The proposed technique consists of a fast NMPC-based path-planner, novel evolutionary programming-based kinematic controller, a dynamic controller with a nonlinear disturbance observer, and a voltage driving circuit. We have performed simulation experiments by considering actual physical parameters to drive the vehicle safely to the goal location. First, a large map of UNSW is used, which is fed to PPQ-Dijkstra algorithm, and NMPC is combined with the map to avoid collisions and non-holonomic constraints. Finally, a test in which the UGV must follow a sizeable circular trajectory under the presence of static and moving objects is performed. The results obtained from these experiments validate that the proposed method can safely drive the vehicle to the goal location. To validate the performance of our proposed technique, we have performed
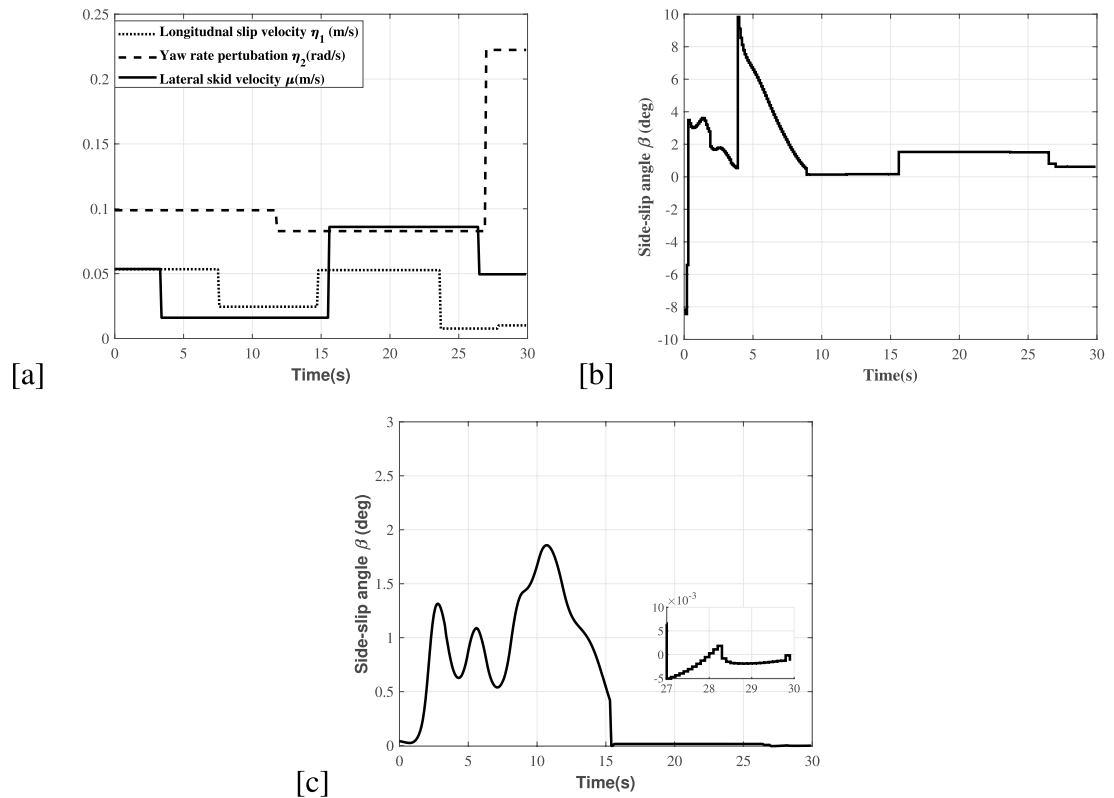
Figure 7. (**a**) PWL function-based slippage and skidding velocities. (**b**) Side slip angle of the front wheels for the simulation 1. (**c**) Side slip angle of the front wheels for the simulation 2.

the numerical multiple-shooting method, being restricted to use only one core and running in multiple cores. Compared to the existing literature, the proposed technique has improved the processing capability of the NMPC to run in a CPU with limited resources and improves the error accuracy.

