Reinforcement Learning DDPG Algorithm Based Wheeled Mobility Aid Robot Control Methods

Junkai Li, Mohd Rizon Mohamad Juhari, Tiang Sew Sun
Faculty of Engineering, Technology and Built Environment, UCSI University, Kuala Lumpur 56000, Malaysia

E-mail: 1002162416@ucsiuniversity.edu.my, mohdrizon@ucsiuniversity.edu.my, tiangss@ucsiuniversity.edu.my
www. ucsiuniversity.edu.my

Abstract

When using wheeled walker robots to help individuals with limited walking ability improve their mobility, the stability of the robot's motion control and trajectory tracking accuracy are critical. In this paper, a new trajectory tracking method for wheeled walking robots is proposed by combining the Deep Deterministic Policy Gradient (DDPG) algorithm in reinforcement learning with a Proportional Integral Differential (PID) controller. The article first analyzes the kinematic model of the chassis of the wheeled walking robot, and then introduces the design principle and structure of the adaptive PID controller that combines the DDPG algorithm and the PID controller. Finally, the effectiveness of the research scheme and control strategy is verified by joint simulation experiments, and the results show that this DDPG-based PID controller can automatically adjust the parameters when tracking the trajectory to ensure the accuracy of the trajectory, and it has a strong anti-interference capability.

Keywords: Wheeled helper robot, trajectory tracking, DDPG algorithm, PID controller.

1. Introduction

In recent years, with increasing aging, the elderly show significant physiological decline, limb flexibility and other basic abilities, increasing the risk of falls. Meanwhile, as people's lifestyles change, there are more and more lower limb motor dysfunctions caused by sports injuries, traffic accidents, or diseases, making there a huge demand for intelligent assistive devices [1]. Walking robots for the elderly and patients with lower limb motor dysfunction can be categorized into two types: exoskeleton robots [2] and wheeled assisted walking robots [3]. However, the center of gravity control of most exoskeleton rehabilitation robots is not satisfactory, and the safety of the rehabilitation process cannot meet the requirements [4]. Wheeled walking robot itself has a stable chassis, not only can well avoid the patient in the training process due to the center of gravity is not stable and caused by the fall and other secondary injuries, but also in a narrow space to achieve any direction of movement, to give the patient a good sense of spatial movement and real walking experience, wheeled walking robot with its stable chassis and flexible spatial movement ability, in terms of safety and practicality with the Advantages.

When using wheeled walking robots for rehabilitation training, due to the complexity of the external environment, the wheeled walking robots are required to follow the desired trajectory as much as possible, which requires the robots to be able to track an ideal trajectory with a time function according to the actual position and motion state, and complete the tracking of each point in the trajectory according to the time requirements Therefore, this paper applies the reinforcement learning algorithm to the traditional proportional-integral-derivative (PID) controller to realize adaptive PID

parameter tuning in order to adjust the robot position more accurately so that the robot can complete trajectory tracking.

Although the traditional PID control has the advantages of simplicity of use, easy to implement, no static error, etc., its disadvantage is that it cannot realize the online adjustment of parameters, so when it encounters a strong interference, it is bound to have the phenomenon of prolonged recovery time and increased overshooting, which affects the stability of the motion of the chassis of wheeled robots. A large number of scholars have conducted research on adaptive PID, and introduced the idea of online parameter adjustment in the traditional PID, which improves the response speed of the system. At present, adaptive PID control methods mainly include: fuzzy PID controller [5], the method requires a large amount of a priori knowledge, there are parameter optimization problems; neural network-based adaptive PID control [6], the method can be achieved without identifying the complex nonlinear system to achieve effective control, but in the use of supervised learning to optimize the parameters, the acquisition of the teacher's signals is relatively difficult; evolutionary algorithm adaptive PID controller [7], the method, although the acquisition of a priori knowledge of the lower requirements, but the computation time is longer, it is difficult to achieve real-time control in practical applications; reinforcement learning adaptive PID controller [8], proposed the Actor-Critic algorithm to achieve adaptive tuning of PID parameters, which makes use of AC algorithms of model-free online learning The algorithm utilizes the model-free online learning capability of AC algorithm, but the convergence speed of AC algorithm is slow and the training time is long.

