SSRG International Journal of Electronics and Communication Engineering
ISSN: 2348-8549 / https://doi.org/10.14445/23488549/IJECE-V10I3P101

Volume 10 Issue 3, 1-9, March 2023

Original Article

Research on Predictive Control for the Damping System of Autonomous Vehicles in the Public Transport on the Basis of Artificial Intelligence

Tran Ngoc Son[1], Lai Khac Lai[2]

[1]*Faculty of Electrical Engineering, University of Economics - Technology for Industries, Viet Nam.*
[2]*Faculty of Electrical Engineering, Thai Nguyen University of Technology, Thai Nguyen City, Vietnam.*

Received: 15 January 2023 Revised: 20 February 2023 Accepted: 02 March 2023 Published: 16 March 2023

Abstract *- Autonomous vehicles in public transport can perform a wide range of real-world tasks such as: moving in factories, in public transport, in search and rescue, etc., thus requiring varying degrees of auto-navigation in response to changes in the environment. This paper presents the predictive model (MPC) for the active suspension system, the vehicle's damping system, combined with Deep Q-network (DQN) algorithm reinforcement learning method applied to control self-driving cars in public transport. Intelligent traffic control system to control movement to avoid fixed obstacles or movable obstacles in the event of external interference, taking into account the comfort of the occupants of the vehicle; taking into account the load, the conveying systems; for the purpose of application in the field of intelligent transportation. The research results, built on Matlab Simulink software, show that autonomous vehicles can safely complete intelligent navigation tasks in an unknown environment and become a real system intelligent with the ability to self-study and adapt well to many different environments and various nonlinear factors.*

Keywords *- Autonomous vehicle, Active car suspension, Damping system, Model predictive control, Artificial intelligence, DQN algorithm.*

1. Introduction

In recent years, advances in power electronics, microelectronics, and permanent magnet materials have enabled significant improvements in the electric drive field of autonomous cars and electric vehicles. More stable performance and dynamic state, reduced weight and mass of the system, unlimited combinations of electronic control systems on cars, high reliability, as well as reduced dimensions are factors important to rationalize towards more general use of electric drives [1-5]. This advancement in the field of technology warrants analysis of the feasibility of implementing existing suspension systems with electromagnetic actuators to increase its performance without increasing costs and energy consumption [6].

Based on the examined findings, the electromagnetic active suspension system is considered to be the future trend of automotive suspension designs due to its energy regeneration, uncomplicated structure, functional high bandwidth capacity, flexible and precise force control, good ride quality as well as handling performance [7- 9]. On that basis, the study and application of predictive control model MPC (Model Prediction Control) and reinforcement learning technology, including the damping factor in the control of autonomous vehicles in the field of public transport. This ensures that autonomous vehicles can learn new skills, improve their self-driving and self-balancing capabilities, and provide them with the ability to make rational decisions and control smartly and safely. In this way, autonomous vehicles can learn to optimally adapt to the uncertainty and unpredictable changes of random and constantly changing environmental factors while safely travelling on the road [10-13]. In the world, a self-driving car driver training method in public transport, based on demonstration data, is a powerful and natural tool for developing controllers for autonomous vehicles taking into account these factors nonlinear factors such as the damping system caused by the vehicle. Some studies also develop techniques based on self-balancing methods based on adaptive control, adaptive fuzzy control, etc., which are collected from the observers and actions of incoming loads, observations and actions of autonomous vehicles [14 - 17]. There is also a wide range of demonstration training methods introduced in previous studies, including studies in the telecommunications field, in direct manipulation of trained agents, grasping objects, in the control of autonomous vehicles in public transport, etc., in addition to applied studies of reinforcement learning techniques and methods of building predictive models of autonomous vehicles when taking into account weaknesses. The vehicle's damping factor is based on the data of the path and obstacles, the load, etc. In recent times, a number of studies have also focused on the application of new control theory [18]-[20] and the basis of artificial intelligence [2] to solve control problems in general and intelligent control for autonomous vehicles and robots in particular. Compared with methods through control data, the nonlinear factor caused by the vehicle, reinforcement learning method is being focused and more flexible, because in some cases, for example, when self-driving cars in hazardous environments, uneven paths, variable loads, etc., without prior information and data, it is necessary to apply and combine with new and suitable models such as control models forecast MPC, [13, 16, 19].

Compared with methods through control data, the nonlinear factor caused by the vehicle, reinforcement learning method is being focused and more flexible, because in some cases, for example, when self-driving cars in hazardous environments, uneven paths, variable loads, etc., without prior information and data, it is necessary to apply and combine with new and suitable models such as control models forecast MPC, [10]. In the document [11, 12] studied and developed a ship rudder system using adaptive law combined with the MPC model, [12] proposed sustainable adaptive control theory combined with intelligence—artificial intelligence to control automatic systems, robot-Camera self-searching and tracking moving objects. In addition, the author Nguyen Tan Luy in [1] also proposed a system control solution for machine learning and intelligent control application of self-propelled robot objects that search for targets and avoid obstacles by control techniques behavior control based on empty space. At the same time, based on Lyapunov's theory to provide conditions to stabilize the process of focusing on the goal of self-propelled robots. The simulation results also show that the self-propelled robot has achieved the goal of tracing the orbit, and the transition time is at the allowable level. The study of reinforcement learning techniques in automatic control combined with the MPC predictive control model for damping systems is considered to be applied to autonomous vehicles' tasks in the autonomous driving public transport field. From there, propose a solution to control self-driving cars using the MPC model for the damping system and apply reinforcement learning techniques to control; aims to enable autonomous vehicles to learn optimally to the uncertainty and unpredictability of random, nonlinear, and constantly changing environmental factors, setting the stage for their application in practice, especially in today's smart traffic, [21-25].

