MDPI

Article

Research on TD3-Based Distributed Micro-Tillage Traction Bottom Control Strategy

Guangxiu Ning [1,†], Lide Su [1,†], Yong Zhang [1,*], Jian Wang [1], Caili Gong [2] and Yu Zhou [1]

1   College of Mechanical and Electrical Engineering, Inner Mongolia Agricultural University, Hohhot 010018, China
2   College of Electronic Information Engineering, Inner Mongolia University, Hohhot 010021, China
*   Correspondence: yongz@imau.edu.cn; Tel.: +86-136-7474-8009
†   These authors contributed equally to this work.

Abstract: Due to its flexibility and versatility, the electric distributed drive micro-tillage chassis can be used more often in the future in Intelligence agriculture scenarios. However, due to the complex working conditions of the agricultural operation environment, it is a challenging task to distribute the torque demand of four wheels reasonably and effectively. In this paper, we propose a drive torque allocation strategy based on deep reinforcement learning to ensure straight-line retention and energy saving, using a distributed electric traction chassis for greenhouses as the research object. The torque assignment strategy can be represented as a Markovian decision process, and the approximate action values and policy functions are obtained through an Actor–Critic network, and the Twin Delayed Deep Deterministic Policy Gradient (TD3) is used to incorporate the vehicle straight-line retention rate into the cumulative reward to reduce energy consumption. The training results under plowing working conditions show that the proposed strategy has a better straight-line retention rate. For typical farming operation conditions, the proposed control strategy significantly improves the energy utilization and reduces the energy by 10.5% and 3.7% compared to the conventional average torque (CAT) distribution strategy and Deep Deterministic Policy Gradient (DDPG) algorithm, respectively. Finally, the real-time executability of the proposed torque distribution strategy is verified by Soil-tank experiments. The TD3 algorithm used in this study has stronger applicability than the traditional control algorithm in dealing with continuous control problems, and provides a research basis for the practical application of intelligent control algorithms in future greenhouse micro-tillage chassis drive control strategies.

Keywords: greenhouse; electric traction chassis; deep reinforcement learning; drive control strategy; soil-trough experiment

1. Introduction

In modern agriculture, the small and confined space in the greenhouse environment requires high requirements for the form factor and exhaust emissions of the agricultural vehicles used for operations [1]. With the development of modern agriculture, the operational vehicles used in the greenhouse environment also need to be versatile, flexible, and highly environmentally friendly. Currently, most of the operating vehicles for greenhouses are small traditional fuel-powered equipment, such as tractors, tricycles, and mobile platforms. Most of these vehicles are equipped with traditional fuel power systems, which generate exhaust gases that can be harmful to the health of users. At the same time, the poor flexibility of traditional operating equipment can significantly reduce productivity in practice [2]. For this reason, it is crucial to develop environmentally friendly, flexible, and energy-sustainable operating equipment for the greenhouse environment [3,4].

With the development of lithium-ion battery technology and electric motor technology, the development of electric drive vehicles has been greatly advanced [5], among which four-wheel-drive electric vehicles equipped with hub motors have gained great attention

due to their flexibility [6,7]. The development of electrification of agricultural machinery in Intelligence agriculture offers the possibility for the development of this vehicle technology in agricultural machinery [8,9]. Therefore, this vehicle technology has great potential to solve the above-mentioned agricultural problems. However, several key issues need to be overcome for the application of such vehicles in future greenhouses, including overdriving of vehicles under plowing conditions [10,11], straight-line travel retention under operating conditions [12], and improved energy efficiency. Therefore, in this paper, an electric multifunctional distributed micro-tillage traction chassis is developed to address the low environmental protection, single function of operating equipment and drive control problems of agricultural facilities for greenhouses, and based on this, the drive control strategy is investigated so that it can meet the operational needs under complex agricultural conditions.

Recently, relevant scholars have carried out many valuable studies on electric vehicle control methods in the field of intelligent agricultural machinery. Wu et al. [13] designed a novel chassis structure for the problem of difficult greenhouse crop transportation, which is able to travel on both ground and track surfaces, and this chassis successfully meets the basic requirements of greenhouse crop transportation. Through the study of its control strategy, this chassis can realize a variety of functions, such as ground driving, track driving, up and down track moving, and automatic track changing. Azmi et al. [14] developed and designed a crop sowing agricultural robot that can achieve autonomous operation; the robot chassis adopts a four-wheel design and can sow 138 seedlings in 5 minutes, the accuracy rate can reach 92%, and the battery life is up to 4 h. Rong et al. [15] developed a greenhouse mushroom-picking robot that integrates the picking unit in parallel on the mobile platform and conducted experimental verification, and the results show that the success rate of mushroom recognition is 95%, and the harvesting success rate is 95%. Shen et al. [16] designed a hybrid chassis with simple structure, high transmission efficiency, and energy saving, which can realize four-wheel independent drive and have better controllability.

In response to the demand for intelligence, some researchers have carried out a series of studies in the field of intelligent control, including low-trajectory tracking, torque distribution strategy, etc. Yorozu et al. [17] proposed a method for trench detection and tracking by measuring the width and interpolation of the trench using the RGB-D camera, which realized the smooth harvesting operation of small agricultural robots in the fields of different widths of the ravine. Zhou et al. [18] took the coordinated control of drive attitude and drive of electric agricultural vehicles as the key research goal, and combined the interactive multi-model (IMM) algorithm with the extended Kalman filter according to the drive and structural characteristics of distributed-drive electric agricultural vehicles to achieve precise control of the driving trajectory tracking of the vehicle platform. Aiming at the complexity of the agricultural greenhouse environment, Ren et al. [19] sent a fuzzy PID path tracking method based on traditional vehicle PID control and applied it to crawler robots, and experiments show that the method has good control performance. Cao et al. [20] considered the changes of load transfer and adhesion characteristics of front and rear axles under vehicle dynamic conditions, and proposed a multi-objective optimal torque distribution strategy, and the simulation results show that the proposed method is superior to the traditional torque distribution strategy.

