IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received 7 August 2023, accepted 16 August 2023, date of publication 21 August 2023, date of current version 1 September 2023.

Digital Object Identifier 10.1109/ACCESS.2023.3307131

RESEARCH ARTICLE

The Optimization of RBFNN Gearshift Controller Parameters for Electric Vehicles Using PILCO Reinforcement Learning

YANWEI LIU[1], JINGLONG ZHANG[1], ZIYUAN LV[1], AND JIE YE[2]
[1]School of Electromechanical Engineering, Guangdong University of Technology, Guangzhou 511006, China
[2]School of Mechatronics Engineering, Foshan University, Foshan 528225, China

Corresponding author: Jie Ye (yejie@fosu.edu.cn)

This work was supported in part by the Natural Science Foundation of Guangdong Province under Grant 2022A1515012080, and in part by the National Science Foundation of Guangdong Province under Grant 2019A1515110562.

ABSTRACT In order to improve the efficiency of gearshift controller parameter optimization and obtain a good gearshift controller control effect, this article proposes an optimization method for electric vehicle transmission gearshift controller, and selects dual clutch transmission as the research object that establishes an 11-degree-of-freedom gearshift dynamics model and a feedforward-feedback gearshift control model. The feedforward control is chosen from the target trajectory given by the Legendre pseudo-spectral approach, and the feedback controller is a Gaussian kernel radial basis function neural network controller. The feedback controller performs parameter optimization by the Probabilistic Inference for Learning Control (PILCO) reinforcement learning algorithm to obtain a control strategy that matches the actual gearshift conditions. By comparing how well the main/secondary moving disk can follow the target trajectory during various optimization iterations, it is verified that the algorithm requires only a few experiments to complete the optimization, and the optimized Radial Basis Function Neural Network (RBFNN) control has a better control effect by comparing the results of different iterations. Applying the learned controller to various slope and load circumstances yields data that demonstrate that all can have an obvious optimization effect with good robustness. Additionally, the reinforcement learning technique suggested in this research can be used for various gearshift controller parameter optimization to assist engineers and technicians in increasing the effectiveness of their Research and Development.

INDEX TERMS Electric vehicle, dual clutch transmission, gearshift control, reinforcement learning.

I. INTRODUCTION

Electric vehicles have the advantages of zero emissions and minimal noise while in operation, and market demand is rising continuously. The transmission control strategy's optimization has a significant impact on how well electric vehicles perform. The transmission is usually controlled by a feedforward controller and a feedback controller. The feedback controller [1], for instance, can adjust the torque signal online depending on the disturbance in the process of shifting so that the whole vehicle operation status remains relatively stable and plays a vital role.

Usually, the optimization of an automotive transmission controller's parameters requires real physical experiments conducted on an experimental bench. In traditional controller parameter optimization, some researchers use the Design-of-Experiments (DOE) method [2], while others propose online adaptation or calibrate the automatic transmission using a powertrain test bench (PTTB) [3], [4], [5] or iterative learning control (ILC) using a hybrid system [6]. However, these methods require engineers to collect experiment data for determining the optimal controller parameters, resulting in high time costs. Therefore, more efficient optimization techniques are needed to reduce the number of experiments required and improve optimization results, warranting further investigation.

The associate editor coordinating the review of this manuscript and approving it for publication was Amin Mahmoudi.

In addition to the optimization algorithm, a suitable controller is also needed to control the gearbox. There are various options for the design of feedback controllers, which have been studied more extensively by domestic and foreign scholars. In [7], dynamic programming method is used to formulate the gearshift operation as a multi-stage decision process, but dynamic programming is often computationally intensive and difficult to apply in real time. In [8], a robust transmission tracking control algorithm is proposed, and a robust feedforward feedback controller is designed. Model errors in the model used by the controller can affect the performance and robustness of the controller. In [9] and [10], a finite time linear quadratic regulator is proposed, but this will lead to high computational complexity. In addition, two independent PID controllers are used in [11]. But the parameter adjustment of PID controller often takes a lot of time. In [12], a backstepping controller based on optimal control input is designed to provide a stable feedback method. The controller has high requirements on the system model and is sensitive to initial conditions. In [13], the controller use a robust multivariable control scheme with H-infinite cyclic shaping, which is difficult to adjust parameters and requires an accurate system model. In [14], a gearshift controller is designed by using the backstepping method, and H∞ robust control is introduced to further improve the adaptive effect of the controller, but there is still a problem that the model uncertainty is difficult to deal with. In [15], the coarse-id control framework is used to solve the dynamic unknown LQR problem, but this still requires a large amount of data and the robustness needs to be improved.

Therefore, it can be seen that the existing research mainly faces problems such as long training time, the need for accurate models, low control accuracy and poor robustness. The development of neural networks and reinforcement learning in recent years has provided additional options for designing feedback controllers [16]. Many researchers have explored the use of these techniques for various aspects of car shifting, such as those in references [17], [18], [19]. Reinforcement learning algorithms can be divided into model-based and model-free categories depending on whether they use a model. Model-based algorithms utilize data obtained through environment interaction to learn an environmental model, which tends to be more efficient than model-free algorithms. Thus, this article considers combining model-based reinforcement learning with neural networks to achieve gearshift control optimization.

To achieve this, this article propose using a Radial Basis Function Neural Network (RBFNN) based on Gaussian function as a feedback controller and a Probabilistic Inference for Learning Control (PILCO) reinforcement learning algorithm to optimize the parameters of the RBFNN controller [20]. The feedback controller part of this article uses the RBFNN, RBFNN is better than the BP neural network in terms of classification ability and learning speed because the parameter initialization has a certain method, not random initialization, and so on, and has a simple structure, simple training, fast learning convergence speed, and can approximate any nonlinear function and overcome the problem of local minima [21], so applied to gearshift control can achieve better torque control. The PILCO algorithm learns a probabilistic transfer model based on the gearshift dynamics model, incorporating model errors, environmental perturbations, and other uncertainties into the modeling process. This approach enables precise and efficient control optimization with only a few experiments [22], [23]. Previous studies have applied PILCO to different engineering disciplines, including an autopilot underwater vehicle control problem [24], [25], constant force control of a robot surface [26], autonomous optimization of PID parameters in the control of a flight attitude simulator [27], and experiments on a planetary gear transmission shift console frame [28]. These studies have shown that PILCO can obtain the ideal controller strategy for the gearbox with a relatively small number of experiments.

This research aims to improve the accuracy and robustness of electric vehicle transmission shift, enhancing shift quality and driving comfort while reducing the cost and time associated with parameter optimization. To achieve this, innovative work has been conducted, including proposing the RBFNN controller as the DCT shift feedback controller and introducing a new method using the PILCO algorithm for optimizing its performance. Comparative experiments and analysis have demonstrated the effectiveness and efficiency of the PilCO-based RBFNN controller, making it a promising alternative to traditional control methods. Overall, this research provides valuable guidance for future research and implementation of improved control strategies in various industrial applications.

This article is structured into five parts. Part I provides an introduction to the development status of electric vehicle gearshift control optimization and analyzes the applications of RBFNN and PILCO algorithms. In Part II, the Dual Clutch Transmission (DCT) is modeled, including both the dynamics model and the gearshift control model. In Part III, the working principle of the selected RBFNN feedback controller is analyzed and the PILCO algorithm is explained in detail, including the algorithm principle and the updating process of the RBFNN controller parameters. In Part IV, the experimental conditions are elucidated, and the PILCO algorithm is verified to be efficient in optimizing the RBFNN through a comparison with a PID controller. Additionally, the robustness of the algorithm is verified by applying the learned controller to different operating conditions. Finally, Part V concludes with key takeaways from this article.

II. GEARSHIFT CONTROL ANALYSIS AND MODELING

A. WORKING PRINCIPLE OF GEAR SHIFTING

The structure sketch is shown in Fig. 1, which mainly consists of clutch 1, clutch 2, first gear pair, second gear pair, input shaft, and output shaft. The process of upshifting goes through two phases, the torque phase, and the inertia phase, and the change of speed and torque in this process is shown in Fig. 2. When the transmission is in first gear, clutch 1 is engaged, and in the process of shifting from first to second

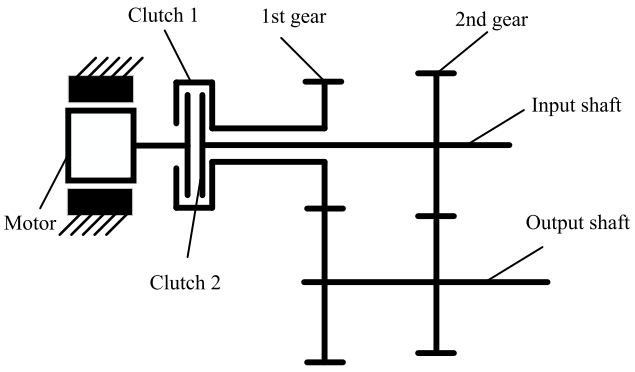
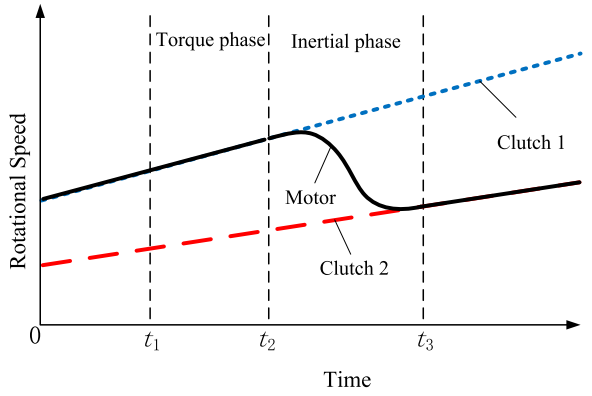
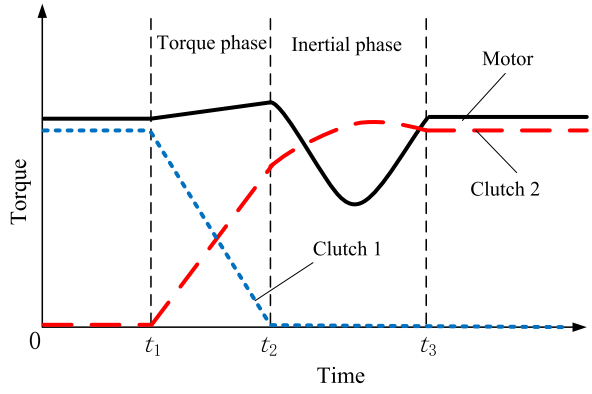


FIGURE 1. Two-speed DCT structure diagram.



(a) Rotational speed trajectory.



(b) Torque trajectory.

FIGURE 2. Diagram of speed and torque trajectory of upshift.

gear, the transmission torque of clutch 1 gradually decreases until it reaches zero and separates, and the transmission torque of clutch 2 gradually rises and at the end of the inertia phase, the primary and secondary rotational speeds of clutch 2 are synchronized, at which time clutch 2 locks and the gearshift ends.