In this paper, the author presents the DQN algorithm to combine with the MPC model to control autonomous vehicles, considering the damping factor of autonomous vehicles in public transport in an unknown environment. The research is conducted based on the new control algorithm and MPC model. More specifically, the author introduces a neural network structure to generalize and approximate the value of all states of the DQN algorithm and simulate and test eggs using Matlab Simulink software and other supporting tools.

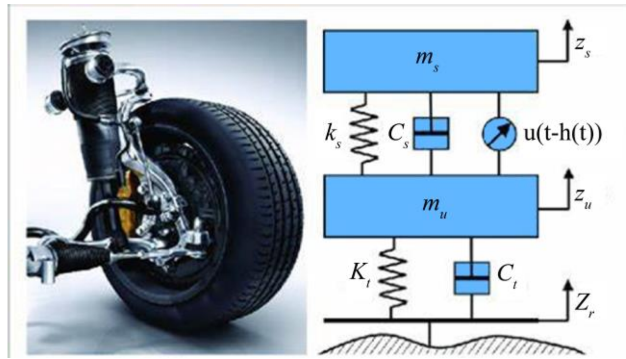

Fig. 1 Model of the active suspension system

2. The Building Models For Control Systems

The suspension determines the position of the wheels relative to the body, and it overcomes the contact forces between the wheel/tire and the road surface. Equipping the car with advanced suspension systems capable of impeding vibrations and noise became necessary. Moreover, the vehicle's speed on bumpy roads was not limited by the propulsion system's performance, which is limited by the quality of the suspension. Evaluating suspension quality in terms of comfort involves taking into account a variety of functional vehicle situations and a number of criteria for determining and measuring suspension quality parameters. In general, the quality of comfort is difficult to quantify; this is a vehicle suspension psychological concept [6, 19].

With stable parameters that can be more easily determined and measured, evaluations are made based on generally accepted criteria, even if these criteria can sometimes become a confusing conflict.

Then, in Figure 1, the vehicle suspension ¼ is presented where ms is the inflated mass, representing the body; mu is the unsprung mass, representing the mass of the wheel assembly; c_s and k_s are the damping and stiffness of the suspension, respectively; k_1 and c_1 represent the compression and damping capacity of the pneumatic tire, respectively; z_s and z_u are displacements of inflated and unsprung masses, respectively; z_r is the displacement input of the line; u control signal is actively fed into the vehicle suspension.

2.1. The Active Suspension Control System

Consider the suspension model shown in figure 2, where: m_s is the mass on the spring, representing the chassis; m_u is the mass under the spring, representing the mass of the wheel assembly; c_t and k_u are the damping and stiffness of the suspension respectively; k_s and c_s represent the compression and damping capacity of the pneumatic tire, respectively; z_r and z_u are the displacements of the unpacked and unpacked masses, respectively; z_s is the displacement input of the line; u is the active input of the suspension.

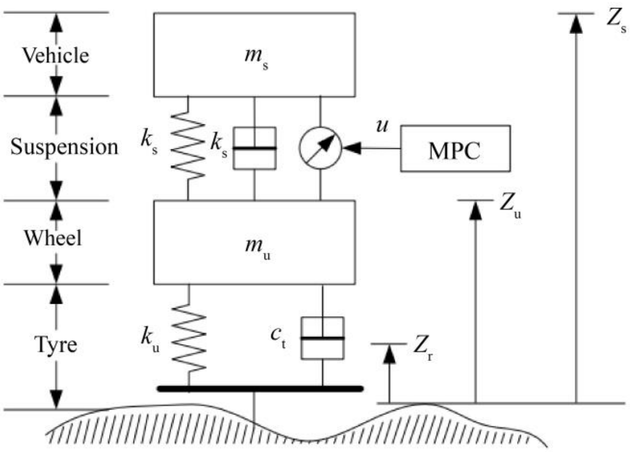

Fig. 2 Model of active suspension system ¼

When designing the control law for an active suspension system, we need to consider the following factors:

Passenger comfort: It is widely accepted that ride comfort is closely related to body acceleration. Therefore, when we design the controller, one of the main goals of the author is to reduce the acceleration of the body, i.e.:

$$\ddot{z}_s = 0 \tag{1}$$

Ensure mechanical strength: due to the mechanical structure, the suspension stroke must not exceed the maximum allowable value, i.e.:

$$|z_s(t) - z_u(t)| \leq z_{max} \tag{2}$$

Where zmax is the maximum allowable value of the suspension system.

Ensure traction: to ensure the safety of the car, it is necessary to ensure that the contact of the wheel with the road is not interrupted, and the dynamic load of the tire should be small, that is:

$$k_s(z_u(t) - z_3(t)) < (m_s + m_u)g$$

$$\Rightarrow \frac{k_s(z_u(t)-z_3(t))}{(m_s+m_u)g} < 1 \tag{3}$$

As a result, much attention has now been paid to active damping control (active suspension); thus, some of the important results based on the control approach in the system active suspension with different control techniques have been proposed; for example, fuzzy logic control method and neural network [3], adaptive control method [7], sustainable control H∞ [3], adaptive sliding control [4], etc. many positive results for this system.

