Robotics and Autonomous Systems 131 (2020) 103570

Contents lists available at ScienceDirect

Robotics and Autonomous Systems

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

Planning the trajectory of an autonomous wheel loader and tracking its trajectory via adaptive model predictive control

Junren Shi, Dongye Sun *, Datong Qin, Minghui Hu, Yingzhe Kan, Ke Ma, Ruibo Chen

State Key Laboratory of Mechanical Transmissions & School of Automotive Engineering, Chongqing University, Chongqing 400044, China

ARTICLE INFO

Article history:
Received 27 April 2019
Received in revised form 10 March 2020
Accepted 19 May 2020
Available online 26 May 2020

Keywords:
Automatic drive
Wheel loader
Trajectory planning
Trajectory tracking
Model predictive control

ABSTRACT

In a typical operation mode, a wheel loader frequently accelerates and decelerates, and the curvature of the driving path is inconsistent. In the past, autonomous vehicle trajectory planning has not considered the related changes in the velocity of the vehicle. Therefore, the trajectory tracking control process has seldom considered the impact of curving paths on the trajectory tracking performance. To address these problems, this study evaluated an autonomous wheel loader based on the trajectory of its non-uniform driving motion and constructed an adaptive model predictive control (AMPC) trajectory tracking system that considers disturbances in the path curvature. The trajectory of the autonomous wheel loader was then tracked using the proposed AMPC system with a planned non-uniform motion trajectory as the target. Its performance was then compared with that of a conventional model predictive control (MPC) trajectory tracking system that does not consider any path curvature disturbances. The maximum displacement error and heading error obtained by the proposed AMPC system were found to be 65.7% and 60%, respectively, smaller than those obtained by the MPC system. The desired trajectory can also be tracked well under different curvature amplitudes using the AMPC trajectory tracking system, ensuring active safety performance of an autonomous wheel loader in the process of trajectory tracking.

1. Introduction

Wheel loaders are a type of flexible and mobile transportation equipment often used to transport materials on uneven terrain such as that found in mines and on construction sites. They consist of a front and rear body, which are connected by a hinge joint point and a swing ring. The front and rear body are connected by hydraulic actuators to steer the wheel loader. There is no relative connection between the wheel and the frame in the steering process; therefore, the steering drive axle and other components were omitted. This structure reduced the turn radius of the wheel loader, improved the maneuverability of the vehicle, and gave it a good adaptability in its operating environment [1,2]. However, owing to this steering structure, the trajectory planning and tracking methods typically used in conventional autonomous passenger vehicles cannot be applied to wheel loaders.

The adverse working environment and frequent acceleration, deceleration, and steering actions of wheel loaders present considerable challenges to the physical and mental state of drivers, making it nearly impossible to guarantee high working efficiency and quality over long periods of operation. In order to improve the potential efficiency of wheel loader operation, automation has been a focus of attention in the field of earthmoving machinery for many years [3–5]. Roberts et al. [6] proposed five stages from manual driving to fully automatic driving for the underground mining vehicles. Frank et al. [7] proposed five similar stages for wheel loaders, consisting of manual operation, auxiliary operation, semi-automatic operation, highly automatic operation, and fully automatic operation. In general, wheel loader automation can be divided into two types: automatic driving and automatic bucket filling. Trajectory planning and trajectory tracking are considered automatic driving activities. The process of trajectory planning and trajectory tracking for automatic driving of a wheel loader involves a variety of sensors, including cameras to detect operational areas, vehicles, and other materials, laser radar and ultrasonic sensors to detect obstacles and take precise distance measurements, and an inertial measurement unit and wheel encoder for dead reckoning. During the automatic driving process, the wheel loader determines its surrounding environment using these various sensors and calculates a feasible trajectory using a planning algorithm. Then, under the action of its control system, it realizes longitudinal and lateral control through steering, acceleration, and deceleration actions to drive in accordance with the established trajectory.

* Corresponding author.
E-mail address: dysun@cqu.edu.cn (D. Sun).

https://doi.org/10.1016/j.robot.2020.103570
0921-8890/

Nomenclature

a	Acceleration of the wheel loader
$\boldsymbol{A}$, $\boldsymbol{B}_u$, $\boldsymbol{C}$	System matrices, discrete time, $\boldsymbol{A} \in \mathbb{R}^{l\times l}$, $\boldsymbol{B}_u \in \mathbb{R}^{l\times m}$, $\boldsymbol{C} \in \mathbb{R}^{n\times l}$
$\boldsymbol{A}_c$, $\boldsymbol{B}_{cu}$, $\boldsymbol{C}_c$	Nonlinear system matrices, continuous time, $\boldsymbol{A}_c \in \mathbb{R}^{l\times l}$, $\boldsymbol{B}_{cu} \in \mathbb{R}^{l\times m}$, $\boldsymbol{C}_c \in \mathbb{R}^{n\times l}$
$\boldsymbol{A}_l$, $\boldsymbol{B}_{lu}$, $\boldsymbol{C}_l$	Linear system matrices, continuous time, $\boldsymbol{A}_l \in \mathbb{R}^{l\times l}$, $\boldsymbol{B}_{lu} \in \mathbb{R}^{l\times m}$, $\boldsymbol{C}_l \in \mathbb{R}^{n\times l}$
$\boldsymbol{A}_{rc}$, $\boldsymbol{B}_{ru}$	System matrices, continuous time, reference, $\boldsymbol{A}_{rc} \in \mathbb{R}^{l\times l}$, $\boldsymbol{B}_{ru} \in \mathbb{R}^{l\times m}$
$\boldsymbol{B}_{cd}$	Nonlinear input disturbance vector, continuous time, $\boldsymbol{B}_{cd} \in \mathbb{R}^{l}$
$\boldsymbol{B}_d$	Input disturbance vector, discrete time, $\boldsymbol{B}_d \in \mathbb{R}^{l}$
$\boldsymbol{B}_{ld}$	Linear input disturbance vector, continuous time, $\boldsymbol{B}_{ld} \in \mathbb{R}^{l}$
$\boldsymbol{B}_{rd}$	Input disturbance vector, continuous time, reference, $\boldsymbol{B}_{rd} \in \mathbb{R}^{l}$
c	Control horizon
C_f, S_f	Fresnel integrals
$C\alpha$	Cornering stiffness coefficient
$C\alpha_f$, $C\alpha r$	Cornering stiffness coefficient of front or rear tire
$C\alpha_1$, $C\alpha_2$	Lateral force coefficient of front or rear tire
$C\gamma$	Torsional damping coefficient
d_r	Integrating disturbance, reference
e_1	Displacement error
e_2	Heading error
$\boldsymbol{E}(k)$	Deviation of the output at time k
$f(\boldsymbol{x}_r, \boldsymbol{u}_r, d_r)$	System function
g	Gravitational acceleration
F_{yf}, F_{yr}	Lateral tire force
$\boldsymbol{H}_k$, $\boldsymbol{G}_k$, $\boldsymbol{P}_k$	Parameter matrices of the objective function $\mathbf{J}(\mathbf{x}(k), \Delta\mathbf{U}(k)) = [\Delta\mathbf{U}(k)^{\mathrm{T}}, \varepsilon]^{\mathrm{T}}\boldsymbol{H}_k[\Delta\mathbf{U}(k)^{\mathrm{T}}, \varepsilon] + \boldsymbol{G}_k[\Delta\mathbf{U}(k)^{\mathrm{T}}, \varepsilon] + \boldsymbol{P}_k$
$\boldsymbol{I}_{n\times n}$	Identity matrix with appropriate dimensions, $\boldsymbol{I}_{n\times n} \in \mathbb{R}^{n\times n}$
I_1, I_2	Yaw inertia at the center of front and rear bodies
$J(\boldsymbol{x}(k), \Delta\mathbf{U}(k))$	MPC objective function
k	Steering curvature or nonnegative integer denoting the sample number
k+1| k	Represents the prediction for k+1 time at time k
K	Driving path curvature
K_r	Driving path curvature, reference
$K\gamma$	Torsional spring stiffness coefficient
l	State dimension
l_a	Distance from front body center of mass to front wheels
l_b	Distance from articulation point to front body center of mass
l_c	Distance from articulation point to rear body center of mass
l_d	Distance from rear wheels to rear body center of mass
L_1, L_2	Length of front body or rear body
m	Input dimension
m_0	Mass of wheel loader
m_1, m_2	Mass of front body or rear body
n	Output dimension
N_{speed}	Number of discretized velocities
N_1, N_2	Front or rear static tire load
p	Prediction horizon
$P_f(x_f, y_f)$, $P_r(x_r, y_r)$	Midpoint of the front and rear axles, respectively
$P_t(x_t, y_t, \theta_t)$, $P_l(x_l, y_l, \theta_l)$, $P_u(x_u, y_u, \theta_u)$	Point pose of turning, loading and unloading, respectively
q_s, q_i, q_j and q_g	Node of the random tree
$\boldsymbol{Q}$, $\boldsymbol{R}$	Pair of weight matrices in the cost function of predictive control, $\boldsymbol{Q} \in \mathbb{R}^{n\times n}$, $\boldsymbol{R} \in \mathbb{R}^{m\times m}$
r	Radius of planning path
r_{min}	Minimum turning radius of the wheel loader
R	Steering radius
$\mathbb{R}$	Real numbers
s	Arc length
$\boldsymbol{S}_x$, $\tilde{\boldsymbol{I}}$, $\boldsymbol{S}_d$, $\boldsymbol{S}_u$	Matrices used in the prediction equation $\boldsymbol{Y}_p\ (k{+}1|\ k) = \boldsymbol{S}_x\Delta\boldsymbol{x}(k) + \tilde{\boldsymbol{I}}\boldsymbol{y}(k) + \boldsymbol{S}_d\ \Delta\boldsymbol{d}(k) + \boldsymbol{S}_u\Delta\mathbf{U}(k)$
t	Time

T_s	Sampling time of the system
$\boldsymbol{u}_r$	Input (manipulated variable) vector, reference, $\boldsymbol{u}_r \in \mathbb{R}^m$
u	Longitudinal speed of vehicle
u_f	Longitudinal speed of front part
v	Lateral speed of vehicle
v_f	Lateral speed of front part
$\boldsymbol{x}$	State vector, $\boldsymbol{x} \in \mathbb{R}^l$
$\dot{x}$	Time derivative of x
$\boldsymbol{x}_r$	State vector, reference, $\boldsymbol{x}_r \in \mathbb{R}^l$
(X, Y)	Reference points that the wheel loader is being steered towards
$(X, Y, \theta)_k$ $(k = start, target, rand, nearest, new)$	Position of the wheel loader
$\boldsymbol{y}$	Output vector, $\boldsymbol{y} \in \mathbb{R}^n$
$\boldsymbol{y}_{ref}$	Reference output vector, $\boldsymbol{y}_{ref} \in \mathbb{R}^n$
$\boldsymbol{Y}_p$	Output set, horizon p
$\boldsymbol{Y}_{ref}$	Reference input vector set
α_f, α_r	Slip angle of front or rear tire
$\alpha_r^{\cdot}$	Equivalent slip angle of rear tire
γ	Articulated angle
ε	Relaxation factor
θ	Heading angle
ρ	Weight factor
ω_f, ω_r	Yaw velocity of front part or rear part
Δ	Delta function
Δs, Δs_{ref}	Distance and reference distance between the loader and the target point
$\Delta \boldsymbol{u}_k^*$	control input increment vector, $\Delta \boldsymbol{u}_k^* \in \mathbb{R}^m$
$\Delta \boldsymbol{U}$	Control increment set
$\Delta \boldsymbol{U}_k^*$	Control input increment set

There are many challenges in the process of wheel loader automation, among which accurate trajectory planning and tracking are quite urgent [8]. The V-shaped work path of the wheel loader planned in previous studies [9–11] consists of a symmetrical "clothoid" path (i.e., a curvature that changes linearly with its length) and a straight line segment. However, during actual operation, the driving path of the loader will change owing to ongoing operational requirements; thus, symmetry cannot always be guaranteed. Therefore, the current path planning algorithm cannot easily meet the actual requirements of on-site operation. Alshaer et al. [12] studied path planning and tracking problems related to an autonomous wheel loader using these V-shaped working conditions. The shortest path algorithm proposed by Reeds and Shepp was improved and extended, and a proportional–integral–derivative (PID) controller was used to track the planning path. Nayl et al. [13] built an online motion planning controller for articulated vehicles based on a model predictive control (MPC) system and analyzed the influence of parameters such as the velocity of the vehicle, maximum allowable change in the articulated steering angle, safe distance from obstacles, and total number of obstacles in the operating arena on the online motion planning algorithm. Another study considered the effect of side angles on the nonlinear kinematic model of a non-holonomic articulated vehicle and developed a switching control scheme based on multi-model predictive controller [14]. A sliding mode controller based on an articulated vehicle nonlinear kinematics model has been constructed and was found to improve the tracking control of the planned trajectory [15]. However, these studies did not consider the tracking control of the vehicle in a non-uniform driving state and did not deeply analyze the influence of path curvature on the time-varying state of the tracking process. Choi and Huhtala [16] used the search-based A* algorithm to optimize the global path of an articulated vehicle under a high-constraint environment to realize obstacle avoidance behavior, but did not analyze the trajectory tracking of the vehicle under variable speed. In another study [17], a driver model for a wheel loader operating under V-shaped working conditions was constructed based on input from the throttle, brake, and steering using model predictive control (MPC). The results indicated that the simulation data and measured speed trends were basically the same. However, the displacement and heading errors, which represent the tracking effect in the simulation process, were not elaborated upon in detail, and the influence of the path curvature on the tracking of the loader trajectory was not considered.

