



Self-Tuning Parameters of a Maglev Control System Based on Q-Learning

Yang Wang¹, Yougang Sun^{2*}, Wen Ji³, Junqi Xu³

¹ Institute of Rail Transit, Tongji University, 201804 Shanghai, China

² Key Laboratory of Railway Industry of Maglev Technology, National Railway Administration of P. R. C., 201804 Shanghai, China

³ Maglev Transportation Engineering R&D Center, Tongji University, 201804 Shanghai, China

* Correspondence: Yougang Sun (1989yoga@tongji.edu.cn)

Received: 04-06-2024

Revised: 06-01-2024

Accepted: 06-15-2024

Citation: Y. Wang, Y. G. Sun, W. Ji, and J. Q. Xu, "Self-tuning parameters of a maglev control system based on Q-learning," *Mechatron. Intell Transp. Syst.*, vol. 3, no. 2, pp. 135–145, 2024. <https://doi.org/10.56578/mits030205>.



© 2024 by the author(s). Published by Acadlore Publishing Services Limited, Hong Kong. This article is available for free download and can be reused and cited, provided that the original published version is credited, under the CC BY 4.0 license.

Abstract: Maglev transportation, as an innovative mode of rail transit, is regarded as an ideal future transportation system due to its wide speed range, low noise, and strong climbing ability. However, the maglev control system faces challenges such as significant nonlinearity, open-loop instability, and multi-state coupling, leading to issues like insufficient tuning and susceptibility to environmental influences. This paper addresses these problems by investigating the self-tuning parameters of a maglev control system using Q-learning to achieve optimal parameter tuning and enhanced dynamic system performance. The study focuses on a basic levitation unit modeled after the simplified control system of an electromagnetic suspension (EMS) train. A Q-learning reinforcement learning environment and Q-learning agent were established for the levitation system, with a forward "anti-deadlock" reward function and discretization of the action space designed to facilitate reinforcement learning model training. Finally, a Q-learning-based method for self-tuning the parameters of the maglev control system is proposed. Simulation results in the Python environment demonstrate that this method outperforms the Linear Quadratic Regulator (LQR) control method, offering better control effects, improved robustness, and higher tracking accuracy under system parameter perturbations.

Keywords: Maglev control system; Proportional-Integral-Derivative (PID) control; Reinforcement learning; Python simulation

1 Introduction

The EMS-type maglev train [1, 2] achieves contactless support, guidance, traction, and partial braking through the electromagnetic attraction or repulsion generated by the interaction of coil currents. It also acquires electrical energy through magnetic induction without relying on a pantograph. This breaks the speed limit imposed by the traditional wheel-rail adhesion relationship, enabling higher speeds. At present, China has successfully completed the prototype and joint debugging of the 600 km/h high-speed maglev system [3], marking a new stage in the research of high-speed maglev transportation in China.

As an indispensable part of ensuring the stable operation of the EMS-type high-speed maglev train, the performance of the maglev control system directly determines the safety and comfort of the train [4]. Research on control systems and their algorithms began in the mid to late 20th century. In 2002, a composite control model combining PID with Fuzzy control is raised [5], which accomplishes PID parameter self-adjusting by employing Fuzzy reasoning, and shows great control effects. Cho and Kim [6] designed a fuzzy PID controller, which combines a traditional PID controller with a fuzzy controller. Based on the characteristics of system deviation, direction, and trend, decisions are made according to Teaching-Learning optimization to automatically adjust PID parameters. To compensate for the mismatch defect of PID control in nonlinear systems, Wang and Zhang [7] puts forward the theory of variable universe fuzzy PID algorithm-by selecting the right expansion factor to realize domain contraction, which can increase the fuzzy control rules, so as to improve the control precision. Additionally, in terms of explaining the PID control parameter adjustment rules for the maglev system, Ryan and Corless [8] proved that for a subclass of uncertain systems, a Maglev suspension control system, the zero state can be (a) rendered 'practically' stable (in

the sense that, given any neighbourhood of the zero state, there exists a control, in the proposed class of continuous feedback controls, which guarantees global uniform ultimate boundedness with respect to that neighbourhood), or (b) rendered globally uniformly asymptotically stable by the proposed discontinuous feedback control.

Moreover, using modern control theory, scholars have proposed a series of nonlinear methods that can solve the stable suspension of the maglev system based on the premise of knowing the accurate model of the maglev system [9–11]. Tang and Yang [12] introduced a boundary variable power Fal function to replace the Sign function and associated the sliding mode gain and hyperplane function to eliminate system chattering instability in their improved sliding mode control algorithm based on Linear Extended State Observer (LESO). Hu et al. [13] improved the suspension performance of the system using Extended State Observer (ESO) technology combined with model reference sliding mode control and improved reaching law sliding mode control strategies. Additionally, this type of nonlinear control method can also improve the disturbance rejection performance of the maglev control system, ensuring suspension performance in the event of sensor failure. For example, Long et al. [14] propose designing Auto-Disturbance-Rejection (ADRC) control algorithm in train automatic operation system. From analyze of simulation of tracing the Automatic Train Operation curve, it shows that the this ADRC algorithm can meet the needs of maglev train automatic control, increasing the system's disturbance rejection.