B. 11-DEGREE-OF-FREEDOM DYNAMICS MODEL

To improve the accuracy of the model, the stiffness and damping of the key components of the driveline, such as the motor, clutch, half-shaft, and wheels, are taken into account according to the electric vehicle driveline configuration, and a more detailed model of the entire vehicle driveline can be obtained, as shown in Fig. 3. The resulting 11-degree-of-freedom dynamics model for gearshift control optimization can be obtained as follows:

$$J_m\ddot{\theta}_m = T_m - k_1\left(\theta_m - \theta_{clu}\right) - c_1\left(\dot{\theta}_m - \dot{\theta}_{clu}\right), \tag{1}$$

$$J_{clu}\ddot{\theta}_{clu} = k_1\left(\theta_m - \theta_{clu}\right) + c_1\left(\dot{\theta}_m - \dot{\theta}_{clu}\right) - T_{cl1} - T_{cl2}, \tag{2}$$

$$J_1\ddot{\theta}_1 = T_{cl1} - k_2\left(\theta_1 - \theta_{g1A}\right) - c_2\left(\dot{\theta}_1 - \dot{\theta}_{g1A}\right), \tag{3}$$

$$J_2\ddot{\theta}_2 = T_{cl2} - k_3\left(\theta_2 - \theta_{g2A}\right) - c_3\left(\dot{\theta}_2 - \dot{\theta}_{g2A}\right), \tag{4}$$

$$\left(i_1^2 J_{g1A} + J_{g1B}\right)\ddot{\theta}_{g1B} = i_1 k_2\left(\theta_1 - \theta_{g1A}\right) + i_1 c_2\left(\dot{\theta}_1 - \dot{\theta}_{g1A}\right) - k_4\left(\theta_{g1B} - \theta_{fA}\right) - c_4\left(\dot{\theta}_{g1B} - \dot{\theta}_{fA}\right), \tag{5}$$

$$\left(i_2^2 J_{g2A} + J_{g2B}\right)\ddot{\theta}_{g2B} = i_2 k_3\left(\theta_2 - \theta_{g2A}\right) + i_1 c_3\left(\dot{\theta}_2 - \dot{\theta}_{g2A}\right) - k_4\left(\theta_{g2B} - \theta_{fA}\right) - c_4\left(\dot{\theta}_{g2B} - \dot{\theta}_{fA}\right), \tag{6}$$

$$\left(J_{fB} + i_0^2 J_{fA}\right)\ddot{\theta}_{fB} = i_0 k_4\left(\theta_{g1B} - \theta_{fA}\right) + i_0 c_4\left(\dot{\theta}_{g1B} - \dot{\theta}_{fA}\right) + i_0 k_4\left(\theta_{g2B} - \theta_{fA}\right) + i_0 c_4\left(\dot{\theta}_{g2B} - \dot{\theta}_{fA}\right) - k_5\left(\theta_{fB} - \theta_w\right) - c_5\left(\dot{\theta}_{fB} - \dot{\theta}_w\right) - k_7\left(\theta_{fB} - \theta_w\right) - c_7\left(\dot{\theta}_{fB} - \dot{\theta}_w\right), \tag{7}$$

$$J_{wl}\ddot{\theta}_{wl} = \left[k_6\left(\theta_{fB} - \theta_{wl}\right) + c_6\left(\dot{\theta}_{fB} - \dot{\theta}_{wl}\right)\right] - \left[k_7\left(\theta_{wl} - \theta_{vl}\right) + c_7\left(\dot{\theta}_{wl} - \dot{\theta}_{vl}\right)\right], \tag{8}$$

$$J_w\ddot{\theta}_{wr} = \left[k_8\left(\theta_{fB} - \theta_{wr}\right) + c_8\left(\dot{\theta}_{fB} - \dot{\theta}_{wr}\right)\right] - \left[k_9\left(\theta_{wr} - \theta_{vr}\right) + c_9\left(\dot{\theta}_{wr} - \dot{\theta}_{vr}\right)\right], \tag{9}$$

$$\frac{J_{vl}}{2}\ddot{\theta}_{vl} = k_7\left(\theta_{wl} - \theta_{vl}\right) + c_7\left(\dot{\theta}_{wl} - \dot{\theta}_{vl}\right) - \frac{T_{vl}}{2}, \tag{10}$$

$$\frac{J_{vr}}{2}\ddot{\theta}_{vr} = k_9\left(\theta_{wr} - \theta_{vr}\right) + c_9\left(\dot{\theta}_{wr} - \dot{\theta}_{vr}\right) - \frac{T_{vr}}{2}. \tag{11}$$

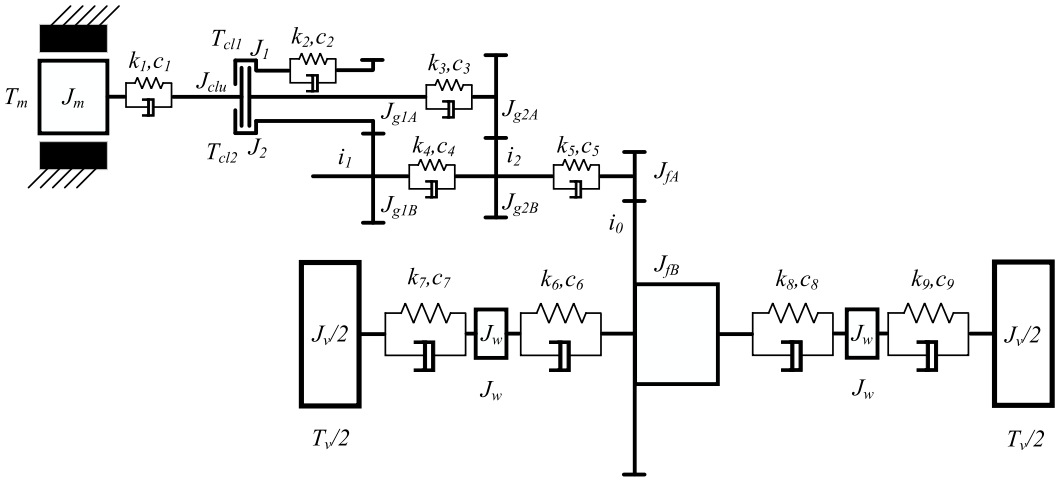


FIGURE 3. Structure diagram of 11-DOF system of vehicle.

The whole vehicle driving resistance torque is

$$T_v = \left(mg\sin\alpha + \mu mg\cos\alpha + \frac{1}{2}\rho C_D A\left(\dot{\theta}_w r_w\right)^2\right) r_w, \tag{12}$$

where k_i and c_i (i=1,2,3,……) denote the stiffness and damping of each rotating shaft, θ_m, θ_{clu}, θ_1, θ_2, θ_{g1A}, θ_{g1B}, θ_{g2A}, θ_{g2B}, θ_{fA}, θ_{fB}, $\theta_w$and θ_v, denote the motor rotation angle, the clutch main disc angle, clutch 1 driven disc angle, clutch 2 driven disc angle, first gear main gear angle, second gear main gear angle, first gear-driven gear angle, second gear-driven gear angle, main reducer main gear angle, main reducer driven gear angle, wheel angle and half car steering angle; θ, $\dot{\theta}$ and $\ddot{\theta}$ indicate the angular displacement of the corresponding rotating parts, rotational speed, and angular acceleration; m is the vehicle mass, g is the gravitational acceleration, α is the slope angle, μ is the rolling resistance coefficient, ρ is the air density, C_D is aerodynamic drag coefficient and A is the windward area.

C. GEARSHIFT CONTROL MODEL

Automotive gearshift control relies on an effective control model and control strategy. The gearshift control model usually selects the gearshift trajectory that meets the conditions from the gearshift strategy according to the vehicle driving conditions and driver's demand and completes the shifting process by adjusting the motor and clutch torque. However, since the gearshift trajectory is obtained from the simplified dynamics model and there are disturbances in the vehicle driving environment, it is difficult to obtain ideal gearshift control results by using the gearshift optimization strategy alone, so the controller in this article chooses to use the feedforward-feedback composite controller, which not only can accurately adjust the torque signal but also can reduce the workload of the feedback controller under the condition that the model accuracy of the feedforward controller is appropriately reduced, thus improving the gearshift quality [29].

The gearshift control model designed in this article is shown in Fig 4. The strategy module provides the reference control trajectory and reference state trajectory to the controller module according to the vehicle speed and throttle opening degree. The feedforward controller sends out control signal U_1 according to the reference control trajectory, and the feedback controller outputs feedback control signal U_2 by comparing the real-time state and the reference state and then outputs the actual control signal U to other systems of the vehicle after combining with U_1. The more important strategy module as well as the controller module in the gearshift control model will be introduced below.

1) STRATEGY MODULE

The gearshift strategy module stores the gearshift optimization strategy based on Legendre pseudo-spectrum method. In the process of gearshift strategy optimization, Legendre pseudo-spectrum method iterative operations are performed according to different accelerator pedal openings and vehicle speeds, and the optimization results contain gearshift optimization strategies for different driving conditions and driver requirements, and the resulting gearshift strategy solution set can meet the performance requirements of different driving conditions. In this article, the economic Jerk-pattern gearshift optimization trajectory will be selected. The gearshift point is determined according to the current vehicle speed and accelerator pedal position, and the trajectory signal that meets the gearshift conditions is sent to the controller module by analyzing the gearshift point and gearshift pattern.

2) CONTROLLER MODULE

The controller module consists of feedforward controller and feedback controller. The feedforward controller is based on the output torque signal of the strategy module obtained by offline optimization, without considering the real-time environmental noise, so the design of the feedback controller is extremely important. In this article, RBFNN is selected as the

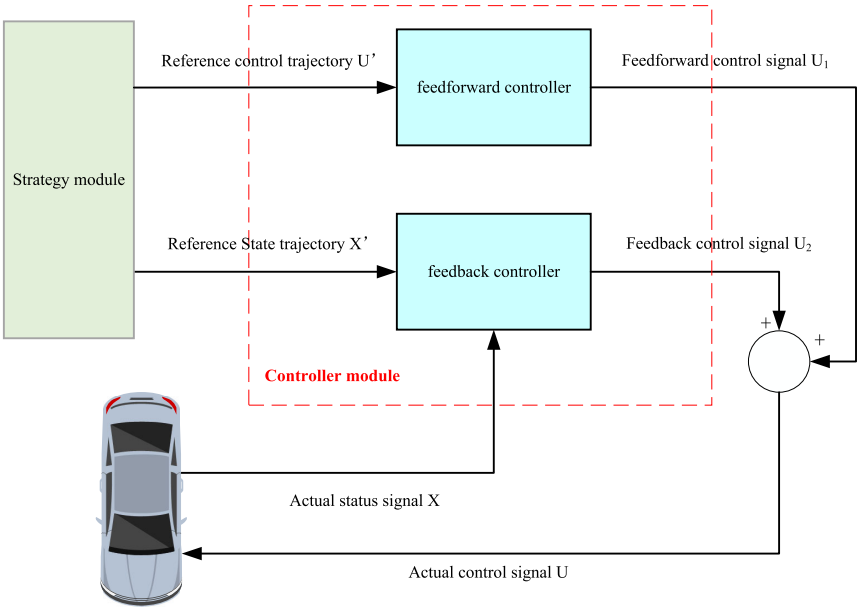


FIGURE 4. Schematic diagram of the gearshift control model, consisting of a controller module and a strategy module.

feedback controller. The real-time performance of RBFNN is good, and the control signal can be generated only by forward propagation in the inference stage, which makes the calculation speed faster. The structure is simple, and the number of parameters that need to be stored is usually small, so the storage requirements are relatively low. The parameters that need to be trained in this article are limited, so a large amount of training time is not required. Therefore, from the point of view of experiment and calculation, the application of RBFNN controller to DCT shift is feasible. Its design and principle are described in detail in Part III.