Given these issues, the following factors remain to be fully considered when planning the trajectory and tracking of an autonomous wheel loader in a V-shaped operation mode. First, because a running wheel loader frequently accelerates and decelerates, the vehicle is often in a non-uniform driving state. Second, as the path curvature of a wheel loader is not constant, in a time-varying state this will have a certain impact on the tracking effect.

Therefore, a non-uniform motion trajectory under forward and reverse driving conditions is proposed in this study by combining the actual operating characteristics and the minimum stable turning radius of a wheel loader based on the optimal rapidly exploring random tree (RRT*) and continuous curvature steer (CC Steer) algorithms. In this manner, a new adaptive model predictive control (AMPC) system is constructed based on the kinematics of the loader and the dynamic deviation model. The AMPC system uses the path curvature in the time-varying state as the disturbance input, the acceleration and articulated angle velocity as control inputs, and the vehicle speed and articulation angle as control outputs. The trajectory tracking of an autonomous wheel loader is then conducted to implement the planned non-uniform trajectory using the constructed AMPC trajectory tracking system. Based on the outcomes of this implementation, the influences of path

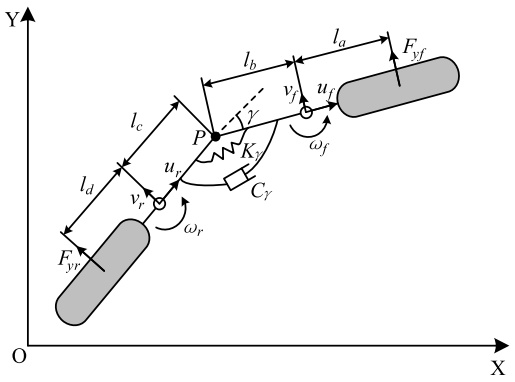


Fig. 1. The wheel loader model.

curvature on the trajectory tracking performance and robustness of the proposed AMPC system are verified for different curvature amplitudes.

Though acceleration and deceleration processes as well as path curvature disturbances are unavoidable during actual vehicle travel, this problem has often been overlooked in previous trajectory planning and tracking control studies. In this study, the entire vehicle trajectory of an autonomous wheel loader under V-mode operation was planned according to its acceleration and deceleration processes to accommodate path curvature disturbances, and the desired trajectory was effectively tracked, addressing such deficiencies in existing research and improving the accuracy of trajectory tracking. This method not only forms a basis for realizing autonomous wheel loader driving functions, but also provides a key technology for improving the active safety of the vehicle. Notably, the method proposed in this paper is not only applicable to wheel loaders, but can also be used as a basis for trajectory planning and tracking research using other types of autonomous vehicles.

The overall structure of this article is as follows. In Section 2, the dynamic and driving dynamic deviation models for the wheel loader are established. Section 3 details the planning of the non-uniform velocity trajectory of the wheel loader, and the AMPC trajectory tracking system is constructed under conditions in which the path curvature is disturbed. Section 4 presents an analysis of the simulation used to verify the influence of the path curvature on the performance and robustness of the AMPC trajectory tracking system when accounting for different curvature amplitudes. Finally, the conclusions are presented in Section 5.

2. Dynamic and dynamic deviation models

2.1. Dynamic model

The dynamic model presented in this paper was primarily used as a predictive model in the model controller. This model must therefore describe the dynamic process as accurately and simply as possible in order to reduce the computational burden of the control algorithm. Accordingly, the following assumptions were made in the dynamic analysis:

(1) Considering the large mass of the wheel loader, changes in longitudinal speed due to the longitudinal components of the tire lateral forces were ignored;

(2) The wheel loader was assumed to be driving on a flat road, so the vertical movement of the vehicle was ignored;

(3) The suspension system and vehicle were assumed to be rigid, ignoring any suspension motion and its influence on the coupling relationship;

(4) The longitudinal and transverse aerodynamics were ignored.

The lateral dynamics model of the wheel loader is depicted in Fig. 1, in which the coordinate system OXY is an inertial coordinate system fixed on the ground. The front and rear bodies are connected by an articulation point P at an articulated angle. The variables shown in Fig. 1 and used in the related equations are defined in the nomenclature. Note that this model only considers the longitudinal motion of the vehicle and the yaw motion of the front and rear bodies.

A linear function is used to approximate the lateral force on the front and rear tires as follows [18]:

$$\begin{cases} F_{yf} = -C_{\alpha f}\alpha_f \\ F_{yr} = -C_{\alpha r}\alpha_r^{:} \end{cases} \tag{1}$$

The cornering stiffness coefficients of the front and rear tires can be described as follows:

$$\begin{cases} C_{\alpha f} = C_{\alpha 1}N_1 \\ C_{\alpha r} = C_{\alpha 2}N_2 \\ N_1 = \dfrac{m_1 g\left(l_b + l_c + l_d\right) + m_2 g l_d}{l_a + l_b + l_c + l_d} \\ N_2 = \left(m_1 + m_2\right) g - N_1 \end{cases} \tag{2}$$

The slip angles of the front and rear tires can be described as:

$$\begin{cases} \alpha_f = -\dfrac{v_f + l_a\omega_f}{u_f} \\ \alpha_r^{:} = \alpha_r - \gamma \\ \alpha_r = -\dfrac{v_f - l_b\omega_f - \left(l_c + l_d\right)\omega_r}{u_f} \end{cases} \tag{3}$$

Finally, the relationship between the heading angular velocity and yaw angular velocity of the front body is as follows:

$$\dot{\theta} = \omega_f \tag{4}$$

In the process of calculation, a small angle hypothesis is proposed, which satisfies the following approximation conditions: $\cos(\gamma)\approx 1$, $\sin(\gamma)\approx\gamma$.

Combining the definitions in Eqs. (1)–(4) with the model presented in Fig. 1, the lateral dynamics equation of the wheel loader is:

$$\boldsymbol{A}_m\dot{\boldsymbol{X}}_m = \boldsymbol{B}_m\boldsymbol{X}_m + \boldsymbol{C}_m U_m \tag{5}$$

where $\boldsymbol{X}_m = \begin{bmatrix} v_f & \omega_f & \omega_r & \theta \end{bmatrix}^T$,

$$\boldsymbol{A}_m = \begin{bmatrix} m_0 & -m_2 l_b & -m_2 l_c & 0 \\ m_1 l_b & I_1 & 0 & C_\gamma \\ -m_2 l_c & m_2 l_b l_c & I_2 + m_2 l_c^2 & -C_\gamma \\ 0 & 0 & 0 & 1 \end{bmatrix},$$

$$\boldsymbol{B}_m = \begin{bmatrix} -\dfrac{C_\alpha}{u_f} & -m_0 u_f - \dfrac{\left(l_a C_{\alpha f} - l_b C_{\alpha r}\right)}{u_f} & \dfrac{L_2 C_{\alpha r}}{u_f} & 0 \\ -\dfrac{L_1 C_{\alpha f}}{u_f} & -m_1 l_b u_f - \dfrac{L_1 C_{\alpha f} l_a}{u_f} & C_\gamma & 0 \\ \dfrac{L_2 C_{\alpha r}}{u_f} & m_2 l_c u_f - \dfrac{L_2 C_{\alpha r} l_b}{u_f} & -\frac{L_2^2 C_{\alpha r}}{u_f} - C_\gamma & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix},$$

$$\boldsymbol{C}_m = \begin{bmatrix} C_{\alpha r} \\ K_\gamma \\ -L_2 C_{\alpha r} - K_\gamma \\ 0 \end{bmatrix}, U_m = \gamma,$$

and, $m_0 = m_1 + m_2$, $L_1 = l_a + l_b$, $L_2 = l_c + l_d$, and $C_\alpha = C_{\alpha f} + C_{\alpha r}$.

The relationship between the actual longitudinal acceleration a_x and the longitudinal velocity u_f, lateral velocity v_f and yaw

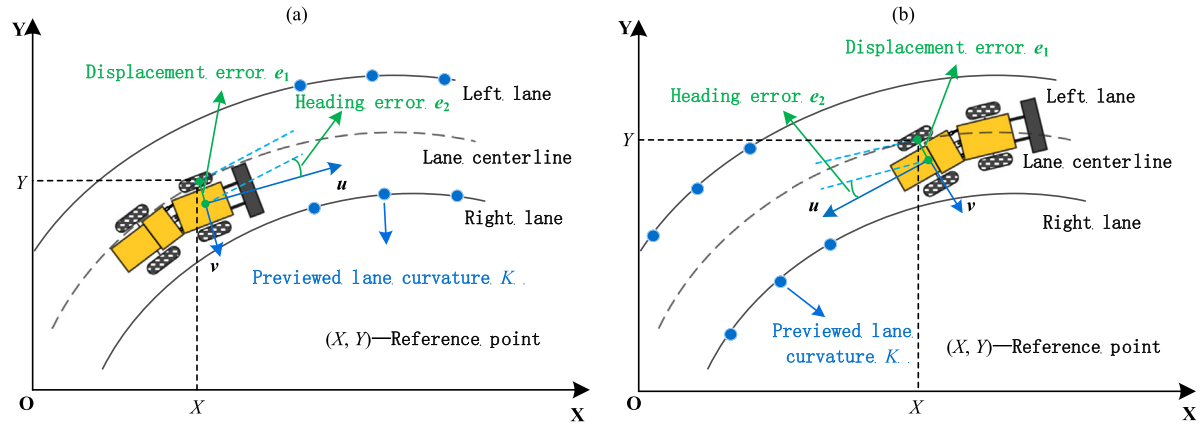


Fig. 2. Wheel loader path tracking schematic diagram under (a) forward motion and (b) reverse motion.

angular velocity ω_f is as follows:

$$\dot{u}_f = \omega_f v_f + a_x \tag{6}$$

Considering the delay characteristics of the entire vehicle power transmission system as the first-order inertia link, the relationship between the actual longitudinal acceleration a_x and the expected longitudinal acceleration a is:

$$\ddot{u}_f = -\frac{1}{\tau}a_x + \frac{1}{\tau}a \tag{7}$$

By combining Eqs. (6) and (7), the longitudinal dynamic equation of loader is obtained as follows:

$$\begin{bmatrix}\dot{u}_f \\ \ddot{u}_f\end{bmatrix} = \begin{bmatrix}\omega_f & 1 \\ 0 & -1/\tau\end{bmatrix}\begin{bmatrix}v_f \\ a_x\end{bmatrix} + \begin{bmatrix}0 \\ 1/\tau\end{bmatrix}a \tag{8}$$

Finally, the conversion relationship between the body coordinate system and inertial coordinate system is as follows:

$$\begin{cases}\dot{X} = u_f \cos\theta - v_f \sin\theta \\ \dot{Y} = u_f \sin\theta + v_f \cos\theta\end{cases} \tag{9}$$

2.2. Dynamic deviation model

In the loader path tracking diagram shown in Fig. 2. When the wheel loader is in forward motion, the reference point is the point on the path that is closest to the center of the vehicle's front axle. When the wheel loader is in reverse motion, the reference point is the point on the path that is closest to the center of the vehicle's rear axle.

The definitions for the error analysis of the dynamic deviation model are as follows [17,19,20]:

(1) The displacement error e_1 is the lateral difference in the position of the registration point of the loader and the reference point on the planning path, given as follows:

$$\dot{e}_1 = v + ue_2 \tag{10}$$

(2) The heading error e_2 is the difference between the heading angle of the positioning point of the wheel loader and the heading angle of the planned path reference point; that is, it is the angle between the direction of the velocity of the loader positioning point and the tangential direction of the planning path reference point, as given by:

$$\dot{e}_2 = \dot{\theta} - uK \tag{11}$$

The dynamic deviation model was constructed to calculate the deviation in trajectory tracking by comparing the future path curvature data with the current vehicle state data. In Section 3, this deviation is used as the basis for feedback control to realize trajectory tracking for the autonomous wheel loader.

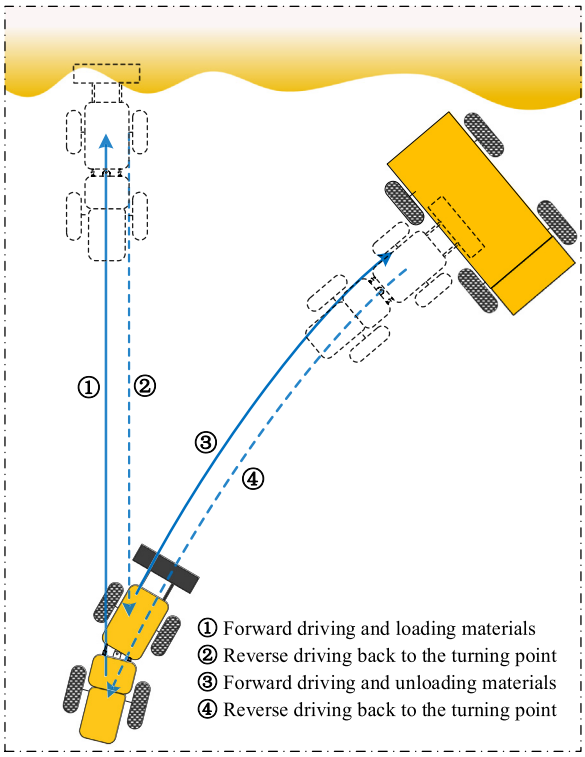


Fig. 3. Schematic diagram of the V- shaped operation mode.

