Technology Introduction

Exploring the Potential of AI for Control Design

SHIBAHARA Anna

1 Introduction

Our society has seen a rapid spread of control systems that use artificial intelligence (AI) [Note 1], such as those used to control humanoid robots and autonomous vehicles. Around ten years ago, right after the beginning of the AI boom, various companies and organizations published proof of concept (PoC) results on AI-based control systems. At that time, most of these control technologies were simple, performing only one function. One example is robot arms grasping various objects. Many of these functions could not be implemented as actual products or services on their own, even though they could hardly be implemented with previous technology.

Since then, AI-based control systems have become increasingly sophisticated and complex. This technology has already begun to be commercialized in the form of products and services. Processing types that were previously difficult to implement are now almost realistic. For instance, it has become possible to teach industrial robots work procedures with minimal effort and deploy them on-site, or enhance a car's autonomous driving system to process a variety of edge cases that humans alone cannot design for. KYB should acquire and utilize these remarkably advanced AI-based control design technologies to develop more sophisticated and efficient control technology.

This research explores the potential of AI for control design with a focus on semi-active suspensions for four-wheeled automobiles (hereinafter, "semi-actives"), one of the company's products, in order to acquire AI-based control design technologies. This report presents the findings of the research.

Note 1) For the purpose of this report, AI is considered as machine learning.

2 Scope of This Research

Fig. 1 provides an overview of the scope of this research. We selected skyhook control to compare with AI-based control. Note that the following

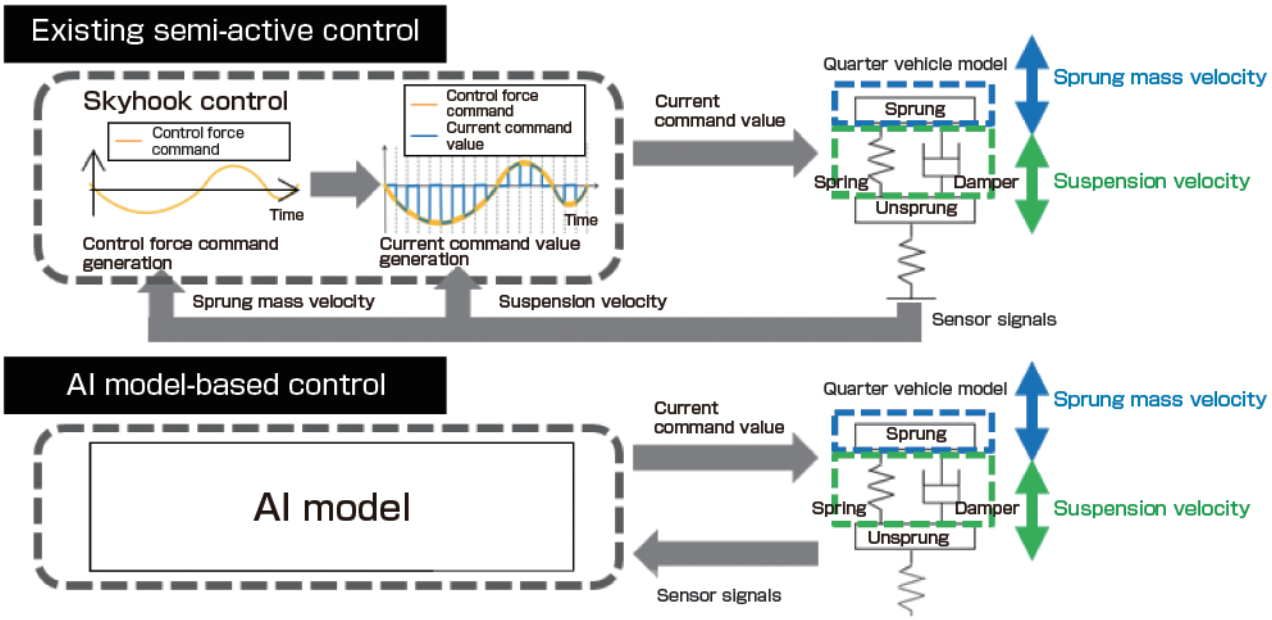

Fig. 1 Scope of this research

discussion assumes existing semi-actives are subject to skyhook control. Semi-actives can only deliver resistance (damping force) against the expansion and compression of the damper. Consequently, there are times when they are unable to damper the sprung mass. During these times, it is desirable to minimize the damping force. Thus, semi-actives involve non-linear control, which makes achieving control technology development quite difficult. In a sense, controlling semi-actives will benefit from the introduction of AI. Furthermore, if this difficult control design is implemented with AI, the control design of other components is highly likely to be implemented with AI as well.