2.2. Reinforcement Learning Methods as Controllers

Previously, in traditional studies, research has been done on control problems of self-propelled robots, industrial electric drive systems, conveyor systems, and CNC metalworking machine tracking systems. ,..v.v... and studied the design and experimented with working on traditional controllers such as PID algorithm, fuzzy PD, PD, PI & LQR. The biggest problem with those methods is that they need to be adjusted manually, which is not yet up to the quality of the controller. Therefore, achieving optimal controller values depends on many trials and occurs with many errors. Many times the optimal values are not achieved at all. The biggest benefit of reinforcement learning algorithms in automatic control is that as a controller, it is considered a self-correcting model to achieve optimal values. Then, with the reinforcement learning technique of the Q Learning algorithm and the Deep Q Network algorithm combined with the MPC predictive model, it has brought great benefits in terms of control quality for the active suspension system of the real car practice in public transport that the author goes to study.

With Q learning algorithm: According to document [21], “this algorithm gives actors the ability to learn how to act optimally in Markovian domains by experiencing the consequences of actions without asking them to build control environment maps for autonomous vehicles in traffic - Active suspension system”. In the Markovian domain, the function Q - algorithmically generated model - computes the expected utility for a given finite state s and any possible finite action a. for agents - which are autonomous vehicles in traffic, in this case - allowing to choose the optimal action with the highest value of Q (s, a), this action selection rule is also called priority policy. Initially, the function values Q (s, a) are assumed to be 0. Then through each training step, the values are updated according to the following equation [2, 22]:

$$Q(s_t, a_t) \leftarrow Q(s_t, a_t) + \alpha \Big[R_t + \\ + \gamma \max_{a \in A} Q(s_{t+1}, a_t) - Q(s_t, a_t) \Big] \tag{4}$$

$$Q(s_t, a_t) \leftarrow R_t + \gamma . \max_{a \in A} Q(s_{t+1}, a) \tag{5}$$

In the DQN algorithm, when training the neural network, there exists an arithmetic coefficient (α) in the backpropagation process, so it is reasonable to omit the learning coefficient (α) in formula (5). By doing this, the calculation of updating values also becomes simpler, and then we have:

$$Q(s_t, a_t) \leftarrow R_t + \alpha.(r + maxQ(s_{t+1}, a)) \tag{6}$$

The Q matrix had 20 columns, each column representing a state and ten rows, each representing every action. Initially, the Q-values were assumed to be 0, and some random actions were specified for every state in the policy π. We trained for 1500 episodes, each episode having 2000 iterations. At the beginning of each episode, the simulation refreshed. Whenever the robot's state exceeded the limit, it was penalized by assigning a reward of -100. The Q Table is updated at each step according to equation (6). This algorithm shows the full algorithm [2, 17, 18].

3. Research and Application of the DQN Algorithm Combined with the MPC Model for an Active Suspension System

3.1. Reinforcement Learning Method with DQN Algorithm

DQN inherits all properties of Q-learning. It is a model in the form of a free model, learned alone and belongs to the group of off-policy algorithms.

As mentioned in documents [2, 3]. The Q-learning and State Action Reward State Action SARSA (State Action Reward State Action) algorithms both have memory problems when storing the evaluation function as a two-dimensional array Q(s, a). When the state space and the action space are very large, about hundreds or thousands, then this storage will have memory problems, not to mention the computational cost of updating the value will increase exponentially.

In addition, the Q-learning algorithm still has another major weakness: the inability to estimate values for unknown states $s_i \notin S\{s_1, s_2, ...s_T\}$ thus, the inability to predict, leading to a lack of generalization. To solve this problem, the DQN algorithm has the ability to remove the two-dimensional Q-Table array and instead build a neural network to approximate this Q-Table algorithm table, as shown in figure 2 below is an illustrative example.

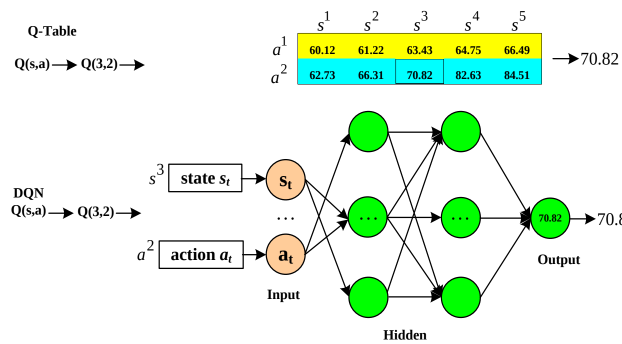

Fig. 3 The from Q-Table to DQN algorithm

The same Q-learning algorithm, the DQN training process is also based on a temporary differential method; DQN Agent updates network parameter θ of Q rating (S, A) at each step of the network training in order to Execute action a, receiving the new algorithm R, then it will significantly improve the performance of the control model for autonomous vehicles in traffic - Active suspension system; when using the control programming using the DQN algorithm.