3. Trajectory planning and tracking control

When a wheel loader performs short-distance shovel loading, its operation is mainly divided into I, V, L, and T-shaped modes according to its loading and running route. Among these modes, the V-shaped operation mode is highly efficient and provides a short working cycle time. Thus, it is the most common mode used in shovel loading operations [1,21]. The V-shaped wheel loader operation mode is shown in Fig. 3. The work process is usually divided into four stages: driving forward and loading materials, driving in reverse back to the turning point, driving forward and unloading the materials, and driving in reverse back to the turning point.

The purpose of wheel loader trajectory planning and tracking control when in the V-shaped operation mode is to generate a motion trajectory that conforms to the actual driving conditions and keeps the vehicle as close to the planned trajectory as

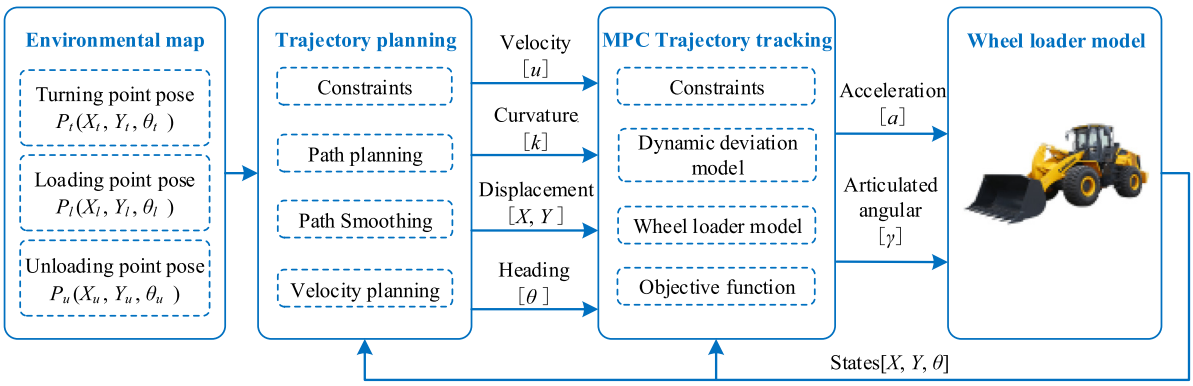


Fig. 4. Overall architecture of trajectory planning and tracking control.

possible. The coordinates for the points at which the loader turns, loads material, and unloads material can be determined by manual manipulation. The overall architecture of the proposed trajectory planning and tracking control is presented in Fig. 4.

For the V-shaped operation mode, the trajectory planning and tracking control of the wheel loader must satisfy the following constraints:

(1) The radius of all planned paths should be greater than the minimum turning radius of the wheel loader, given the constraints of the loader actuator;

(2) The planning path and its curvature should be continuous to provide steering stability of the loader;

(3) When the loader is at the loading and unloading points, the articulation angle should be as close to zero as possible to avoid rollover of the vehicle;

(4) Considering actual operational situation, the maximum planned velocity of the loader should not exceed 3m/s and rapid acceleration and deceleration should be avoided.

3.1. Planning the trajectory

Given the mathematical model of the vehicle, the distribution of environmental obstacles, and the initial and target states, the task of trajectory planning is to provide the trajectory tracking controller with a series of control inputs that drive the vehicle from its initial state to the target state.

In recent years, representative algorithms for unmanned vehicle motion planning include the search-based state lattice algorithm [22,23] and rapidly exploring random tree (RRT) [24] based on random sampling. The state lattice algorithm is analytically complete and even analytically optimal, but it must process a large number of grids for high-dimensional state space, resulting in low efficiency. The RRT method has probabilistic completeness, does not require precise connections between states, and has been successfully applied to unmanned vehicles in the 2007 DARPA challenge [25]. The optimal rapidly exploring random tree algorithm (RRT*) [26] is an improvement of the standard RRT based on the theory of random geometric graphs that compensates for the insufficiency of the standard RRT algorithm in ensuring that the optimal solution is obtained.

Trajectory generation is a basic process in all types of unmanned vehicle motion planning algorithms. Its purpose is to construct a trajectory in the state space to connect two given states and is generally divided into two steps: path generation and velocity generation. For path generation, the continuous bounded path generated by the CC Steer algorithm [27] is not only suitable for wheel loaders in reverse, but also for vehicle motion planning at low speeds. For velocity generation, the literature [28] studied the problem of velocity generation through a given path in a given time, but this method is prone to the problem of large rate of acceleration change. Xu et al. [29], however, directly used the cubic polynomial of the velocity with respect to the path length to generate the velocity, not only avoiding acceleration rate issues, but making the desired velocity easier to realize.

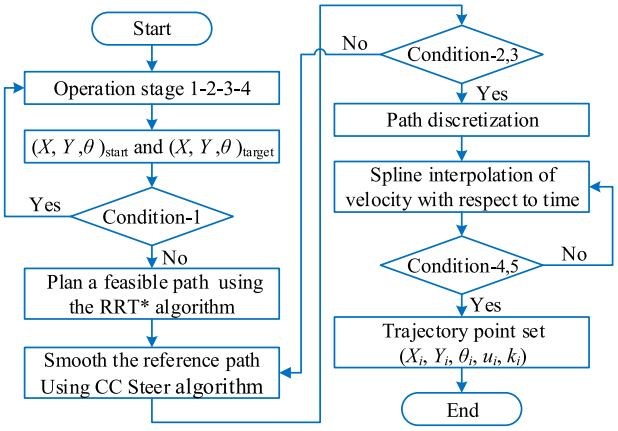
Table 1
Wheel loader initial and target positions according to operation stage.

Operation stage	Initial pose $(X,Y,\theta)_{start}$	Target pose $(X,Y,\theta)_{target}$
1	$P_t(X_t, Y_t, \theta_t)$	$P_l(X_l, Y_l, \theta_l)$
2	$P_l(X_l, Y_l, \theta_l)$	$P_t(X_t, Y_t, \theta_t)$
3	$P_t(X_t, Y_t, \theta_t)$	$P_u(X_u, Y_u, \theta_u)$
4	$P_u(X_u, Y_u, \theta_u)$	$P_t(X_t, Y_t, \theta_t)$

In the trajectory planning algorithm, the initial and target positions of the wheel loader, $(X, Y, \theta)_{start}$ and $(X, Y, \theta)_{target}$, respectively, change with each operation stage as defined in Table 1.

A flow chart of the wheel loader motion trajectory plan algorithm for the V-shaped working mode and its constraints are provided in Fig. 5. According to the operation stage of the wheel loader, the initial position $(X, Y, \theta)_{start}$ and the target position $(X, Y, \theta)_{target}$ of the inertial coordinate system are defined, and the reference distance between the vehicle and the target point is used as Condition-1 to determine whether or not the next stage of the operation should be initiated. Using the RRT* algorithm [26] to plan an optimal collision-free path in the current environment will lead to sudden changes in the articulated angle when the wheel loader is driven under actual operation conditions because the path is connected by several segments, and the curvature at the junctions of these segments is discontinuous. Therefore, to ensure wheel loader steering stability during trajectory tracking, the CC Steer algorithm [27] is applied to smooth the existing path using Condition-2 (the minimum turning radius of the loader) and Condition-3 (the path curvature) as constraints. Finally, the cubic polynomial [29] is constructed using the velocity, acceleration, and length of the discretized path to determine the velocity plan of the global path.

Algorithm 1 is the path planning algorithm, in which the environment data model is first established using the environment() function, and the starting and target points are determined and added to the tree. The algorithm then judges whether or not the tree node has reached the target point through the while loop; if it has not, it begins to expand the point. The function Random_state() is used to generate random sampling points, search for the point nearest to the random point in the tree node, and to extend the new node and add it to the random tree. Finally, the path generated is obtained using the function $(X, Y, \theta)_{rand}$.



Div	Contents	Condition
Condition-1	Distance	$\Delta s \le \Delta s_{ref}$
Condition-2	Turning radius	$r \ge r_{\min}$
Condition-3	Road curvature	K(t) continuous
Condition-4	Velocity	$u \le u_{\max}$
Condition-5	Acceleration	$a \le a_{\max}$

Δs : Distance between the loader and the target point
Δs_{ref} : Reference distance between the loader and the target point
r : Radius of planning path
$r_{\min}$: Minimum turning radius of the loader
K(t) : The curvature of the planning path
u : Longitudinal speed of wheel loader
a : Wheel loader acceleration

Fig. 5. Flow chart and constraints of loader trajectory planning algorithm.

Algorithm 1: Body of Path Planning Algorithm

```
environment();
Tree.init ((X, Y, θ)start , (X, Y, θ)target)
while distance (Tree( ))> Δs
(X, Y, θ)rand ← Random_state();
Extend (Tree, (X, Y, θ)rand);
end
Path = getPath ( );
```

The expressions for the steering curvature and rate of change of curvature of the wheel loader under steady-state steering are as follows:

$$k = \frac{\dot{\theta}}{u_f}, \dot{\gamma} = \dot{k} = \frac{\ddot{\theta}}{u_f} \tag{12}$$

The path curvature and its rate of change can then be obtained using Eq. (12) as follows:

$$|k| \le k_{\max}, \left|\dot{k}\right| \le \dot{k}_{\max} \tag{13}$$

As shown in Fig. 6, under the assumption that the initial node of the random tree is $q_s = (0, 0, 0, 0)$ and that the target point is at the upper left, the wheel loader would need to make a left turn to reach the target point. The node expansion method used, as given in Fraichard and Scheuer [27], is:
(1) Follow a clothoid arc of length $k_{\max}/\dot{k}_{\max}$ and sharpness $\dot{k}_{\max}$ until reaching q_i, defined as:

$$q_i = \begin{cases} X_i = \sqrt{\pi/\dot{k}_{\max}} C_f\left(\sqrt{k_{\max}^2/\pi\dot{k}_{\max}}\right) \\ Y_i = \sqrt{\pi/\dot{k}_{\max}} S_f\left(\sqrt{k_{\max}^2/\pi\dot{k}_{\max}}\right) \\ \theta_i = k_{\max}^2/2\dot{k}_{\max} \\ k_i = k_{\max} \end{cases} \tag{14}$$

(2) Follow a clothoid arc of radius $k_{\max}^{-1}$ until reaching $q_j = (X_j, Y_j, \theta_j, k_{\max})$;
(3) Follow a clothoid arc of sharpness $-\dot{k}_{\max}$ until reaching the goal configuration $q_g = (X_g, Y_g, \theta_g)$.

Algorithm 2 represents the process of extending a new node. To effectively apply the planned path to the movement of the loader, that is, to track the path of the loader, the path curvature and its rate of change constraints are introduced during planning at this stage. In this part of the algorithm, $(X, Y, \theta)_{nearest}$, which is the point nearest to the random point, is taken from the search tree after the random points $(X, Y, \theta)_{rand}$ in space are obtained. The nearest point expands the new node $(X, Y, \theta)_{new}$ along the direction of this random point. The maximum curvature of the new node is then detected to determine whether or not it should be added to the tree.

Algorithm 2: Extend (Tree,$(X, Y, \theta)_{rand}$)

```
(X, Y, θ)nearest ← Nearest_State (Tree, (X, Y, θ)rand, P);
(X, Y, θ)new ← New_State ((X, Y, θ)nearest, StepSize);
if (k_maxCheckReturn ((X, Y, θ)new, k))
    if(dk_maxCheckReturn ((X, Y, θ)new, dkList))
    newnode Tree++;
    Addnewnode (Tree, (X, Y, θ)new);
        if ((X, Y, θ)new==(X, Y, θ)target);
            Treepath ← FinalPath (Tree);
        end
    end
end
```

After the path planning is completed, the vehicle velocity is discretized and fitted to the basis of the length of the previous spanning tree node using a polynomial equation. The multi-project equations is defined as follows [29]

$$u(s) = \rho_0 + \rho_1 s + \rho_2 s^2 + \rho_3 s^3 \tag{15}$$

The velocity state is defined as $E = (s, u, a)$, where s is the arc length, u is the velocity, and a is the acceleration. For each path, the start velocity state is defined as $E_{init} = (s_0, u_0, a_0)$ from the start point of the path and the end velocity state $E_{goal} = (s_1, u_1, a_1)$ is defined from the end point of the path. For all vertices, u_0 and u_1 are taken from the corresponding discretized vertices and a_0 and a_1 are set to 0. The maximum and minimum velocity are denoted $u_{\max}$ and $u_{\min}$, respectively, and the number of discretized velocities are denoted N_{speed}; the velocities at each discretization can be determined as:

$$u_i = u_{\min} + \frac{u_{\max} - u_{\min}}{N_{speed} - 1} i,\ i = 0, 1, \ldots, n_u - 1 \tag{16}$$

By setting $s_0 = 0$, the following parameters can be obtained:

$$\begin{cases} \rho_0 = u_0 \\ \rho_1 = a_0 \\ \rho_2 = -\dfrac{2a_0}{s_1} - \dfrac{a_1}{s_1} - \dfrac{3u_0}{s_1^2} + \dfrac{3u_1}{s_1^2} \\ \rho_3 = \dfrac{a_0}{s_1^2} + \dfrac{a_1}{s_1^2} + \dfrac{2u_0}{s_1^3} - \dfrac{2u_1}{s_1^3} \end{cases} \tag{17}$$