In this paper, we adopts the deep deterministic policy gradient (DDPG) algorithm, which is based on the Actor-Critic (AC) framework, to enhance the performance of the PID controller and the deep Q-network (DQN) algorithm is added on the basis of the deterministic policy gradient (DPG) algorithm. This algorithm can not only update in a single step like DQN, but also has the advantages of high data utilization and fast convergence of DPG. In order to realize the adaptive adjustment of PID parameters, a PID controller based on DDPG algorithm is proposed. In the simulation experiments, the kinematic model of the omnidirectional chassis of the wheeled walking robot in our laboratory is used to verify the superiority and generality of the proposed method.

2. Proposed Method

This section will introduce the kinematic model of the wheeled walking robot chassis. Then, it will describe the design principles and structure of the adaptive PID controller that combines the DDPG algorithm and the PID controller.

2.1. *Kinematic modeling of assistive robots*

The schematic and kinematic model of the robot's omnidirectional wheel chassis is shown in Fig. 1. Taking the world coordinate system $x_w o_w y_w$ as a reference, the robot moves in the motion coordinate system xoy with a velocity of magnitude v toward an angle α from the Y_w axis. V_x, V_y denote the horizontal and vertical travel speeds of the robot relative to the motion coordinate system, respectively [9].

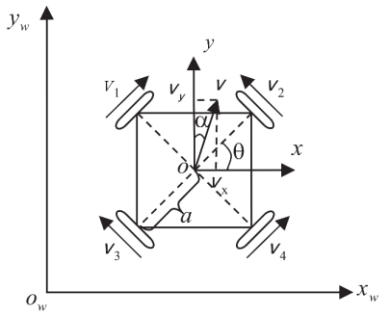

Fig. 1 Kinematic modeling of assistive robots

When the robot is moving in any direction, e.g., at an angle α from the y-direction with velocity v, the magnitudes of the horizontal and vertical velocities of the robot respectively:

$$\begin{aligned} V_x &= v sin\alpha \\ V_y &= v cos\alpha \end{aligned} \tag{1}$$

Let the angle between the omnidirectional wheels and the x-axis, measured from the center of the robot chassis, be θ ,and the distance from the center of the chassis to the four wheels be a , then the wheel speed of each omnidirectional wheel is degree is:

$$\begin{aligned} V_1 &= -sin\theta V_x + cos\theta V_y + \dot{\theta} a \\ V_2 &= \ \ sin\theta V_x + cos\theta V_y - \dot{\theta} a \\ V_3 &= \ \ sin\theta V_x + cos\theta V_y - \dot{\theta} a \\ V_4 &= -sin\theta V_x + cos\theta V_y + \dot{\theta} a \end{aligned} \tag{2}$$

Where V_1, V_2, V_3, V_4 denote the linear velocities of the four omni-directional wheels respectively and $\dot{\theta}$ denotes the angular velocity of the robot writing Eq. (1) in matrix form gives:

$$\begin{bmatrix} V_1 \\ V_2 \\ V_3 \\ V_4 \end{bmatrix} = \begin{bmatrix} -sin\theta & cos\theta & a \\ cos\theta & sin\theta & a \\ -sin\theta & cos\theta & a \\ cos\theta & sin\theta & a \end{bmatrix} \quad (3)$$

Let the transformation matrix R be:

$$R = \begin{bmatrix} -sin\theta & cos\theta & a \\ cos\theta & sin\theta & a \\ -sin\theta & cos\theta & a \\ cos\theta & sin\theta & a \end{bmatrix} \quad (4)$$