In the era of intelligence, research in artificial intelligence has gradually penetrated the field of maglev control [15]. To reject the disturbance and parameter perturbations, an adaptive neural-fuzzy sliding mode controller is presented [16], which employs a sliding mode control, adaptive-fuzzy approximator, and the neural-fuzzy switching law, whose controller significantly reduces the impact of the disturbance and parameter perturbations, showing good control system performance in simulations and experiments. Zhao et al. [17] used deep reinforcement learning methods, utilizing a double Q-learning scheme to reduce estimation errors in the Q function and used a PID controller to accelerate convergence, achieving suspension control and optimization based on a medium and low-speed maglev train dataset.

Therefore, for the maglev train suspension control system, the future development trend is bound to be towards high intelligence, excellent robustness, and good maintainability. Conducting adaptive reinforcement learning research on self-tuning control system parameters is of certain significance for consolidating maglev technology reserves and developing a new generation of highly intelligent high-speed maglev trains.

2 Levitation System Modeling

As shown in Figure 1, the topology of the maglev system of the EMS-type maglev train consists of four units: control unit, drive unit, motion unit, and feedback unit. These units can complete the levitation functions from lifting off to levitating and then to maintaining stability. A carriage of the EMS-type maglev train is composed of several levitation bogies, each of which consists of four basic levitation units, as shown in Figure 2. It is not difficult to find that there is consistency in functionality between the topological structure and the basic levitation unit, and the study of the basic levitation unit is representative to a certain extent [18].

Assumptions: Ignore the possible magnetic leakage in the electromagnet windings; the magnetic field generated by the electromagnet is uniform in the action area and conforms to infinite magnetic permeability; ignore adverse phenomena such as magnetic field saturation, hysteresis, and residual magnetism in electromagnetic phenomena:

$$L(c, i) = \frac{N}{i(t)} \varphi_T = \frac{\mu N^2 A}{2c(t)} \quad (1)$$

where, $L(c, i)$ represents the inductance value corresponding to the determined coil current i and the determined gap value c ; N is the number of turns of the electromagnet winding; φ_T is the main pole magnetic flux; A is the core area; μ is the vacuum permeability; $i(t)$ is the control current at time t ; $c(t)$ represents the gap value at time t .

Then combining the inductance formula with the relationship between magnetic induction intensity and main pole magnetic flux, we have:

$$B \approx \frac{\varphi_T}{A} \quad (2)$$

$$F_e(i, c) = \frac{\mu N^2 A}{4} \left(\frac{i(t)}{c(t)} \right)^2 \quad (3)$$

where, $F_e(i, c)$ represents the electromagnetic attraction corresponding to the determined coil current i and the determined gap value c .

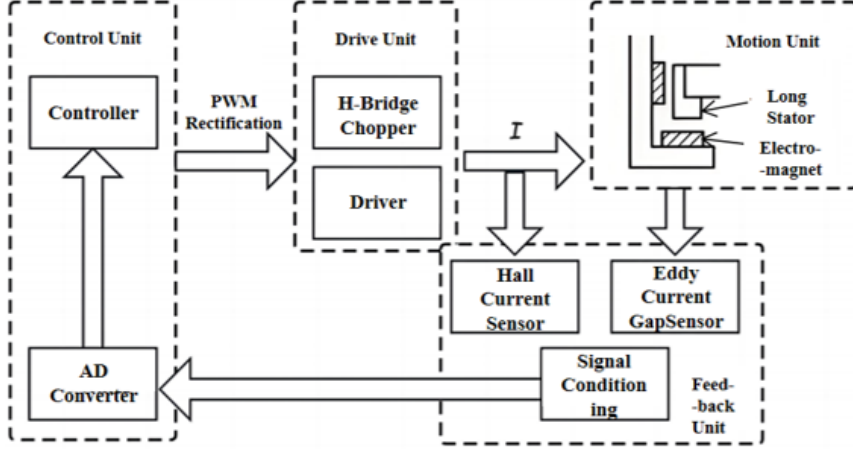


Figure 1. Topology of the high-speed maglev system

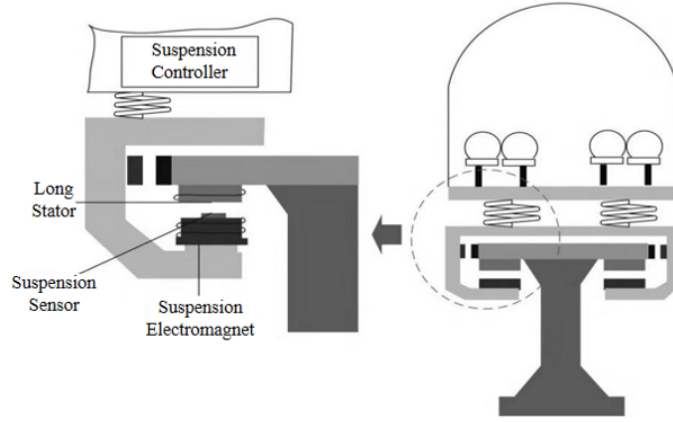


Figure 2. Topology of the high-speed maglev system [19, 20]

In the vertical direction, according to Newton's second law, the kinematic equation of the levitation electromagnet can be expressed as:

$$m \frac{d^2 c(t)}{dt^2} = mg + f_d(t) - F_e(i, c) \quad (4)$$

where, m represents the weight allocated to a single levitation unit, including the weight of the vehicle body; g represents the local gravitational acceleration; $f_d(t)$ represents the external force disturbance at time t .