3.2. Tracking and controlling the trajectory

The purpose of trajectory tracking control is to drive the loader along the trajectory established by the planning algorithm using

$$\dot{\boldsymbol{x}} = \boldsymbol{A}_c\boldsymbol{x} + \boldsymbol{B}_{cu}[\gamma, a]^T + \boldsymbol{B}_{cd}K \tag{18a}$$

$$\boldsymbol{y} = \boldsymbol{C}_c\boldsymbol{x} \tag{18b}$$

where

$$A_c = \begin{bmatrix} 0 & 0 & 0 & -\sin\theta & 0 & 0 & \cos\theta & 0 & 0 & 0 \\ 0 & 0 & 0 & \cos\theta & 0 & 0 & \sin\theta & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & a_{44} & a_{45} & a_{46} & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & a_{54} & a_{55} & a_{56} & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & a_{64} & a_{65} & a_{66} & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & \omega_f & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & -\frac{1}{\tau} & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & e_2 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \quad \boldsymbol{B}_{cu} = \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ b_{41} & 0 \\ b_{51} & 0 \\ b_{61} & 0 \\ 0 & 0 \\ 0 & \frac{1}{\tau} \\ 0 & 0 \\ 0 & 0 \end{bmatrix}, \boldsymbol{B}_{cd} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ -u_f \end{bmatrix}, \boldsymbol{C}_c = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}^T$$

,

$$a_{44} = -\frac{m_2e^2C_\alpha I_1 - m_2l_cC_{\alpha r}L_2I_1 + C_\alpha I_1I_2 + m_2l_bC_{\alpha f}I_2L_1}{u_f\left(m_1m_2l_b^2I_2 - m_2^2l_c^2I_1 + m_0m_2l_c^2I_1 + m_0I_1I_2\right)}$$

$$a_{45} = -\frac{m_1m_2l_b^2l_2u_f^2 - m_2l_bl_c^2C_{\alpha r}I_1 + m_2l_bl_cC_{\alpha r}L_2I_1 + m_2l_bC_\gamma I_2u_f + m_2l_al_bC_{\alpha f}L_1I_2 - l_bC_{\alpha r}I_1I_2 - m_2^2l_c^2I_1u_f^2}{u_f\left(m_1m_2l_b^2I_2 - m_2^2l_c^2I_1 + m_0m_2l_c^2I_1 + m_0I_1I_2\right)}$$
$$- \frac{m_0m_2l_c^2I_1u_f^2 + m_2l_al_c^2C_{\alpha f}I_1 - m_2l_cC_\gamma I_1u_f + m_0I_1I_2u_f^2 + l_aC_{\alpha f}I_1I_2}{u_f\left(m_1m_2l_b^2I_2 - m_2^2l_c^2I_1 + m_0m_2l_c^2I_1 + m_0I_1I_2\right)}$$

$$a_{46} = \frac{-m_2l_cC_{\alpha r}L_2^2I_1 + m_2l_c^2C_{\alpha r}L_2I_1 + C_{\alpha r}L_2I_1I_2 - m_2l_cC_\gamma I_1u_f + m_2l_bC_\gamma I_2u_f}{u_f\left(m_1m_2l_b^2I_2 - m_2^2l_c^2I_1 + m_0m_2l_c^2I_1 + m_0I_1I_2\right)}$$

$$a_{54} = \frac{m_2^2l_c^2C_{\alpha f}L_1 - m_0C_{\alpha f}L_1I_2 + m_1l_bC_\alpha I_2 - m_0m_2l_c^2C_{\alpha f}L_1 + m_1m_2l_bl_c^2C_\alpha - m_1m_2l_bl_cC_{\alpha r}L_2}{u_f\left(m_1m_2l_b^2I_2 - m_2^2l_c^2I_1 + m_0m_2l_c^2I_1 + m_0I_1I_2\right)}$$

$$a_{55} = -\frac{m_1l_b^2C_{\alpha r}I_2 - m_2^2l_c^2C_\gamma u_f + m_0C_\gamma I_2u_f + m_1m_2l_b^2l_c^2C_{\alpha r} + m_0l_aC_{\alpha f}L_1I_2 - m_1l_al_bC_{\alpha f}I_2 + m_0m_2l_c^2C_\gamma u_f}{u_f\left(m_1m_2l_b^2I_2 - m_2^2l_c^2I_1 + m_0m_2l_c^2I_1 + m_0I_1I_2\right)}$$
$$- \frac{-m_2^2l_al_c^2C_{\alpha f}L_1 - m_1m_2l_al_bl_c^2C_{\alpha f} + m_1m_2l_bl_cC_\gamma u_f + m_0m_2l_al_c^2C_{\alpha f}L_1 - m_1m_2l_b^2l_cC_{\alpha r}L_2}{u_f\left(m_1m_2l_b^2I_2 - m_2^2l_c^2I_1 + m_0m_2l_c^2I_1 + m_0I_1I_2\right)}$$

$$a_{56} = \frac{m_1m_2l_bl_cC_{\alpha r}L_2^2 - m_1m_2l_bl_c^2C_{\alpha r}L_2 - m_1l_bC_{\alpha r}L_2I_2 - m_2^2l_c^2C_\gamma u_f + m_0m_2l_c^2C_\gamma u_f + m_1m_2l_bl_cC_\gamma u_f + m_0C_\gamma I_2u_f}{u_f\left(m_1m_2l_b^2I_2 - m_2^2l_c^2I_1 + m_0m_2l_c^2I_1 + m_0I_1I_2\right)}$$

$$a_{64} = \frac{m_0C_{\alpha r}L_2I_1 - m_2l_cC_\alpha I_1 - m_2^2l_bl_cC_{\alpha f}L_1 + m_1m_2l_b^2C_{\alpha r}L_2 - m_1m_2l_b^2l_cC_\alpha + m_0m_2l_bl_cC_{\alpha f}L_1}{u_f\left(m_1m_2l_b^2I_2 - m_2^2l_c^2I_1 + m_0m_2l_c^2I_1 + m_0I_1I_2\right)}$$

$$a_{65} = \frac{m_0C_\gamma I_1u_f - m_0l_bC_{\alpha r}L_2I_1 - m_2l_al_cC_{\alpha f}I_1 + m_2l_bl_cC_{\alpha r}I_1 - m_1m_2l_b^3C_{\alpha r}L_2 + m_1m_2l_b^3l_cC_{\alpha r} - m_2^2l_bl_cC_\gamma u_f}{u_f\left(m_1m_2l_b^2I_2 - m_2^2l_c^2I_1 + m_0m_2l_c^2I_1 + m_0I_1I_2\right)}$$
$$+ \frac{m_1m_2l_b^2C_\gamma u_f - m_1m_2l_al_b^2l_cC_{\alpha f} + m_0m_2l_bl_cC_\gamma u_f - m_2^2l_al_bl_cC_{\alpha f}L_1 + m_0m_2l_al_bl_cC_{\alpha f}L_1}{u_f\left(m_1m_2l_b^2I_2 - m_2^2l_c^2I_1 + m_0m_2l_c^2I_1 + m_0I_1I_2\right)}$$

$$a_{66} = -\frac{m_1m_2l_b^2C_{\alpha r}L_2^2 + m_0C_{\alpha r}L_2^2I_1 - m_1m_2l_b^2l_cL_2C_{\alpha r} - m_2l_cL_2C_{\alpha r}I_1 + m_1m_2l_b^2C_\gamma u_f - m_2^2l_bl_cC_\gamma u_f + m_0m_2l_bl_cC_\gamma u_f + m_0C_\gamma I_1u_f}{u_f\left(m_1m_2l_b^2I_2 - m_2^2l_c^2I_1 + m_0m_2l_c^2I_1 + m_0I_1I_2\right)}$$

$$b_{41} = \frac{C_{\alpha r}I_1I_2 + m_2l_c^2C_{\alpha r}I_1 + m_2l_bK_\gamma I_2 - m_2l_cK_\gamma I_1 - m_2l_cC_{\alpha r}L_2I_1}{m_1m_2l_b^2I_2 - m_2^2l_c^2I_1 + m_0m_2l_c^2I_1 + m_0I_1I_2}$$

$$b_{51} = \frac{m_0K_\gamma I_2 - m_2^2l_c^2K_\gamma + m_0m_2l_c^2K_\gamma - m_1l_bC_{\alpha r}I_2 + m_1m_2l_bl_cK_\gamma - m_1m_2l_bl_c^2C_{\alpha r} + m_1m_2l_bl_cL_2C_{\alpha r}}{m_1m_2l_b^2I_2 - m_2^2l_c^2I_1 + m_0m_2l_c^2I_1 + m_0I_1I_2}$$

$$b_{61} = -\frac{m_0K_\gamma I_1 - m_2^2l_bl_cK_\gamma + m_1m_2l_b^2K_\gamma + m_0C_{\alpha r}I_1I_2 - m_2l_cC_{\alpha r}I_1 + m_0m_2l_bl_cK_\gamma + m_1m_2l_b^2C_{\alpha r}L_2 - m_1m_2l_b^2l_cC_{\alpha r}}{m_1m_2l_b^2I_2 - m_2^2l_c^2I_1 + m_0m_2l_c^2I_1 + m_0I_1I_2}$$

Box I.

its lateral and longitudinal controls [30]. At present, the most widely used control algorithms include PID [31], fuzzy logic [32], and robust control [33]. These algorithms are highly dependent on their parameters and the operational environment and thus cannot easily meet the kinematic and executive mechanism constraints of moving vehicles. Because an MPC system has a strong

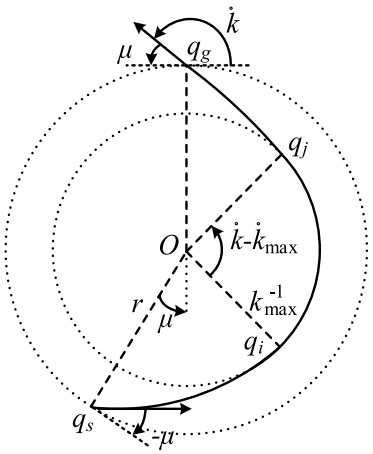


Fig. 6. Node expansion diagram.

ability to predict future trajectories and address multi-objective constraints [34], this algorithm was used to track the trajectory control of the loader.

(1) Constructing the Prediction Model

To design the AMPC trajectory tracking system, the wheel loader state space equation used for prediction was first derived. The state space vector consists of the midpoint coordinates (X, Y) of the front axle of the wheel loader, the heading angle θ, the yaw angular velocities ω_f and ω_r at the center of mass of the front and rear bodies, respectively, the longitudinal speed u_f, the rate of change of the longitudinal speed $\dot{u}_f$, the displacement error e_1, and the heading error e_2. The control input is the articulated angle γ and the expected longitudinal acceleration a. The external disturbance is the product of the longitudinal speed u_f and the driving path curvature K. By controlling the articulation angle γ and the longitudinal speed u_f, the displacement error e_1 and the heading error e_2 tend to zero, thereby enabling the tracking of the planned trajectory. Using this definition, the state–space vector was obtained as:

$$\boldsymbol{x} = \left[X, Y, \theta, v_f, \omega_f, \omega_r, u_f, \dot{u}_f, e_1, e_2\right]^T$$

The state equations can then be written based on Eqs. (5), (8), (9), (10), and (11), derived in Section 2, as equations given in Box I.

Among which e=e_1+e_2, $m_0 = m_1$+m_2, $L_1 = l_a$+l_b, $L_2 = l_c$+l_d, and $C_\alpha = C_{\alpha f}$+$C_{\alpha r}$.

Note that Eq. (18) is a nonlinear system and therefore cannot be used directly in the predictive control of linear time-varying models. This equation must accordingly be linearized.

A nonlinear system can be approximated as a linear time-varying system using either approximate linearization or exact linearization methods. Approximate linearization methods are simple and practical, but difficult to use when the control accuracy requirements are high. Exact linearization methods are generally not universal, and as a result are often analyzed specifically for a single system. Accordingly, approximate linearization methods are generally adopted in model predictive control applications [35].

For the planned reference trajectory, the approximate linearization of Eq. (18) can be described in terms of the trajectory of the reference vehicle. Each point on this trajectory then satisfies the equation of motion of the vehicle, and r represents the reference quantity, then:

$$\dot{\boldsymbol{x}}_r = \boldsymbol{A}_{rc}\boldsymbol{x}_r + \boldsymbol{B}_{ru}[\gamma, a]_r^T + \boldsymbol{B}_{rd}K_r \tag{19}$$

The general form of which is given by:

$$\dot{\boldsymbol{x}}_r = f\left(\boldsymbol{x}_r, \boldsymbol{u}_r, d_r\right) \tag{20}$$

where, $\boldsymbol{x}_r = \left[X, Y, \theta, v_f, \omega_f, \omega_r, u_f, \dot{u}_f, e_1, e_2\right]_r^T$, $\boldsymbol{u}_r = [\gamma, a]_r^T$, $d_r = K_r$.