Let the positional attitude of the robot at $Y_w O_w X_w$ be $P = [x, y, \theta]^T$, and the differential of this positional attitude be $\dot{P}$, and the velocities of each of the robot's wheels$V_b = [V_1, V_2, V_3, V_4,]$, then the equations of kinematics of the robot's differential omnidirectional wheels are:

$$\dot{P} = R^{-1} \cdot V_b \quad (5)$$

2.2. *Incremental PID Control Principle*

Digital PID control can be categorized into two types: positional PID and incremental PID. incremental PID does not require the use of the accumulated value of past deviations, which effectively reduces the system's computational error. Incremental PID is an algorithm that realizes PID control by controlling the increment. The formula is as follows:

$$u(t) = u(t-1) + K_i(t)e(t) + K_p\Delta e(t) + K_d(t)\Delta^2 e(t) \quad (6)$$

$$\begin{aligned} e(t) &= y_d(t) - y(t) \\ \Delta e(t) &= e(t) - e(t-1) \\ \Delta^2 e(t) &= e(t) - 2e(t-1) + e(t-2) \end{aligned} \quad (7)$$

In these equations, $y_d(t)$ denotes the current actual signal value, $y(t)$ denotes the current system output value of the current system, $e(t)$ denotes the output error of the system, $\Delta e(t)$ denotes the first difference of the error, and $\Delta^2 e(t)$ denotes the second difference of the error.

The use of incremental PID controllers in control system design can optimize the use of computational resources. The method relies only on the last three sampling points to determine the control increments, thus reducing the computational burden and storage requirements. This facilitates fast training of the DDPG algorithm and storage of sample data. In addition, the incremental PID controller ensures stable operation in the event of a system failure, and since its output is the changing value of the control quantity, it makes the rewards obtained in a reinforcement learning environment more stable, which in turn accelerates the convergence of the algorithm.

In summary, the incremental PID algorithm is used to realize the trajectory tracking control of the wheeled robot, which requires the design of two PID controllers, i.e., the transverse position X controller and the longitudinal position Y controller, and the block diagram of the control system is shown in Fig. 2. The input of the transverse PID controller is x_e , i.e., the transverse deviation of the wheeled robot system; the output is v_x, the transverse speed of the wheeled robot system. the output of the PID controller goes to the speed distribution controller, which calculates the speeds of the four omnidirectional wheels according to Eq. (2) to correct the transverse deviation of the wheeled robot. The principle of longitudinal position Y control is the same as the principle of transverse position X control, i.e., y_e is the longitudinal error, v_y is the longitudinal velocity.

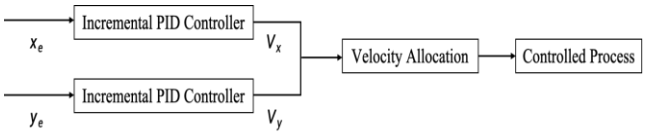

Fig.2 Block diagram of two PID controller systems

2.3. *DDPG algorithm*

Reinforcement learning is a machine learning method that learns behavioral strategies through trial and error, while deep learning is a machine learning method that performs high-level abstraction and feature extraction through multi-layer neural networks. Deep Reinforcement Learning combines the feature extraction capability of Deep Learning with the decision-making capability of Reinforcement Learning to autonomously learn and improve the performance of decision-making strategies. DDPG (Deep Deterministic Policy Gradient) algorithm is an optimization algorithm based on the Actor-Critic algorithm [5] framework, which is able to better select the optimal policy in continuous actions such as robot movement. The DDPG algorithm is based on deterministic policy gradient and refers to the experience pool replay mechanism and Double-Depth Q-Network (DDQN) objective network method to update the network parameters and realize the self-tuning of PID parameters, and its algorithm structure is shown in Fig. 3.

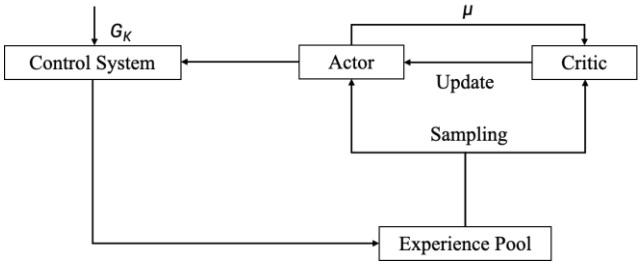

Fig.3. Block diagram of DDPG algorithm structure

Actor gives a new parameter μ according to the strategy network, denoted as $a_t = \mu(S_t|\theta^\mu) + N_t$, where θ^μ denotes the parameter of the neural network, G_k denotes the noise.After one update iteration, the state changes from S_t to S_{t+1}, and obtains the reward R_t. (S_t, A_t, S_{t+1}, R_t) is stored in the experience pool. Finally, samples are drawn from the experience pool to train the strategy network and evaluation network.