3) GEARSHIFT CONTROL MODELING

From the above analysis, it can be seen that the motor torque includes the feedforward torque $\bar{T}_m$ and the feedback torque T'_m, the transfer torque of clutch 1 is only the feedforward torque $\bar{T}_{c1}$, and the transfer torque of clutch 2 includes the feedforward torque $\bar{T}_{c2}$ and the feedback torque T'_{c2}, where the feedback torque of the motor and clutch 2 are the control variables. The state variable $\boldsymbol{x}$ is expressed as follows:

$$\boldsymbol{x} = \left[\dot{\theta}_m, \dot{\theta}_{clu}, \dot{\theta}_1, \dot{\theta}_2, \dot{\theta}_{g1B}, \dot{\theta}_{g2B}, \dot{\theta}_{fB}, \dot{\theta}_{wl}, \dot{\theta}_{wr}, \dot{\theta}_{vl}, \dot{\theta}_{vr}\right].$$

The feedforward control variable $\bar{\boldsymbol{u}}$ and the feedback control variable $\boldsymbol{u}'$ are denoted as follows:

$$\bar{\boldsymbol{u}} = \left[\bar{T}_m, \bar{T}_{c1}, \bar{T}_{c2}\right]^\top,$$
$$\boldsymbol{u}' = \left[T'_m, 0, T'_{c2}\right]^\top.$$

Then the control variable u is denoted as:

$$\boldsymbol{u} = \left(\bar{\boldsymbol{u}} \oplus \boldsymbol{u}'\right)^\top.$$

The state space equations of the shift dynamics model can be obtained by reorganizing the variables of the above equation:

$$\dot{\boldsymbol{x}} = (\boldsymbol{A} \oplus \boldsymbol{B})\boldsymbol{x} + \boldsymbol{C}\left(\bar{\boldsymbol{u}} + \boldsymbol{u}'\right) + \boldsymbol{D}, \tag{13}$$

where $\boldsymbol{A}$,$\boldsymbol{B}$,$\boldsymbol{C}$,$\boldsymbol{D}$ are the matrices with damping, stiffness, control variables, and driving torque parameters, respectively.

The optimization target of the gearshift control is to adjust the feedback torque of the motor and clutch 2 so that the actual speed difference trajectory of the main and secondary discs of clutch 2 follows the reference trajectory, ensuring that the actual speed difference at the end of the shifting process tends to zero. Therefore, the actual state variable to be controlled is $\left(\dot{\boldsymbol{x}}_{[2]} - \dot{\boldsymbol{x}}_{[4]}\right)$ and the control objective is $\left(\dot{\boldsymbol{x}}_{[2]} - \dot{\boldsymbol{x}}_{[4]}\right) - \left(\dot{\theta}_m - i_0 i_2 \dot{\theta}_v\right)$.

III. CONTROLLER DESIGN AND PARAMETERS OPTIMIZATION

A. RBFNN FEEDBACK CONTROLLER DESIGN

In 1985, Powell proposed the radial basis function method for multivariate interpolation [30]. The radial basis function is a real-valued function that depends only on the distance from the origin, that is, $\varphi(x) = \varphi(\|x\|)$, or can also be the distance to any point c, c is called the center point, that is, $\varphi(x, c) = \varphi(\|x - c\|)$. Any function φ that satisfies the characteristic $\varphi(x) = \varphi(\|x\|)$ is called a radial basis function, and the common Gaussian kernel function chosen in this article is of the following form

$$k(\|x - c\|) = \exp\left(-\frac{\|x - c\|^2}{(2 * \sigma)^2}\right), \tag{14}$$

where c is the center of the kernel function; σ is the width parameter of the function, which controls the radial range of action of the function.

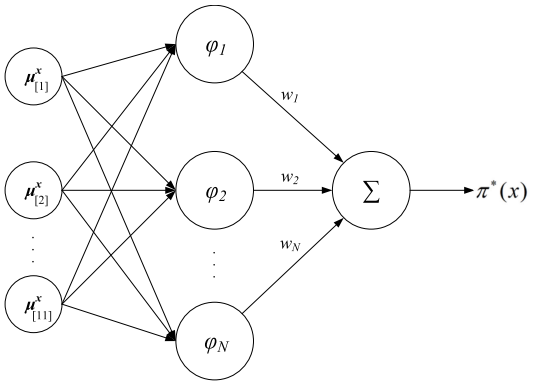


FIGURE 5. Radial basis function neural networks.

Fig. 5 visualizes the structure of a radial basis function neural network, which is a three-layer neural network consisting of an input layer, a linear output layer, and a hidden layer with a specific nonlinear RBF activation function. In this article, the input layer is an 11-dimensional vector $\left[\boldsymbol{\mu}^{\boldsymbol{x}}_{[1]}, \cdots, \boldsymbol{\mu}^{\boldsymbol{x}}_{[11]}\right]$, i.e., the state distribution. The hidden layer uses multiple Gaussian functions $[\varphi_1, \cdots, \varphi_N]$ as the neurons of the activation function, whose role is to map the low-dimensional nonlinearly differentiable input to a high-dimensional linearly differentiable space. This results in the output value of the implicit function layer as:

$$\boldsymbol{\varphi}_n = \exp\left(-\frac{1}{2}\left(\boldsymbol{\mu}^{\boldsymbol{x}} - \boldsymbol{c}_n\right)^{\top} \boldsymbol{W}\left(\boldsymbol{\mu}^{\boldsymbol{x}} - \boldsymbol{c}_n\right)\right), \tag{15}$$

where $\boldsymbol{\varphi}_n$ is the Gaussian kernel function; $\boldsymbol{c}_n$ is the centroid of the function, the initial centroid is sampled from the initial state distribution and can be obtained by training samples to meet the specific values required, and the Gaussian kernel function responds locally to the input, when the input is close to the centroid of the basis function, the hidden layer nodes will produce a larger output; when it is far from the centroid, the output will be exponentially decayed; $\boldsymbol{\mu}^{\boldsymbol{x}}$ is the mean value of the state variables; $\boldsymbol{W}$ is the covariance matrix, the magnitude of its value determines the magnitude of the influence of different input values on the neuron, and the influence of a state variable on the corresponding neuron can be eliminated by setting the corresponding value in $\boldsymbol{W}$ to zero.

The control strategy $\pi^*(\boldsymbol{x})$ of the output layer output is a weighted sum of the output values of each neuron in the hidden function layer

$$\pi^*(\boldsymbol{x}) = \sum_{n=1}^{N} \varphi_n w_n, \tag{16}$$

where w_n is the weight coefficient, which, like $\boldsymbol{c}_n$, can be obtained from training samples to meet the specific value required.

In addition, due to physical factors, both the motor output torque and the transfer torque of clutch 2 are limited, i.e., $\boldsymbol{u} \in [-\boldsymbol{u}_{\max}, \boldsymbol{u}_{\max}]$, so the final feedback controller outputs the control strategy $\pi(\boldsymbol{x})$ as:

$$\pi(\boldsymbol{x}) = \boldsymbol{u}_{\max}\rho\pi^*(\boldsymbol{x}), \tag{17}$$

where ρ is the sinusoidal compression function that can control $\pi(\boldsymbol{x})$ within the restriction range $[-\boldsymbol{u}_{\max}, \boldsymbol{u}_{\max}]$; $\pi^*(\boldsymbol{x})$ is the non-compression control strategy.

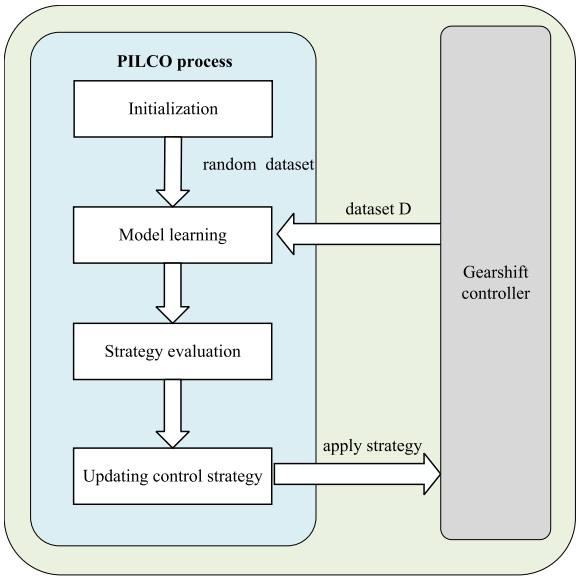


FIGURE 6. PILCO reinforcement learning algorithm process.

B. PILCO OPTIMIZATION ALGORITHM

In order for the RBFNN to achieve the desired control effect, the values of the function's centroid $\boldsymbol{c}_n$, the covariance matrix $\boldsymbol{W}$, and the weight coefficients w_n need to be optimized. For this purpose, it is crucial to obtain sufficient training samples and to select the appropriate algorithm for controller parameter optimization.

Reinforcement learning algorithms have demonstrated their effectiveness in training neural networks [31], and reinforcement learning is usually divided into two categories, model-based methods, and model-free methods. In the former, the agent gradually builds an internal model of the environment(learning) and then uses that model to design a control policy (planning). In the latter, the agent learns the control policy directly from the interaction with the environment. Model-free approaches often require more interaction with the environment, i.e., extensive experiments in the real world.To improve the efficiency of the gearshift controller parameter optimization, the model-based PILCO algorithm is chosen to perform this work in this article.

The main flow of PILCO algorithm is shown in Fig. 6. After initializing the program, a random data set is first generated and a Gaussian procedure (GP) is performed using that data set. The uncertainty factors such as model error and environmental disturbance are included in the modeling process, and the nonlinear probabilistic dynamics model of the vehicle satisfying the Gaussian distribution is learned. Approximate

TABLE 1. PILCO algorithm pseudo-code.