3.2. Research on the Controller based on the Active Suspension System

From the system model, as shown in figure 1 and figure 2, we have the ideal dynamic equations of the multi-mass components on the vehicle as follows:

The kinetic equations for the upper and lower springs are:

$$\begin{cases} m_1\ddot{z}_1(t) + b_1[\dot{z}_1(t) - \dot{z}_2(t)] + k_1[z_1(t) - z_2(t)] = u(t) \\ m_2\ddot{z}_2(t) - b_1[\dot{z}_1(t) - \dot{z}_2(t)] - k_1[z_1(t) - z_2(t)] + k_2[z_2(t) - z_3(t)] + \\ +b_2[\dot{z}_2(t) - \dot{z}_3(t)] = -u(t) \end{cases} \quad (7)$$

When designing the control law for an active suspension system, we need to consider the following factors: in expression (7), z_1 is z_r, z_2 is z_u, z_3 is z_s, k_1 is k_s, k_2 is k_u, c_1 is c_s, c_2 is c_t.

Ensuring riding comfort: It is widely accepted that riding comfort is closely related to body acceleration. Therefore, when we designed the controller, one of our main goals was to reduce the body acceleration, means $\ddot{z}_1 = 0$.

$$\ddot{z}_1 = 0 \quad (8)$$

Ensure mechanical strength: due to mechanical construction, the suspension stroke should not exceed the maximum allowable level, which means:

$$|z_1(t) - z_2(t)| \le z_{max} \quad (9)$$

Where, z_{max} is the maximum deflection of the suspension.

In addition, to ensure the safety of the car, we need to ensure that the contact of the wheel with the road surface is not interrupted, and the dynamic load of the tire must be small, that is:

$$k_1(z_2(t)-z_3(t)) < (m_1+m_2)g$$

$$\Rightarrow [k_1(z_2(t)-z_3(t))/(m_1+m_2)g] < 1 \quad (10)$$

$$|u(t)| \le u_{max} \quad (11)$$

From there, the author will design a predictive controller to generate signal u(t) acting on the vehicle suspension system so that the expressions (8), (9), (10) are satisfied with the condition binding (11)

First, we need to represent expression (1) in the state space in a form suitable for the design of the MPC controller. Set the state variables as follows:

$$\begin{cases} x_1(t) = (z_1 - z_2) \\ x_2(t) = (z_2 - z_3) \\ x_3(t) = \dot{z}_1 \\ x_4(t) = \dot{z}_2 \\ x(t) = [x_1(t) \quad x_2(t) \quad x_3(t) \quad x_4(t)]^T \end{cases} \quad (12)$$

where, $x_1(t)$ represents the suspension deflection; $x_2(t)$ is the tire deflection; $x_3(t)$ is the mass release rate, and $x_4(t)$ represents the uninflated mass rate. The noise input is defined $w(t) = \dot{z}_3(t)$. The dynamics equations (7) can be written as follows:

$$\dot{x}(t) = A\begin{bmatrix} x_1(t) \\ x_2(t) \\ x_3(t) \\ x_4(t) \end{bmatrix} + B\begin{bmatrix} u(t) \\ w(t) \end{bmatrix} \quad (13)$$

in there,

$$A = \begin{bmatrix} 0 & 0 & 1 & -1 \\ 0 & 0 & 0 & 1 \\ -\frac{k_1}{m_1} & 0 & -\frac{b_1}{m_1} & \frac{b_1}{m_1} \\ \frac{k_1}{m_2} & -\frac{k_2}{m_2} & \frac{b_1}{m_2} & -\frac{b_1+b_2}{m_2} \end{bmatrix}; B = \begin{bmatrix} 0 & 0 \\ 0 & -1 \\ 1/m_1 & 0 \\ -1/m_2 & b_2/m_2 \end{bmatrix}$$

$$C = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & \frac{k_1}{(m_1+m_2)g} & 0 & 0 \end{bmatrix}; D = \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 0 & 0 \end{bmatrix}.$$

From the above equations, we build a predictive controller model according to the model consisting of two inputs and three outputs, including the first input value, u(t), is a measurable input, simple position N; the second input value: pavement undulation r(t) is random noise, this is an unmeasured component, unit (m); output components include: first output: body acceleration $\ddot{z}_1$ (m/s^2) is a measured quantity; the second output is suspension travel (z_1 - z_2) (m) which is an unmeasurable quantity; and the third output is road grip (kgm/s^2) (this is the relative dynamic load of the tire); assuming the constraint is $|u(t)| \le u_{max}$.

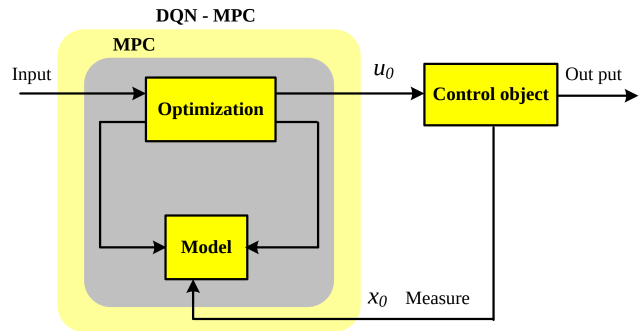

Fig. 4 Vehicle suspension control diagram with MPC model combined with reinforcement jurisprudence