References

1. Choset, H. M. *et al. Principles of robot motion: Theory, algorithms, and implementation* (MIT press, 2005).
2. Kanayama, Y., Kimura, Y., Miyazaki, F. & Noguchi, T. A stable tracking control method for a non-holonomic mobile robot. In *IROS*, 1236–1241 (1991).
3. Sun, H. *et al.* Application of the Udwadia–Kalaba approach to tracking control of mobile robots. *Nonlinear Dyn.* **83**, 389–400 (2016).
4. Biswas, J. & Veloso, M. Depth camera based indoor mobile robot localization and navigation. In *2012 IEEE International Conference on Robotics and Automation*, 1697–1702 (IEEE, 2012).
5. Thrun, S. Learning metric-topological maps for indoor mobile robot navigation. *Artif. Intell.* **99**, 21–71 (1998).
6. Bailey, T. *Mobile robot localisation and mapping in extensive outdoor environments* (Citeseer, 2002).
7. Hagras, H., Callaghan, V. & Collry, M. Outdoor mobile robot learning and adaptation. *IEEE Robot. Autom. Mag.* **8**, 53–69 (2001).
8. Dong, W. & Kuhnert, K.-D. Robust adaptive control of nonholonomic mobile robot with parameter and nonparameter uncertainties. *IEEE Trans. Robot.* **21**, 261–266 (2005).
9. Low, C. B. & Wang, D. Gps-based tracking control for a car-like wheeled mobile robot with skidding and slipping. *IEEE/ASME Trans. Mechatron.* **13**, 480–484 (2008).
10. Odry, Á., Fuller, R., Rudas, I. J. & Odry, P. Kalman filter for mobile-robot attitude estimation: Novel optimized and adaptive solutions. *Mech. Syst. Signal Process.* **110**, 569–589 (2018).
11. Zuo, S.-l. Mechanical fault diagnosis and repair of the mobile robot based on "four diagnosis methods". *DEStech Trans. Comput. Sci. Eng.* (2019).
12. He, W., Li, Z. & Chen, C. P. A survey of human-centered intelligent robots: Issues and challenges. *IEEE/CAA J. Autom. Sinica* **4**, 602–609 (2017).
13. Niloy, A. et al. Critical design and control issues of indoor autonomous mobile robots: A review. *IEEE Access* (2021).
14. Mohanan, M. & Salgoankar, A. A survey of robotic motion planning in dynamic environments. *Robot. Auton. Syst.* **100**, 171–185 (2018).
15. Rubio, F., Valero, F. & Llopis-Albert, C. A review of mobile robots: Concepts, methods, theoretical framework, and applications. *Int. J. Adv. Robot. Syst.* **16**, 1729881419839596 (2019).
16. Qu, Z., Wang, J., Plaisted, C. E. & Hull, R. A. Global-stabilizing near-optimal control design for nonholonomic chained systems. *IEEE Trans. Autom. Control* **51**, 1440–1456 (2006).
17. Morin, P. & Samson, C. Control of nonholonomic mobile robots based on the transverse function approach. *IEEE Trans. Robot.* **25**, 1058–1073 (2009).