	Algorithm: PILCO algorithm
1	Initialization: set random dataset for the first rollout.
2	**For i = 1: m**
3	Apply the policy π in the system and collect Data D for the first time.
4	Train an unknown dynamic model f_unkown with GP.
5	**For j = 1: n**
6	Rollout system with RBF Neural Network and get expected long-term cost $L\left(\boldsymbol{x}_{[0]}\right)$
7	Update policy π
8	**End**
9	**End**

inference for policy evaluation to get the expected long term cost. The gradient descent method is used to minimize the value function, and the optimal strategy under the current GP model is found. And update the control policy by updating the controller parameters. Applying the optimized control strategy to the shift control results in a new data set D. After that, the PILCO algorithm completes the first iteration and performs the next iteration using the new data set D. The results of the shift control optimization are recorded in the results of each iteration.

Table 1 shows the pseudo-code of this algorithm, m denotes the number of custom iterations and n denotes the number of searches for the control strategy. The PILCO algorithm is characterized by high learning efficiency and good robustness so that the controller parameter matrix satisfying the practical requirements is usually obtained after a few iterations. In this article, m is taken to be 10, and n is taken to be 150 in the gearshift control optimization study.

Fig. 7 shows the schematic diagram of the gearshift control model optimized based on the PILCO algorithm. The feedback part uses a radial basis function neural network controller with offline learning and online control functions. The gearshift strategy module stores the Legendre pseudo-spectral optimization strategy, and the state estimation module collects the sensor signals and transmits the data set D required for feedback regulation to the radial basis function neural network controller. The feedback torque signal is output from the radial basis function neural network controller, summed with the feedforward torque signal and transmitted to the torque actuator module and finally transformed into the torque actuation signal to achieve gearshift control.

The computing process of the PILCO algorithm is divided into model learning, strategy evaluation, and controller parameter updating.

1) MODEL LEARNING

Learning the probabilistic dynamics model using a GP is an important part of the bottom layer of the PILCO algorithm. the GP uses a scalar output approximation function from which the uncertainty model $f_d(\boldsymbol{x}, \boldsymbol{u})$ is learned and the corresponding probability distribution is obtained. the GP actually learns the difference between the reference dynamics model and the actual dynamics model, and the process is stochastic. The gearshift dynamics model can be expressed as follows:

$$\boldsymbol{x}_{[t]} = f\left(\boldsymbol{x}_{[t-1]}, \boldsymbol{u}_{[t-1]}\right), \tag{18}$$

where the state variable $\boldsymbol{x}_{[t]}$ of the current step is jointly determined by the state variable $\boldsymbol{x}_{[t-1]}$ of the previous step and the control quantity $\boldsymbol{u}_{[t-1]}$.

To model the probabilistic dynamics, the PILCO algorithm introduces the difference variable

$$\Delta_{[t]} = \left(\boldsymbol{x}_{[t]} - \boldsymbol{x}_{[t-1]}\right) + \boldsymbol{\alpha}, \tag{19}$$

where $\boldsymbol{\alpha} \sim N\left(0, \boldsymbol{\Sigma}_\alpha\right)$, $\boldsymbol{\Sigma}_\alpha = diag\left(\left[\boldsymbol{\sigma}_{\alpha_1}, \ldots, \boldsymbol{\sigma}_{\alpha_N}\right]\right.$, and $\boldsymbol{\sigma}$ is the variance.

To obtain the gearshift probability dynamics model, PILCO learns the fitted relationship between the input $\boldsymbol{x}_{[t]}$ and the output $\Delta_{[t]}$. Define the training data set as:

$$\mathcal{D} := \left\{\boldsymbol{x} := \left[\boldsymbol{x}_{[1]}, \cdots, \boldsymbol{x}_{[n]}\right]^\top, \mathbf{Y} := \left[\Delta_{[1]}, \cdots, \Delta_{[n]}\right]^\top\right\}. \tag{20}$$

Assuming $z = \left[\boldsymbol{x}^\top, \boldsymbol{u}^\top\right]^\top$, the GP is characterized by the state variable mean $m(z)$ and the kernel function $k\left(z, z^{'}\right)$, and $m(z) = 0$ is chosen, then $k\left(z, z^{'}\right)$ is denoted as

$$k\left(z, z^{'}\right) = \sigma_f^2 \exp\left(-\frac{1}{2}\left(z - z^{'}\right)^\top \Lambda^{-1}\left(z - z^{'}\right)\right), \tag{21}$$

where σ_f^2 is the signal variance. Λ is a diagonal matrix composed of feature lengths, which are hyper-parameters of GP, where the last hyper-parameter value denotes the noise variance σ_ϵ^2. Since the controller parameter matrix δ' is randomly generated when running PILCO for the first time, the GP needs to learn the entire gearshift dynamics model with data set D.

For the test point z_*, the output value $\Delta(z_*)$ will be Gaussian distributed with the mean and variance

$$\mu_d(z_*) = K_{z_* z}\left(K_{zz} + \sigma_\epsilon^2 I\right)^{-1} y_d, \tag{22}$$

$$\Sigma_d(z_*) = k(z_*, z_*) - K_{z_* z}\left(K_{zz} + \sigma_\epsilon^2 I\right)^{-1} K_{z_* z}^\top, \tag{23}$$

where $K_{zz} = \begin{bmatrix} k\left(z_{[1]}, z_{[1]}\right) & \cdots & k\left(z_{[1]}, z_{[n]}\right) \\ \vdots & \ddots & \vdots \\ k\left(z_{[n]}, z_{[1]}\right) & \cdots & k\left(z_{[n]}, z_{[n]}\right) \end{bmatrix}$,

$K_{z_* z} = \left[k\left(z_*, z_{[1]}\right) \ldots k\left(z_*, z_{[n]}\right)\right]$.

At this point, the learning of the probabilistic gearshift dynamics model is completed.

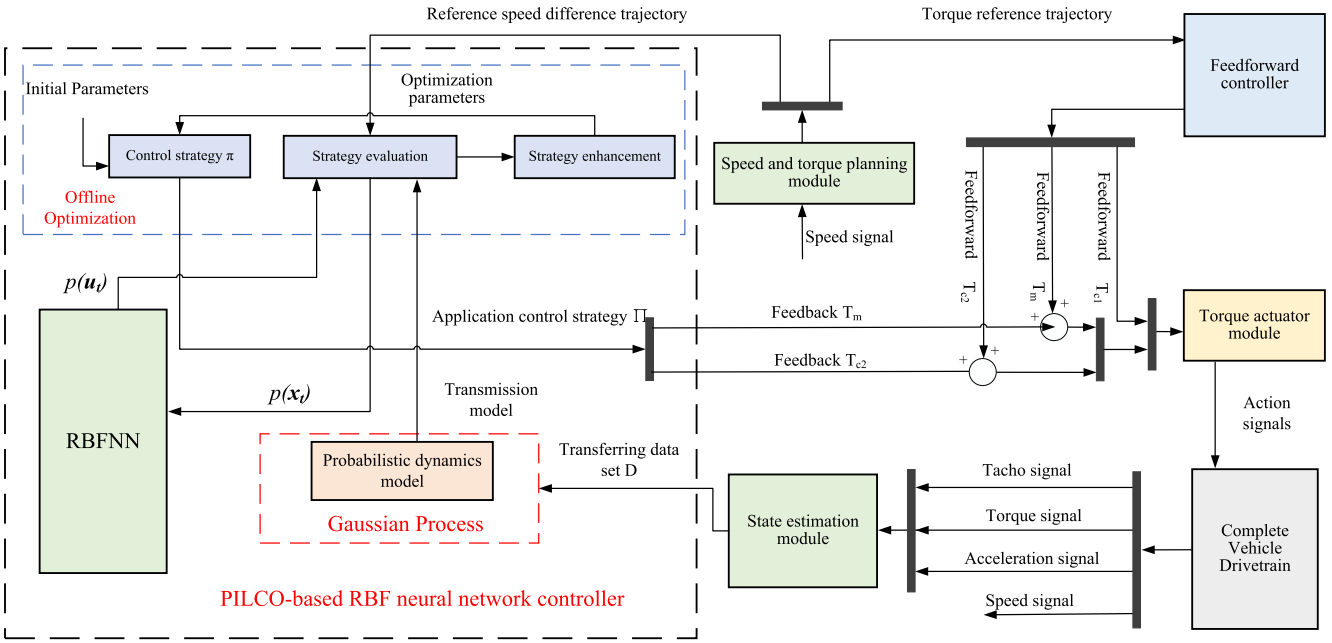


FIGURE 7. Schematic diagram of gearshift control model based on PILCO algorithm.

2) STRATEGY EVALUATION

With the PILCO bottom operation, the GP learns a probabilistic dynamics model satisfying a Gaussian distribution with the following expression:

$$\dot{\boldsymbol{x}}_{[t]} = (\boldsymbol{A}_d \oplus \boldsymbol{B}_d)\,\boldsymbol{x}_{[t-1]} + \boldsymbol{C}\left(\bar{\boldsymbol{u}}_{[t-1]} + \boldsymbol{u}'_{[t-1]}\right) + f\left(\boldsymbol{x}_{[t-1]}, \boldsymbol{u}_{[t-1]}\right) + \boldsymbol{D}, \tag{24}$$

$$\boldsymbol{u}_{[t-1]} = \bar{\boldsymbol{u}}_{[t-1]} + \pi\left(\boldsymbol{x}_{[t-1]}\right). \tag{25}$$

The system model contains the nominal dynamics model $\left(A_d\boldsymbol{u}_{[t-1]} + \boldsymbol{C}\right)$ and the uncertainty model $f\left(\boldsymbol{x}_{[t-1]}, \boldsymbol{u}_{[t-1]}\right)$ learned through the underlying. $\boldsymbol{u}_{[t-1]}$ is the control variable, including the feedforward control variable $\bar{\boldsymbol{u}}_{[t-1]}$ and the feedback control variable $\pi\left(\boldsymbol{x}_{[t-1]}\right)$. From the expression, it is clear that the system model needs to be run step-by-step.

In the middle layer, the PILCO algorithm uses the initial state $\boldsymbol{x}_{[0]}$ and time t=0 as the simulation starting point, and the target state $\boldsymbol{x}_{[t]}$ and time $t = T$ as the simulation end point, and uses the system model to compute the value function $L\left(\boldsymbol{x}_{[0]}\right)$ to evaluate the system model for the policy. the expression of $L\left(\boldsymbol{x}_{[0]}\right)$ is as follows:

$$L\left(\boldsymbol{x}_{[0]}\right) = \sum_{i=0}^{T} \int \left[c\left(\boldsymbol{x}_{[t]}\right) p\left(\boldsymbol{x}_{[t]}\right)\right] d\boldsymbol{x}_{[t]}, \tag{26}$$

$$c\left(\boldsymbol{x}_{[t]}\right) = 1 - \exp\left(-\frac{1}{2}\left(\boldsymbol{x}_{[t]} - \bar{\boldsymbol{x}}_{[t]}\right)^{\top} \boldsymbol{W}\left(\boldsymbol{x}_{[t]} - \bar{\boldsymbol{x}}_{[t]}\right)\right), \tag{27}$$

where $c\left(\boldsymbol{x}_{[t]}\right)$ is the artificially set reward function, $x_{[0]}$ is the initial state,$\boldsymbol{x}_{[t]} \sim \mathcal{N}\left(\boldsymbol{\mu}^{\boldsymbol{x}}_{[t]}, \Sigma^{\boldsymbol{x}}_{[t]}\right)$, $\boldsymbol{\mu}^{\boldsymbol{x}}_{[t]}$ and $\Sigma^{x}_{[t]}$ denote the mean and variance of the state distribution at the current step, respectively. The matrix $\boldsymbol{W}$ is the weight matrix, which is an irreversible matrix with a dimension equal to the number of state variables.