Traditional control methods require complex and accurate modeling to improve the accuracy of the algorithm, and cannot completely solve the nonlinear problem of the complex working conditions of electric vehicles [21]. With the development of artificial intelligence technology, reinforcement learning has been more widely used in the control field to solve control problems under complex working conditions because it does not rely on complex modeling. Shayan et al. [22] designed a torque vectoring controller based on the reinforcement learning (RL) algorithm of DDPG, which adjusts the control parameters in different driving environments through reinforcement learning, which significantly improves the performance and stability of vehicles under extreme driving conditions. Qi et al. [23] designed a plug-in hybrid energy management strategy based on deep rein-

forcement learning to solve the energy consumption problem of plug-in hybrid vehicles, and the vehicle can independently learn the energy allocation strategy from the working environment. The test data show that the algorithm can save 16.3% of fuel compared to the traditional control strategy.

In the field of agricultural machinery, due to the complex working conditions (such as typical ploughing operations), there are many factors affecting the stable operation of the operating equipment, and complex environmental factors need to be considered, and the equipment needs to be multifunctional. For the electric distributed micro-cultivation traction chassis mentioned above, this paper adopts the modular design concept, integrates the drive system and steering system, and the drive system adopts a wheel hub motor drive, which greatly simplifies the transmission system structure. The chassis is equipped with ternary lithium-ion batteries, which can be mounted with different operating equipment to meet different operational needs. In order to adapt to complex road conditions, the chassis is also designed with a lifting system to adapt to different operating environments. The chassis is expected to be used in narrow greenhouse environments, enabling ploughing, low-speed transportation, etc. when busy, and performing plant protection tasks when idle.

Based on the developed distributed traction chassis, a driving torque control strategy under ploughing conditions is proposed to reduce energy consumption under ploughing conditions and ensure the linear stability of the vehicle. To this end, the dynamic model of the vehicle with seven degrees of freedom is established, and the dynamic formula of the chassis is derived. In order to improve the control accuracy of the four drives, the TD3 algorithm was used to learn the torque distribution control strategy under ploughing conditions, and the feasibility of the control strategy was verified offline through the model, and finally, the experimental verification was carried out in the soil trough. This paper provides two main contributions to the field of intelligent agricultural equipment:

- A torque distribution strategy based on TD3 is proposed for a distributed micro-traction chassis for greenhouses to solve the control problem of electric equipment under complex agricultural operating conditions;
- Under the proposed control method, the energy utilization and straight-line driving stability of the chassis are improved.

2. Undercarriage Model Building

2.1. Overall Structure

The distributed micro-tillage traction chassis structure is shown in Figure 1. It mainly consists of a chassis frame, drive system, steering system, battery system, power distribution unit (PDU), central controller, farm equipment hookup interface, and intelligent kit hookup mechanical interface. The drive system consists of the wheel motor and lifting system; the wheel motor is connected to the chassis frame through the upper and lower swing arm; and the lifting cylinder is connected between the lower swing arm and chassis bracket to lift the chassis, thus increasing the terrain passability. The chassis frame is designed as a hollow type, and the hollow part of the front part of the bracket is used to install the steering system, PDU, and central controller. Considering the weight distribution of the entire traction chassis and the offset of the center of gravity during traction operations, the lithium-ion battery is mounted at the front of the chassis frame.

The electrical system of the chassis is shown in Figure 2, and the central controller is connected to the motor controller and hydraulic valve via CAN communication to control the movement of the four hub motors and lifting cylinders. The central controller integrates a 4G wireless module, through which the remote control handle can be used to remotely control the chassis movement. Under the control of the intelligent control system, the chassis can realize remote control drive, chassis lifting, and steering functions. This paper mainly studies the drive control strategy of the traction chassis for straight driving under the working conditions of a ploughing operation.
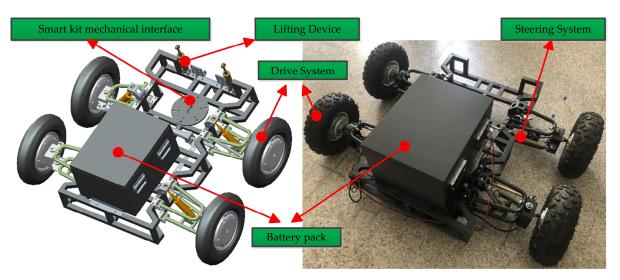
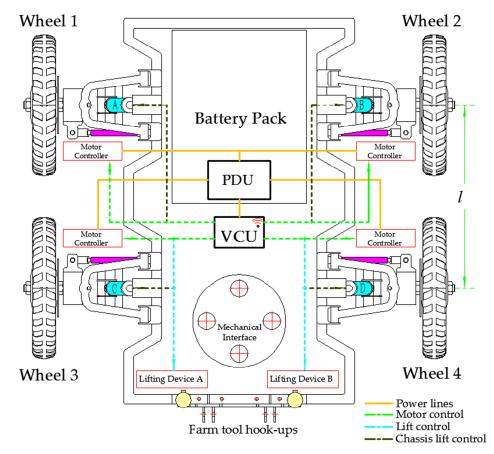**Figure 1.** Basic structure of electric traction chassis.

Figure 2. Electrical connection diagram of traction chassis.

2.2. Longitudinal Dynamics Model of the Chassis

In order to make the control strategy design and validation more accurate, the seven-degree-of-freedom model established by ignoring the pitch and tilt motion of the above model is shown in Figure 3 [24].
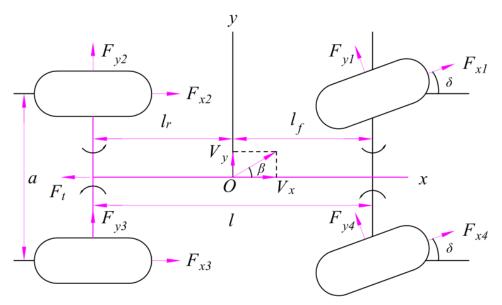
Figure 3. The 7-DOF model of traction chassis.