Then, Eq. (18) is expanded by a Taylor series at the reference trajectory point. Only the first order term is preserved and the higher order terms are neglected, yielding the following equation:

$$\begin{aligned}\dot{\boldsymbol{x}} = f\left(\boldsymbol{x}_r, \boldsymbol{u}_r, d_r\right) &+ \frac{\partial f\left(\boldsymbol{x}, \boldsymbol{u}\right)}{\partial \boldsymbol{x}}\Big|_{\substack{\boldsymbol{x}=\boldsymbol{x}_r \\ \boldsymbol{u}=\boldsymbol{u}_r}}\left(\boldsymbol{x}-\boldsymbol{x}_r\right) \\ &+ \frac{\partial f\left(\boldsymbol{x}, \boldsymbol{u}\right)}{\partial \boldsymbol{u}}\Big|_{\substack{\boldsymbol{x}=\boldsymbol{x}_r \\ \boldsymbol{u}=\boldsymbol{u}_r}}\left(\boldsymbol{u}-\boldsymbol{u}_r\right) + \boldsymbol{B}_{rd}K_r\end{aligned} \tag{21}$$

Subtracting Eq. (21) from Eq. (20), and using Eq. (18b), the continuous time model state equation can be obtained as follows:

$$\dot{\boldsymbol{x}}\left(t\right) = \boldsymbol{A}_l\boldsymbol{x}\left(t\right) + \boldsymbol{B}_{lu}\boldsymbol{u}\left(t\right) + \boldsymbol{B}_{ld}d\left(t\right) \tag{22a}$$

$$\boldsymbol{y}\left(t\right) = \boldsymbol{C}_l\boldsymbol{x}\left(t\right) \tag{22b}$$

Because controlled systems are generally modeled by discrete state equations, Eq. (22) still needs to be discretized. The transformational relationship relating the continuous and discrete time models is the equation [36].

$$\boldsymbol{A} = e^{\boldsymbol{A}_l T_s} \tag{23a}$$

$$\boldsymbol{B}_u = \int_{kT_s}^{(k+1)T_s} e^{\boldsymbol{A}_l[(k+1)T_s-\tau]}d\tau \cdot \boldsymbol{B}_{lu} \tag{23b}$$

$$\boldsymbol{B}_d = \int_{kT_s}^{(k+1)T_s} e^{\boldsymbol{A}_l[(k+1)T_s-\tau]}d\tau \cdot \boldsymbol{B}_{ld} \tag{23c}$$

where $e^{\boldsymbol{A}_l T_s}$ is an exponential matrix and T_s is the sampling time of the system.

The discretized state space model is thus:

$$\boldsymbol{x}(k+1) = \boldsymbol{A}\boldsymbol{x}(k) + \boldsymbol{B}_u\boldsymbol{u}(k) + \boldsymbol{B}_d\boldsymbol{d}(k) \tag{24a}$$

$$\boldsymbol{y}(k) = \boldsymbol{C}\boldsymbol{x}(k) \tag{24b}$$

where $\boldsymbol{A}$, $\boldsymbol{B}_u$, $\boldsymbol{B}_d$, and $\boldsymbol{C}$ are the state matrix of the discrete state space equation, the control input matrix, the external interference matrix, and the controlled output matrix, respectively, and

$$\boldsymbol{y}\left(k\right) = \begin{bmatrix} u_f \\ e_1 \\ e_2 \end{bmatrix}, \boldsymbol{C} = \boldsymbol{C}_c.$$

To introduce integrals and reduce or eliminate dynamic errors, Eqs. (24a) and (24b) are then expressed as incremental models:

$$\Delta\boldsymbol{x}\left(k+1\right) = \boldsymbol{A}\Delta\boldsymbol{x}\left(k\right) + \boldsymbol{B}_u\Delta\boldsymbol{u}\left(k\right) + \boldsymbol{B}_d\Delta\boldsymbol{d}\left(k\right) \tag{25a}$$

$$\boldsymbol{y}\left(k\right) = \boldsymbol{C}\Delta\boldsymbol{x}\left(k\right) + \boldsymbol{y}\left(k-1\right) \tag{25b}$$

where

$$\begin{aligned}\Delta\boldsymbol{x}\left(k\right) &= \boldsymbol{x}\left(k\right) - \boldsymbol{x}\left(k-1\right) \\ \Delta\boldsymbol{u}\left(k\right) &= \boldsymbol{u}\left(k\right) - \boldsymbol{u}\left(k-1\right) \\ \Delta\boldsymbol{d}\left(k\right) &= \boldsymbol{d}\left(k\right) - \boldsymbol{d}\left(k-1\right)\end{aligned}$$

It is assumed that the current time is k, the prediction horizon is p, the control horizon is c, that $c < p$, and that $\Delta\boldsymbol{x}(k)$ can be used as a starting point for predicting the future dynamics of the system. Then, the states at times k + 1 to k + p are predicted by

Eq. (25) as follows:

$$\begin{aligned}
\Delta \boldsymbol{x}(k+1|k) &= \boldsymbol{A}\Delta \boldsymbol{x}(k) + \boldsymbol{B}_u \Delta \boldsymbol{u}(k) + \boldsymbol{B}_d \Delta \boldsymbol{d}(k) \\
\Delta \boldsymbol{x}(k+2|k) &= \boldsymbol{A}^2 \Delta \boldsymbol{x}(k) + \boldsymbol{A}\boldsymbol{B}_u \Delta \boldsymbol{u}(k) \\
&\quad + \boldsymbol{B}_u \Delta \boldsymbol{u}(k+1) + \boldsymbol{A}\boldsymbol{B}_d \Delta \boldsymbol{d}(k) \\
&\vdots \\
\Delta \boldsymbol{x}(k+c|k) &= \boldsymbol{A}^c \Delta \boldsymbol{x}(k) + \boldsymbol{A}^{c-1}\boldsymbol{B}_u \Delta \boldsymbol{u}(k) + \cdots \\
&\quad + \boldsymbol{B}_u \Delta \boldsymbol{u}(k+c-1) + \boldsymbol{A}^{c-1}\boldsymbol{B}_d \Delta \boldsymbol{d}(k) \\
&\vdots \\
\Delta \boldsymbol{x}(k+p|k) &= \boldsymbol{A}^p \Delta \boldsymbol{x}(k) + \boldsymbol{A}^{p-1}\boldsymbol{B}_u \Delta \boldsymbol{u}(k) + \cdots \\
&\quad + \boldsymbol{A}^{p-c}\boldsymbol{B}_u \Delta \boldsymbol{u}(k+c-1) \\
&\quad + \boldsymbol{A}^{p-1}\boldsymbol{B}_d \Delta \boldsymbol{d}(k)
\end{aligned} \tag{26}$$

In Eq. (26), $k+1|k$ represents the prediction for time $k+1$ at time k. The output states at times $k+1$ to $k+p$ can be predicted by the output of Eq. (25b), given as Eq. (27) below:

$$\begin{aligned}
\boldsymbol{y}(k+1|k) &= \boldsymbol{C}\boldsymbol{A}\Delta \boldsymbol{x}(k) + \boldsymbol{C}\boldsymbol{B}_u \Delta \boldsymbol{u}(k) \\
&\quad + \boldsymbol{C}\boldsymbol{B}_d \Delta \boldsymbol{d}(k) + \boldsymbol{y}(k) \\
\boldsymbol{y}(k+2|k) &= \left(\boldsymbol{C}\boldsymbol{A}^2 + \boldsymbol{C}\boldsymbol{A}\right)\Delta \boldsymbol{x}(k) \\
&\quad + \left(\boldsymbol{C}\boldsymbol{A}\boldsymbol{B}_u + \boldsymbol{C}\boldsymbol{B}_u\right)\Delta \boldsymbol{u}(k) \\
&\quad + \boldsymbol{C}\boldsymbol{B}_u \Delta \boldsymbol{u}(k+1) \\
&\quad + \left(\boldsymbol{C}\boldsymbol{A}\boldsymbol{B}_d + \boldsymbol{C}\boldsymbol{B}_d\right)\Delta \boldsymbol{d}(k) + \boldsymbol{y}(k) \\
&\vdots \\
\boldsymbol{y}(k+c|k) &= \sum_{i=1}^{c} \boldsymbol{C}\boldsymbol{A}^i \Delta x(k) + \sum_{i=1}^{c} \boldsymbol{C}\boldsymbol{A}^{i-1}\boldsymbol{B}_u \Delta \boldsymbol{u}(k) \\
&\quad + \sum_{i=1}^{c-1} \boldsymbol{C}\boldsymbol{A}^{i-1}\boldsymbol{B}_u \Delta \boldsymbol{u}(k+1) + \cdots \\
&\quad + \boldsymbol{C}\boldsymbol{B}_u \Delta \boldsymbol{u}(k+c-1) \\
&\quad + \sum_{i=1}^{c} \boldsymbol{C}\boldsymbol{A}^{i-1}\boldsymbol{B}_d \Delta \boldsymbol{d}(k) + \boldsymbol{y}(k) \\
&\vdots \\
\boldsymbol{y}(k+p|k) &= \sum_{i=1}^{p} \boldsymbol{C}\boldsymbol{A}^i \Delta \boldsymbol{x}(k) + \sum_{i=1}^{p} \boldsymbol{C}\boldsymbol{A}^{i-1}\boldsymbol{B}_u \Delta \boldsymbol{u}(k) \\
&\quad + \sum_{i=1}^{p-1} \boldsymbol{C}\boldsymbol{A}^{i-1}\boldsymbol{B}_u \Delta \boldsymbol{u}(k+1) + \cdots \\
&\quad + \sum_{i=1}^{p-c+1} \boldsymbol{C}\boldsymbol{A}^{i-1}\boldsymbol{B}_u \Delta \boldsymbol{u}(k+c-1) \\
&\quad + \sum_{i=1}^{p} \boldsymbol{C}\boldsymbol{A}^{i-1}\boldsymbol{B}_d \Delta \boldsymbol{d}(k) + \boldsymbol{y}(k)
\end{aligned} \tag{27}$$

The p-step prediction output vector and the c-step input vector are defined as follows:

$$\boldsymbol{Y}_p(k+1|k) = \begin{bmatrix} \boldsymbol{y}(k+1|k) \\ \boldsymbol{y}(k+2|k) \\ \vdots \\ \boldsymbol{y}(k+p|k) \end{bmatrix}_{p\times 1} \tag{28a}$$

$$\Delta \boldsymbol{U}(k) = \begin{bmatrix} \Delta \boldsymbol{u}(k) \\ \Delta \boldsymbol{u}(k+1) \\ \vdots \\ \Delta \boldsymbol{u}(k+c-1) \end{bmatrix}_{c\times 1} \tag{28b}$$

Therefore, the future p-step prediction of the system can be calculated using the following prediction equation:

$$\begin{aligned}
\boldsymbol{Y}_p(k+1|k) = \boldsymbol{S}_x \Delta \boldsymbol{x}(k) + \widetilde{\boldsymbol{I}}\boldsymbol{y}(k) + \boldsymbol{S}_d \Delta \boldsymbol{d}(k) \\
+ \boldsymbol{S}_u \Delta \boldsymbol{U}(k)
\end{aligned} \tag{29}$$

where

$$\boldsymbol{S}_x = \begin{bmatrix} \boldsymbol{C}\boldsymbol{A} \\ \sum_{i=1}^{2} \boldsymbol{C}\boldsymbol{A}^i \\ \vdots \\ \sum_{i=1}^{p} \boldsymbol{C}\boldsymbol{A}^i \end{bmatrix}_{p\times 1}, \quad \widetilde{\boldsymbol{I}} = \begin{bmatrix} \boldsymbol{I}_{n\times n} \\ \boldsymbol{I}_{n\times n} \\ \vdots \\ \boldsymbol{I}_{n\times n} \end{bmatrix}_{p\times 1}, \quad \boldsymbol{S}_d = \begin{bmatrix} \boldsymbol{C}\boldsymbol{B}_d \\ \sum_{i=1}^{2} \boldsymbol{C}\boldsymbol{A}^{i-1}\boldsymbol{B}_d \\ \vdots \\ \sum_{i=1}^{p} \boldsymbol{C}\boldsymbol{A}^{i-1}\boldsymbol{B}_d \end{bmatrix}_{p\times 1}$$

$$\boldsymbol{S}_u = \begin{bmatrix} \boldsymbol{C}\boldsymbol{B}_u & 0 & 0 & \cdots & 0 \\ \sum_{i=1}^{2} \boldsymbol{C}\boldsymbol{A}^{i-1}\boldsymbol{B}_u & \boldsymbol{C}\boldsymbol{B}_u & 0 & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \sum_{i=1}^{c} \boldsymbol{C}\boldsymbol{A}^{i-1}\boldsymbol{B}_u & \sum_{i=1}^{c-1} \boldsymbol{C}\boldsymbol{A}^{i-1}\boldsymbol{B}_u & \cdots & \cdots & \boldsymbol{C}\boldsymbol{B}_u \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \sum_{i=1}^{p} \boldsymbol{C}\boldsymbol{A}^{i-1}\boldsymbol{B}_u & \sum_{i=1}^{p-1} \boldsymbol{C}\boldsymbol{A}^{i-1}\boldsymbol{B}_u & \cdots & \cdots & \sum_{i=1}^{p-c+1} \boldsymbol{C}\boldsymbol{A}^{i-1}\boldsymbol{B}_u \end{bmatrix}_{p\times c}$$

Since the model has three outputs, n in matrix $\widetilde{\boldsymbol{I}}$ is 3.