18. Li, Z. *et al.* Trajectory-tracking control of mobile robot systems incorporating neural-dynamic optimized model predictive approach. *IEEE Trans. Syst. Man Cybern. Syst.* **46**, 740–749 (2015).
19. Shi, D., Zhang, J., Sun, Z., Shen, G. & Xia, Y. Composite trajectory tracking control for robot manipulator with active disturbance rejection. *Control Eng. Pract.* **106**, 104670 (2021).
20. Zhang, X., Wang, R., Fang, Y., Li, B. & Ma, B. Acceleration-level pseudo-dynamic visual servoing of mobile robots with backstepping and dynamic surface control. *IEEE Trans. Syst. Man Cybern. Syst.* **49**, 2071–2081 (2017).
21. Xie, Y. *et al.* Coupled fractional-order sliding mode control and obstacle avoidance of a four-wheeled steerable mobile robot. *ISA Trans.* **108**, 282–294 (2021).
22. Saenz, A. *et al.* Velocity control of an omnidirectional wheeled mobile robot using computed voltage control with visual feedback: Experimental results. *Int. J. Control Autom. Syst.* **19**, 1089–1102 (2021).
23. Fateh, M. M. & Arab, A. Robust control of a wheeled mobile robot by voltage control strategy. *Nonlinear Dyn.* **79**, 335–348 (2015).
24. Sathya, A. et al. Embedded nonlinear model predictive control for obstacle avoidance using panoc. In *2018 European control conference (ECC)*, 1523–1528 (IEEE, 2018).
25. Hu, Y. *et al.* Nonlinear model predictive control for mobile medical robot using neural optimization. *IEEE Trans. Ind. Electron.* (2020).
26. Lu, Q. et al. Targeting posture control with dynamic obstacle avoidance of constrained uncertain wheeled mobile robots including unknown skidding and slipping. *IEEE Trans. Syst. Man Cybern. Syst.* (2020).
27. Patle, B. *et al.* A review: On path planning strategies for navigation of mobile robot. *Def. Technol.* **15**, 582–606 (2019).
28. Chang, L., Shan, L., Jiang, C. & Dai, Y. Reinforcement based mobile robot path planning with improved dynamic window approach in unknown environment. *Auton. Robots* **45**, 51–76 (2021).
29. Xiao, H. & Chen, C. P. Time-varying nonholonomic robot consensus formation using model predictive based protocol with switching topology. *Inf. Sci.* **567**, 201–215 (2021).
30. Dao, P. N., Nguyen, H. Q., Nguyen, T. L. & Mai, X. S. Finite horizon robust nonlinear model predictive control for wheeled mobile robots. *Math. Probl. Eng.* **2021** (2021).
31. Sun, Z., Xia, Y., Dai, L., Liu, K. & Ma, D. Disturbance rejection mpc for tracking of wheeled mobile robot. *IEEE/ASME Trans. Mechatron.* **22**, 2576–2587 (2017).
32. Khan, S., Guivant, J. & Li, X. Design and experimental validation of a robust model predictive control for the optimal trajectory tracking of a small-scale autonomous bulldozer. *Robot. Auton. Syst.* 103903 (2021).
33. Khan, S. & Guivant, J. Nonlinear model predictive path-following controller for a small-scale autonomous bulldozer for accurate placement of materials and debris of masonry in construction contexts. *IEEE Access* **9**, 102069–102080 (2021).
34. Khanpoor, A., Khalaji, A. K. & Moosavian, S. A. A. Modeling and control of an underactuated tractor-trailer wheeled mobile robot. *Robotica* **35**, 2297–2318 (2017).
35. Juang, C.-F., Jeng, T.-L. & Chang, Y.-C. An interpretable fuzzy system learned through online rule generation and multiobjective aco with a mobile robot control application. *IEEE Trans. Cybern.* **46**, 2706–2718 (2015).
36. Ma, H., Chen, M., Yang, H., Wu, Q. & Chadli, M. Switched safe tracking control design for unmanned autonomous helicopter with disturbances. *Nonlinear Anal. Hybrid Syst.* **39**, 100979 (2021).
37. Khan, A. H., Li, S. & Luo, X. Obstacle avoidance and tracking control of redundant robotic manipulator: An rnn-based metaheuristic approach. *IEEE Trans. Ind. Inform.* **16**, 4670–4680 (2019).
38. Wang, D. & Low, C. B. Modeling skidding and slipping in wheeled mobile robots: Control design perspective. In *2006 IEEE/RSJ International Conference on Intelligent Robots and Systems*, 1867–1872 (IEEE, 2006).
39. Wei, S., Zefran, M., Uthaichana, K. & DeCarlo, R. A. Hybrid model predictive control for stabilization of wheeled mobile robots subject to wheel slippage. In *Proceedings 2007 IEEE International Conference on Robotics and Automation*, 2373–2378 (IEEE, 2007).
40. Chen, M. Robust tracking control for self-balancing mobile robots using disturbance observer. *IEEE/CAA J. Autom. Sinica* **4**, 458–465 (2017).
41. Li, L., Wang, T., Xia, Y. & Zhou, N. Trajectory tracking control for wheeled mobile robots based on nonlinear disturbance observer with extended Kalman filter. *J. Frankl. Inst.* **357**, 8491–8507 (2020).
42. Chen, M., Shi, P. & Lim, C.-C. Robust constrained control for mimo nonlinear systems based on disturbance observer. *IEEE Trans. Autom. Control* **60**, 3281–3286 (2015).
43. Andersson, J. A., Gillis, J., Horn, G., Rawlings, J. B. & Diehl, M. Casadi: A software framework for nonlinear optimization and optimal control. *Math. Program. Comput.* **11**, 1–36 (2019).
44. Jin, X., Yu, J., Qin, J., Zheng, W. X. & Chi, J. Adaptive perturbation rejection control and driving voltage circuit designs of wheeled mobile robots. *J. Frankl. Inst.* **358**, 1185–1213 (2021).
45. Robledo, A., Guivant, J. & Cossell, S. Pseudo priority queues for real-time performance on dynamic programming processes applied to path planning. In *Australasian Conference on Robotics and Automation* (2010).
46. Wang, D. & Low, C. B. Modeling and analysis of skidding and slipping in wheeled mobile robots: Control design perspective. *IEEE Trans. Robot.* **24**, 676–687 (2008).
47. Chen, M. Disturbance attenuation tracking control for wheeled mobile robots with skidding and slipping. *IEEE Trans. Ind. Electron.* **64**, 3359–3368 (2016).
48. Fierro, R. & Lewis, F. L. Control of a nonholonomic mobile robot using neural networks. *IEEE Trans. Neural Netw.* **9**, 589–600 (1998).
49. Slotine, J.-J.E. *et al. Applied Nonlinear Control* Vol. 199 (Prentice hall, Englewood Cliffs, 1991).
50. Chen, W.-H., Ballance, D. J., Gawthrop, P. J. & O'Reilly, J. A nonlinear disturbance observer for robotic manipulators. *IEEE Trans. Ind. Electron.* **47**, 932–938 (2000).

Acknowledgements

Author contributions

S.K. prepared Model Predictive Control, Kinematic and Dynamic control, simulation results, figures, and tables. J.G. prepared PPQ-Dijkstra algorithm, provided the UNSW Map, and suggested the idea of Simulation 1. All authors reviewed the manuscript.

Competing interests

The authors declare no competing interests.

Additional information

Correspondence and requests for materials should be addressed to S.K.