The strategy network and evaluation network are updated as shown in Fig. 4. The actor and the Critic consist of two identical networks, denoted as: actor evaluation network $\mu(s|\theta^\mu)$, actor estimation network $\mu' = (s_t|\theta^{\mu'})$, Critic evaluation network $Q(s, a|\theta^Q)$, Critic estimation network $Q'(s, a|\theta^{Q'})$.

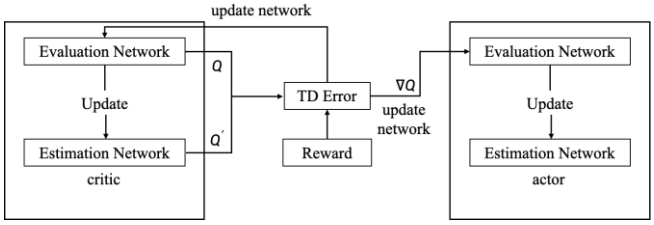

Fig.4. updating network parameters

The actor evaluation network updates the parameters according to the following objective function:

$$\mu = \frac{1}{N}\sum_t \left(\nabla_a Q(s, a|\theta^Q)|_{S=s_t, A=\mu(S_t)} \cdot \nabla_{\theta^\mu}\mu(s|\theta^\mu)|_{S=s_t}\right) \quad (8)$$

where $\nabla_a Q(s, a|\theta^Q)$ is the gradient of the Critic evaluation network, and the gradient is updated by computing the loss function. The loss function is computed as follows:

$$L = \frac{1}{N}\sum_t \left(\mathrm{y}_t - Q(s_t, a_t|\theta^Q)\right)^2 \quad (9)$$

where y_t is referred to as the TD target, which consists of an immediate reward and an estimated discounted reward. The formula is as follows:

$$\mathrm{y}_t = \mathrm{r}_t + \gamma Q'\left(s_{t+1}, \mu'\left(s_{t+1}|\theta^{\mu'}\right)|\theta^{Q'}\right) \quad (10)$$

Then update the Critic evaluation network according to $\nabla_a Q(s, a|\theta^Q)$, and at the same time pass $\nabla_a Q(s, a|\theta^Q)$ to the actor, update the actor evaluation network according to Eq. (9), and finally update the parameters of the target network as follows:

$$\begin{cases} \theta^{Q'} \leftarrow \tau\theta^Q + (1-\tau)\theta^{Q'} \\ \theta^{\mu'} \leftarrow \tau\theta^\mu + (1-\tau)\theta^{\mu'} \end{cases} \quad (11)$$

2.4. *General Structure of Adaptive PID Controller Based on DDPG Algorithm*

The design idea is to combine the DDPG algorithm on the basis of incremental PID controller, so the structure design is shown in Fig. 5.

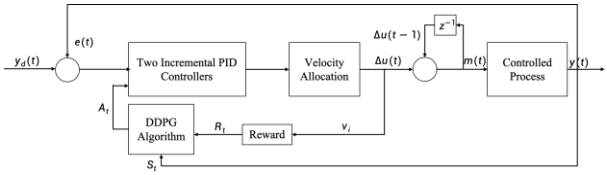

Fig.5. adaptive PID controller based on DDPG algorithm

The system block diagram is based on the traditional PID with the addition of closed-loop control by the DDPG algorithm. The speed allocation module allocates the speeds of the four wheels according to Eq. (2), and the scores are obtained through a self-designed reward function, and output to the DDPG algorithm. According

to the state obtained by the system output, the DDPG algorithm continuously carries out trial and error training until it selects the action group with the highest score, which is the optimal PID parameter, and outputs it to the two PID controllers, adaptively adjusts the parameter, and the error under the parameter is minimized, and the parameter outputs it to the speed allocation module, and then outputs the speeds of the four wheels to the wheeled walking robots' chassis system, which realizes the trajectory tracking control. The controller uses the parameter vector of the PID controllers, i.e. the 6 parameters to be tuned by the two PID controllers, as the action space.

