Ball and Beam Control using Adaptive PID based on Q-Learning

Brilian Putra Amiruddin
Department of Electrical Engineering
Institut Teknologi Sepuluh Nopember
Surabaya, Indonesia
brilianamiruddin.17071@mhs.its.ac.id

Rusdhianto Effendi Abdul Kadir
Department of Electrical Engineering
Institut Teknologi Sepuluh Nopember
Surabaya, Indonesia
ditto@ee.its.ac.id

***Abstract*—The ball and beam system is one of the most used systems for benchmarking the controller response because it has nonlinear and unstable characteristics. Furthermore, in line with the increasing of computation power availability and artificial intelligence research intensity, especially the reinforcement learning field, nowadays plenty of researchers are working on a learning control approach for controlling systems. Due to that, in this paper, the adaptive PID controller based on Q-Learning (Q-PID) was used to control the ball position on the ball and beam system. From the simulation result, Q-PID outperforms the conventional PID and heuristic PID controller technique with the swifter settling time and lower overshoot percentage.**

Keywords—ball and beam, reinforcement learning, adaptive PID controller, q-learning

I. Introduction

The ball and beam system is one of the popular systems used for representing and learning the control systems from classical control engineering through the modern control engineering regime. This system behavior is nonlinear and unstable, due to these characteristics, this system can be used as a benchmark platform of control design performance effectiveness. It is composed of two main parts which are a beam and a ball. The beam is designed to whirl in the vertical plane by giving torque to it and as well utilized for the place of the ball to spin throughout it [1]. The main control objective of the ball and beam system is to move the ball towards the desired position.

In line with the rapid growth of the artificial intelligence field and the availability of vast computing resources. Lately, various control algorithms are combined with machine learning methods such as reinforcement learning (RL), one of the reinforcement learning algorithms which frequently employed towards the control task is Q-Learning. Due to that, Pandian et al. [2] has applied Q-Learning and Q-Learning with policy function approximation to control the ball and beam system. The Q-Learning with policy approximation is implementing an Artificial Neural Network (ANN) to obtain the policy function for each iteration and use the ANN as the controller. Yet, the response still did not meet the expectations. Both of the controller settling times are relatively slow, 10.21 s for Q-Learning controller, and 11.17 s for Q-Learning with the policy approximation controller. Besides, the steady-state error from those two controllers is also high, 2 cm to Q-Learning controller, then 1 cm to Q-Learning with the policy approximation controller. Furthermore, the system is also experiencing large oscillations during the simulation.

From that background, in this paper, instead of merely either using the Q-Learning itself or combined with an artificial neural network for controlling the ball position. An adaptive PID controller based on the Q-Learning (Q-PID) algorithm is proposed to control the ball and beam system to solve the prior research drawbacks. To the best of the author's knowledge, there are no results in the literature regarding the implementation of the Q-PID controller to control the ball and beam system.

Later, the performance and the capability of the Q-PID controller are compared and analyzed with two other controllers, which are conventional PID controller and heuristic controller based on the Adaptive Neuro-Fuzzy Inference System inverse control-PID (ANFIS-PID). Thus from the performance analysis results, novel insight about the effectiveness of the Q-PID controller for the ball and beam system will be gotten. The upcoming section of the paper was ordered as follows, Section II is presenting the related work of ball and beam control problem. Section III is giving a basic understanding of ball and beam system modeling, the Q-PID controller, and the controller performance analysis workflow. Section IV is prepared for the simulation and performance comparison results and the discussion of it, and Section V is used for the conclusions and the future works.

II. Related Works

Few researchers had been working on ball and beam control systems, applying several methods of the control algorithm to control the ball and beam system from the conventional control algorithms to the modern one. Ali et al. [3] have implemented Linear Quadratic Gaussian (LQG) for tracking control of the ball and beam system, and the response results were excellent. Another researcher [4], have used the Fuzzy PID control algorithm to control the ball and beam system, the controller applied 49 fuzzy rules based on the Mamdani Fuzzy Inference System to adjust the PID controller gain. Also, Kharola et al. [5] have employed a different fuzzy or soft computing control approach, control the ball and beam system using the ANFIS controller. Ezzabi et al. [6] are using the nonlinear robust adaptive fuzzy backstepping controller to control the ball position.

In [7], using Nonlinear Model Predictive Control, they successfully control the ball and beam systems, which achieved superior time domain response compared to the

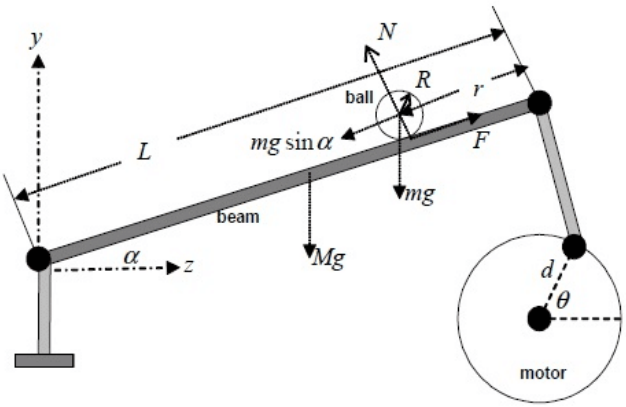

Fig. 1. The Ball and Beam Systems Free Body Diagram [12]