The equation of longitudinal motion of the vehicle is:

$$m\ddot{x} = (F_{x1} + F_{x4})\cos\delta + F_{x2} + F_{x3} - (F_{y1} + F_{y4})\sin\delta - \sum F_T \tag{1}$$

$$\sum F = F_a + F_t + F_r \tag{2}$$

where x is the longitudinal displacement of the vehicle; m is the mass of the vehicle; F_{x1}, F_{x2}, F_{x3}, and F_{x4} represent the longitudinal traction of the left front wheel, left rear wheel, right rear wheel, and right front wheel of the vehicle, respectively; F_{y1} and F_{y2} represent the lateral force of the left front wheel and right front wheel of the vehicle, respectively; $\sum F_T$ is the wheel turning angle; $\sum F$ is the resistance of the vehicle during travel; F_a is the air resistance; F_t is the plowing resistance; and F_r is the rolling resistance received during operation.

$$F_a = \frac{1}{2}C_d A_F \rho v_x^2 \tag{3}$$

$$F_t = (1.1 \sim 1.2)k \cdot z \cdot B_n \cdot h \tag{4}$$

$$F_r = mgf \tag{5}$$

where C_d is the wind resistance coefficient, A_F is the windward area, ρ is the air mass density, $v_x = x$ is the operating speed, k is the soil specific resistance, B_n is the plowing width, h is the plowing depth, f is the rolling resistance coefficient in a greenhouse soil environment, and g is the gravitational acceleration.

Since this paper only studies the straight-line driving motion of the traction chassis under the plowing operation condition, the steering operation can be ignored to keep the straight-line driving only. At the same time, the operating speed is designed to be 5 km/h, and the air resistance can be ignored. Simplifying the above equation, the longitudinal equation of motion of the tractor chassis driving in a straight line under plowing conditions is:

$$m\ddot{x} = F_{x1} + F_{x4} + F_{x2} + F_{x3} - (F_t + F_r) \tag{6}$$

In the operating mode of the traction chassis, the driving force mainly comes from the four hub motors. In order to better study the drive control strategy, it is necessary to conduct a force analysis of the hub motor wheels. Under greenhouse plowing conditions,

tire deformation and wheel forces while rolling on the operating road are ignored. The equation of force for each wheel is:

$$F_x = \frac{1}{r}\left(T_e - B_m\omega_m - J\frac{d\omega_m}{dt} - M_f\right) \quad (x = 1, 2, 3, 4) \tag{7}$$

In the equation, F_x is the longitudinal tire force, r is the motorized wheel radius, T_e is the electromagnetic torque of the motor, B_m is the viscous friction damping factor, ω_m is the angular velocity of the motorized wheel, J is the rotational inertia, and M_f is the rolling resistance moment. $x = 1, 2, 3, 4$ represents the left front wheel, left rear wheel, right rear wheel, and right front wheel of the chassis.

The longitudinal wheel force can be expressed as:

$$F_{xi} = f_i(\lambda_i)F_{yi} \quad (i = 1, 2, 3, 4) \tag{8}$$

where f_i is the road adhesion coefficient, which is a nonlinear function of the slip rate λ_i; and F_{yi} is the vertical load on the wheel.

Where the slip rate is the proportion of the wheel sliding in the course of motion:

$$\lambda_i = \frac{v_x - \omega_i r_i}{\max(v_x, \omega_i R_i)} \tag{9}$$

Under plowing conditions, the calculation of the vertical load on the chassis will directly affect the control effect of the control strategy, and its value is not only related to the mass of the traction chassis, but is also related to the longitudinal acceleration; ignoring the pitch and tilt motion of the chassis, the vertical load on the wheels can be expressed as:

$$\begin{aligned}
F_{y1} &= \tfrac{1}{2}mg \cdot \tfrac{l_r}{l} - \tfrac{1}{2}\tfrac{ma_x \cdot h}{l} \\
F_{y2} &= \tfrac{1}{2}mg \cdot \tfrac{l_r}{l} + \tfrac{1}{2}\tfrac{ma_x \cdot h}{l} \\
F_{y3} &= \tfrac{1}{2}mg \cdot \tfrac{l_r}{l} + \tfrac{1}{2}\tfrac{ma_x \cdot h}{l} \\
F_{y4} &= \tfrac{1}{2}mg \cdot \tfrac{l_r}{l} - \tfrac{1}{2}\tfrac{ma_x \cdot h}{l}
\end{aligned} \tag{10}$$

where l is the front and rear axle distance, l_f is the distance from the center of mass to the front axle, l_r is the distance from the center of mass to the rear axle, g is the acceleration of gravity, and h is the height from the center of mass to the road surface.

3. TD3-Based Control Strategy

3.1. Description of the Reinforcement Learning Algorithm

A Markov decision process (MDP) is one in which the decision maker makes a decision sequentially by observing a stochastic dynamic system with Markovianity periodically or continuously. That is, the decision is made by selecting an action from the set of available actions based on the observed state at each moment, and the next (future) state of the system is random, and its state transfer probability is Markovian. The decision maker makes a new decision based on the newly observed state, and so on, iteratively. The MDP formulation is defined as a five-tuple, $MDP = (S, A, P, R, \gamma)$, where S represents the state space, A represents the action space, P represents the probability density function of the state transfer process, R is the reward set, and γ is the discount factor [25].

3.2. Q-Learning Algorithm and DQN Algorithm

As a foundation for reinforcement learning algorithms, this section focuses on the basic Q-learning algorithm and DQN architecture. The Q-learning algorithm is an offline policy-based reinforcement learning algorithm that is model-independent. An intelligent body relies on a real-time updated Q matrix to select the next action and obtain a reward [26]. The core of the Q-learning algorithm is to find the optimal action value function $Q*$ from all

the policy functions *P*. However, as the state space and action space increase, the Q-learning algorithm is prone to "dimensional catastrophe".

The DQN algorithm takes a function approximation approach to estimate the state function based on the Q-learning algorithm, while the target network and empirical replay are used for model training. The training objectives is:

$$\text{argmin}_\theta(q(s,a) - \hat{q}(s,a,\theta))^2 \tag{11}$$

where θ is denoted as the parameter matrix, and the corresponding gradient is updated as follows:

$$\theta = \theta + \alpha(q(s,a) - \hat{q}(s,a,\theta))\Delta\hat{q}(s,a,\theta) \tag{12}$$

DQN employs a separate target network and uses this network to calculate TD errors, avoiding the problem of training instability due to data correlation. The parameter matrix update can be expressed as follows:

$$\theta = \theta + \alpha[r + \gamma\text{max}_{a'}Q(s',a';\theta^-) - Q(s,a;\theta)]\Delta Q(s,a;\theta) \tag{13}$$

where θ^- is the parameter of the target network. Therefore, the total loss function can be expressed in terms of the TD error, as follows:

$$L(\theta) = \frac{1}{2}\left(\underbrace{y_t - Q(s_t,a_t;\theta)}_{\text{TD error}}\right)^2$$

$$= \frac{1}{2}\left(\underbrace{r_t + \lambda\text{max}Q(s_{t+1},a;\theta)}_{\text{TD target}} - Q(s_t,a_t;\theta)\right)^2 \tag{14}$$

The optimal action value function can be obtained by using gradient descent to search for the global minimum of the loss function of the above equation.