This research evaluated the feasibility of the AI-based control design by determining if an AI-based control design (AI model) could achieve a performance level equal to or higher than skyhook control. All experiments in this research were conducted on a simulation platform. The system being controlled is a quarter vehicle model. The control provides current command to the semi-active damper to control the quarter vehicle model. The damping force of the semi-active damper is determined by these current command.

Skyhook control uses suspension velocity and sprung mass velocity as sensor signals. Sprung mass velocity is used to generate control force commands. The control force commands and suspension velocity are then used to generate the current command. These values are used to control the control target, i.e., the quarter vehicle model.

Meanwhile, the AI model-based control covers the range of the skyhook control domain, which is enclosed by a dotted line. Specifically, the AI model receives sensor signals from the target (the quarter vehicle model) and returns current command. For evaluation purposes, the performance of this AI model-based control is compared with that of the existing semi-active control.

3 AI Methods

3. 1 Reinforcement Learning

There are several different types of AI methods. The underlying problem settings and applications of AI vary by method. Groups of AI methods for certain problem settings correspond to learning paradigms. In this context, AI methods can be broadly divided into three types of learning paradigms: supervised, unsupervised, and reinforcement. Groups of methods belonging to each learning paradigm include more than one method that can learn control. To implement the AI model described in Chapter 2, AI methods capable of learning and acquiring control must be used. However, not all AI methods that can acquire control are acceptable. Some methods may be suitable for practical use, while others may not. One example of an unsuitable method is one that requires human-designed controls for the elements used to train the AI model.

Typical groups of AI methods that fall into this category include supervised learning. In supervised learning, the AI model uses a training dataset to output values that correspond to the input data. In many cases, preparing the necessary teaching data for supervised learning requires human intervention. An AI model trained with such a dataset essentially provides a control design similar to a human-designed one. Sophisticated control design is difficult to implement. Furthermore, this method requires additional processes to create a dataset and train the AI model, on top of human-designed control. Applying control design by humans to training may negate the benefits of using AI. Therefore, this research adopted another AI method classified as reinforcement learning, in which the AI model training does not involve human-designed control.

The basic concept of reinforcement learning is that the AI model accumulates experience through repeated trial and error. Based on this experience, the model derives an optimal action policy (i.e., control in this case). Fig. 2 shows the general mechanism of reinforcement learning.

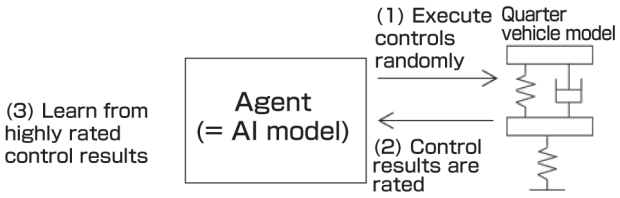

Fig. 2 Reinforcement learning mechanism

The problem setting in this research is the control of semi-actives. The reinforcement learning terminology corresponding to this problem setting is as follows:

(1) Control = Policy
(2) The output of the control results (the current command in this case) = Action
(3) The target to be controlled (the plant; the quarter vehicle model in this case) = Environment
(4) Sensor signals obtained by observing the environment (sprung mass velocity and suspension velocity obtained from quarter vehicle model simulation results in this case) = State

The following explains the reinforcement learning mechanism. The entity that executes actions for reinforcement learning is called an agent. First, the agent executes actions according to a random policy. Next, the actions are applied

to the environment, and the resulting "states" are rated. These ratings are called "rewards." The agent learns from policies that maximize rewards. The agent continues to learn from a combination of the learned good policies and random policies, thereby refining policies. Good and random policies are called exploration and exploitation, respectively. Exploration involves trial and error, while exploitation involves the agent achieving the best possible control. Exploration and exploitation cannot be executed at the same time, creating a trade-off between them. The agent should acquire a better policy by executing these two actions with a good balance.

Fig. 3 shows a conceptual diagram of learning programs necessary for implementing the reinforcement learning mechanism. There are three modules to be designed. The first module is the reward function, which evaluates the agent's actions. The design of this reward function determines which controls the agent will learn. The second module is the input to the agent. A feature extractor must be designed to select the type of input signals and process them so that the agent can easily learn. The third module is the agent itself. Among the various reinforcement learning methods, this research uses deep reinforcement learning, which uses a neural network to express policies. The agent's ability to learn and its performance limitations depend on the neural network's structure. Therefore, it is essential to determine the network structure.