Critics rate each reinforcement learning cycle using the system's state variables and the defined reward function R_t to generate the TD error $\delta_{TD}(t)$ and evaluate the value function Q_t, where $\delta_{TD}(t)$ is provided directly to the actor and Critic, and the reward R_t is used to evaluate the quality of the current behavior.

$$\begin{cases} R_1 = -0.00001 \\ R_2 = -10, e(t) \geq 1 \times 10^{-5} \\ R_3 = 0, e(t) < 1 \times 10^{-5} \\ R_4 = -200, e(t) > 0.5 \\ R_t = R_1 + R_2 + R_3 + R_4 \end{cases} \tag{12}$$

R_1 is the reward for the speed of the four wheels, which is set to a score that has less impact on the system since the main reward comes from the error rather than the amount of control. R_2 and R_3 denote the scores when the error is lower or higher than the error tolerance interval, respectively, and R_4 denotes the score of -200 when the system error is more than 0.5. Finally, the four scores are summed up to form the final reward function. Based on the continuous trial-and-error training, the maximum reward value is obtained, and thus the optimal PID parameter values are obtained.

3. Simulation experiment

In this paper, trajectory tracking control simulation experiments are carried out in python environment. Based on the chassis modeling of the wheeled walking robot, the system simulation model was established in Simulink. In the simulation process, the initial position of the controlled object is set as [0,1], the desired trajectory is set as a cosine curve along the x-axis, and the sampling time is set as 1ms. The parameters of the adaptive PID control based on the DDPG algorithm are listed in Table 1 and Table 2.

Table 1. DDPG Algorithm Parameters

Parameter	Value
Actor learning rate	0.005
Critic learning rate	0.001
Discount rate	0.99
Soft update parameter	0.001
Total training steps	50000
Experience pool capacity	1000000

Table 2. PID controller Parameters

Parameter	Value
K_{p1}	5.0
K_{p2}	5.0
K_{i1}	2.0
K_{i2}	2.0
K_{d1}	0.2
K_{d2}	0.2

4. Result

Experiment shows the simulation results of trajectory tracking realized by traditional PID controller in the Fig. 6. The solid line shows the desired trajectory and the dashed line shows the tracked trajectory. It can be seen that the method can also realize the trajectory tracking, the overshoot is 4.5%, and the tracking trajectory is far away from the target trajectory, and the error is large. Fig. 7 shows the simulation results of trajectory tracking control using the PID controller based on DDPG algorithm proposed in this paper. The overshoot of this method is 2.7%, and the trajectory is closer to the target trajectory with less error than the traditional PID control.

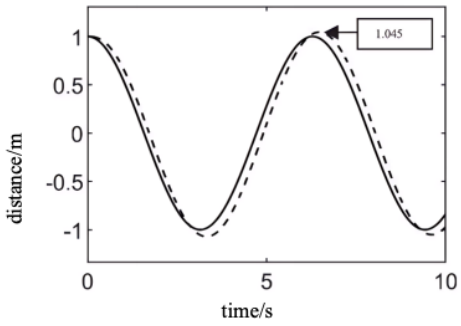

Fig.6 Traditional PID for trajectory tracking

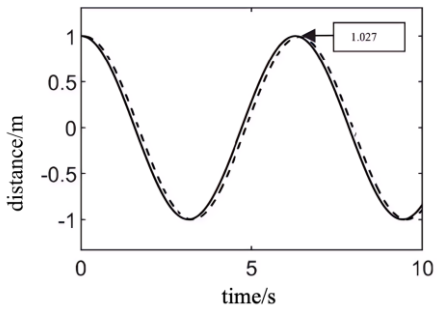

Fig.7 PID control for trajectory tracking based on DDPG algorithm

Compared with the traditional PID controller, the maximum error of this controller is reduced by 48.12%, as listed in Table 3. In addition, the controller's overshoot is reduced by 40% compared to a traditional PID controller, which improves user safety during assisted walking and training.