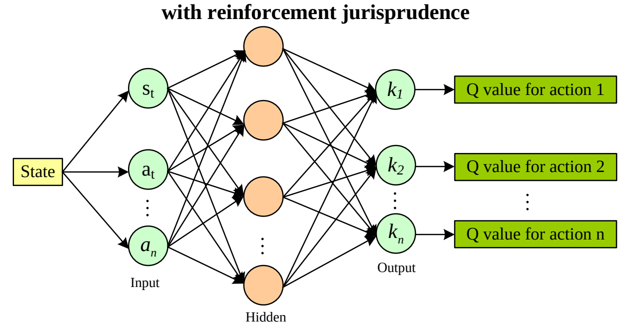

Fig. 5 The schematic diagram of the reinforcement learning model

3.3. The MPC Predictive Control Combined with the Damping System of Autonomous Vehicles using the DQN algorithm

The DQN algorithm was initially introduced as an AI agent to design for computational values with the action space the agent causes [2, 3]. The DQN algorithm Incorporating the MPC control model accomplished different results with the same algorithm. Each new value is considered a new state in which the agent can perform an action; since there are so many possible states and actions, it is impossible to explore all of them or use conventional algorithms to solve the problem posed, presented in figure 4.

The deep Q network algorithm starts by exploring the action space and gradually learns the mechanics of those values; the more these actors learn, the more they can get and the higher scores they can get. In this work, DQN will explore the relationship between changing geometric properties and the effect of action space on the final result through simulation and then use that knowledge to design structures that produce the optical response we desire. First, the environment is set up. This includes the initial structural design and simulation environment. Second, the agent's actions to change the structure are decided, and finally, a defined reward system. The DQN algorithm that connects all these parts together is shown in figure 5.

The decision of which action to take in a given state is decided by an updated neural network based on what it has learned (autonomous vehicle action spaces). To improve the performance of the DQN, an auxiliary model used in conjunction with this network is used to select actions for the performing agents, while the primary DQN network is used to predict the Q value of the pair state-action. This prevents overestimation, which is a problem in DQN in general. At each iteration, two models are trained on the network, and the weight of the target model is obtained from a combination of the main model weights and the target model weights. This method helped to overestimate due to using only one model. The backend network is periodically updated with the parameters of the DQN algorithm.

The implementation of the DQN algorithm on our Robot model: is similar to the Q-Learning algorithm. However, there are some exceptions. Initially, a model is initialized instead of Q-Matrix Initialization. Instead of choosing an action based on policy π, Q values are computed by the model in a policy with a value component. At the end of each set of state data values, the component is trained using random small batches of experience. Initially, an architecture with two hidden layers of 20 units was chosen, while the last layer was a Linear Density layer with ten units. With the γ of 0.9999 and very high precision probability values. DQN algorithm deployed on self-driving car model with input and output; shows that for the purpose of evaluating the quality and efficiency of the algorithm when the estimator is used in the whole process, the objective function helps to create separation for improvement, which improves the stability. Moreover, the neural network training process is kept constant for computation and updated gradually. The network training process is somewhat similar to supervised learning.

The performance of the implemented DQN algorithm is very satisfactory. One of the main reasons why this algorithm offers omnidirectional robot control is to develop an algorithm that can be used to control autonomous and industrial robots in homes and industrial machines. In industrial environments such as the 5S environment, Japan's 6S environment is applied to all factories. Comparing the DQN algorithm with other algorithms shows that the optimization of RL, Q-learning, etc., was also successful. Several tests have been implemented to control the process and clearly show how well the DQN algorithm works in different cases. Therefore, in order to build a complete DQN algorithm, the needs must always be met: from selecting control actions to multicast mobile robots, executing actions, getting the reward, storing and sampling to train the algorithm, and then going to the objective function calculation, from there update the model parameter by minimizing the loss function on all the selected samples, teach, followed by selecting the method of updating the neural network parameters and the objective and finally updating the control coefficients with uncertainty, [2, 3].

From there, we build the DQN algorithm with the MPC model as follows:

Algorithm: DQN Algorithm; MPC for self-driving car;

```
Initialize self-driving car;
Initialize model MPC;
Initialize the reward value;
for number of episodes do
    Reset simulation ;
    Wait for 1 second ;
    Pause simulation ;
    Read the pitch angle ϕ of the self-driving car ;
    state ← ϕ ;
    start simulation ;
    for number of iterations do
        Generate a random number rand;
        if rand ≤ δ then
            take random action ;
        end
        else
            Q ← M(state);
            action ← action formax(Q);
        end
        state_new ← ϕ;
        Pause simulation; self-driving car;
        if absolute value of state_new ≥ limit then
            if reward_total ≤ Target then
                reward ← pen;
                experience ←
                 (state, reward, action, state_new);
                Add Exp to Memory;
            end
        end
        else
            reward ← 1;
            experience ←
             (state, reward, action, state_new);
            Add Exp to Memory;
            state ← state_new
        end
    end
    Take radom minibatch of Experience;
    if reward = = pen then
        Q_pred ← reward;
    end
    else
        Q_pred ←
        reward + γmax(Q(state_new, action))
    end
    ;
    Train the model according to loss
    abs(Q_pred(state, action) – Q_pred(state, action));
end
```

4. Results and Discussion

After studying, calculating, and setting up the controller for the active suspension system, combined with the MPC model and DQN reinforcement learning algorithm, the author modeled the system with the dynamic structure of autonomous vehicles applied in traffic as above. We proceed to build a simulation model on Matlab Simulink 2021; The network training process is run on a high-spec Dell Gaming G3 3590 processor: Core i7 9750H, graphics card: Nvidia GTX1660 TI Ram 8GB DDR5; Version 21H1, with the following parameters:

Table 1. Parameters of ¼ car suspension system

m_1	m_2	k_1	k_2	b_1	b_2
973 kg	114 kg	92720 N/m	101115 N/m	1095 Ns/m	14.6 Ns/m

From there, we go to build a simulation model of the active suspension system combined with MPC predictive control and conduct the simulation; we have the following results:

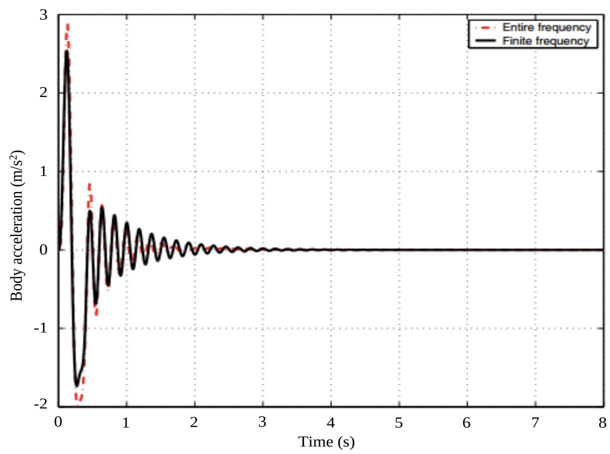

Fig. 6 Response to body acceleration with active suspension

Figure 6 shows the time-domain response of acceleration when considering body comfort for active suspension. Where the red line (Entire frequency) dotted represents the Entire frequency, and the solid black line (Finite frequency) represents a finite frequency.

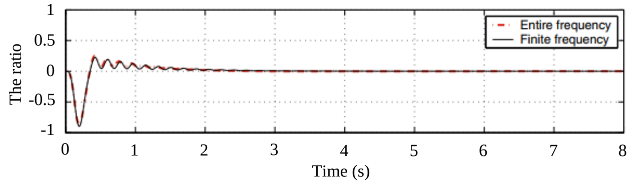

Fig. 7 The deflection of the active suspension system

Taking into account the ratio of deflection of the active suspension and the maximum limit for the system is shown in figure 7 above.

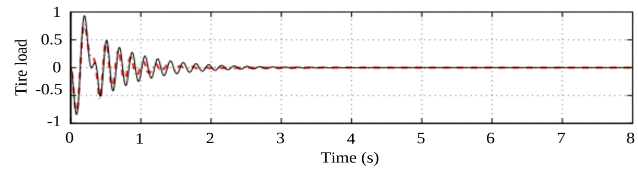

Fig. 8 Tire load relationship (traction)

Taking into account the tire load relationship (traction) for the vehicle suspension is shown in figure 8 above.

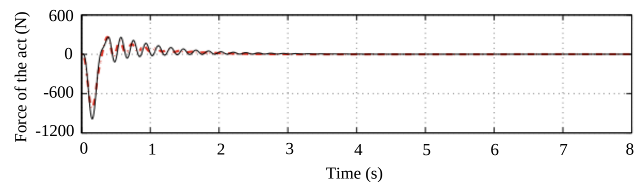

Fig. 9 Force of the actuator

When taken into account, the force relationship of the transmission (N) of the vehicle suspension is shown in figure 9 above.

When simulating and implementing reinforcement learning techniques in automatic control. We proceed to build a simulation model in Matlab Simulink 2021 (as shown in Figure 10) with the following parameter values: training episodes number is 3000; the average reward is 500; In fact, we can get more than 4000 episode numbers to know the accuracy of autonomous vehicles with DQN algorithm and MPC predictive control model. From there, we have the simulation results shown in Figure 11 with the following Training episodes and Average reward processes.

Figure 11 depicts the values obtained during training and the learning process. Despite the volatility due to the changing complexity of the environment and the efficiency of the control algorithms for the active suspension, traction; vehicle comfort; damping factor; load factor; tire problems; etc., these problems also show that the total reward grows by the total reward during the period in training. The red line (DQN) is the result of training the network; the blue line (CDQN) is the line that performs the comparison process based on the red line (DQN); the dark blue line (DQN1), and the purple line (DQN2) are the results of training the values on a two-stage basis with different components.