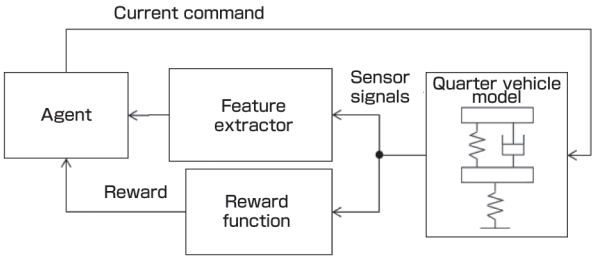

Fig. 3 Conceptual diagram of learning programs for reinforcement learning

3. 2 Development Environment

The key elements to be implemented in this research are quarter vehicle model simulation and reinforcement learning. These have all been implemented using MATLAB® and Simulink®, both of which are popular software programs used to build control models[Note 2].

Note 2) MATLAB® and Simulink® are registered trademarks of The MathWorks, Inc.

3. 3 Learning Algorithms Used in This Research

We used Proximal Policy Optimization (PPO) [1], a type of learning algorithm for deep reinforcement learning. Designed by OpenAI to achieve stability and efficiency in learning, PPO was released in 2017.

PPO is based on the actor-critic method. This method uses two models: The actor, which executes actions (a neural network in this case), and the critic, which evaluates the action value (the expected value of cumulative reward from actions taken in a given state according to the current policy). When the PPO algorithm is used, the agent has two neural networks.

In this research, the action spaces (a mathematical structure for defining a set of possible actions) are continuous, and the actor's policy follows a Gaussian distribution.

3. 4 Design of Reinforcement Learning System

We designed the three modules described in Section 3.1 on a trial basis. The resulting modules are as follows.

3. 4. 1 Reward Function

The reward function is defined to provide higher rewards when the maximum value of the group of the last set of sprung mass accelerations over a fixed interval is smaller.

3. 4. 2 Feature Extractor

Features, which are input to the agent, use sprung mass velocity and suspension velocity. These two values are adjusted in scale and added with their historical values before being input to the agent.

3. 4. 3 Agent

The actor and critic models use the same network structure with three internal layers (128, 64, and 32 units).

4 Training and Evaluation

4. 1 Progress of Training

Fig. 4 shows changes in the return on training (total reward per episode[Note 3]). The graph indicates that the return increased steadily before reaching a state of near convergence. It can be concluded that the training progressed stably.

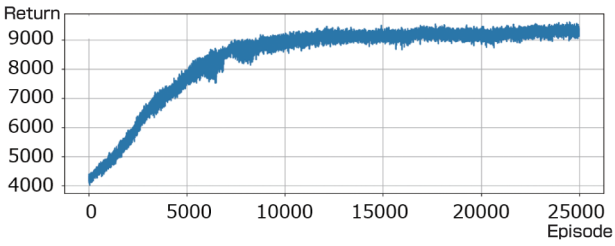

Fig. 4 Changes in return of training

Note 3) Episode: A unit indicating how many times the reinforcement training is repeated.

4. 2 Evaluation Methodology

After the episodes shown in Section 4.1, the training has produced a trained AI model (corresponding to the agent described in Section 3.1). This research uses data obtained through inference[Note 4)] using the trained AI model (a group of sprung mass velocity). Fig. 5 shows the AI model evaluation methodology used in this research. We used Partial Over All (POA), the value representing overall vibration magnitude within a specific frequency range, to evaluate performance. The evaluation procedure is as follows: First, we simulate driving the quarter vehicle model on a test road using the trained AI model for inference. The sprung mass velocity obtained from this simulation are used to calculate the power spectral density (PSD), which is the vibration magnitude for each frequency. To compare the performance of the AI model, we also simulated test driving under skyhook control conditions and fixed current command (no control). The PSD is also determined from the results of this simulation. The PSD values obtained in this way are used to determine the POA for the low and intermediate frequency ranges. The resulting POA values are plotted on two axes, one for each frequency range, for performance comparison. In the POA plot on the right side of Fig. 5, the red area indicates the POA for the fixed current command, and the blue area shows the POA for skyhook control. For the fixed current command, three points are plotted: soft damping force, hard damping force, and the intermediate point between soft and hard damping forces. For skyhook control, 12 points are plotted at different skyhook control gain levels (-3,000, -5,000, -10,000, -20,000, ……, -100,000). The POA plot shows higher performance in the lower left of the diagram. Therefore, the learning program for the AI model has been designed so that the POA of the evaluation results is positioned as far as possible in the lower left of the diagram.

Note 4) Inference: Refers to controlling semi-actives without updating or exploring policies.

4. 3 Evaluation Results

Fig. 6 shows the POA of sprung mass velocity for the AI model, skyhook control, and fixed current command. Fig. 7 shows a one-second excerpt of the trend in current command for the AI model and a skyhook control that has the POA closest to that of the AI model (-30,000 gain). According to Fig. 6, the POA for the AI model is nearly equal to the POA for the skyhook control with a gain of -30,000. Additionally, as shown in Fig. 7, the AI model outputs current command in a trend close to that of the relevant skyhook control. These results demonstrate that the AI model exhibits control behavior similar to the skyhook control in terms of control performance and current command trends. In other words, we believe that control design using AI is feasible for achieving control performance comparable to that of existing semi-actives controls.