At the equilibrium position, there is:

$$\begin{cases} m \frac{d^2 c(t)}{dt^2} = mg + f_d(t) - \frac{\mu N^2 A}{4} \left(\frac{i(t)}{c(t)} \right)^2 \\ F_e(i_0, c_0) = mg + f_d(t) \end{cases} \quad (5)$$

Ignoring the disturbance term and performing approximate linearization, we have the Taylor expansion at the equilibrium point, ignoring higher-order terms:

$$\begin{cases} \frac{d^2 c(t)}{dt^2} = k_1 (c - c_0) - k_2 (i - i_0) \\ k_1 = \frac{\mu N^2 A i_0^2}{2m c_0^3}; k_2 = \frac{\mu N^2 A i_0}{2m c_0^2} \end{cases} \quad (6)$$

The open-loop transfer function of the entire system is:

$$G(s) = \frac{X(s)}{Y(s)} = \frac{-k_2}{s^2 - k_1} \quad (7)$$

Obviously, the characteristic equation of the system has a zero point on the right half-plane of s , making the system open-loop unstable and requiring external feedback control.

3 Design of the Levitation Control System Parameter Tuning Algorithm Based on Q-Learning

Reinforcement learning is a type of learning that originates from the field of artificial intelligence. Its core idea is to enable an agent to gradually learn how to make optimal or near-optimal decisions in a complex environment through a series of trial-and-error processes, guided by a specific reward function [16, 21]. Q-learning is a type of reinforcement learning that is based on the temporal difference method. By using the maximization of the subsequent state's Q value as the target value, it iteratively updates the Q function and ultimately converges to the optimal Q function to obtain the optimal policy [22]. The parameter training framework of the maglev control system based on reinforcement learning is shown in Figure 3.

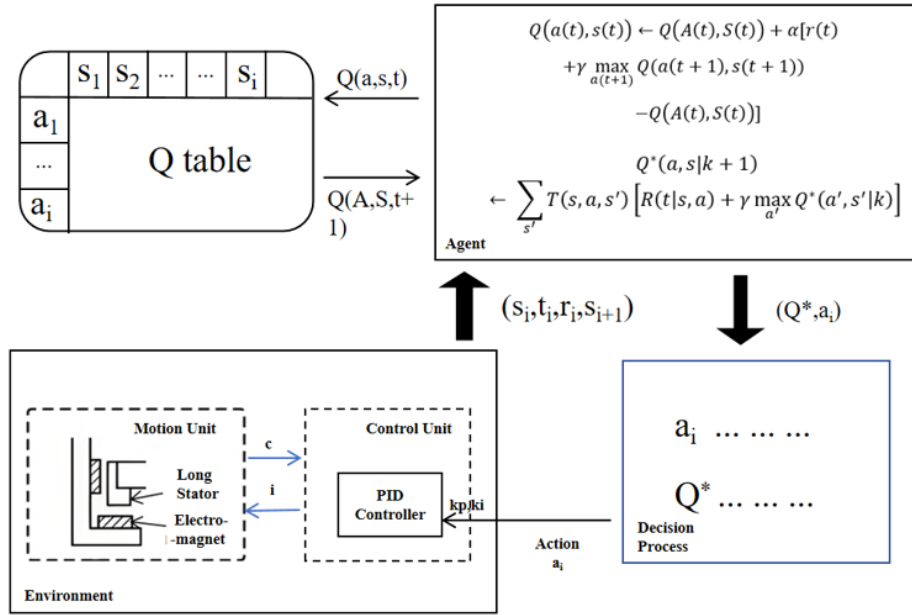


Figure 3. Framework of maglev control system parameter training based on reinforcement learning

3.1 Q-Learning

3.1.1 Q-function

$Q(U, a | s)$ represents the expected total sum of future cumulative reward functions obtained by executing all subsequent actions using policy U after taking action a under state s .

3.1.2 Value function

$V(U | s)$ represents the expected total sum of future cumulative reward functions obtained by executing a series of actions using policy U under state s .

3.1.3 Bellman equation relationship

Naturally, we know from the Bellman equation that $Q(U, a | s)$ and $V(U | s)$ satisfy:

$$V(U | s) = \sum_a U(a | s) \sum_{s'} T(s, a, s') [R(s, a) + \gamma V(U | s')] \quad (8)$$

$$Q(U, a | s) = \sum_{s'} T(s, a, s') \left[R(s, a) + \gamma \sum_{a'} U(a' | s') Q(U, a' | s') \right] \quad (9)$$

3.1.4 Q-learning algorithm

As previously mentioned, the Q-learning algorithm is a learning interaction process based on the optimal Q-function. Its update strategy can be derived from Eqs. (8) and (9):

$$Q(a(t), s(t)) \leftarrow Q(A(t), S(t)) + \alpha \left[r(t) + \gamma \max_{a(t+1)} Q(a(t+1), s(t+1)) - Q(A(t), S(t)) \right] \quad (10)$$

In the equations, α represents the learning rate, which quantifies the impact of each learning iteration; γ is the discount factor, which measures the importance of future rewards.

At the same time, the optimal Q-function $Q^*(a, s)$ also satisfies the Bellman equation, and its transformation relationship is:

$$Q^*(a, s | k+1) \leftarrow \sum_{s'} T(s, a, s') \left[R(t | s, a) + \gamma \max_{a'} Q^*(a', s' | k) \right] \quad (11)$$

In the equations, k represents the number of iterations, and s' and a' represent the values of S and A at the next time step in the iteration.

3.2 Self-Tuning Algorithm for Levitation Control Parameters Based on Q-Learning

3.2.1 Agent setup

For the single-point maglev model agent, its operating range must have definite upper and lower limits, including but not limited to the upper and lower limits of speed $c(t)$, the upper and lower limits of levitation gap $c(t)$, the upper limit of adjustment time t_s (mechanism model time limit), and the upper and lower limits of control current i , etc.