3.3. DDPG Algorithm and TD3 Algorithm

3.3.1. DDPG Algorithm

DDPG is a combination of the Actor–Critic network and DQN algorithm, which can compensate for the deficiency of the DQN algorithm in dealing with the continuous control action problem [27]. In practice, the DDPG algorithm creates four networks for the policy network and the Q network, namely, the Actor-online network, the Critic-online network, the Actor-target network, and the Critic-target network. The loss function of the Critic network is defined as follows:

$$L(\theta_Q) = \frac{1}{M}\sum_{i=1}^{M}[y_t - Q(s_t,a_t|\theta_Q)]^2 \tag{15}$$

where θ_Q is the parameter of the Critic current network, M is the number of learning samples selected from the experience pool buffer, y_t is the Q value of Critic-target network, and Q is the Critic current network.

y_t is calculated as follows:

$$y_t = r_t + \gamma Q'\left[s_{t+1}, \mu'\left(s_{t+1}\middle|\theta_{\mu'}\right)\middle|\theta_{Q'}\right] \tag{16}$$

where r_t is the immediate reward, and γ is the discount factor. Q' and $Q_{Q'}$ are the Critic-target network and its parameters, respectively; μ' and $\theta_{\mu'}$ are the Actor-target network

and its parameters, respectively. The parameters of the Actor network are updated by the gradient back-propagation method training, and the loss gradient is:

$$\Delta_{\theta_\mu}\mu \approx \frac{1}{M}\sum_{t=1}^{M}\left[\Delta_a Q\left(s_t, a|\theta_Q\right)\Big|_{a=\mu(s_t|\theta_\mu)}\cdot\Delta_{\theta_\mu}\mu\left(s_t|\theta_\mu\right)\right] \tag{17}$$

where Δ is the gradient, θ_μ is the parameter of the actor's current network, and μ is the Actor current network. The parameter θ_μ of the online Actor network is updated using the learning rate α^μ in the following methods:

$$\theta_\mu \leftarrow \theta_\mu + \alpha^\mu\cdot\Delta_{\theta^\mu}\mu \tag{18}$$

In the DDPG algorithm, the TD error is also introduced in the Critic network, and the parameters are updated using the gradient descent method, so the loss function with the TD mean-square error is:

$$
\begin{aligned}
L\left(s_t, a_t; \theta_Q\right) &= \tfrac{1}{2}\left(y_t - Q(s_t, a_t; \theta)\right)^2 \\
&= \tfrac{1}{2}\left(\underbrace{r_t + \lambda Q\left(s_{t+1}, \mu\left(s_{t+1}; \theta_\mu\right); \theta_Q\right)}_{TD\ target} - Q\left(s_t, a_t; \theta_Q\right)\right)
\end{aligned}
\tag{19}
$$

In the target network, the loss function is:

$$
\begin{aligned}
L\left(s_t, a_t; \theta_Q\right) &= \tfrac{1}{2}\left(y_t' - Q(s_t, a_t; \theta)\right)^2 \\
&= \tfrac{1}{2}\left(\underbrace{r_t + \lambda Q'\left(s_{t+1}, \mu'\left(s_{t+1}; \theta_{\mu'}\right); \theta_{Q'}\right)}_{TD\ target} - Q\left(s_t, a_t; \theta_Q\right)\right)^2
\end{aligned}
\tag{20}
$$

Its gradient equation is:

$$\Delta L_{\theta_Q}\left(\theta_Q\right) = E_{s,a}\left[y_t' - Q\left(s_t, a_t; \theta_Q\right)\cdot\Delta_{\theta_Q}Q\left(s_t, a_t; \theta_Q\right)\right] \tag{21}$$

$$y_t' = r_t + \lambda Q'\left(s_{t+1}, \mu'\left(s_{t+1}; \theta_{\mu'}\right); \theta_{Q'}\right) \tag{22}$$

where Δ is the gradient, θ_Q is the parameter of the Critic current network, and μ' is the Actor current network. The parameter of the Critic-target network θ_Q is updated using the learning rate α^Q in the following methods:

$$\theta_Q \leftarrow \theta_Q + \alpha^Q\cdot\Delta_{\theta_Q}L\left(\theta_Q\right) \tag{23}$$

Finally, the target network parameters $\theta_{Q'}$ and $\theta_{\mu'}$ of critic and actor can be updated with the soft update method:

$$
\begin{aligned}
\theta_{Q'} &= \tau\theta_Q + (1-\tau)\theta_{Q'} \\
\theta_{\mu'} &= \tau\theta_\mu + (1-\tau)\theta_{\mu'}
\end{aligned}
\tag{24}
$$

where τ is the soft update factor, and $0 < \tau \ll 1$. The purpose is to make the target network change smoothly so as to improve the stability of learning.

3.3.2. TD3 Algorithm

The classical DDPG algorithm can have too-high Q values when calculating certain continuous actions, thus causing the bias to increase and making it difficult to find the

optimal strategy. In addition, the learning process of DDPG especially relies on the design of hyperparameters, and unreasonable hyperparameter design easily leads to unstable convergence. To address the above problems, TD3 optimizes the model structure based on the DDPG algorithm, including three aspects of shear double-Q learning, delayed policy updating, and target policy smoothing, as follows:

- Clipped double-Q learning

In order to solve the Q-value overestimation problem during the iteration of the algorithm, the TD3 algorithm optimizes two sets of critic networks, $\theta_{Q_1'}$ and $\theta_{Q_2'}$, and only the smallest value of both is selected for the target policy update during the training process. Further, the smallest Q value is selected to update the TD target in each iteration, so that the estimated Q value is closer to the actual value [28].

- Delayed policy updates

To avoid inaccurate actor network iterations due to the constantly changing Q parameters, the update frequency of the actor network is set to be lower than that of the critic network in the TD3 algorithm.