TABLE 2. Iterative calculation process.

$p\left(\boldsymbol{x}_{[t]}\right)$ iterative calculation process
1. Obtain the state distribution $p\left(\boldsymbol{x}_{[t-1]}\right)$ at moment t-1, Calculate the control volume distribution $p\left(\boldsymbol{u}_{[t-1]}\right)$) at moment t-1
2. Calculate the joint state-control volume distribution $p\left(\boldsymbol{u}_{[t-1]} \mid \boldsymbol{x}_{[t-1]}\right)$ at moment $t-1$
3. Calculate the output distribution $p\left(\Delta\boldsymbol{x}_{[t-1]}\right)$ at moment t-1
4. Calculate the state distribution $p\left(\boldsymbol{x}_{[t]}\right)$) at moment t

According to the above equation, the value function $L\left(\boldsymbol{x}_{[0]}\right)$ of the initial state can be solved only by computing the probability distribution $p\left(\boldsymbol{x}_{[t]}\right)$ of the state $\boldsymbol{x}_{[t]}$ for all steps. The Probability distribution for each step is given by (28), as shown at the bottom of the next page.

It follows that $p\left(\boldsymbol{x}_{[t]}\right)$ needs to be obtained before calculating $p\left(\boldsymbol{x}_{[t]} \mid \boldsymbol{x}_{[t-1]}, \boldsymbol{u}_{[t-1]}\right)$, $p\left(\boldsymbol{u}_{[t-1]} \mid \boldsymbol{x}_{[t-1]}\right)$ and $p\left(\boldsymbol{x}_{[t-1]}\right)$. The process can be divided into 4 steps, as shown in Table 2.

The state distribution $p\left(\boldsymbol{x}_{[1]}\right), \ldots, p\left(\boldsymbol{x}_{[T]}\right)$ for all steps can be solved by repeating the above four steps, and the value function $L\left(\boldsymbol{x}_{[0]}\right)$ of the initial state can be calculated by using the resulting state distribution and performing the integration operation according to equation (28), as shown at the bottom of the next page.

3) CONTROLLER PARAMETER UPDATE

The feedback torque optimization objective of the RBFNN controller is to minimize the value function $L\left(\boldsymbol{x}_{[0]} \mid \pi_\delta\right)$ under the condition of simulation time T and having the

number of steps dt. The method for updating the controller parameter matrix is to use pytorch's auto-gradient function to implement a gradient descent-based policy search by searching for the optimal parameter δ^* such that the control policy π satisfies the following equation:

$$\delta^* \in \arg\min_{\pi\in\Pi} L\left(\boldsymbol{x}_{[0]}\middle|\pi_\delta\right). \tag{29}$$

The expression for the gradient of $L\left(\boldsymbol{x}_{[0]}\middle|\pi_\delta\right)$ with respect to δ is as follows:

$$\frac{dL\left(\boldsymbol{x}_{[0]}\right)}{d\delta} = \sum_{t=0}^{T} \frac{d}{d\delta}\mathbb{E}_{\boldsymbol{x}_{[t]}}\left[c\left(\boldsymbol{x}_{[t]}\right)\middle|\pi_\delta\right], \tag{30}$$

where the state quantity $\boldsymbol{x}_{[t]}$ at each moment satisfies the Gaussian distribution and the controller parameters δ are in the mean and covariance. The feasible methods to calculate the gradient formula include analytic differentiation and automatic differentiation, and in this article, the analytic method is chosen to calculate the gradient formula, and the corresponding gradient formula is the extended formula using the chain rule is given by (31), as shown at the bottom of the page.

where the mean and variance derivatives of the state distribution associated with the controller parameters δ can be further expanded as the following expressions:

$$\frac{d\boldsymbol{\mu}^{\boldsymbol{x}}_{[t]}}{d\delta} = \frac{\partial\boldsymbol{\mu}^{\boldsymbol{x}}_{[t]}}{\partial\boldsymbol{\mu}^{\boldsymbol{x}}_{[\boldsymbol{x}-1]}}\frac{d\boldsymbol{\mu}^{\boldsymbol{x}}_{[t-1]}}{d\delta} + \frac{\partial\boldsymbol{\mu}^{\boldsymbol{x}}_{[t]}}{\partial\Sigma^{\boldsymbol{x}}_{[t-1]}}\frac{d\boldsymbol{\Sigma}^{\boldsymbol{x}}_{[t-1]}}{d\delta} + \frac{\partial\boldsymbol{\mu}^{\boldsymbol{x}}_{[t]}}{\partial\delta}, \tag{32}$$

$$\frac{d\boldsymbol{\Sigma}^{\boldsymbol{x}}_{[t]}}{d\delta} = \frac{\partial\boldsymbol{\Sigma}^{\boldsymbol{x}}_{[t]}}{\partial\mu^{\boldsymbol{x}}_{[t-1]}}\frac{d\mu^{\boldsymbol{x}}_{[t-1]}}{d\delta} + \frac{\partial\boldsymbol{\Sigma}^{\boldsymbol{x}}_{[t]}}{\partial\boldsymbol{\Sigma}^{\boldsymbol{x}}_{[t-1]}}\frac{d\boldsymbol{\Sigma}^{\boldsymbol{x}}_{[t-1]}}{d\delta} + \frac{\partial\boldsymbol{\Sigma}^{\boldsymbol{x}}_{[t]}}{\partial\delta}. \tag{33}$$

The sum of the gradients of all steps is the gradient of the value function to δ. The controller parameter matrix control is updated using the gradient descent method to obtain a new control strategy. The parameter matrix of the controller $\boldsymbol{P}_{hyp}$ is as follows:

$$\boldsymbol{P}_{hyp} = \begin{bmatrix} \delta_{1,1}\ \delta_{1,2}\ldots\delta_{1,12}\ \delta_{1,13}\ \delta_{1,s}\ \delta_{1,n} \\ \delta_{2,1}\ \delta_{2,2}\ldots\delta_{2,12}\ \delta_{2,13}\ \delta_{2,s}\ \delta_{2,n} \end{bmatrix} \tag{34}$$

where $\boldsymbol{P}_{hyp}(1:13, 1:2)$ is the state parameter matrix, $\boldsymbol{P}_{hyp}(14, 1:2)$ denotes the signal variance, and $\boldsymbol{P}_{hyp}(15, 1:2)$ denotes the noise variance.

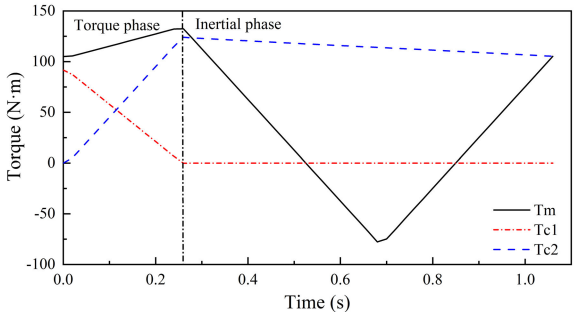


FIGURE 8. Feedforward control value.

IV. SIMULATION EXPERIMENTS AND ANALYSIS

It should be noted that the whole vehicle driveline used in this article is a simplified 11-degree-of-freedom system and does not fully reflect the vehicle's transmission. It is assumed that the whole vehicle is under steady state conditions at the beginning of the experiment and that the constant operating conditions are maintained throughout the experiment. And there is no power loss during the whole gear shift process.

A. SIMULATION CONDITIONS

To verify the optimization efficiency and effect of the RBFNN gearshift controller using the PILCO algorithm, the Jerk-pattern gearshift optimization results obtained through the pseudo-spectrum method, as shown in Fig. 8, are used as feedforward control variables. Subsequently, simulation experiments are conducted under 0 slope conditions, no load, and a 40% accelerator pedal opening. The vehicle parameters for the simulation process are set according to Table 3. To illustrate the optimization effect, the PID controller is selected as a comparison in this article, which usually takes a lot of time to adjust parameters. To compare the efficiency of the PILCO algorithm laterally, a group of parameters with better control effects in 20 adjustments are selected as simulation parameters in this article, and P, I and D parameters were set at 0.1, 3.0 and 0.01, respectively.

B. SIMULATION RESULTS AND ANALYSIS

The simulation results of shift control based on initial random controller parameters are shown in Fig. 9, and the results of four iterations of training are shown in Fig. 10 to Fig. 13, in which (a) (b) record the motor and clutch 2 torque trajectories and clutch 2 main and secondary moving disc speed difference trajectory difference during the simulation, respec-

$$p\left(\boldsymbol{x}_{[t]}\right) = \iint \left[p\left(\boldsymbol{x}_{[t]} \mid \boldsymbol{x}_{[t-1]}, \boldsymbol{u}_{[t-1]}\right) p\left(\boldsymbol{u}_{[t-1]} \mid \boldsymbol{x}_{[t-1]}\right) p\left(\boldsymbol{x}_{[t-1]}\right)\right] d\boldsymbol{x}_{[t-1]} d\boldsymbol{u}_{[t-1]}. \tag{28}$$

$$\frac{d}{d\delta}\mathbb{E}_{\boldsymbol{x}_{[t]}}\left[c\left(\boldsymbol{x}_{[t]}\right) \mid \pi_\delta\right] = \left(\frac{\partial}{\partial\boldsymbol{\mu}^{\boldsymbol{x}}_{[t]}}E_{\boldsymbol{x}_{[t]}}\left[c\left(\boldsymbol{x}_{[t]}\right) \mid \pi_\delta\right]\right)\frac{d\boldsymbol{\mu}^{\boldsymbol{x}}_{[t]}}{d\delta} + \left(\frac{\partial}{\partial\boldsymbol{\Sigma}^{\boldsymbol{x}}_{[t]}}\mathbb{E}_{\boldsymbol{x}_{[t]}}\left[c\left(\boldsymbol{x}_{[t]}\right) \mid \pi_\delta\right]\right)\frac{d\boldsymbol{\Sigma}^{\boldsymbol{x}}_{[t]}}{d\delta}. \tag{31}$$

TABLE 3. System parameters for powertrain model.