The following values are specified:

$$c(t) = \begin{cases} 0, \dot{c}(t) < 0, Fe < 0 \\ 0, \dot{c}(t) > c_{\max}, Fe > 0 \\ \dot{c}(t), \text{ others} \end{cases} \quad (12)$$

$$\begin{cases} 0 < c(t) < 0.0024 \\ 0 < t_s < t_{s \max} \\ 0 < i < i_{\max} \end{cases} \quad (13)$$

Additionally, the action space for the self-tuning algorithm of levitation control parameters based on Q-learning meets:

$$\begin{cases} Kp \in \left[i_1, \frac{i_1}{n} (Kp_{\max} - Kp_{\min}) \right]; i_1 \in [1, n] \\ Ki \in \left[i_2, \frac{i_2}{n} (Ki_{\max} - Ki_{\min}) \right]; i_2 \in [1, n] \\ Kd \in \left[i_3, \frac{i_3}{n} (Kd_{\max} - Kd_{\min}) \right]; i_3 \in [1, n] \end{cases} \quad (14)$$

The action space here cannot be simply explained as the values or changes of Kp , Ki , and Kd , but rather the sum of the default initial values and the changes. It is important to note that during the Q-learning process, there is an expected cumulative reward value for the reward function. To express this value, the action space in Q-learning must ignore the default initial value to estimate the impact of its change on the cumulative expectation of the reward function and the effect of increasing the change on the cumulative expectation of the reward function.

For the reward function, we cannot only calculate the expected gap value because we require the overall steady-state performance of the system. Therefore, the reward function $r(i, s)$ is expressed as follows. To avoid the ‘‘deadlock phenomenon’’ caused by negative reward functions, which may lead to poor convergence during training, a positive reward function is adopted here. It uses an exponential form to expand the positive reward function [14], reducing or avoiding fatigue phenomena in reinforcement learning.

$$r(i, s) = w^{k_c(c-c_o)^2} + k_s w^{t_s} + k_o w^{0_c} \quad (15)$$

where, O_c represents the overshoot value under a set of control parameters. t_s is the adjustment time. w is a positive coefficient, an important parameter for designing the positive function; k_c , k_s , and k_o are the reward coefficients for the error value, adjustment time, and steady-state error, respectively. Among them, k_o is a value less than 0 because the smaller the overshoot, the better the steady-state performance.

3.2.2 Basic levitation unit Gym environment model

The Python mechanism modeling of the single-point maglev control system should be divided into the following steps:

(1) Construct a PID controller (simulator). When given the controller parameters (K_p , K_i , K_d) and the system's previous action (speed, position information), the PID's calculation output value can be fed back under this set of parameters.

Gym Model Code Based on the Single-Point Levitation Mathematical Model
Class single-levitation Functions:
Initialize internal parameters, reset supervisor parameters
Define input functions, input controller parameters K_p , K_i , K_d
Design PID controller, calculate PID output at time step
Define step operation function, discretize single-point levitation model
Define supervision function, supervise step operation time and effect
Define reward function for direct calling
Define reset function to reset internal parameters for next input
Define render function to graphically display some set values
Define close function to ensure resource utilization
Output: PID instability bool, system rise time, adjustment time, overshoot, steady-state error, reward function value

Q-Learning-Based Self-Tuning Algorithm for Levitation Control Parameters
Initialize the Q-table and reward function, reset the environment.
Randomly assign initial control parameters $Km = [K_p, K_i, K_d]$.
For each episode step do
Randomly initialize the environment
For each environment step do
Based on the state $s_t(Km)$ obtained in the previous episode, randomly select action $a_t(Km)$ according to the exploration rate.
Execute action $a_t(Km)$ in the mechanistic environment.
Obtain reward $r_t(i, s)$, episode end signal g_t , and the next state s_{t+1} .
Store (s_t, Km, r_t, s_{t+1}) into the Q-table.
If g_t indicates the end of the episode or exceeds the maximum steps W , end the episode.
End for
End for
For gradient update steps (according to Bellman equation) do
Select different Q-table update measures based on (s_t, Km, r_t, s_{t+1}) information.
End for
Output: Policy network U

(2) Construct a numerical feedback model in Python for the single-point maglev mechanism as a single environment step. After obtaining the PID calculation output value at this time, complete the output feedback of the single-point maglev mathematical system and output the system's next action information (speed, position information).

(3) Construct a monitor and termination program with certain monitoring capabilities. Under stable system conditions, output the steady-state indicators of the single-point levitation system mechanism model under given controller parameters, including rise time, adjustment time, overshoot, and steady-state error. Meanwhile, under unstable conditions, output a calibration condition to prevent the system from encountering singularities during reinforcement learning.

For convenience in subsequent research and development and standardizing the code, the Gym model framework based on neural learning cases was referenced when modeling the single-point levitation model. The code implementation is shown in the above figure.

The construction of the Gym environment is mainly based on Eq. (5). Given the system parameters, based on the transfer function relationship between the control quantity (current) and the controlled quantity (gap), Python solver tools simulate the environment feedback during a given single environment duration. It provides real-time transient information such as gap value and speed and dynamic performance information like system stability indicators, adjustment time, and steady-state error. The Q-learning system in this paper mainly uses dynamic performance information for training and simulation.