- Target strategy smoothing

Due to the function approximation error and the target variance, the Q-value calculation still suffers from overfitting. To avoid the overfitting problem, noise obeying the truncated normal distribution is added to each action.

After the above three corrections, the objective values of the two critic networks calculated by the TD3 model are optimized as follows:

$$\begin{cases} y_1 = r + \gamma Q_1'\left(s_{t+1}, \widetilde{a}|\theta_{Q_1'}\right) \\ y_2 = r + \gamma Q_2'\left(s_{t+1}, \widetilde{a}|\theta_{Q_2'}\right) \end{cases} \tag{25}$$

where y_1 and y_2 are the target values of the two Critic-target networks, Q_1' and Q_2' are the parameters of the two target networks, and $\widetilde{a}$ is the joined normally distributed noise action.

After selecting the minimum of the two target values, the loss function is calculated by bringing in the Bellman equation, as follows:

$$y_i = r_i + \gamma \min_{k=1,2} Q_k'\left(s_{i+1}, \widetilde{a}|\theta_{Q_k'}\right) \tag{26}$$

$$L\left(\theta_{Q_k}\right) = \frac{1}{M}\sum_{i=1}^{M}\left[y_i - Q_k\left(s_i, a_i|\theta_{Q_k}\right)\right]^2 \tag{27}$$

where Q_k and θ_{Q_k} are the critic current network and its parameters, respectively. After the final correction of the added action noise, the target was updated, as follows:

$$\widetilde{a} \leftarrow \mu'\left(s_{i+1}|\theta_{\mu'}\right) + \varepsilon, \varepsilon \sim clip(N(0,\sigma), -c, c), c > 0 \tag{28}$$

where ε is the noise obeying the truncated normal distribution, σ is the variance, and c is the shear amplitude.

3.4. Micro-Tillage Chassis Drive Strategy Design

3.4.1. Overall Control Strategy Design

Distributed drive systems are used in agricultural operations and should distribute the torque of all four wheels rationally under complex operating conditions to improve vehicle stability and energy efficiency. However, in a typical plowing environment, the non-linear variation of land conditions and plowing resistance will pose a great challenge for the control system to distribute the torque properly. Therefore, the torque distribution in a plowing environment is a nonlinear multi-objective optimization problem. In this study,

the torque allocation problem of the traction chassis can be abstracted as the *MDP* decision process of the traction chassis with state S, torque vector command A_t, and reward value R_t, where the states are defined as the real-time state information of the chassis, such as the power (SOC), the output power (P), the traction variable resistance (F), and the chassis operating speed (v). The torques (T_1, T_2, T_3, T_4) of the four wheels are defined as actions. They are expressed as follows:

$$S = \{SOC, P, F, v\}$$
$$A = \{T_1, T_2, T_3, T_4\} \tag{29}$$

In the study of deep reinforcement learning control strategies, the design of the reward function is directly related to the learning effect of the intelligence; the reward can guide the intelligence to explore the best control strategy in learning [29]. For the control object of this study, two rewards are introduced—*R1*: the straight-line offset rate of the traction chassis and *R2*: the reward related to energy consumption per unit time.

$$R_t(S_t, A_t) = R_1 + R_2 \tag{30}$$

$$R_1 = 2C_1 \cdot V_x - |V_y| \tag{31}$$

$$R_2 = -\left(C_2 \cdot \max \frac{V - V_\omega}{V} 100\% + C_3 |V - V_{\lim}| + C_4 \sum T_{ij} \cdot \omega_{ij} \eta_{ij}^k \right) \tag{32}$$

where the chassis is located is considered as a plane two-dimensional coordinate system; V_x is the horizontal movement distance of the chassis in the forward direction; V_y is the distance of left and right offset; V is the forward speed of the chassis; V_ω is the wheel speed; and $V_{\lim}$ is the required maximum operating speed. In order to better control the energy-saving effect of the traction chassis, the total power is calculated with $P_{ij} = T_{ij} \cdot \omega_{ij} \eta_{ij}^k$; η_{ij} is the wheel motor efficiency—when the chassis is forward, $k = 1$, when backward, $k = -1$; C1~C4 are scale factors related to the reward straight (0~1).

The framework of the TD3-based distributed micro-tillage chassis torque distribution strategy is shown in Figure 4.
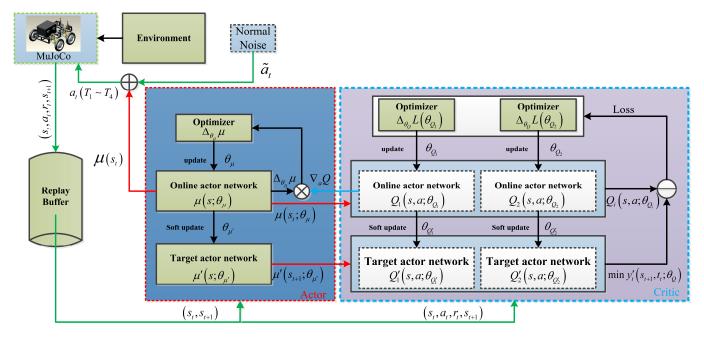

Figure 4. Framework of control strategy.

3.4.2. Intelligent Body Learning Environment Design

The objective of this study is to learn the optimal control strategy for the micro-tillage traction chassis, which is a reinforcement learning intelligence, to drive in a straight line under plowing operation. As the typical operation of this traction chassis is to mount a single-share plow for plowing in a greenhouse, the designed traction force varies in the range of 2300~2900 *N*. In this study, the environment is established by the MuJoCo simulator, and the load of the intelligent body is restricted to varying randomly within the design range by introducing a stochastic normal distribution function.

4. Results and Analysis

The proposed TD3 micro-tillage chassis-based torque distribution strategy is implemented on the Python–MuJoCo platform. The traction chassis is driven by four independent hub motors, and the intelligent body model parameters are shown in Table 1. The control strategy is validated in a straight-line driving environment for the intelligent body under plowing operation conditions. Further, in order to verify the effectiveness of the proposed control strategy, it was compared with the conventional average torque distribution strategy and the DDPG algorithm. In addition, an indoor simulation site was built to verify that the strategy is actually executable.

Table 1. Parameters of Distributed drive electric Traction chassis and the In-Wheel motor.