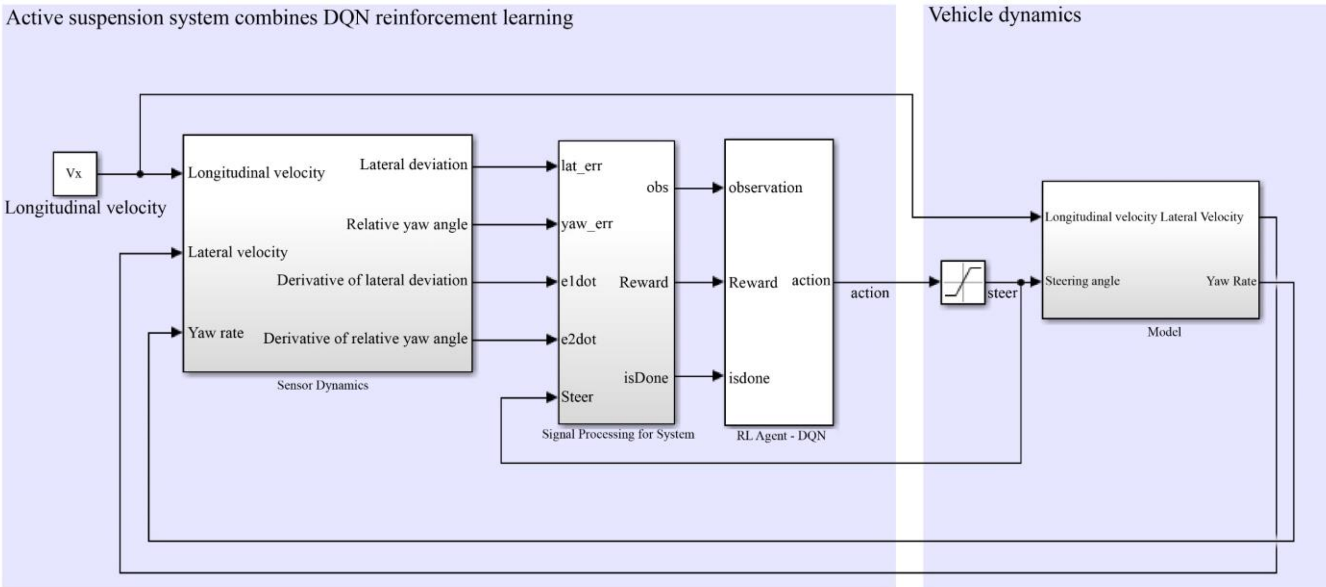

Fig. 10 The MPC predictive control model and DQN reinforcement learning techniques in automatic control

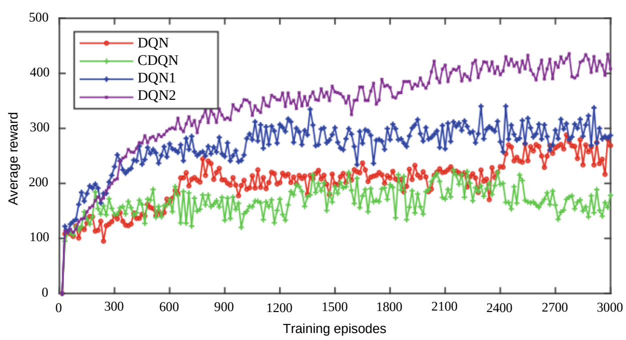

Fig. 11 Example of a small figure

5. Conclusion

The article has studied the theory of reinforcement learning techniques combined with building the MPC predictive control model, combined with DQN, enhanced jurisprudence in automatic control for the active suspension system of autonomous vehicles applied in public transport. The obtained results show that the correctness of the studied model is appropriate. From information about control problems, environment, operating position of autonomous vehicles, load problems, the comfort of occupants, transmission forces, and obstacles are determined, the combines DQN algorithm with the MPC model to calculate the motion trajectory for autonomous vehicles in the intelligent traffic navigation system to safely reach the destination without any obstacles. The simulation results on Matlab simulink software with the proposed algorithm have shown the practical effectiveness of the intelligent automatic navigation system for autonomous vehicles that the author has built. The development direction of the research problem is desired, which will be implemented on some practical four-wheeled self-propelled vehicles in the field of public transport and industrial plants today.

Acknowledgments

I would like to acknowledge my gratefulness and render my warmest thanks to my supervisor Associate Professor Lai Khac Lai, who did this work research. Associate Professor Lai Khac Lai's friendly guidance and expert advice are invaluable at all stages of the work.

References

[1] N. T. Luy, "*Textbook of Machine Learning and Intelligent Control Application*," Publishing House of Industrial University of Ho Chi Minh City. Ho Chi Minh, 2019.

[2] N. T. Tuan, *Base Deep learning*, The Legrand Orange Book. Version 2, last update, 2020.

[3] Charu C. Aggarwal, "Neural Networks and Deep Learning", Springer International Publishing AG, part of Springer Nature, 2018.

[4] Boris Houska, Hans Joachim Ferreau, and Mortiz Diehl, "An Auto-generated Realtime Iteration Algorithm for Nonlinear MPC in the Microsecond Range," *Automatica*, vol. 47, no. 10, pp. 2279-2285, 2011. [CrossRef] [Goggle Scholar] [Publisher link]

[5] Hiep Do Quang et al., "Design a Nonlinear MPC Controller for Autonomous Mobile Robot Navigation System Based on ROS," *International Journal of Mechanical Engineering and Robotics Research*, vol. 11, no. 6, pp. 379 - 388, 2022. [CrossRef] [Goggle Scholar] [Publisher link]

[6] Lai Thi Thanh Hoa, K. L. Lai, and Tran Ngoc Son, "Model Predictive Control for Vehicle Active Suspension Systems," *Advance in Engineering Research and Applications*, pp. 574–580, 2022. [CrossRef] [Goggle Scholar] [Publisher link]

[7] P. H. D. Duc, "Predictive control model and application of adaptive control of ship direction," Journal of Science & Technology of Technical Universities, no. 1(45), pp. 101-106, 2008.