3.2.3 Implementation of maglev control parameter self-tuning based on Q-learning

To implement the model functions, in addition to the single-point levitation mechanism model, it is necessary to design a training algorithm for the maglev PID control model based on Q-learning. The code is shown in the above figure. It should be noted that there is an exploration rate in the Q-learning algorithm, which was not mentioned earlier. This is because even with a positive reward function to prevent deadlock, the input-output relationship between the system and the reward function and the Q function cannot be purely linear, so “potential traps” exist. The reward function setting cannot fully guarantee crossing these “potential traps”, where a certain non-optimal control parameter’s Q value and the corresponding Q values within the positive and negative action range during Q-learning may both be greater. However, this does not indicate that the control parameter is optimal. The existence of the exploration rate effectively solves the problem of “potential traps”.

4 Simulation Results Validation and Analysis

To validate the performance of the Q-learning-based tuning levitation control system, this method was compared with the traditional LQR method for control performance. The Q-learning algorithm and the traditional LQR method were compared under conditions of initial startup to stable levitation and system load change.

Key parameters in the Q-learning model include:

Total environment training update steps: 100000 times, maximum steps per episode: $W=1200$. Discrete PID sampling time: 0.001 s, discretization coefficient of control parameters: $n=500$.

Due to the program design, the total training steps in the environment cannot be infinitely large, so the steps were derived based on preliminary simulation and estimation.

The parameters used for comparison with LQR control are $Q = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 150 \end{bmatrix}$, $R = 4$, step time: 0.001 s.

Here, the parameter Q represents the state weighting matrix, where its value can relatively reflect the speed at which the LQR control system’s state approaches the target value; R is the control weighting matrix, where the larger the value, the slower the system’s state decays.

Additionally, the basic physical quantities referenced in the algorithm are shown in Table 1.

Table 1. Reference values for basic physical quantities required by the algorithm model

Parameter	Unit	Symbol	Reference Value
Single track load mass	Kg	M	750
Coil turns	-	N	270
Target gap	m	c_0	0.008
Electromagnet area	m^2	A	0.115
Vacuum permeability	H/m	μ	$4\pi \times 10^{-7}$
Equilibrium current	A	i_0	19.1

4.1 Simulation Training Results

(1) Steady-state levitation conditions Here, we only considered the first 20,000 environment update steps. The variation images of the Action (i.e., coefficients in Km) are shown in Figure 4. It can be observed that at approximately 9,800 steps and 8,000 steps, K_p and K_i have both reached a nearly stable value, namely:

$$KpQ = 65400, KiQ = 1522 \quad (16)$$

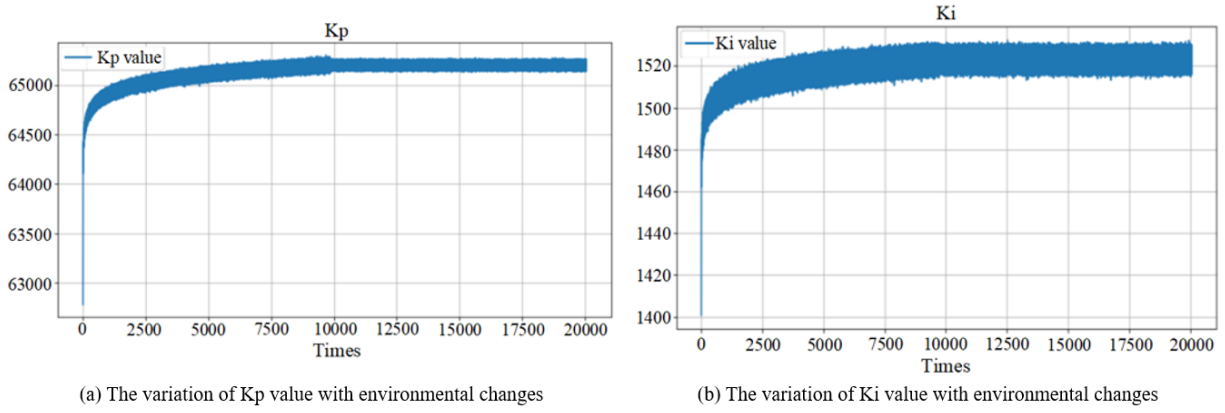


Figure 4. Variation of system control parameters with environmental changes

Additionally, it should be noted that the Kd parameter was not considered in Q-learning and was set to 0 by default.

(2) Simulation Training results under system load variation

For changes in system load, we modify the average load M to $M'=820$ kg as described above. After training the Q-learning reinforcement learning model, the output values of Km' are shown in Figure 5.

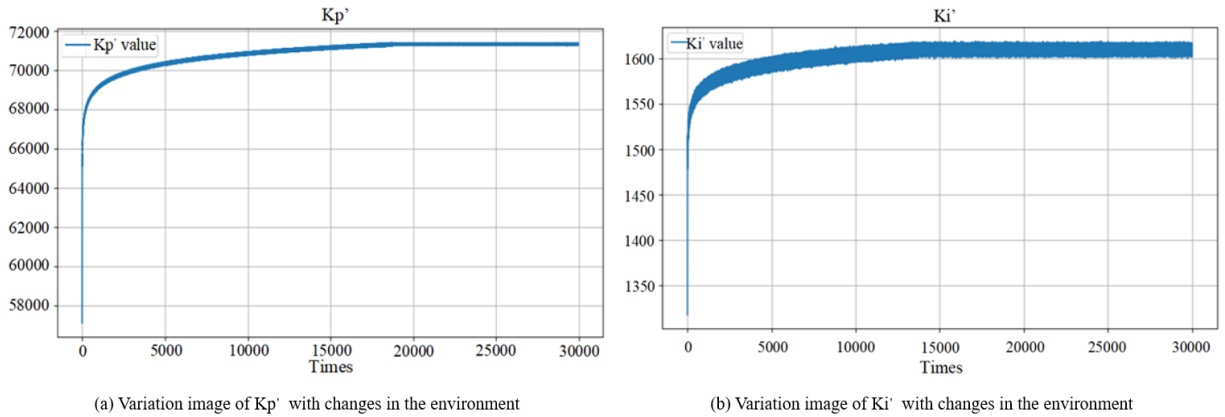


Figure 5. System control parameters with changes in the environment under varying load conditions