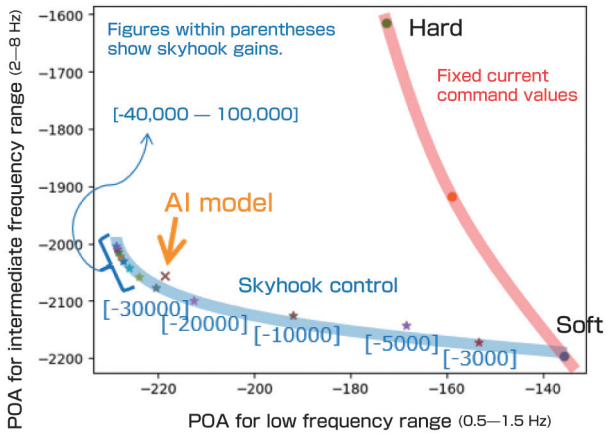

Fig. 6 POA of sprung mass velocity for AI model, skyhook control, and fixed current command

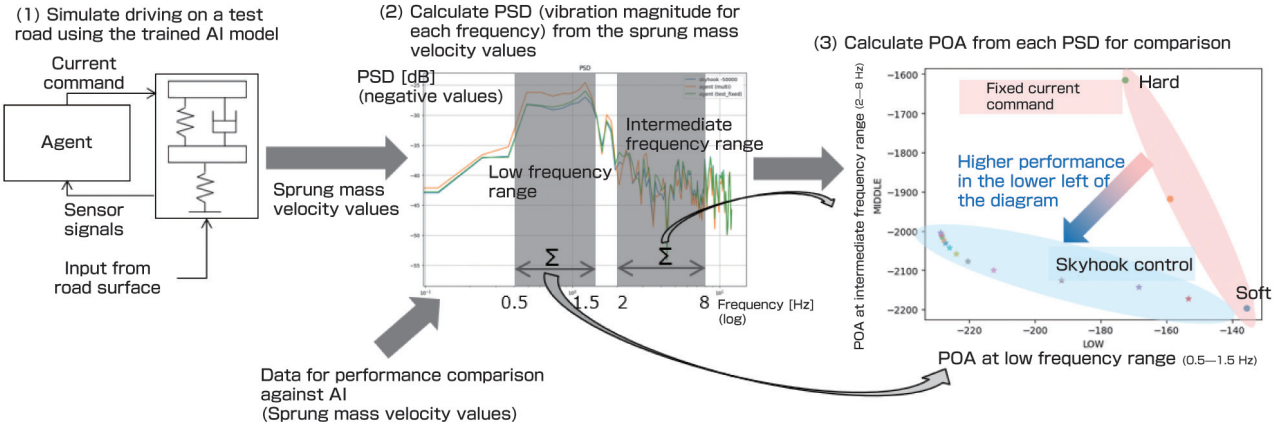

Fig. 5 AI model evaluation methodology

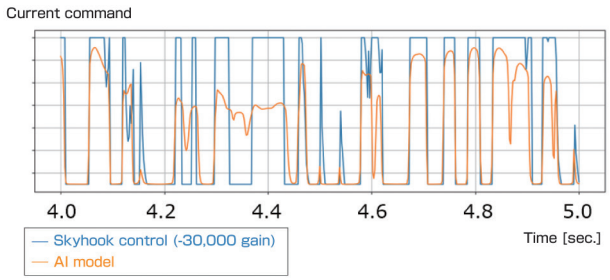

Fig. 7 Trend in current command for the AI model and skyhook control

However, it should be noted that this research used only one type of road surface, raising concerns about overfitting. To address this, our future research will test controls in various scenes according to the target to be controlled.

5 In Closing

This research explored the potential of AI for control design, focusing on the design of semi-actives. As a result, we obtained an AI model that delivers performance nearly equal to the skyhook control, which we selected as the target for comparison. This model also outputs current command similar to those of the skyhook control. These results provide basic technology for control design using AI.

This research was conducted within a limited scope. We will transition to more practical research to effectively apply this technology to actual control designs.

Reference

1) Scholarly articles on PPO are available on the journal submission platform arXiv: https://arxiv.org/abs/1707.06347.

Author

SHIBAHARA Anna

Joined the company in 2015.
Motion Control R&D Sect., Basic Technology R&D Center, Engineering Div.
Engaged in research and development of control technology using AI.