Symbol	Name	Value	Symbol	Name	Value
J_m	Motor	6.57×10^{-2} kg · m^2	J_{clu}	Clutch drum	0.30 kg · m^2
J_1	Clutch one	7.25×10^{-3} kg · m^2	J_2	Clutch two	1.29×10^{-2} kg · m^2
J_{g1A}	Gear 1	1.51×10^{-2} kg · m^2	J_{g1B}	Pinion 1	7.52×10^{-3} kg · m^2
J_{g2A}	Gear 2	9.40×10^{-3} kg · m^2	J_{g2B}	Pinion 2	6.33×10^{-3} kg · m^2
J_{fA}	Final drive 1	3.04×10^{-3} kg · m^2	J_{fB}	Final drive 2	1.16 kg · m^2
J_w	Wheel hubs	7.52×10^{-1} kg · m^2	J_v	Whole vehicle	143.04 kg · m^2
r_w	Tyre radius	0.23 m	m	Vehicle mass	1500 kg
μ	Rolling resistance	0.01	ρ	Air density	1.13 kg / m
C_D	Drag coefficient	0.30	A	Windward area	2.08 m^2
η_t	Transmission efficiency	0.94	k_1	Motor output	1×10^5 Nm/rad
k_2	Primary shaft 1	1×10^5 Nm/rad	k_3	Lay shaft 1	1×10^5 Nm/rad
k_4	Primary shaft 2	1×10^5 Nm/rad	k_5	Lay shaft 2	1×10^5 Nm/rad
k_6	Half shaft	1.10×10^4 Nm/rad	k_7, k_9	Tyre stiffness	2.60×10^3 Nm/rad
k_8	Half shaft	9.39×10^3 Nm/rad	c_1	Motor output	10 Nm · s/rad
c_2	Primary shaft 1	100 Nm · s/rad	c_3	Lay shaft 1	100 Nm · s/rad
c_4	Primary shaft 2	100 Nm · s/rad	c_5	Lay shaft 2	100 Nm · s/rad
c_6	Half shaft	90 Nm · s/rad	c_7, c_9	Tyre damping	50 Nm · s/rad
c_8	Half shaft	90 Nm · s/rad	m_a	Full load mass	1900 kg

tively. Under the initial random strategy simulation conditions, the speed difference between the main and secondary discs of clutch 2 is significantly higher than the reference speed trajectory throughout the shifting process, and the actual motor torque is significantly smaller than its feedforward torque as shown by the torque trajectory, indicating that the motor output torque is reduced, which also cannot make the actual speed difference trajectory follow the reference trajectory. Therefore, in the first iteration, the transfer torque of the clutch rises sharply in the inertial phase, as shown in Fig. 10, but then the transfer torque is too large, resulting in the speed difference following the target trajectory only in the middle moments, while in the end moments, it is difficult to reduce rapidly, and finally, the actual speed difference is smaller than the estimated speed difference instead.

Based on the data recorded in the first iteration, the second iteration was started, as shown in Fig. 11. In this iteration, both the instances when the motor output torque was lower than the feedforward torque and when the clutch transfer torque exceeded the feedforward torque were advanced compared to the previous iteration. This adjustment was made to avoid the situation where the speed difference at the end of the last iteration became too small. The final speed difference is slightly greater than 0 at the end of the gearshift. However, at the beginning moment, the actual speed difference is still larger than the ideal speed difference, and this problem will be solved in the next iteration.

As can be seen in Fig. 12, in the third iteration, the motor output torque decreases significantly in the torque phase, and the actual speed difference can track the reference trajectory better, while the clutch transfer torque is also reduced, making the subsequent speed difference unaffected. According to the results of this iteration, it can be seen that the trajectory fit of the actual speed difference and the reference speed difference is better, and the actual speed difference of the clutch tends to be 0 at the end of the gearshift. according to the comparison of Fig. 12 and Fig. 13, it can be seen that the results of the fourth iteration are similar to those of the third iteration, except that iteration 3 is further optimized and the trajectory of the actual speed difference between the main and secondary discs of clutch 2 is closer to the reference trajectory, and the speed reaches 0 at the end of the gearshift.

Considering the existence of stiffness and damping in the driveline, the speed of each link in the shifting process has obvious fluctuations, so when calculating the trajectory error, the actual trajectory and the reference trajectory are discretized separately and split into 1000 segments with a step length of only 0.00106s, which can reflect the speed fluctuation more accurately. The following equations are then used to calculate the mean square error (MSE), mean absolute error (MAE), and root mean square error (RMSE), respectively:

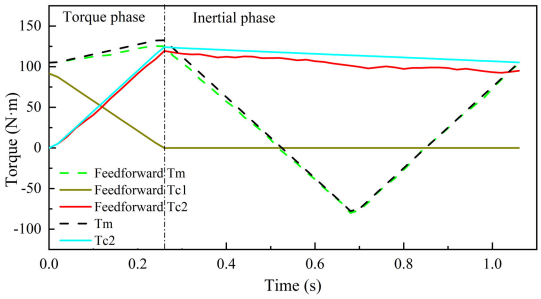
$$MSE = \frac{1}{N}\sum_{i=1}^{N}\left(y_{actu} - y_{ref}\right)^2, \tag{35}$$

$$MAE = \frac{1}{N}\sum_{i=1}^{N}\left|y_{actu} - y_{ref}\right|, \tag{36}$$

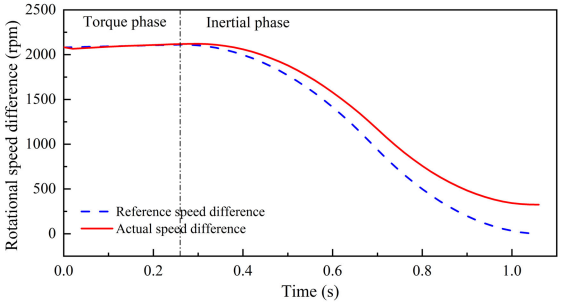
$$RMSE = \sqrt{\frac{1}{N}\sum_{i=1}^{N}\left(y_{actu} - y_{ref}\right)^2}. \tag{37}$$

where N is the number of time segments of the shifting process, y_{actu} is the actual value of the speed difference in each segment, and y_{ref} is the reference value of the speed difference in each segment.

The trajectory errors of the four iterations are shown in Table 4. Fig. 14 visualizes the trend of trajectory error. When the PILCO algorithm is used to optimize the RBFNN

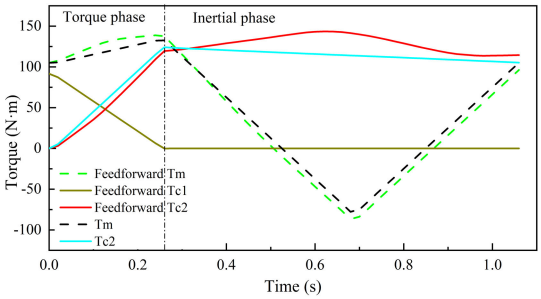


(a) Torque trajectory.

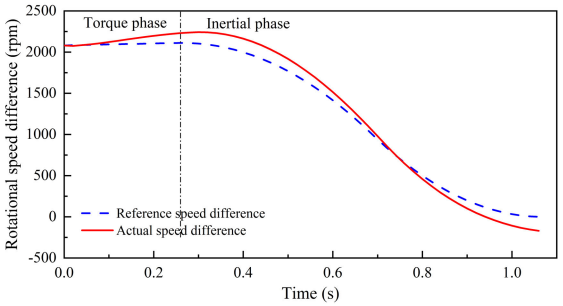


(b) Clutch 2 master and slave disc speed difference trajectory.

FIGURE 9. Gearshift control based on initial random controller parameters.

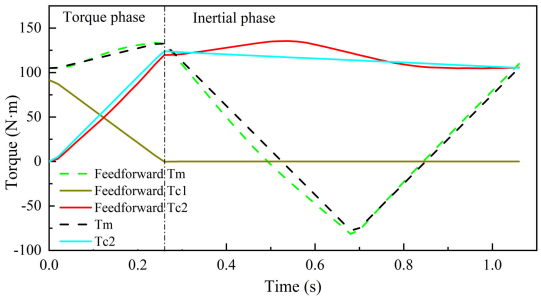


(a) Torque trajectory.

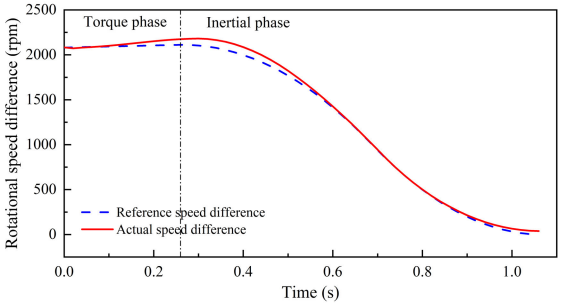


(b) Clutch 2 master and slave disc speed difference trajectory.

FIGURE 10. The first iteration of gearshift control.

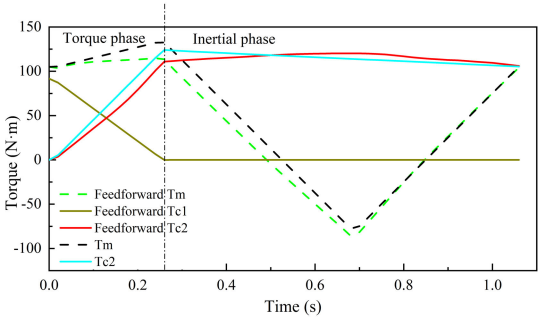


(a) Torque trajectory.

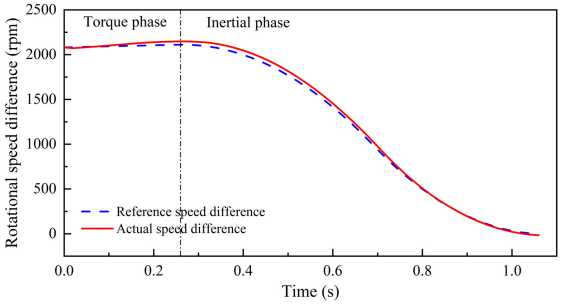


(b) Clutch 2 master and slave disc speed difference trajectory.

FIGURE 11. The second iteration of gearshift control.



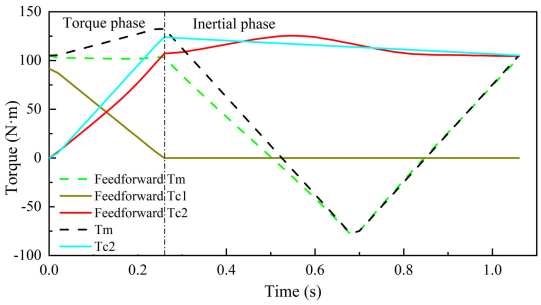
(a) Torque trajectory.



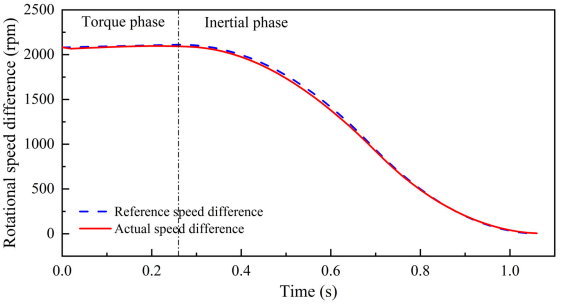
(b) Clutch 2 master and slave disc speed difference trajectory.