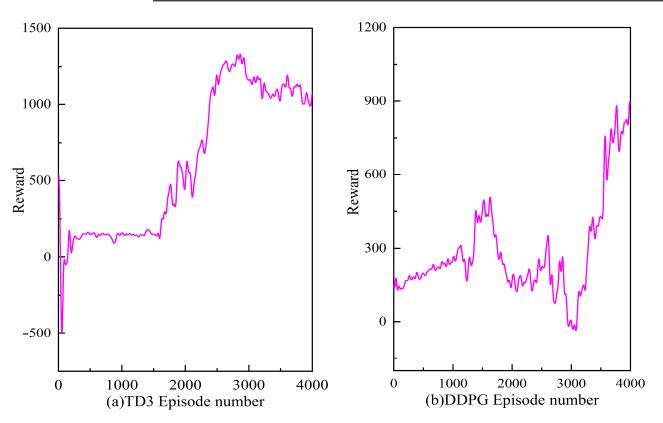| Symbol	Description	Values
M	Total Traction chassis mass	225 kg
$L \times W \times H$	Overall dimensions about chassis (Length $\times$ Wide $\times$ High)	$1.2 \times 1 \times 0.6$
r	Radius of tire	270 mm
h	Height of lift	80 mm
l	Tracking between front wheel-axle and rear wheel-axle	1.2 m
l_f	Tracking between front wheel-axle and O	0.36 m
V	Voltage of battery system	48 V
V_x	Speed of operation	5 km/h
V_{max}	Maximum driving speed	55 km/h
P	Power of motors	1.5 kW
T	In-Wheel motor torque	54 N·m

The learning environment of the intelligent body in this study is built based on the DELL T7820 workstation; in order to make the training effect better, the chassis system model is created using MuJoCo, the control strategy model is built in Python, and the working speed of the intelligent body is set to 5 km/h. In the training environment, an elliptical model approximating a single plough is dragged by the intelligent body, and the traction resistance is randomly generated by a normal distribution function with a standard deviation of 0.1. The Python toolkit versions used are shown in Table 2.

The DDPG and TD3 algorithms were trained in a simulation environment, and their learning effects are shown in Figure 5. The reward curve during the learning process shows that the DDPG algorithm is more likely to get the highest reward at the beginning (1600 episodes), but in the next iterations, the reward curve changes significantly, making the exploration of the intelligence very difficult. After 3800 episodes, the DDPG reward "trap behavior" makes the learning process unstable. The slow exploration (1600 episodes) at the beginning of the TD3 algorithm is due to the "double Q shear learning" measure introduced in the TD3 algorithm, which avoids the overestimation of the Q value at the

beginning of the exploration. The algorithm achieves the highest reward (1300 episodes) in 2800 episodes, while the reward curve is relatively flat, indicating that the algorithm has better learning stability than DDPG.

Table 2. The Python toolkit versions.

Python Toolkit	Version
gym	0.21.0
matplotlib	3.4.2
Mujoco_py	2.1.2.14
numpy	1.19.5
pandas	1.2.5
torch	1.9.0

Figure 5. Reward curve with different algorithms.

Figure 6 shows the driving speed curves of the intelligent body under the 2 algorithms, and the red line in the figure is the designed theoretical operating speed (5 km/h.) Under the DDPG algorithm, the driving speed of the intelligent body deviates from the designed speed throughout the exploration process, and the TD3 algorithm can reach the designed speed in 1700 episodes, and the driving speed tends to be stable after 2500 episodes, and the deviation from the designed speed is maintained at 6%. This traction chassis has a more stable operating speed under the TD3 algorithm, which highlights the advantages of this algorithm.

Figure 6. Speed profile with different algorithms.

Figure 7a shows the torque distribution results of the four wheels of the intelligent body under the DDPG algorithm. Although the torques of the front and rear wheels achieve symmetric distribution throughout the learning process, the torque output of the intelligent body is not stable throughout the process and fails to maintain the efficient output range of the motor, which will increase the energy consumption during the operation.
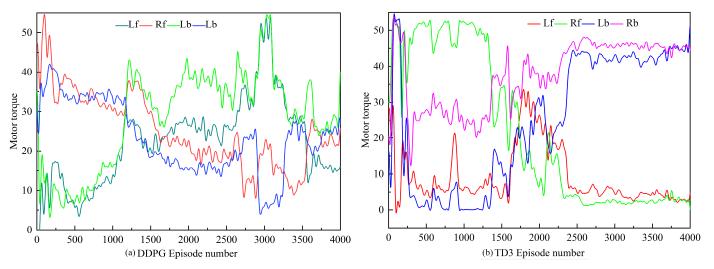

Figure 7. Motor output torque with DDPG and TD3.

Figure 7b shows the results of the torque distribution of the four wheels of the intelligent body under the TD3 algorithm. Although it will show the same instability as the DDPG algorithm until the first 2300 episodes, the torque output tends to stabilize during the learning process afterwards and stays in the efficient range of the motor. The jump in torque variation after 2500 episodes is due to the variable drag force on the spar plow under plowing operation.

Figure 8 shows the curve of the intelligent body wheel slip rate under the optimal control strategy of the DDPG algorithm and TD3 algorithm; through the curve, it can be seen that under the same external environment, the performance of the intelligent body slip rate under the optimal control strategy of DDPG is lower than that of the TD3 algorithm, and under the operating condition of 3600 s, the performance of the intelligent body under the control strategy of DDPG in the first 1800 s is still relatively smooth. Under the operating conditions of 3600 s, the first 1800 s smart body performed relatively smoothly under the DDPG control strategy, but after that, the slip rate fluctuated more, and the output torque varied drastically on the smart body. As a comparison, under the TD3 algorithm control strategy, the slip rate (the proportion of wheel slip components under operating conditions) of the smart body was maintained between 0.5% and 1% after 300 s, although the change was large at the beginning.
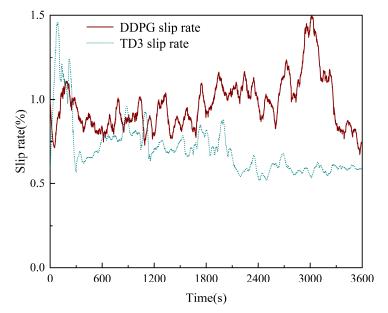

Figure 8. Slip rate with different algorithms.