Eq. (29) shows that the state and output quantities in the prediction time domain can be calculated using the current state quantity $\Delta \boldsymbol{x}(k)$ of the system, the control increment $\Delta \mathbf{U}(k)$ in the control time domain, and the external interference quantity $\Delta \boldsymbol{d}(k)$; this acts as the "predictive" function.

(2) Optimizing the solution

Notably, in the current application, the control increment of the system is unknown. Only by setting the appropriate optimization target and solving it can the control sequence be obtained in the control time domain. In reference [35], the model predictive control problem is transformed into a constrained quadratic programming problem. By solving the optimization objective with such constraints, the control sequence in the future can be obtained. A common and proven effective method for preventing a situation in which there is no feasible solution in the control process is to add the softening constrains [37–40] to the optimization objective. In [40], a method for dealing with soft constraints is proposed, and the equivalence between the proposed method and the maximum allowable set is proved, which ensures the recursive feasibility for this type of control strategies. The objective function is as follows:

$$\begin{aligned}
J\left(\boldsymbol{x}(k), \Delta \boldsymbol{U}(k)\right) &= \sum_{i=1}^{p} \left\| \boldsymbol{y}(k+i|k) - \boldsymbol{y}_{ref}(k+i|k) \right\|_{\boldsymbol{Q}}^2 \\
&\quad + \sum_{i=1}^{c-1} \left\| \Delta \boldsymbol{u}(k+i|k) \right\|_{\boldsymbol{R}}^2 + \rho \varepsilon^2
\end{aligned} \tag{30}$$

where $\boldsymbol{Q}$ and $\boldsymbol{R}$ are the weight matrices, ρ is the weight factor, and ε is the relaxation factor.

Substituting Eq. (28) into the optimization objective Eq. (29) and expressing the deviation of the output in the predicted time domain results yields:

$$\boldsymbol{E}(k) = \boldsymbol{Y}_{ref}(k) - \boldsymbol{S}_x \Delta \boldsymbol{x}(k) - \widetilde{\boldsymbol{I}}\boldsymbol{y}(k) - \boldsymbol{S}_d \Delta \boldsymbol{d}(k) \tag{31}$$

where $\boldsymbol{Y}_{ref} = \left[\boldsymbol{y}_{ref}(k+1|k), \ldots, \boldsymbol{y}_{ref}(k+p|k)\right]^T$.

After the corresponding matrix calculation is made, the optimization goal can be adjusted to the following:

$$J(\boldsymbol{x}(k), \Delta\boldsymbol{U}(k)) = \left[\Delta\boldsymbol{U}(k)^T, \varepsilon\right]^T \boldsymbol{H}_k \left[\Delta\boldsymbol{U}(k)^T, \varepsilon\right] + \boldsymbol{G}_k \left[\Delta\boldsymbol{U}(k)^T, \varepsilon\right] + \boldsymbol{P}_k \tag{32}$$

where

$$\boldsymbol{H}_k = \begin{bmatrix} \boldsymbol{S}_u^T \boldsymbol{Q}^T \boldsymbol{Q} \boldsymbol{S}_u + \boldsymbol{R}^T \boldsymbol{R} & 0 \\ \boldsymbol{0} & \rho \end{bmatrix}$$

$$\boldsymbol{G}_k = \left[2\boldsymbol{E}(k)^T \boldsymbol{Q}^T \boldsymbol{Q} \boldsymbol{S}_u^T \quad \boldsymbol{0}\right], \boldsymbol{P}_k = \boldsymbol{E}(k)^T \boldsymbol{Q}^T \boldsymbol{Q} \boldsymbol{E}(k)$$

The expression in Eq. (32) considers the constraints on the control quantity limit, the control increment, and the output quantity in the control process. The control quantities, control increments, and output expressions are defined, respectively, as follows:

$$\boldsymbol{u}_{\min}(k+i) \le \boldsymbol{u}(k+i) \le \boldsymbol{u}_{\max}(k+i) \quad i = 0, 1, \ldots, c-1 \tag{33a}$$

$$\Delta\boldsymbol{u}_{\min}(k+i) \le \Delta\boldsymbol{u}(k+i) \le \Delta\boldsymbol{u}_{\max}(k+i) \quad i = 0, 1, \ldots, c-1 \tag{33b}$$

$$\boldsymbol{y}_{\min}(k+i) \le \boldsymbol{y}(k+i) \le \boldsymbol{y}_{\max}(k+i) \quad i = 1, 2, \ldots, p \tag{33c}$$

In Eq. (32), $\boldsymbol{P}_k$ is a constant; therefore, solving the AMPC problem at each step is equivalent to solving the following quadratic programming problem:

$$\begin{aligned} &\min_{\Delta\boldsymbol{U}_k, \varepsilon} \left[\Delta\boldsymbol{U}(k)^T, \varepsilon\right]^T \boldsymbol{H}_k \left[\Delta\boldsymbol{U}(k)^T, \varepsilon\right] + \boldsymbol{G}_k \left[\Delta\boldsymbol{U}(k)^T, \varepsilon\right] \\ &s.t. \Delta\boldsymbol{U}_{\min} \le \Delta\boldsymbol{U}(t) \le \Delta\boldsymbol{U}_{\min}, t = k, \ldots, k+c-1 \\ &\boldsymbol{U}_{\min} \le \boldsymbol{u}(k-1) + \textstyle\sum_{i=k}^{t} \Delta\boldsymbol{U}(i) \le \boldsymbol{U}_{\max}, t = k, \ldots, k+c-1 \\ &\boldsymbol{Y}_{\min} - \varepsilon \le \boldsymbol{S}_x \Delta\boldsymbol{x}(k|k) + \boldsymbol{S}_u \Delta\boldsymbol{U}(k) \le \boldsymbol{Y}_{\max} + \varepsilon \\ &\varepsilon > 0 \end{aligned} \tag{34}$$

Solving Eq. (34) in each control cycle produces a series of control input increments and relaxation factor in the control time domain in the form of:

$$\left[\Delta\boldsymbol{U}_k^*, \varepsilon_k^*\right] = \left[\Delta\boldsymbol{u}_k^*, \Delta\boldsymbol{u}_{k+1}^*, \ldots, \Delta\boldsymbol{u}_{k+c-1}^*, \varepsilon_k^*\right]^T \tag{35}$$

The first element in the control sequence is then applied to the system as the following actual control input increment:

$$\boldsymbol{u}(k) = \boldsymbol{u}(k-1) + \Delta\boldsymbol{u}_k^* \tag{36}$$

After entering the next cycle, the above process is repeated, thus ensuring that the cycle tracks the control of the wheel loader.

(3) Stability analysis

Since constrained MPC systems are nonlinear, the second stability theorem of Lyapunov is commonly used to prove the nominal stability of such systems [37,41].

Theorem. *For the plant to be controlled (i.e. Eq. (18)), the discretized state space model (24) and Eq. (34) are considered. The weight matrices* $\boldsymbol{Q}$ *and* $\boldsymbol{R}$*, a prediction horizon* $p \ge 1$*, and a control horizon* $1 \le c \le p$ *are selected such that the optimal solution Eq. (35) of Eq. (34) holds. Let the optimization objective* $J^*(\boldsymbol{x}(k), \Delta\boldsymbol{U}(k))$ *be a Lyapunov function* $V^*(k)$ *at time k. If* $V^*(k+1) \le V^*(k)$ *is satisfied at time k+1, then the optimal solution (37) guarantees the nominal stability of the plant (i.e. Eq. (18)).*

Proof. In solution Eq. (35) of Eq. (34) with softening constrains, let $\Delta\boldsymbol{u}^*(k+1|k)$ and $\varepsilon^*(k)$ be the optimal control input increment and optimal relaxation factor, respectively; further, let $\boldsymbol{u}^*(k+1|k)$ be the optimal control input. The optimization objective $J^*(\boldsymbol{x}(k), \Delta\boldsymbol{U}(k))$ is denoted as a Lyapunov function $V^*(k)$, as follows:

$$\begin{aligned} V^*(k) &= \min J(\boldsymbol{x}(k), \Delta\boldsymbol{U}(k)) \\ &= \min \left\{ \sum_{i=1}^{p} \left\| \boldsymbol{y}(k+i|k) - \boldsymbol{y}_{ref}(k+i|k) \right\|_{\boldsymbol{Q}}^2 \right. \\ &\left. + \sum_{i=1}^{c-1} \left[\left\| \Delta\boldsymbol{u}(k+i|k) \right\|_{\boldsymbol{R}}^2 + \rho\varepsilon^2(k) \right] \right\} \end{aligned} \tag{37}$$

The optimization function (37) satisfies $V^*(0) = 0$ for $k = 0$ and $V^*(k) > 0$ for arbitrary $k \ne 0$. For model (18) with external disturbance, the optimal control input increments $\Delta\boldsymbol{u}(k+1+i|k+1)$ and optimal relaxation factors $\varepsilon(k)$ are given by:

$$\begin{aligned} \Delta\boldsymbol{u}(k+1+i|k+1) &= \begin{bmatrix} \Delta\boldsymbol{u}(k+1|k+1) \\ \Delta\boldsymbol{u}(k+2|k+1) \\ \cdots \\ \Delta\boldsymbol{u}(k+c|k+1) \end{bmatrix} \\ &= \begin{bmatrix} \Delta\boldsymbol{u}^*(k+1|k) \\ \Delta\boldsymbol{u}^*(k+2|k) \\ \cdots \\ \Delta\boldsymbol{u}^*(k+c|k) \end{bmatrix} \end{aligned} \tag{38}$$

$$\varepsilon(k+1) = \varepsilon^*(k) \tag{39}$$

The control input $\boldsymbol{u}(k+1+i|k+1)$ is written as follows:

$$\begin{aligned} \boldsymbol{u}(k+1+i|k+1) &= \begin{bmatrix} \boldsymbol{u}(k+1|k+1) \\ \boldsymbol{u}(k+2|k+1) \\ \cdots \\ \boldsymbol{u}(k+c|k+1) \end{bmatrix} \\ &= \begin{bmatrix} \boldsymbol{u}^*(k+1|k) \\ \boldsymbol{u}^*(k+2|k) \\ \cdots \\ \boldsymbol{u}^*(k+c|k) \end{bmatrix} \end{aligned} \tag{40}$$

It can be readily verified that Eqs. (38)–(40) are a group of feasible solutions to the Eq. (34). The control input increments $\Delta\boldsymbol{u}(k+1+i|k+1)$ and the control inputs $\boldsymbol{u}(k+1+i|k+1)$ are belong to the constraint sets Eq. (32).

The relation between $J(\boldsymbol{x}(k+1), \Delta\boldsymbol{U}(k+1))$ and $V^*(k)$, obtained from Eqs. (38)–(40), is shown as follows:

Table 2
Basic parameters of the wheel loader model.

Unloading	Loading
$m_1 = 7010$ [kg]	$m_1 = 9510$ [kg]
$m_2 = 8590$ [kg]	$m_2 = 8590$ [kg]
$I_1 = 7010$ [kg m^2]	$I_1 = 9510$ [kg m^2]
$I_2 = 8590$ [kg m^2]	$I_2 = 8590$ [kg m^2]
$l_a = 0.75$ [m]	$l_a = 0.45$ [m]
$l_b = 0.75$ [m]	$l_b = 1.05$ [m]
$l_c = 1.00$ [m]	$l_c = 1.00$ [m]
$l_d = 0.80$ [m]	$l_d = 0.80$ [m]
$C_\gamma = 2{\cdot}10^8$ [Nm/rad]	$C_\gamma = 2{\cdot}10^8$ [Nm/rad]
$K_\gamma = 0.0$ [Nms/rad]	$K_\gamma = 0.0$ [Nms/rad]
$C_{\alpha 1} = 6.0$ [rad^{-1}]	$C_{\alpha 1} = 6.0$ [rad^{-1}]
$C_{\alpha 2} = 6.0$ [rad^{-1}]	$C_{\alpha 2} = 6.0$ [rad^{-1}]

$$
\begin{aligned}
J\left(\boldsymbol{x}\left(k+1\right),\Delta\boldsymbol{U}\left(k+1\right)\right) &= \sum_{i=1}^{p}\left\|\boldsymbol{y}\left(k+1+i|k+1\right)\right. \\
&\quad \left.-\boldsymbol{y}_{ref}\left(k+1+i|k+1\right)\right\|_{\boldsymbol{Q}}^{2} \\
&\quad +\sum_{i=1}^{c-1}\left[\left\|\Delta\boldsymbol{u}\left(k+1+i|k+1\right)\right\|_{\boldsymbol{R}}^{2}\right. \\
&\quad \left.+\rho\varepsilon^{2}\left(k+1+i|k+1\right)\right] \\
&= \sum_{i=1}^{p}\left\|\boldsymbol{y}\left(k+i|k\right)-\boldsymbol{y}_{ref}\left(k+i|k\right)\right\|_{\boldsymbol{Q}}^{2} \\
&\quad +\sum_{i=1}^{c-1}\left[\left\|\Delta\boldsymbol{u}^{*}\left(k+i|k\right)\right\|_{\boldsymbol{R}}^{2}\right. \\
&\quad \left.+\rho\varepsilon^{*2}\left(k+i|k\right)\right] \\
&= \sum_{i=1}^{p}\left\|\boldsymbol{y}\left(k+i|k\right)-\boldsymbol{y}_{ref}\left(k+i|k\right)\right\|_{\boldsymbol{Q}}^{2} \\
&\quad +\sum_{i=0}^{c-1}\left[\left\|\Delta\boldsymbol{u}^{*}\left(k+i|k\right)\right\|_{\boldsymbol{R}}^{2}\right. \\
&\quad \left.+\rho\varepsilon^{*2}\left(k+i|k\right)\right] \\
&\quad -\left\|\boldsymbol{y}\left(k|k\right)-\boldsymbol{y}_{ref}\left(k|k\right)\right\|_{\boldsymbol{Q}}^{2} \\
&\quad -\left\|\Delta\boldsymbol{u}^{*}\left(k|k\right)\right\|_{\boldsymbol{R}}^{2}-\rho\varepsilon^{2}\left(k|k\right) \\
&= V^{*}\left(k\right)-\left\|\boldsymbol{y}\left(k|k\right)-\boldsymbol{y}_{ref}\left(k|k\right)\right\|_{\boldsymbol{Q}}^{2} \\
&\quad -\left\|\Delta\boldsymbol{u}^{*}\left(k|k\right)\right\|_{\boldsymbol{R}}^{2}-\rho\varepsilon^{2}\left(k|k\right)
\end{aligned}
\tag{41}
$$