FIGURE 12. The third iteration of gearshift control.

controller, with the increase of iteration times, the average trajectory error after the first iteration decreases by 52.83% compared with no feedback, and the optimization effect is obvious. After the second optimization iteration, it dropped



(a) Torque trajectory.



(b) Clutch 2 master and slave disc speed difference trajectory.

FIGURE 13. The fourth iteration of gearshift control.

TABLE 4. Trajectory error between actual speed difference and the reference speed difference. The average decline from the no feedback mode is displayed in the last column of the table.

Control mode	MSE	RMSE	MAE	Reduction
PILCO Iteration 1	1.16×10^4	107.86	94.35	52.83%
PILCO Iteration 2	1.94×10^3	44.03	32.72	84.36%
PILCO Iteration 3	9.82×10^2	31.34	26.33	88.63%
PILCO Iteration 4	3.80×10^2	19.49	16.48	93.12%
No feedback	3.76×10^4	193.83	171.90	-

to 84.36%, which is still a significant improvement compared with the first optimization. The third time is slightly optimized to 88.63%, and the actual velocity difference trajectory gradually conforms to the reference trajectory. After the fourth optimization, the average trajectory error is reduced to 93.12%, which is not obvious compared with the third optimization. This shows that when the RBFNN controller is optimized with the PILCO algorithm, only a few iterations are needed to complete the optimization of the RBFNN controller parameters, and the optimized controller is able to make the actual speed difference trajectory completely close to the reference trajectory. This proves the effectiveness and efficiency of the PILCO algorithm for optimizing the RBFNN controller parameters.

C. ANALYSIS OF DIFFERENT SLOPES AND LOADINGS

To further demonstrate the effectiveness and robustness of the PILCO algorithm for controller optimization. This section continues to use the Jerk-pattern optimization strategy with 40% accelerator pedal opening as the feedforward control signal, and the RBFNN controller obtained by the PILCO algorithm in the previous section is used as the feedback controller for the shifting simulation at +3.0% slope operating condition, full load operating condition and +3.0% slope full load operating condition, respectively.

Fig. 15, Fig. 16 and Fig. 17 show the optimized gearshift control results for +3.0% slope no load and 0% slope full load and +3.0% slope full load conditions, respectively. Compared with the no-ramp working condition, as the increase of load

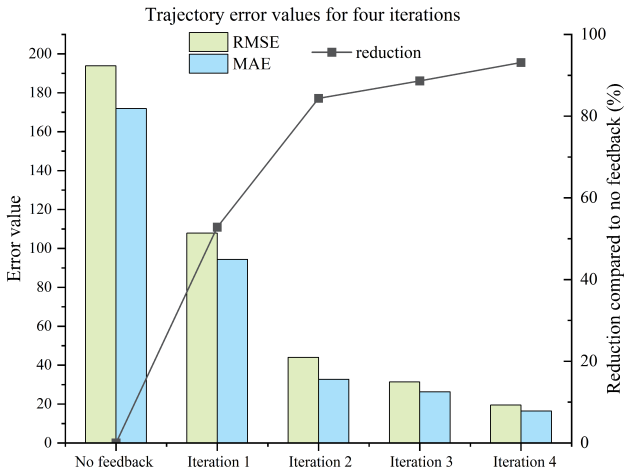
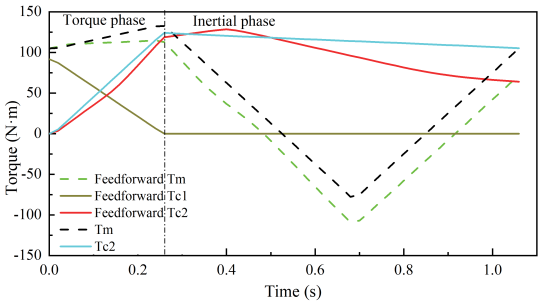


FIGURE 14. Trajectory error values for the four iterations of PILCO.

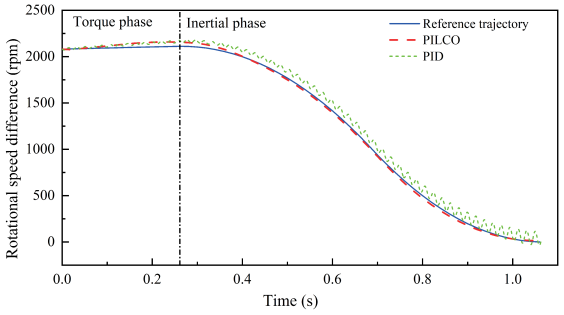
or uphill will increase the longitudinal resistance torque of the whole vehicle, the tendency of the clutch secondary to decelerate is more obvious, so the average clutch transmission torque increases further when the load increases or uphill shifting, and the motor output torque decreases further so that the active disc speed decreases more quickly to ensure the speed difference tends to 0 at the end of the gearshift. It can be observed that in the final moment, the speed difference between the three operating conditions tends to zero, and the performance is better than the PID control scheme. In order to visually demonstrate the optimization effect of PILCO on the controller, an analysis of the result data will be conducted.

Since the ideal trajectory of the main and secondary discs in the shifting process is not the same under different load and slope conditions, the shifting evaluation indexes are chosen here to measure the control effects of the RBFNN controller and PID controller. The gearshift quality of the car is an important indicator to characterize the performance of the vehicle, after careful consideration, the impact degree and slip work are chosen as the evaluation indexes of gearshift quality:

$$g_1 = \int_{t_1}^{t_2} j^2 dt, \tag{38}$$

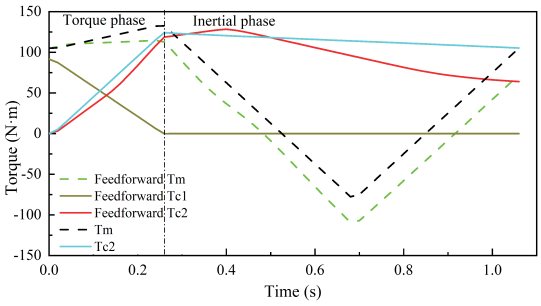


(a) Torque trajectory.

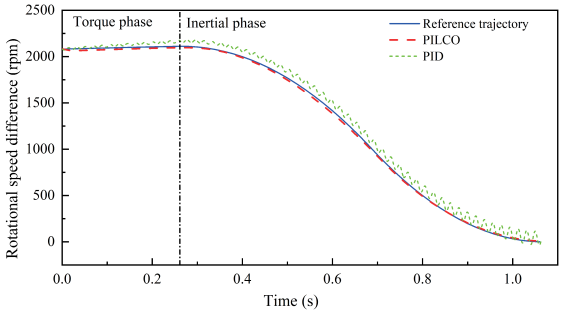


(b) Clutch 2 master and slave disc speed difference trajectory.

FIGURE 15. Shift control based on no-load, +3% ramp conditions.

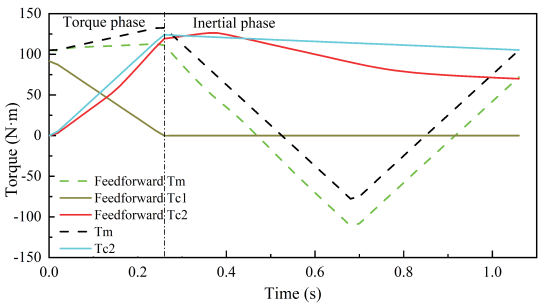


(a) Torque trajectory.

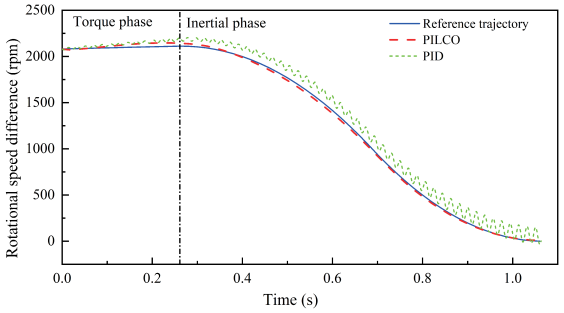


(b) Clutch 2 master and slave disc speed difference trajectory.

FIGURE 16. Shift control based on full load, 0% ramp conditions.



(a) Torque trajectory.



(b) Clutch 2 master and slave disc speed difference trajectory.

FIGURE 17. Shift control based on full load, +3% ramp conditions.

$$j = \frac{d\dot{a}}{dt}, \tag{39}$$

$$\dot{a} = \frac{d\omega_v}{dt} r, \tag{40}$$

$$g_2 = \int_{t_1}^{t_2} \left[\left| T_{c1} \left(\omega_{clu} - \omega_1 \right) \right| + \left| T_{c2} \left(\omega_{clu} - \omega_2 \right) \right| \right] dt, \tag{41}$$

where, g_1 is the sum of the square of the shock degree during the gearshift, j is the shock, a is the acceleration, ω_v is the wheel angular velocity, r is the wheel radius, t_1 and t_2 is the start and end of the gearshift; g_2 is the sum of the slipping work during the gearshift.

To comprehensively consider the impact of shock and slip work on the gearshift quality, max (g_1) and g_2 are selected as the final comparison indexes. Table 5 shows the values of each index when the PID controller and RBFNN controller are used as feedback controllers under three different operating conditions, respectively. From the table, it can be seen that the RBFNN controller optimized by the PILCO algorithm reduces max (g_1) by 91.212%, 41.395% and 91.798%, and g_2 by 4.250%, 8.891% and 4.500% under three different working conditions of slope and load. From Fig. 18, it can be seen more intuitively that under different working conditions, the jerk has been significantly reduced, the slipping work has been well optimized, and the gearshift quality has been significantly improved to meet the gearshift requirements, which proves that the RBFNN controller has a good control

TABLE 5. Trajectory error between actual speed difference and the reference speed difference.

Vehicle wight(kg)	Slope(%)	Control model	g1(m/s3)	max(g1)(m/s3)	Reduction of max(g1)(%)	g2(kj)	Reduction of g2(%)
1500	+3	RBFNN	0.064	2.237	-	12.863	-
		PID	97.677	25.457	91.212	13.434	4.250
1900	0	RBFNN	0.136	3.233	-	12.666	-
		PID	9.702	5.636	41.395	13.902	8.891
1900	+3	RBFNN	0.024	1.771	-	13.029	-
		PID	139.210	21.592	91.798	13.643	4.500

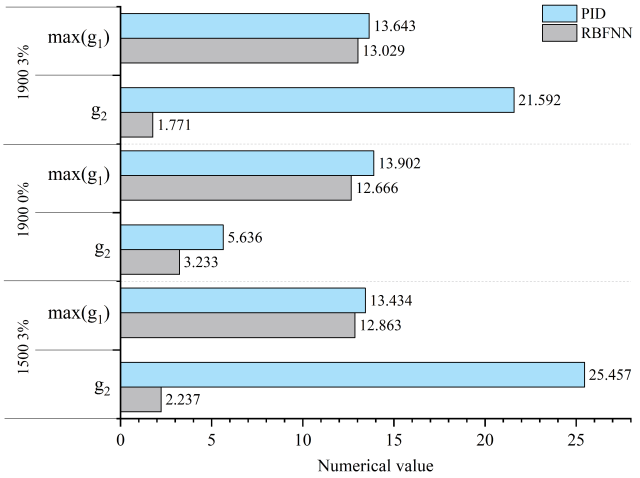


FIGURE 18. Trajectory error values for different operating conditions and control modes.

effect as well as the effectiveness and robustness of PILCO algorithm.