5. Test Verification

5.1. Experimental Equipment and Methods

In order to further verify the executability of the proposed control strategy in terms of the energy consumption and operational straight-line retention of the tractor chassis, the proposed control strategy was compiled into the central controller of this tractor chassis, and actual operational simulations were performed. The central controller includes a microcontroller module (STM32F103ZE), a wheel motor manufactured by Faraday, and a motor controller by Asiacon (AQMD6040BLS-Ex). The data acquisition system is developed based on Labview and communicates with the central controller through CAN-USB. The whole test system is built as in Figure 9.

The test site was selected from the soil trough laboratory of Inner Mongolia Agricultural University, which had a length of 70 m and a width of 4 m. Before the experiment started, the position was well marked at the starting point of the chassis. The experiment was repeated 20 times under the condition that the mechanical parts of the chassis were connected completely, and the control system and data acquisition system were working normally, and the distance of each forward driving was 20 m, and the road was re-leveled after each test was completed to ensure the same test environment each time. Finally, the accumulated energy consumption and the average driving route yaw rate data obtained are analyzed.
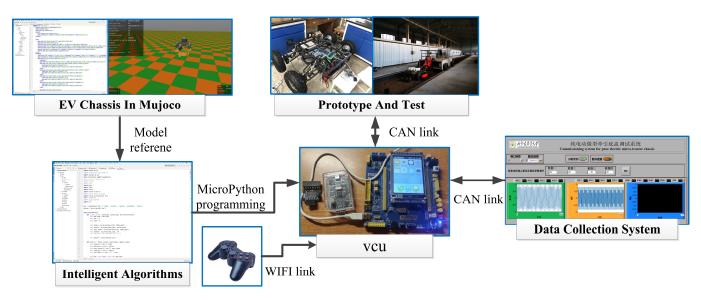
Figure 9. Soil-tank test system.

5.2. Analysis of Experimental Results

As can be seen from Figure 10, in terms of cumulative energy consumption at the end of the experiment, TD3 consumed the least amount of energy, 1.26 kWh; and the DDPG algorithm and the conventional direct torque distribution algorithm consumed 1.32 kWh and 1.68 kWh, respectively; and the TD3 algorithm proposed in this study improved the energy-saving efficiency compared with the DDPG algorithm and the conventional torque averaging algorithm by 3.7% and 10.5%, respectively. Under the conventional torque distribution algorithm, DDPG algorithm, and TD3 algorithm, the straight-line driving deflection of the traction chassis is 0.33 m, 0.24 m, and 0.21 m, respectively, and the TD3 algorithm proposed in this study further demonstrates its advantage in controlling the straight-line driving stability of the traction chassis.
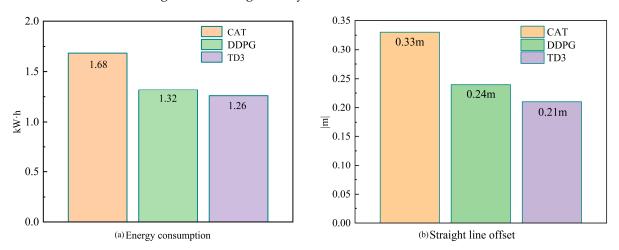

Figure 10. Energy consumption and line offset statistics.

6. Conclusions

In this work, a distributed micro-traction chassis is designed and developed based on the needs of the electric micro-traction chassis for greenhouses, and a TD3-based distributed torque distribution strategy is proposed to improve the driving stability and reduce the energy consumption of this chassis based on the micro-traction chassis. The torque distribution is denoted as MDP, which incorporates the straight-line retention and energy consumption of this chassis into the cumulative reward. The critic network and actor

network are used to approximate the action and strategy value functions, respectively. The results of this study can be summarized as follows:

1. The Actor–Critic network in the TD3 algorithm can effectively cope with the torque distribution problem of the micro-tillage traction chassis under complex operating conditions. The reward curves show that the adopted double-delay algorithm has higher learning efficiency and stability than the traditional deep deterministic policy gradient algorithm.

2. Under the TD3-based torque distribution strategy, the micro-tillage traction chassis can effectively cope with the operational requirements in complex environments and maximize the reduction of energy consumption, while maintaining the chassis in a straight line. The TD3 algorithm improves energy utilization by 3.7% and 10.5%, respectively, compared with the DDPG algorithm and the traditional average torque distribution strategy.

3. The Soil-tank experimental verification shows that the TD3 algorithm can not only reasonably distribute the driving torque of the four wheels of the micro-tillage traction chassis under plowing conditions, but also effectively suppress the wheel slip rate, maximizing energy consumption, while ensuring the straight-line driving retention rate.

4. The outdoor experiments verified the real-time executability of the control algorithm. In the future, we will continue our in-depth research to take more factors affecting torque distribution into account and conduct further experiments within the greenhouse environment.

Author Contributions: Conceptualization, methodology, writing—original draft preparation, and investigation, G.N. and L.S.; validation, formal analysis, and writing—review and editing, C.G. and Y.Z. (Yu Zhou); writing—review and editing, supervision, Y.Z. (Yong Zhang) and J.W. All authors have read and agreed to the published version of the manuscript.

Institutional Review Board Statement: Not applicable.