It can be seen that due to the change in system load, the environment stabilization steps increase. At approximately 17,500 steps and 13,000 steps, Kp' and Ki' have both reached a stable learning value:

$$Kp' = 71320, Ki' = 1610 \quad (17)$$

4.2 Comparison with Traditional LQR Algorithm

(1) Static levitation

Figure 6 presents the gap values and velocity (gap derivative) values for Q-learning.

It is evident that the Q-learning PID achieves a steady-state adjustment time t of 0.17 s, compared to the 0.9 s adjustment time for LQR control, representing an improvement rate of 81.1%. Moreover, the slope of the rise curves for the two control systems indicates that the Q-learning PID responds much faster than the LQR control system, which is also visually reflected in the peak sizes of b.

It is clear from the figure that the Q-learning strategy is more effective and reliable. The control parameters reflected by the Q-learning PID control strategy show more efficient characteristics in steady-state levitation. Under the condition of satisfying control size constraints, the control current effect is utilized to achieve faster levitation and stabilization.

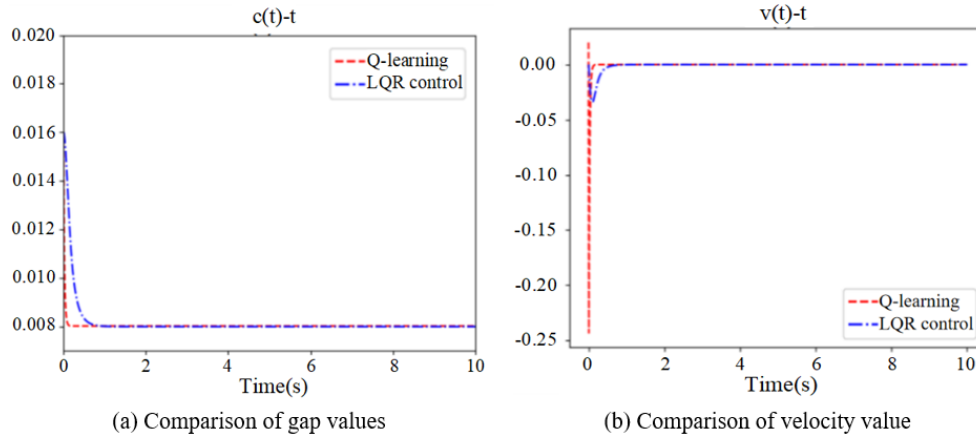


Figure 6. The steady-state levitation comparison between Q-learning PID and LQR control

(2) System load variation

During the operation of magnetic levitation trains, the load on the bogie changes continuously due to the boarding and alighting of passengers. Figure 7 compares the levitation performance of Q-learning PID and LQR control when the system load changes from 750 kg to 820 kg.

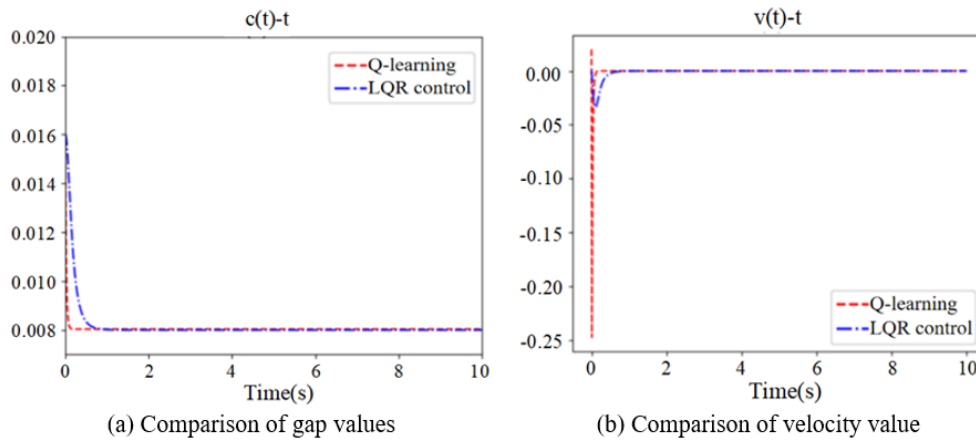


Figure 7. Performance comparison of Q-learning PID and LQR control when system load changes

It can be observed that when the system load changes, the LQR control method experiences a decrease in control capability, with an increased rise time and a significantly smaller curve slope compared to the slope of the curve obtained after retraining the Q-learning reinforcement learning model.

From the information in Figure 7, it is evident that due to the lack of reconfiguration of LQR system parameters, the adjustment time increases significantly from 0.9 s to approximately 1.2 s, representing a growth rate of 33.3%. This indicates that when using an LQR controller, adjustments to the QR values are necessary during system load changes. For the Q-learning PID control algorithm, the adjustment time only increases from 0.17 s to approximately 0.20 s after load changes. The system control parameters exhibit strong robustness and adaptability to system parameter perturbations.