Owing to the optimality of Eq. (34), the optimization objective $J^{*}(\boldsymbol{x}(k+1), \Delta\boldsymbol{U}(k+1))$ is not less than $V^{*}(k+1)$, namely:

$$
\begin{aligned}
&V^{*}\left(k+1\right)\leq J\left(\boldsymbol{x}\left(k+1\right),\Delta\boldsymbol{U}\left(k+1\right)\right) \\
&\leq V^{*}\left(k\right)-\left\|\boldsymbol{y}\left(k\right)-\boldsymbol{y}_{ref}\left(k\right)\right\|_{\boldsymbol{Q}}^{2}-\left\|\Delta\boldsymbol{u}^{*}\left(k\right)\right\|_{\boldsymbol{R}}^{2}-\rho\varepsilon^{2}\left(k\right)
\end{aligned}
\tag{42}
$$

which leads to:

$$
V^{*}\left(k+1\right)\leq V^{*}\left(k\right) \tag{43}
$$

Thus, the nominal stability of Eq. (18) is proven.

Remark. The Lyapunov function (37) satisfies $V^{*}(0)=0$ when $k=0$ and $V^{*}(0)>0$ when $k\neq 0$. Furthermore, this Lyapunov function (37) decreases monotonically, i.e., $V^{*}(k+1)\leq V^{*}(k)$ holds along the entire reference trajectory. Thus, Eq. (18) is nominally stable.

4. Simulation analysis

The V-shaped operation of the wheel loader was evaluated using a simulation in which the control target traveled along a desired path at a planned velocity. The control target tracked its trajectory by continuously reducing any deviations in the path and velocity from the reference trajectory. Two different simulation conditions were designed to comprehensively analyze the feasibility of the proposed control method. The upper driving speed limit for the desired trajectory was set to 3 m/s [42], and the basic parameters of the control system were set as follows: $T_s = 0.05$ s, $p = 20$, $c = 15$, -38 deg $\leq \gamma \leq 38$ deg, -4.5 m/s^2 $\leq a \leq 4.5$ m/s^2. The weight matrix and relaxation factor were set as $\boldsymbol{Q} = \mathrm{diag}(150, 50, 150)$, $\boldsymbol{R} = 5000\boldsymbol{I}_{(2\times 2)}$ and $\rho = 1000$. The basic parameters of wheel loader were as quoted in reference [43] and shown in Table 2.

4.1. The influence of path curvature on trajectory tracking

In order to determine the influence of path curvature on the trajectory tracking performance of an autonomous wheel loader, the behavior of the loader using the MPC system, which does not consider path curvature disturbances, was compared with that using the proposed AMPC system, which can adapt to path curvature disturbances.

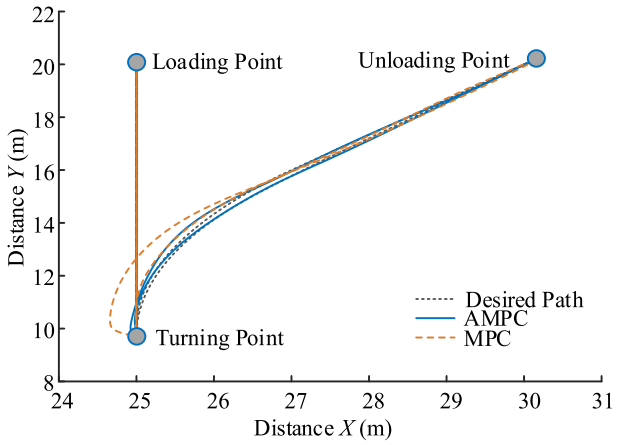


Fig. 7. Comparison between the desired and actual paths of a wheel loader using the proposed AMPC and conventional MPC methods.

The simulation results are shown in Figs. 7 to 9. As shown in Fig. 7, when the reference path is a straight line, the autonomous wheel loader can accurately follow the path when using either control system. However, when the reference path is a curve, the two different control systems resulted in a considerable difference in behavior. Whether in the forward or reverse driving stages, under the action of MPC the actual driving path is quite different from the expected path. The longitudinal control of the wheel loader is shown in Fig. 8, in which it can be seen that both the acceleration a and the longitudinal speed u_f were within the ranges of the constraints when using either system. The planned velocities exhibit obvious acceleration and deceleration processes that are in line with actual operation conditions. The actual velocities under the action of either control system tracked the expected velocities well, where the positive and negative values of u_f represent the forward and reverse driving phases, respectively. A comparison of the lateral control of the wheel loader is shown in Fig. 9. This is the last stage of the loader, which returns from the unloading point to the turning point at $t = 20$ s. The results shown in Fig. 9, along with those in Fig. 7, demonstrate that the driving path at this stage has a large curvature. This is because the effect of path curvature on trajectory tracking is not taken into account in MPC, which leads to a sudden increases of the heading error at this stage. In comparison, AMPC, which considers the path curvature disturbance is taken into consideration in AMPC, thereby providing a competitive trajectory tracking performance. As can be seen from Fig. 9(a), during the trajectory tracking process, the articulated angle γ remained within the constraints, ensuring that the output of the control system could be executed smoothly by the actuator. Under the action of the proposed AMPC, Figs. 9(b) and (c) indicate that the maximum values of the displacement error e_1 and heading error e_2 were 0.12 m and -8 deg, respectively.

Compared with the results using MPC, these maximum values for e_1 and e_2 were reduced by 65.7% and 60%, respectively. Thus, based on the results of this analysis, the proposed AMPC system considering the disturbance of the path curvature was clearly able to track the trajectory of the loader quite well, showing reduced errors when compared to the MPC system that did not consider any trajectory disturbance.

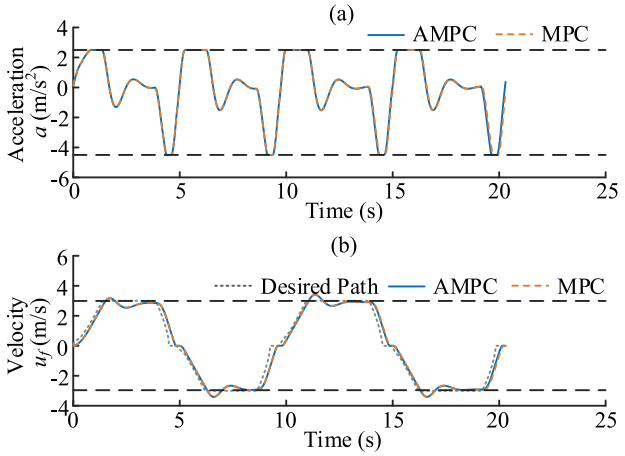


Fig. 8. Comparison of the longitudinal control of a wheel loader under the proposed AMPC and conventional MPC methods in terms of (a) acceleration and (b) longitudinal speed.

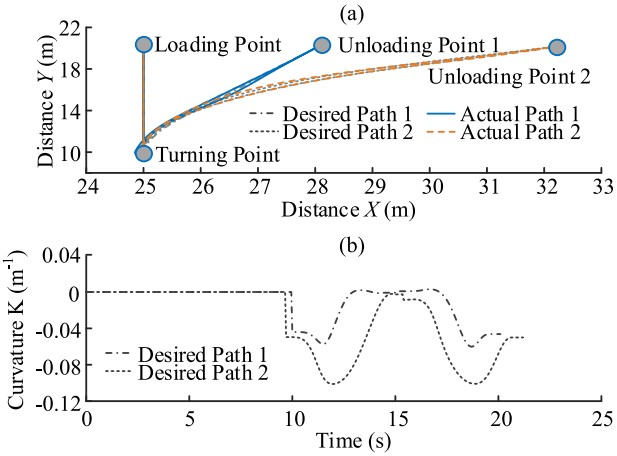


Fig. 10. Tracking performance of AMPC in terms of (a) desired path and actual path and (b) path curvature.

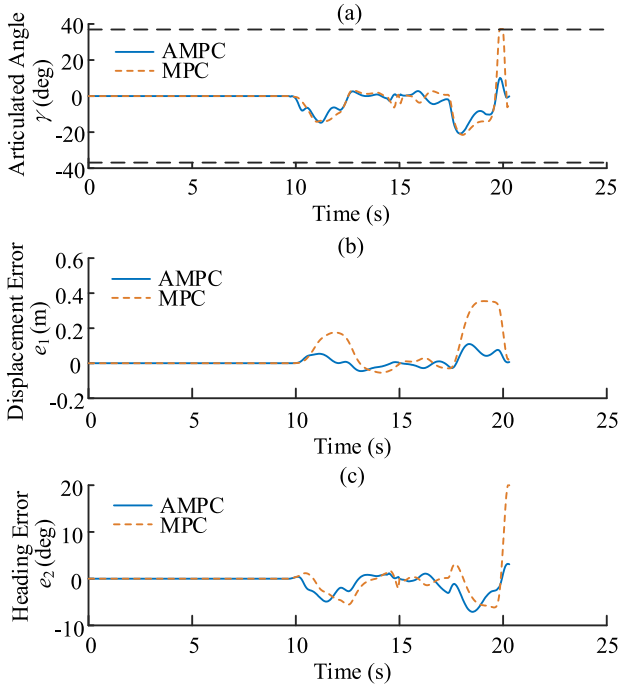


Fig. 9. Comparison of the lateral control of a wheel loader under the proposed AMPC and conventional MPC methods in terms of (a) articulated angle, (b) displacement error, and (c) heading error.

4.2. The influence of different curvature amplitudes on trajectory tracking

In general, the unloading points will change under V-shaped wheel loader operation mode according to the operational requirements; therefore, different driving paths must be determined for different unloading points and the path curvature amplitude must change in the curvature amplitude of the reference path poses potential challenges to the performance of the control system. A simulation was therefore used to track the trajectory control of an autonomous wheel loader using the AMPC system for different curvature amplitude conditions and analyze its robustness accordingly.

The simulation results are shown in Figs. 10 to 12. Fig. 10 shows the tracking results for different desired paths and the corresponding path curvatures. Note there is a significant difference in the curvature of the two desired paths: the maximum curvature amplitude of Path 2 is twice that of Path 1, but the autonomous wheel loader using AMPC performs well at trajectory tracking, showing a strong robustness under changing curvature amplitude. Fig. 11 shows the longitudinal control of the wheel loader, in which it can be seen that the actual velocity of the wheel loader under different curvature amplitudes tracked the expected vehicle velocity well, and that the vehicle velocity, acceleration a, and longitudinal speed u_f were within the ranges of the constraints. Fig. 12 shows the lateral control of the wheel loader. As can be seen from Fig. 12(a), during the trajectory tracking process, the articulated angle γ was always within the constraints, ensuring that the output of the control system could be executed smoothly by the actuator. As can be seen from Fig. 12(b) and (c), there is no significant difference between the maximum displacement error e_1 and heading error $e_2$in wither path tracking process. In Fig. 12(b), there is a local step in e_1 at 15 s, but this was the result of the change in the vehicle registration point after the loader arrived at the target point. Therefore, the proposed AMPC system was able to accurately track the desired trajectory under different curvature amplitudes, ensuring the safe and active performance of the entire vehicle during the process of tracking the trajectory.

The algorithm and simulation program are implemented in 64-bit Windows 10 system. On the quad-core Intel CPU with the main frequency of 3.40 GHz, the time to solve the quadratic programming is less than 10 ms (In fact, previous tests have shown that 100 Hz communication frequency can meet the needs of most tasks). This algorithm can meet the requirements of future hardware experiments.

5. Conclusion

A non-uniform driving trajectory satisfying the conditions of driving forwards and in reverse using a path-velocity decomposition method was planned and combined with the characteristics of a wheel loader with a minimum stable turning radius and functioning in a typical operation mode.