Table 3. Comparison of results

Controller	Maximum error	Overshoot
Traditional PID	0.2097	4.5%
DDPG+PID	0.1088	2.7%

Comparative results show that the controller is able to track the desired trajectory more accurately when the user uses the wheeled mobility aid robot for rehabilitation training. It also has a strong anti-interference ability due to the trial-and-error mechanism of reinforcement learning, which increases the safety and comfort of training.

5. Conclusion

In order to ensure that the elderly and patients with lower limb dysfunction can move accurately according to the pre-set desired trajectory when they use wheeled walking robots for rehabilitation training in complex environments, this paper proposes a reinforcement learning method that combines the DDPG algorithm with a PID controller. This method not only solves the problem that the traditional PID cannot adjust the parameters online, but also reduces the storage space requirement of the controlled system, thus reducing the computation time of the system. Simulation results show that the PID control based on the DDPG algorithm has the advantages of high tracking accuracy, small overshooting amount and strong adaptability, which enables the wheeled walking robot to realize accurate trajectory tracking control. The method has good generality and generalization.

References

1. Hou Zeng-Guang, Zhao Xin-Gang, Cheng Long, Wang Qi-Ning, Wang Wei-Qun. Recent Advances in Rehabilitation Robots and Intelligent Assistance Systems. ACTA AUTOMATICA SINICA, 2016
2. Zhou, Jinman, Shuo Yang, and Qiang Xue. "Lower limb rehabilitation exoskeleton robot: A review." Advances in Mechanical Engineering 13.4 ,2021
3. Neuhaus, Peter, and H. Kazerooni. "Design and control of human assisted walking robot." Proceedings 2000 ICRA. Millennium Conference. IEEE International Conference on Robotics and Automation. Symposia Proceedings (Cat. No. 00CH37065). Vol. 1. IEEE, 2000
4. Nokata M, Ikuta K, Ishii H. 11 Safety Evaluation Method of Rehabilitation Robots. In: BIEN Z Z, STEFANOV D. Advances in Rehabilitation Robotics[M]. Lecture Notes in Control and Information Science, 2006, 306: 187-198.
5. Savran A. A Multivariable Predictive Fuzzy PID Control System [J]. Applied Soft Computing,2013, 13(5): 2658-2667.
6. Chen J H, Huang T C. Applying Neural Networks to on-line Updated PID Controllers for Nonlinear Process Control [J]. Journal of Process Control,2004, 14(2): 211-230.
7. Saad M S, Jamaluddin H, Darus I Z M. Implementation of PID controller tuning using differential evolution and genetic algorithms [J]. International Journal of Innovative Computing, Information and Control, 2012, 8(11): 7761-7779.
8. Pomerleau A, Desbiens A, Hodouin D. Development and evaluation of an auto-tuning and adaptive PID controller [J]. Automatica, 1996, 32(1): 71-82.
9. Mollaret C, Mekonnen A A, Lerasle F, et al. A multi-modal perception based assistive robotic system for the elderly [J]. Computer Vision and Image Understanding, 2016, 149: 78-97.

Authors Introduction

Mr. Junkai Li

He is currently pursuing Doctor of Philosophy (Engineering) in Faculty of Engineering, Technology and Built Environment, UCSI University, Malaysia. His research interests are machine learning, optimization algorithm and optical measurement.

Prof. Dr. Mohd Rizon Mohamad Juhari

He is a Professor in Faculty of Engineering at UCSI University in Malaysia. He received his PhD in Engineering from Oita University, Japen in 2002. His research interests are face analysis, pattern recognition and vision for mobile robot.

Assistant Professor Ir Ts Dr Tiang Sew Sun

She is an Assistant Professor in Faculty of Engineering at UCSI University in Malaysia. She received her PhD in Electrical and Electronic Engineering from Universiti Sains Malaysia in 2014. Her research interests are optimization and antenna design.