(3) System parameter perturbations

During operation, magnetic levitation trains might encounter vertical perturbing forces when passing through tunnels and other environments. Figure 8 compares the system performance of Q-learning PID control and LQR control under an external perturbing force of 1 m/s^2 .

In subgraph (a) of Figure 8, it can be seen that LQR control results in a significant gap variation when a constant external force F is applied, changing from 0.008 m to 0.020 m, which exceeds the control limit range. Furthermore, LQR control cannot correct the gap error, which remains at 0.020 m, indicating an imbalanced condition. In contrast, the Q-learning PID system performance is significantly better than LQR, with only about 0.7 mm of error.

This indicates that under the same external force, the effect of acceleration changes due to Q-learning control parameters is much smaller compared to LQR control.

In summary, the Q-learning PID control method demonstrates better robustness and faster response speed

compared to LQR control. For situations involving external load changes or external forces, which result in significant perturbations to system parameters, the Q-learning-based magnetic levitation system parameter self-tuning PID control exhibits superior dynamic performance, robustness, and adaptability compared to LQR control.

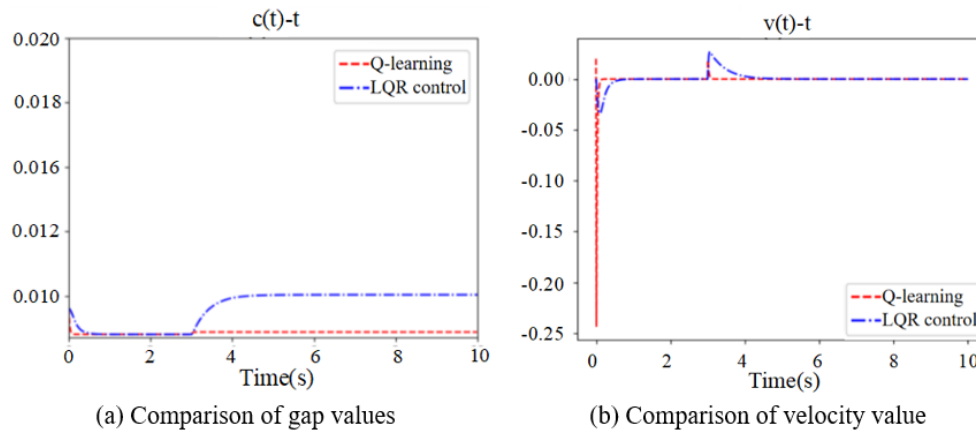


Figure 8. Performance comparison of Q-learning PID and LQR control under an external perturbing force of 1 m/s^2

5 Conclusion

For the problem of parameter tuning in magnetic levitation control systems and the associated issues, a self-tuning method based on reinforcement learning Q-learning was proposed. A standardized Gym reinforcement learning environment for the basic levitation unit was established, a forward “anti-deadlock” reward function was set, and a Q-learning algorithm path was used to find the optimal Q function. A Q-learning algorithm for self-tuning magnetic levitation control system parameters under discretized action space of control parameters was designed.

Compared to traditional LQR control methods, the proposed Q-learning PID control demonstrated superior performance. The steady-state levitation time for Q-learning PID control is only 18.9% of that for LQR control; during system load changes, the rise time for Q-learning PID control is 29.8% of that for LQR control; and under external perturbations, the steady-state error of Q-learning PID control is only 5.8% of that for LQR control. This confirms that the Q-learning-based magnetic levitation control system parameter self-tuning research can achieve optimization of control system parameters and dynamic performance. It provides the magnetic levitation control system with excellent parameter tuning capability, improved interference resistance, and robustness.

As the current Q-learning magnetic levitation system control parameter self-tuning algorithm is a simplified model and does not consider coupling situations or the self-tuning of K_d parameters, further detailed research will be conducted in the next phase to address these issues and achieve more outstanding self-tuning of magnetic levitation system control parameters.

Data Availability

The data used to support the findings of this study are available from the author upon request.

Conflicts of Interest

The author declares no conflicts of interest regarding this work.

References

- [1] H. W. Lee, K. C. Kim, and J. Lee, “Review of maglev train technologies,” *IEEE Trans. Magn.*, vol. 42, no. 7, pp. 1917–1925, 2006. <https://doi.org/10.1109/TMAG.2006.875842>
- [2] R. Nian, J. Liu, and B. Huang, “A review on reinforcement learning: Introduction and applications in industrial process control,” *Comput. Chem. Eng.*, vol. 139, p. 106886, 2020. <https://doi.org/10.1016/j.compchemeng.2020.106886>
- [3] Z. Deng, Z. Liu, H. Li, and W. Zhang, “Development status and prospect of maglev train,” *J. Southwest Jiaotong Univ.*, vol. 57, no. 3, pp. 455–474, 2022. <https://doi.org/10.3969/j.issn.0258-2724.20220001>
- [4] F. Ni, S. Mu, J. Kang, and J. Xu, “Robust controller design for maglev suspension systems based on improved suspension force model,” *IEEE Trans. Transport. Electrification*, vol. 7, no. 3, pp. 1765–1779, 2021. <https://doi.org/10.1109/TTE.2021.3058137>