In addition, if the optimized trajectories of the above three cases are compared with their respective ideal trajectories, it is not difficult to think of the same analytical results, i.e., there is an obvious optimization than PID, which is limited to space and will not be expanded here. In the experiment, it is also found that the longitudinal drag torque increases significantly when the vehicle is going uphill, and the increase of the load at this time will further increase the longitudinal drag torque, resulting in the speed difference trajectory error increasing with the increase of the longitudinal drag torque. Since the controller parameters are derived from no-load, no-gradient shift simulation iterations, the same controller parameters have limited applicability. In the case of certain controller parameters, the longitudinal drag torque of the whole vehicle is too large to affect the performance of the controller.

V. CONCLUSION

A deep reinforcement learning method for shift control of dual clutch transmissions in electric vehicles is proposed in this work. The method uses RBFNN as the feedback controller and PILCO as the optimization algorithm. By conducting experiments to analyze the training iteration process of the optimization algorithm on the RBFNN controller, it is evident that the algorithm can complete the optimization learning of the RBFNN controller with only a few experiments, and the learned controller meets the gearshift requirements and performs well. Furthermore, the RBFNN controller optimized by the reinforcement learning algorithm is applied to different ramps and loads to compare its control effect with the PID feedback controller. The results show that the RBFNN feedback controller has good robustness under complex and variable driving conditions.

It is worth mentioning that in the high noise environment, such as potholes, slippery and other extreme road conditions and data transmission delays, the uncertainty estimation of the model or the occasional local suboptimal solution may affect the control effect. In addition, the research can be extended to different transmissions, providing efficient shift control strategies for electric vehicles. Incorporating other learning and planning techniques like safe reinforcement learning and hybrid reinforcement learning can further enhance the algorithm's performance and applicability.

REFERENCES

[1] B. Z. Gao, H. Chen, K. Sanada, and Y. Hu, "Design of clutch-slip controller for automatic transmission using backstepping," *IEEE/ASME Trans. Mechatronics*, vol. 16, no. 3, pp. 498–508, Jun. 2011.

[2] D. R. Cox and N. Reid, *The Theory of the Design of Experiments*. Boca Raton, FL, USA: CRC Press, 2000.

[3] K. D. Mishra and K. Srinivasan, "Improved integrated powertrain control of gearshifts using linear parameter varying control," in *Proc. Amer. Control Conf. (ACC)*, Jul. 2019, pp. 4553–4558.

[4] F. Boissinot, J. Bellavoine, A. Shabashevich, and S. Puster, "Automated calibration for transmission on powertrain dynamometers," SAE Tech. Paper 2015-01-1625, 2015.

[5] S. Krishnaswamy and M. Kirti, "Method for automated calibration and adaptation of automatic transmission controllers," OHIO State Innov. Found., U.S. Patent 11 261 961 B2, Mar. 2021.

[6] K. D. Mishra, G. Cardwell, and K. Srinivasan, "Automated calibration of gearshift controllers using iterative learning control for hybrid systems," *Control Eng. Pract.*, vol. 111, Jun. 2021, Art. no. 104786.

[7] A. Haj-Fraj and F. Pfeiffer, "Optimal control of gear shift operations in automatic transmissions," *J. Franklin Inst.*, vol. 338, nos. 2–3, pp. 371–390, Mar. 2001.

[8] T. Ouyang, Y. Lu, S. Li, R. Yang, P. Xu, and N. Chen, "An improved smooth shift strategy for clutch mechanism of heavy tractor semi-trailer automatic transmission," *Control Eng. Pract.*, vol. 121, Apr. 2022, Art. no. 105040.

[9] G. Li and D. Görges, "Optimal control of the gear shifting process for shift smoothness in dual-clutch transmissions," *Mech. Syst. Signal Process.*, vol. 103, pp. 23–38, Mar. 2018.

[10] I. Cvok, J. Deur, V. Ivanovic, Y. Zhang, and Y. Fujii, "An LQR approach of automatic transmission upshift control including use of off-going clutch within inertia phase," *SAE Int. J. Adv. Current Practices Mobility*, vol. 2, no. 4, pp. 2081–2091, Apr. 2020.

[11] M. Roozegar and J. Angeles, "A two-phase control algorithm for gear-shifting in a novel multi-speed transmission for electric vehicles," *Mech. Syst. Signal Process.*, vol. 104, pp. 145–154, May 2018.

[12] M. S. R. Mousavi, A. Pakniyat, T. Wang, and B. Boulet, "Seamless dual brake transmission for electric vehicles: Design, control and experiment," *Mech. Mach. Theory*, vol. 94, pp. 96–118, Dec. 2015.

[13] S. Kim and S. B. Choi, "Cooperative control of drive motor and clutch for gear shift of hybrid electric vehicles with dual-clutch transmission," *IEEE/ASME Trans. Mechatronics*, vol. 25, no. 3, pp. 1578–1588, Jun. 2020.

[14] K. Cheng, D. Qin, J. Jian, and B. Wu, "Adaptive gearshift control of wet dual clutch transmission based on extended state observer and H∞ robust control," *Proc. Inst. Mech. Eng. C, J. Mech. Eng. Sci.*, vol. 235, no. 22, pp. 6586–6598, Nov. 2021.

[15] Z. Wang and T. Hong, "Reinforcement learning for building controls: The opportunities and challenges," *Appl. Energy*, vol. 269, Jul. 2020, Art. no. 115036.

[16] X. Zhang, W. Lei, Y. Dai, Q. Tang, X. Yuan, Z. Xiao, and G. Lv, "Parameter optimization of multiple resonant controller: A deep reinforcement learning approach," in *Proc. IEEE 9th Int. Power Electron. Motion Control Conf. (IPEMC-ECCE Asia)*, Nov. 2020, pp. 2578–2581.

[17] K. Ogawa and T. Aihara, "Seamless shifting of a two-speed dual clutch transmission for electric vehicles using machine learning," *Mech. Eng. Lett.*, vol. 7, no. 2021, p. 21, Dec. 2021.

[18] K. Deuschl, T. Doster, G. Gaiselmann, and S. Studer, "Automated functional development for automatic transmissions using deep reinforcement learning," *ATZ Electron. Worldwide*, vol. 15, no. 9, pp. 8–13, Sep. 2020.

[19] G. Gaiselmann, S. Altenburg, S. Studer, and S. Peters, "Deep reinforcement learning for gearshift controllers in automatic transmissions," *Array*, vol. 15, Sep. 2022, Art. no. 100235.

[20] M. P. Deisenroth, D. Fox, and C. E. Rasmussen, "Gaussian processes for data-efficient learning in robotics and control," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 37, no. 2, pp. 408–423, Feb. 2015.

[21] A. G. Bors, "Introduction of the radial basis function (RBF) networks," in *Proc. Online Symp. Electron. Eng.*, 1996, pp. 1–7.

[22] B. Bischoff, D. Nguyen-Tuong, H. van Hoof, A. McHutchon, C. E. Rasmussen, A. Knoll, J. Peters, and M. P. Deisenroth, "Policy search for learning robot control using sparse data," in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, May 2014, pp. 3882–3887.

[23] Y. Li, S. Guo, L. Zhu, T. Mukai, and Z. Gan, "Enhanced probabilistic inference algorithm using probabilistic neural networks for learning control," *IEEE Access*, vol. 7, pp. 184457–184467, 2019.

[24] W. A. Ramirez, Z. Q. Leong, H. D. Nguyen, and S. G. Jayasinghe, "Exploration of the applicability of probabilistic inference for learning control in underactuated autonomous underwater vehicles," *Auto. Robots*, vol. 44, no. 6, pp. 1121–1134, Jul. 2020.

[25] T. Zhang, M. Xiao, and Y. B. Zou, "Guidance control of a wheeled mobile robot with human interaction based on force control," *J. Zhejiang Univ., Eng. Sci.*, vol. 53, pp. 1865–1873, Jan. 2019.

[26] Y. F. Yang, K. Deng, Y. Q. Zuo, X. J. Ban, and X. L. Huang, "Parameter design and optimization of a flight attitude simulator system based on pilco framework," *Opt. Precis. Eng.*, vol. 27, no. 11, pp. 2365–2373, 2019.

[27] Y. Liu, Y. Chen, Z. Li, K. Zhao, and Z. Lin, "Multi-objective optimal gearshift control for multispeed transmission electric vehicles," *IEEE Access*, vol. 8, pp. 129785–129798, 2020.

[28] M.-A. Beaudoin and B. Boulet, "Improving gearshift controllers for electric vehicles with reinforcement learning," *Mech. Mach. Theory*, vol. 169, Mar. 2022, Art. no. 104654.

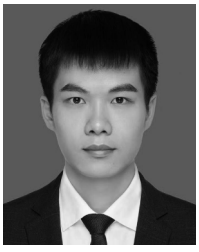
[29] X. Zhu, H. Zhang, J. Xi, J. Wang, and Z. Fang, "Optimal speed synchronization control for clutchless AMT systems in electric vehicles with preview actions," in *Proc. Amer. Control Conf.*, Jun. 2014, pp. 4611–4616.

[30] M. Powell, "The theory of radial basis function approximation in 1990," *Adv. Numer. Anal.*, vol. 2, pp. 105–210, 1992.

[31] T. P. Lillicrap, J. J. Hunt, A. Pritzel, N. Heess, T. Erez, Y. Tassa, D. Silver, and D. Wierstra, "Continuous control with deep reinforcement learning," 2015, *arXiv:1509.02971*.

YANWEI LIU received the Ph.D. degree in vehicle engineering from the South China University of Technology, Guangzhou, China, in 2012. He is currently an Associate Professor with the Guangdong University of Technology. His current research interests include control and optimization of powertrain systems and energy management strategy of hybrid energy storage systems for EV.

JINGLONG ZHANG received the engineering degree from the Zhongyuan University of Technology, Zhengzhou, China, in 2021. He is currently pursuing the joint M.S. degrees in vehicle engineering and mechanical engineering with the Zhongyuan University of Technology and the Guangdong University of Technology. His research interest includes optimization of gear shifting processes using reinforcement learning algorithms.



ZIYUAN LV received the engineering degree from Jiangxi Agricultural University, Jiangxi, China, in 2021. He is currently pursuing the M.S. degree in mechanical engineering with the Guangdong University of Technology. His research interest includes multiobjective optimization of gearshift for electric vehicle.

JIE YE received the Ph.D. degree in vehicle engineering from the South China University of Technology, Guangzhou, China, in 2018. He is currently a Lecturer with Foshan University. His research interest includes control and optimization of powertrain systems for EV.