Data Availability Statement: Not applicable.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Cheng, C.; Fu, J.; Su, H.; Ren, L. Recent Advancements in Agriculture Robots: Benefits and Challenges. *Machines* **2023**, *11*, 48. [CrossRef]
2. Kondoyanni, M.; Loukatos, D.; Maraveas, C.; Drosos, C.; Arvanitis, K.G. Bio-Inspired Robots and Structures toward Fostering the Modernization of Agriculture. *Biomimetics* **2022**, *7*, 69. [CrossRef] [PubMed]
3. Ghobadpour, A.; Monsalve, G.; Cardenas, A.; Mousazadeh, H. Off-Road Electric Vehicles and Autonomous Robots in Agricultural Sector: Trends, Challenges, and Opportunities. *Vehicles* **2022**, *4*, 843–864. [CrossRef]
4. Bagagiolo, G.; Matranga, G.; Cavallo, E.; Pampuro, N. Greenhouse Robots: Ultimate Solutions to Improve Automation in Protected Cropping Systems—A Review. *Sustainability* **2022**, *14*, 6436. [CrossRef]
5. Idoko, H.C.; Akuru, U.B.; Wang, R.-J.; Popoola, O. Potentials of Brushless Stator-Mounted Machines in Electric Vehicle Drives—A Literature Review. *World Electr.* **2022**, *13*, 93. [CrossRef]
6. Hongxing, G.; Liqing, M. Analysis of the development status of electric micro-tiller. *South. Agric. Mach.* **2022**, *53*, 155–157.
7. Zhao, Y.; Zhang, C.; Ni, Y.; He, S.; Wen, X. Development of Multifunctional Greenhouse Agricultural Robot. In Proceedings of the 2019 2nd International Conference on Informatics, Control and Automation (ICA 2019), Hangzhou, China, 26–27 May 2019; pp. 181–186.
8. Li, Z.; Khajepour, A.; Song, J. A comprehensive review of the key technologies for pure electric vehicles. *Energy* **2019**, *182*, 824–839. [CrossRef]
9. Shuyou, Y.; Wenbo, L.; Yi, L. Steering stability control of four-wheel independent drive electric vehicles. *Control Theory Appl.* **2021**, *38*, 719–730.
10. Ani, O.A.; Uzoejinwa, B.B.; Ezeama, A.O.; Onwualu, A.P.; Ugwu, S.N.; Ohagwu, C.J. Overview of soil-machine interaction studies in soil bins. *Soil Tillage Res.* **2018**, *175*, 13–27. [CrossRef]

11. Tangarife, H.I.; Díaz, A.E. Robotic Applications in the Automation of Agricultural Production under Greenhouse: A Review. In Proceedings of the 2017 IEEE 3rd Colombian Conference on Automatic Control (CCAC), Cartagena, Colombia, 18–20 October 2017; pp. 1–6.
12. Yang, X.; Ma, Y. Current situation and development trend of vegetable mechanized seedling transplanting in facilities. *J. Agric. Mech. Res.* **2022**, *44*, 8–13+32.
13. Wu, C.; Tang, X.; Xu, X. System Design, Analysis, and Control of an Intelligent Vehicle for Transportation in Greenhouse. *Agriculture* **2023**, *13*, 1020. [CrossRef]
14. Azmi, H.N.; Hajjaj, S.S.H.; Gsangaya, K.R.; Sultan, M.T.H.; Mail, M.F.; Hua, L.S. Design and fabrication of an agricultural robot for crop seeding. *Mater. Today Proc.* **2021**, *81*, 283–289. [CrossRef]
15. Rong, J.; Wang, P.; Yang, Q.; Huang, F. A Field-Tested Harvesting Robot for Oyster Mushroom in Greenhouse. *Agronomy* **2021**, *11*, 1210. [CrossRef]
16. Shen, Y.; Zhang, B.; Liu, H.; Cui, Y.; Hussain, F.; He, S.; Hu, F. Design and Development of a Novel Independent Wheel Torque Control of 4WD Electric Vehicle. *Mechanics* **2019**, *25*, 210–218. [CrossRef]
17. Yorozu, A.; Ishigami, G.; Takahashi, M. Human-Following Control in Furrow for Agricultural Support Robot. In *IAS: 2021: Lecture Notes in Networks and Systems*; Ang, M.H., Jr., Asama, H., Lin, W., Foong, S., Eds.; Springer: Cham, Switzerland, 2021; Volume 412. [CrossRef]
18. Zhou, X.; Zhou, J. Optimization of autonomous driving state control of low energy consumption pure electric agricultural vehicles based on environmental friendliness. *Environ. Sci. Pollut. Res.* **2021**, *28*, 48767–48784. [CrossRef]
19. Ren, Q. Intelligent Control Technology of Agricultural Greenhouse Operation Robot Based on Fuzzy Pid Path Tracking Algorithm. *INMATEH-Agric. Eng.* **2020**, *62*, 181–190.
20. Cao, K.; Hu, M.; Wang, D.; Qiao, S.; Guo, C.; Fu, C.; Zhou, A. All-Wheel-Drive Torque Distribution Strategy for Electric Vehicle Optimal Efficiency Considering Tire Slip. *IEEE Access* **2021**, *9*, 25245–25257. [CrossRef]
21. Kong, H.; Fang, Y.; Fan, L.; Wang, H.; Zhang, X.; Hu, J. A novel torque distribution strategy based on deep recurrent neural network for parallel hybrid electric vehicle. *IEEE Access* **2019**, *7*, 65174–65185. [CrossRef]
22. Taherian, S.; Kuutti, S.; Visca, M.; Fallah, S. Self-adaptive Torque Vectoring Controller Using Reinforcement Learning. In Proceedings of the 2021 IEEE International Intelligent Transportation Systems Conference (ITSC), Indianapolis, IN, USA, 19–22 September 2021.
23. Qi, X.; Luo, Y.; Wu, G.; Boriboonsomsin, K.; Barth, M. Deep reinforcement learning enabled self-learning control for energy efficient driving. *Transp. Res. Part C Emerg. Technol.* **2019**, *99*, 67–81. [CrossRef]
24. Peng, H.; Wang, W.; Xiang, C.; Li, L.; Wang, X. Torque coordinated control of four in-wheel motor independent-drive vehicles with consideration of the safety and economy. *IEEE Trans. Veh. Technol.* **2019**, *68*, 9604–9618. [CrossRef]
25. Zou, Y.; Liu, T.; Liu, D.; Sun, F. Reinforcement learning-based real-time energy management for a hybrid tracked vehicle. *Appl. Energy* **2016**, *171*, 372–382. [CrossRef]
26. Srouji, M.; Zhang, J.; Salakhutdinov, R. Structured control nets for deep reinforcement learning. *arXiv* **2018**, arXiv:1802.08311.
27. Tan, H.; Zhang, H.; Peng, J.; Jiang, Z.; Wu, Y. Energy management of hybrid electric bus based on deep reinforcement learning in continuous state and action space. *Energy Convers. Manag.* **2019**, *195*, 548–560. [CrossRef]
28. Fujimoto, S.; Hoof, H.; Meger, D. Addressing function approximation error in actor-critic methods. *arXiv* **2018**, arXiv:1802.09477.
29. Wang, X.; Han, Y.; Leung, V.C.M.; Niyato, D.; Yan, X.; Chen, X. Convergence of Edge Computing and Deep Learning: A Comprehensive Survey. *IEEE Commun. Surv. Tutor.* **2020**, *22*, 869–904. [CrossRef]