- [5] D. S. Liu, L. M. Yin, and L. H. She, "Application of fuzzy-PID algorithm in maglev system," *Comput. Autom. Meas. Control*, vol. 10, no. 6, pp. 375–377, 2002. <https://doi.org/10.3969/j.issn.1671-4598.2002.06.011>
- [6] J. H. Cho and Y. T. Kim, "Design of TLBO-based optimal fuzzy PID controller for magnetic levitation system," *Trans. KIEE*, vol. 66, no. 4, pp. 701–708, 2017. <https://doi.org/10.5370/KIEE.2017.66.4.701>
- [7] X. L. Wang and G. M. Zhang, "Magnetic levitating bearing control system simulation based on variable universe fuzzy pid algorithm," *Instrum. Tech. Sens.*, no. 12, pp. 144–147, 2012. <https://doi.org/10.3969/j.issn.1002-1841.2012.12.049>
- [8] E. P. Ryan and M. Corless, "Ultimate boundedness and asymptotic stability of a class of uncertain dynamical systems via continuous and discontinuous feedback control," *IMA J. Math. Control Inf.*, vol. 1, no. 3, pp. 223–242, 1984. <https://doi.org/10.1093/imamci/1.3.223>
- [9] F. Ni, S. Mu, J. Kang, and J. Xu, "Robust controller design for maglev suspension systems based on improved suspension force model," *IEEE Trans. Transp. Electr.*, vol. 7, no. 3, pp. 1765–1779, 2021. <https://doi.org/10.1109/TTE.2021.3058137>
- [10] J. Xu, Y. Du, Y. H. Chen, and H. Guo, "Adaptive robust constrained state control for non-linear maglev vehicle with guaranteed bounded airgap," *IET Control Theory Appl.*, vol. 12, no. 11, pp. 1573–1583, 2018. <https://doi.org/10.1049/iet-cta.2017.1348>
- [11] Y. G. Sun, S. Xie, J. Q. Xu, and G. B. Lin, "A robust levitation control of maglev vehicles subject to time delay and disturbances: Design and hardware experimentation," *Appl. Sci.*, vol. 10, no. 3, p. 1179, 2020. <https://doi.org/10.3390/app10031179>
- [12] Y. Tang and J. Yang, "Improved sliding mode magnetic levitation control algorithm based on LESO," *J. Ordn. Equip. Eng.*, vol. 44, no. 11, pp. 262–269, 2023. <https://doi.org/10.11809/bqzbgcxb2023.11.033>
- [13] K. Hu, Q. Jiang, C. Ji, J. Yang, and F. Li, "Model reference sliding mode control of magnetic levitation system based on improved ESO," *J. Eng. Des.*, vol. 29, no. 1, pp. 82–91, 2022. <https://doi.org/10.3785/j.issn.1006-754X.2022.00.004>
- [14] Z. Long, Y. Li, and X. Wang, "On maglev train automatic operation control system based on auto-disturbance-rejection control algorithm," in *2008 27th Chinese Control Conference*, Kunming, China, 2008, pp. 681–685. <https://doi.org/10.1109/CHICC.2008.4605180>
- [15] G. Cui, B. Cai, B. Su, and X. Chu, "Radial basis function neural network-based adaptive sliding mode suspension control for maglev yaw system of wind turbines," *Proc. Inst. Mech. Eng., Part I: J. Syst. Control Eng.*, vol. 236, no. 1, pp. 66–75, 2022. <https://doi.org/10.1177/09596518211022068>
- [16] Y. Sun, J. Xu, H. Qiang, and G. Lin, "Adaptive neural-fuzzy robust position control scheme for maglev train systems with experimental verification," *IEEE Trans. Ind. Electron.*, vol. 66, no. 11, pp. 8589–8599, 2019. <https://doi.org/10.1109/TIE.2019.2891409>
- [17] F. Zhao, K. You, S. Song, W. Zhang, and L. Tong, "Suspension regulation of medium-low-speed maglev trains via deep reinforcement learning," *IEEE Trans. Artif. Intell.*, vol. 2, no. 4, pp. 341–351, 2021. <https://doi.org/10.1109/TAI.2021.3097313>
- [18] Y. Liu, K. Fan, and Q. Ouyang, "Intelligent traction control method based on model predictive fuzzy PID control and online optimization for permanent magnetic maglev trains," *IEEE Access*, vol. 9, pp. 29 032–29 046, 2021. <https://doi.org/10.1109/ACCESS.2021.3059443>
- [19] K. Hu, J. Zu, Z. Liu, and G. Lin, "Reinforcement learning-based suspension control for electromagnetic suspension maglev trains," *J. Tongji Uni.*, vol. 51, no. 3, pp. 332–340, 2023. <http://doi.org/10.11908/j.issn.0253-374x.23006>
- [20] L. Beauloye and B. Dehez, "Permanent magnet electrodynamic suspensions applied to MAGLEV transportation systems: A review," *IEEE Trans. Transp. Electrification*, vol. 9, no. 1, pp. 748–758, 2022. <https://doi.org/10.1109/TTE.2022.3193296>
- [21] Y. Sun, J. Xu, G. Lin, W. Ji, and L. Wang, "RBF neural network-based supervisor control for maglev vehicles on an elastic track with network time delay," *IEEE Trans. Ind. Inform.*, vol. 18, no. 1, pp. 509–519, 2020. <https://doi.org/10.1109/TII.2020.3032235>
- [22] H. Pan, H. Wang, C. Yu, and J. Zhao, "Displacement-constrained neural network control of maglev trains based on a multi-mass-point model," *Energies*, vol. 15, no. 9, p. 3110, 2022. <https://doi.org/10.3390/en15093110>