[8] V.T. Minh, and A.A. Rashid, "Modeling and Model Predictive Control for Hybrid Electric Vehicles," *International Journal of Automotive Technology*, vol. 13, pp. 477– 485, 2012. [CrossRef] [Goggle Scholar] [Publisher link]

[9] Ahmad Reda, Ahmed Bouzid, and Jozsef Vasarhelyi, "Model Predictive Control for Automated Vehicle Steering," *Acta Polytechnica Hungarica*, vol. 17, no. 7, pp. 163-182, 2020. [CrossRef] [Goggle Scholar] [Publisher link]

[10] Hiep Do Quang et al., "An Approach to Design Navigation System for Omnidirectional Mobile Robot Based on ROS," *International Journal of Mechanical Engineering and Robotics Research*, vol. 9, no. 11, pp. 1502-1508, 2020. [CrossRef] [Goggle Scholar] [Publisher link]

[11] S. Di Cairano, and I. Kolmanovsky, "Automotive applications of model predictive control, Handbook of Model Predictive Control," *Control Engineering*, pp. 493-527, 2019.

[12] Veronica Adetola, and Martin Guay, "Robust Adaptive MPC for Constrained Uncertain Nonlinear Systems," *International Journal of Adaptive Control and Signal Processing*, vol. 25, no. 2, pp. 155-167, 2011. [CrossRef] [Goggle Scholar] [Publisher link]

[13] Nguyen Tien Ban, "The TUBE-MPC learning predictive control for the uncertain LURE system with the Lipschitz continuum", Journal of Science, Hai Phong University, no. 46, pp. 45-51, 2021.

[14] Mohammad Ali Nekoui, and Parisa Hadavi, "Optimal Control of an Active Suspension System," *Proceedings of 14th International Power Electronics and Motion Control Conference, EPE-PEMC 2010*, pp. 60–63, 2010. [CrossRef] [Goggle Scholar] [Publisher link]

[15] Huijun Gao, Weichao Sun, and Peng Shi, "Robust Sampled-data Control for Vehicle Active Suspension System," *IEEE Transactions Control Systems Technology*, vol. 18, no. 1, pp. 238–245, 2010. [CrossRef] [Goggle Scholar] [Publisher link]

[16] L.T.H. Linh, L. K. Loi, and N. T. M. Huong, "Model-based Predictive Control for Nonlinear Systems with a Forecasting Range of One," *Journal of Science and Technology of Thai Nguyen University*, no. 112(12)/2, p. 55-61, 2015.

[17] Xiaogang Ruan et al., "Mobile Robot Navigation based on Deep Reinforcement Learning," *2019 Chinese Control and Decision Conference*, 2019. [CrossRef] [Goggle Scholar] [Publisher link]

[18] Hiep Do Quang et al., "Mapping and Navigation with Four-wheeled Omnidirectional Mobile Robot Based on Robot Operating System," *2019 International Conference on Mechatronics Robotics and Systems Engineering*, pp. 54–59, 2019. [CrossRef] [Google Scholar] [Publisher link]

[19] Nguyen Thi Mai Huong et al., "Research on Optimization Strategies in Predictive Control," *Journal of Science and Technology of Thai Nguyen University*, vol. 113, no. 13, pp. 115 - 121, 2013.

[20] Hong Che, and Kong-Hui Guo, "Constrained Control of Active Suspensions: An LMI Approach," *IEEE Transactions Control Systems Technology*, vol. 13, no. 3, pp. 412–421, 2005. [CrossRef] [Google Scholar] [Publisher link]

[21] Yuankai Wu et al., "Deep Reinforcement Learning of Energy Management with Continuous Control Strategy and Traffic Information for a Series-Parallel Plug-in Hybrid Electric Bus," *Applied Energy*, vol. 247, pp. 454-466, 2019. [CrossRef] [Google Scholar] [Publisher link]

[22] Pablo Marin-Plaza et al., "Global and Local Path Planning Study in a ROS Based Research Platform for Autonomous Vehicles," *Journal of Advanced Transportation*, pp. 1-10, 2018. [CrossRef] [Goggle Scholar] [Publisher link]

[23] Haiping Du, and Nong Zhang, "Control of Active Vehicle Suspensions with Actuator time Delay," *Journal of Sound and Vibration*, vol. 301, no. 1-2, pp. 236–252, 2007. [CrossRef] [Google Scholar] [Publisher link]

[24] Anh Tuan Vo, Thanh Nguyen Truong, and Hee-Jun Kang, "A Novel Prescribed-Performance-Tracking Control System with Finite-Time Convergence Stability for Uncertain Robotic Manipulators," *Sensors*, vol. 22, no. 7, 2022. [CrossRef] [Google Scholar] [Publisher link]

[25] Hamid Khatibi, Alireza Karimi, and Roland Longchamp, "Fixed-order Controller Design for Polytopic Systems using LMIs," IEEE Transactions on Automatic Control, vol. 53,no. 1, pp. 428–434, 2008. [CrossRef] [Google Scholar] [Publisher link]

[26] Wen-Kung Tseng, and Hou-Yu Chen, "The Study of Tracking Control for Autonomous Vehicle," *SSRG International Journal of Mechanical Engineering*, vol. 7, no. 11, pp. 57-62, 2020. [CrossRef] [Publisher link]

[27] P. Mahalakshmi, and R. Vallikannu, "New Miniature Dual Mode Rat-race Coupler Design for Autonomous Vehicles," *International Journal of Engineering Trends and Technology*, vol. 70, no. 1, pp. 222-233, 2022. [CrossRef] [Publisher link]