An adaptive model predictive control (AMPC) system operating under conditions in which the path curvature was perturbed was constructed based on a kinematic model of the loader and a dynamic driving deviation model. In the proposed AMPC system, the path curvature in a time-varying state is employed as the disturbance input, and the acceleration and articulated angle

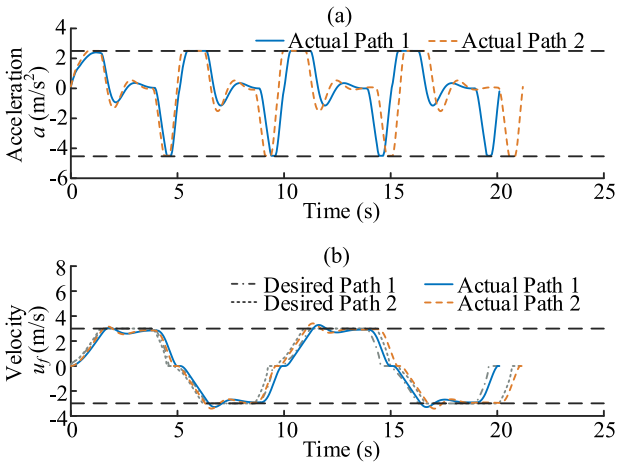


Fig. 11. Longitudinal control of the wheel loader using AMPC in terms of (a) acceleration and (b) longitudinal speed.

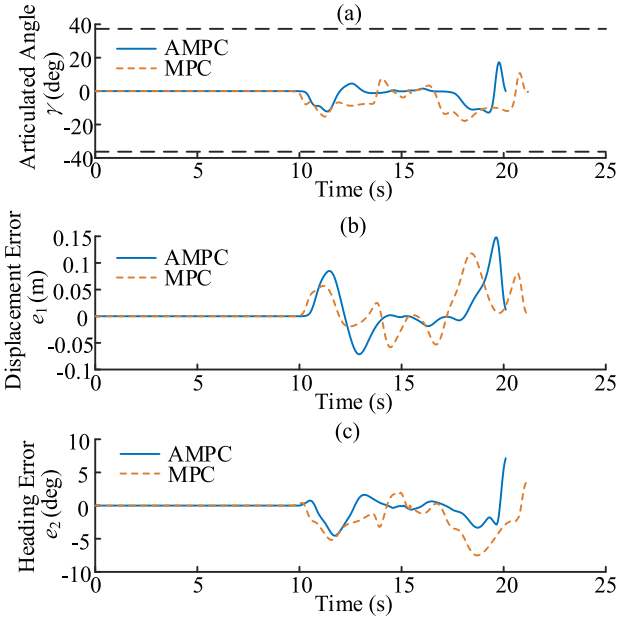


Fig. 12. Lateral control of the wheel loader using AMPC in terms of (a) articulated angle, (b) displacement error, and (c) heading error.

velocity are used as control inputs. The vehicle velocity and articulated angle are the control outputs from the AMPC system.

Taking a planned non-uniform motion trajectory as a target, the trajectory of an autonomous wheel loader was tracked using the proposed AMPC system and its performance was compared with that of a conventional MPC system that did not consider path disturbances.

The results show that the maximum displacement and heading errors resulting from the proposed AMPC system were 65.7% and 60% lower, respectively, than those produced by the conventional MPC system. The corresponding trajectory tracking effect was also observed to be significantly improved. Using the AMPC trajectory tracking system proposed in this paper, the desired trajectory was tracked under different curvature amplitude conditions with good results, guaranteeing the safe and active performance of the autonomous wheel loader vehicle in terms of trajectory tracking.

Declaration of competing interest

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

Acknowledgments

This work was sponsored by the National Natural Science Foundation of China (Grant No. 51875055); Scientific and Technological Research Program of Chongqing Municipal Education Commission, China (Grant No. KJQN201800718); Scientific and Technological Research Program of Chongqing Municipal Education Commission, China (Grant No. KJQN201803106).

References

[1] K. Oh, S. Yun, K. Ko, Gear ratio and shift schedule optimization of wheel loader transmission for performance and energy efficiency, Automat. Constr. 69 (2016) 89–101.
[2] B. Frank, J. Kleinert, R. Filla, Optimal control of wheel loader actuators in gravel applications, Automat. Constr. 91 (2018) 1–14.
[3] P.A. Mikhirev, Theory of the working cycle of automated rock-loading machines of periodic action, Sov. Min. Sci. 19 (1983) 515–522.
[4] A. Hemami, F. Hassani, An overview of autonomous loading of bulk material, in: Proceedings of the 26th ISARC, Austin TX, U.S.A, 2009, pp. 405–411.
[5] S. Dadhich, F. Sandin, U. Bodin, Field test of neural-network based automatic bucket-filling algorithm for wheel-loaders, Automat. Constr. 97 (2019) 1–12.
[6] J.M. Roberts, E.S. Duff, P.I. Corke, Reactive navigation and opportunistic localization for autonomous underground mining vehicles, Inform. Sci. 145 (2002) 127–146.
[7] B. Frank, L. Skogh, R. Filla, On increasing fuel efficiency by operator assistant systems in a wheel loader, in: International Conference on Advanced Vehicle Technologies and Integration, Changchun, China, 2012, pp. 155–161.
[8] S. Dadhich, U. Bodin, U. Andersson, Key challenges in automation of earth-moving machines, Autom. Constr. 68 (2016) 212–222.
[9] S. Sarata, Y. Weeramhaeng, T. Tsubouchi, Planning of scooping position and approach path for loading operation by wheel loader, in: International Symposium on Automation and Robotics in Construction ISARC, 2005, pp. 1–6.
[10] S. Sarata, Y. Weeramhaeng, T. Tsubouchi, Approach path generation to scooping position for wheel loader, in: Proceeding IEEE International Conference on Robotics and Automation, 2005, pp. 1809–1814.
[11] S. Sarata, N. Koyachi, T. Tubouchi, Development of autonomous system for loading operation by wheel loader, in: International Symposium on Automation and Robotics in Construction, 2006, pp. 406–471.
[12] B.J. Alshaer, T.T. Darabseh, M.A. Alhanouti, Path planning, modeling and simulation of an autonomous articulated heavy construction machine performing a loading cycle, Appl. Math. Model. 37 (2013) 5315–5325.
[13] T. Nayl, G. Nikolakopoulos, T. Gustafsson, Effect of kinematic parameters on MPC based on-line motion planning for an articulated vehicle, Robot. Auton. Syst. 70 (2015) 16–24.
[14] T. Nayl, G. Nikolakopoulos, T. Gustafsson, A full error dynamics switching modeling and control scheme for an articulated vehicle, Int. J. Control Autom. 13 (2015) 1221–1232.
[15] T. Nayl, G. Nikolakopoulos, T. Gustafsson, Design and experimental evaluation of a novel sliding mode controller for an articulated vehicle, Robot. Auton. Syst. 103 (2018) 213–221.
[16] J. Choi, K. Huhtala, Constrained global path optimization for articulated steering vehicles, IEEE Trans. Veh. Technol. 65 (2016) 1868–1879.
[17] K. Oh, H. Kim, P. Kim, Integrated wheel loader simulation model for improving performance and energy flow, Autom. Constr. 58 (2015) 129–143.
[18] N.L. Azad, Dynamic modelling and stability controller development for articulated steer vehicles, 2006.
[19] G.M. Hoffmann, C.J. Tomlin, M. Montemerlo, Autonomous automobile trajectory tracking for off-road driving: controller design, experimental validation and racing, in: 2007 American Control Conference, IEEE, 2007, pp. 2296–2301.
[20] P. Ridley, P. Corke, Autonomous control of an underground mining vehicle, in: 2nd Annual Dynamic Systems and Control Conference, 2001, pp. 26–31.

[21] R. Filla, Quantifying Operability of Working Machines, Linköping University, 2011.
[22] M. Pivtoraiko, A. Kelly, Efficient constrained path planning via search in state lattices, in: International Symposium on Artificial Intelligence, Robotics, and Automation in Space, 2005, pp. 1–7.
[23] T.M. Howard, C.J. Green, A. Kelly, State space sampling of feasible motions for high performance mobile robot navigation in complex environments, J. Field Robot. 25 (2008) 325–345.
[24] S.M. LaValle, J.J. Kuffner, Randomized kinodynamic planning, Int. J. Robot. Res. 20 (2001) 473–479.
[25] J. Leonard, J. How, S. Teller, A perception-driven autonomous urban vehicle, J. Field Robot. 25 (2008) 727–774.
[26] S. Karaman, E. Frazzoli, Incremental sampling-based algorithms for optimal motion planning, Int. J. Robot. Res. 30 (2010) 5326.
[27] T. Fraichard, A. Scheuer, From Reeds and Shepp's to continuous-curvature paths, IEEE Trans. Robot. 20 (2004) 1025–1035.
[28] C.G.L. Bianco, Minimum-jerk velocity planning for mobile robot applications, IEEE Trans. Robot. 29 (2013) 1317–1326.
[29] W. Xu, J. Wei, J.M. Dolan, A real-time motion planner with trajectory optimization for autonomous vehicles, in: 2012 IEEE International Conference on Robotics and Automation, IEEE, 2012, pp. 2061–2067.
[30] A.P. Aguiar, J.P. Hespanha, P.V. Kokotovic, Path-following for nonminimum phase systems removes performance limitations, IEEE Trans. Autom. Control 50 (2005) 234–239.
[31] H. Talebi Abatari, A. Dehghani Tafti, Using a fuzzy pid controller for the path following of a car-like mobile robot, in: 2013 First RSI/ISM International Conference on Robotics and Mechatronics (ICRoM), IEEE, 2013, pp. 189–193.
[32] F. Rovira-Más, Q. Zhang, Fuzzy logic control of an electrohydraulic valve for auto-steering off-road vehicles, Proc. Inst. Mech. Eng. D 222 (2008) 917–934.
[33] K. Nam, S. Oh, H. Fujimoto, Robust yaw stability control for electric vehicles based on active front steering control through a steer-by-wire system, Int. J. Auto. Tech.-Kor. 13 (2012) 1169–1176.
[34] G.V. Raffo, G.K. Gomes, J.E. Normey-Rico, A predictive controller for autonomous vehicle path tracking, IEEE Trans. Intell. Transp. 10 (2009) 92–102.
[35] F. Kuhne, W.F. Lages, J. M. G. D. Silva Jr., Model predictive control of a mobile robot using linearization, in: Prodeedings of Mechatronics and Robototics, 2004, pp. 525–530.
[36] L. Wang, Model Predictive Control System Design and Implementation using MATLAB, Springer Verlag London Limited, London, 2009.
[37] L. Shengbo, W. Jianqiang, L. Keqiang, Stabilization of linear predictive control systems with softening constraints, J. Tsinghua Univ. 50 (2010) 1848–1852.
[38] H. Yang, M. Guo, Y. Xia, Trajectory tracking for wheeled mobile robots via model predictive control with softening constraints, IET Control Theory Appl. 12 (2018) 206–214.
[39] J.M. Maciejowski, Pridictive Control with Constraints, Pearson Education, London, 2002.
[40] B. Kouvaritakis, M. Cannon, S.V. Raković, Explicit use of probabilistic distributions in linear predictive control, Automatica 46 (2010) 1719–1724.
[41] H. Yang, M. Guo, Y. Xia, Trajectory tracking for wheeled mobile robots via model predictive control with softening constraints, IET Control Theory Appl. 12 (2018) 206–214.
[42] M. Gottschalk, G. Jacobs, A. Kram, Test method for evaluating the energy efficiency of wheel loaders, ATZoffhighway Worldw. 11 (2018) 44–49.
[43] Y. He, A. Khajepour, J. McPhee, Dynamic modeling and stability analysis of articulated frame steer vehicles, Int. J. Heavy Veh. Syst. 12 (2005) 28–59.

Junren Shi received the M.S. degree in mechanical and electronic engineering from Chongqing University of Technology, China, in 2016. He is currently a Ph.D. student in vehicle engineering at the Chongqing University, China. His current research activities focusing in the areas of: Model Predictive Control, Trajectory planning and tracking, Data mining.

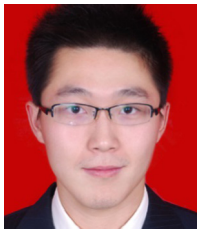
Dongye Sun received Ph.D. degree in mechanical engineering from Jilin University, China, in 1996. Now he is a Professor at State Key Laboratory of Mechanical Transmission, Chongqing University, China. His research areas of interest include power transmission and integrated control, hybrid powertrain design theory and control methods.

Datong Qin received Ph.D. degree in Mechanical Engineering Department from Chongqing University, China, in 1993. Now he is a Professor at State Key Laboratory of Mechanical Transmission, Chongqing University, China. His research areas of interest include structural design and optimization, mechanical transmissions and control, especially in gear dynamics.

Minghu Hu received Ph.D. degree in Mechanical Engineering Department from Chongqing University, China, in 2008. Now he is a Professor at State Key Laboratory of Mechanical Transmission, Chongqing University, China. His research areas of interest include structural design and optimization, mechanical transmissions and control.

Yingzhe Kan received the M.S. degree in vehicle engineering from Chongqing University of Technology, China, in 2018. He is currently working toward the Ph.D. degree in the School of Vehicle Engineering at Chongqing University, Chongqing, China. His research interests include hydro-mechanical transmission system and battery management system.

Ke Ma received the B.S. degree from Southwest Jiaotong University and M.S. degree from Chongqing University of Technology, China, in 2015 and 2018 respectively. He is currently working toward the Ph.D. degree in the School of Vehicle Engineering at Chongqing University, China. His research interests include electro-hydraulic intelligent control technology and new power transmission technology.



Ruibo Chen received the B.S. and M.S. degrees in mechanical engineering from Xinjiang University, China, in 2015 and 2018 respectively. Now he is a Ph.D. student studying in the College of Mechanical Engineering of Chongqing University. His research area is mechanical system dynamics, transmission system electromechanical integration technology.