TABLE I. SYSTEM PARAMETERS

Name	System Variables and State	
	Value or Range	
Ball Mass (m)	0.1	kg
Ball Radius (R)	0.015	m
Level Arm Offset (d)	0.03	m
Beam Length (L)	0.8	m
Gravitational Acceleration (g)	9.8	m/s^2
Moment Inertia of Ball (I)	9.99 x 10^{-6}	kgm^2
Ball Position Coordinate (r)	[-0.4, 0.4]	m
Ball Velocity ($\dot{r}$)	[-15, 15]	m/s
Beam Angle Coordinate (α)	[-π/4, π/4]	rad
Beam Angle Coordinate Change ($\dot{\alpha}$)	[-15, 15]	rad/s

Neural Network controller. The experiment by Liqing et al. [8], proposed a Back Propagation Neural Network based controller to control the ball and beam systems, the controller was trained and designed from the root locus controller response. As well, the control algorithm for the ball and beam using robust sliding mode control methods have been developed by Soni et al. [9]. Later, the other research by Aburakhis et al. [10] have applied fractional-order adaptive law and fractional order PID controller instead of only using the PID controller to cope with the ball and beam system. Ding et al. [11], on their research, have analyzed the effectiveness of active disturbance rejection control towards position control on the ball and beam system.

III. MATERIAL AND METHODS

A. Ball and Beam System

The ball and beam system consists of two main parts which are the ball and the beam. The ball and beam system was modeled using Lagrangian dynamics. The slipping and the friction among the ball were presumed to zero. The ball and beam system used was nonlinear and unstable in nature. Fig. 1 shows the ball and beam system free body diagram, and Table I shows the system parameters, including the system variables value and the system states and its range. The ball and beam system equation and state were derived as (1) – (7),

Total torque which applied to the ball rotation

$$I\ddot{\alpha} = F.R \tag{1}$$

The ball rotation angle denoted as

$$\alpha = -\frac{r}{R} \tag{2}$$

Substituting (1) to (2) yield

$$F = -\frac{I}{R^2}\ddot{r} \tag{3}$$

The total force working on the ball

$$F - mg\sin\alpha = m(\ddot{r} - r\dot{\alpha}^2) \tag{4}$$

Thus, by substituting (4) to (3) the Lagrangian equation for the system yield

$$0 = \left(\frac{I}{R^2} + m\right)\ddot{r} + mg\sin\alpha - mr\dot{\alpha}^2 \tag{5}$$

Because the torque was applied immediately to the beam from the servo, the servo gear angle should be converted to beam angle to obtain the system state as (6)

$$\alpha = \frac{d}{L}\theta \tag{6}$$

So, from that then the system state could be defined as follows,

$$\begin{bmatrix} \dot{r}(t) \\ \ddot{r}(t) \\ \dot{\alpha}(t) \\ \ddot{\alpha}(t) \end{bmatrix} = \begin{bmatrix} \dot{r}(t) \\ \dfrac{mg\sin\alpha(t) - mr\dot{\alpha}(t)^2}{\left(\frac{I}{R^2} + m\right)} \\ \dot{\alpha}(t) \\ 0 \end{bmatrix} \tag{7}$$

B. Adaptive PID Q-Learning Based Controller

PID controller is a straightforward control algorithm that is widely used in many real-world applications because of its simplicity and capability to control a bunch of systems with easy tunability. The equation of the PID controller is written on (8),

$$u(k) = K_P e(k) + K_I \sum_{i=0}^{k} e(k)\Delta_k + K_D \frac{e(k) - e(k-1)}{\Delta_k} \tag{8}$$

Where K_P is the proportional gain, K_I the integral gain, K_D the derivative gain, Δ_k is the sampling time of the controller, and $e(k)$ is the system error compared to reference on k-th sampling time, respectively. Equation (9) is denoting the error equation

$$e(k) = r_{ref} - r_k \tag{9}$$

With the latter swift expansion and growth of reinforcement learning field, marked with several prominent

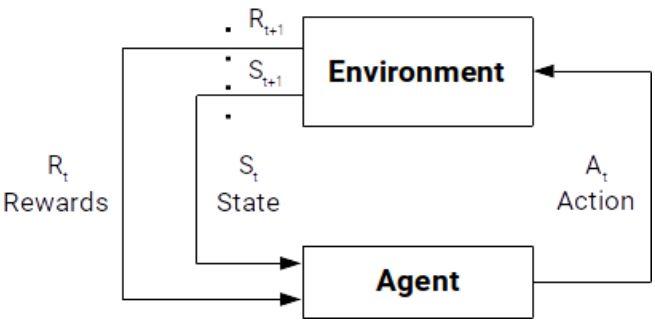

Fig. 2. Markov Decision Processes

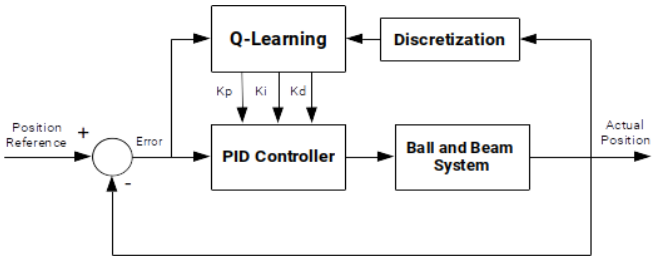

Fig. 3. The Q-PID Controller Block Diagram

research carried out by many researchers, for instance, the success of Google DeepMind AlphaGo to beat the European Go champion Fan Hui [13]. It also won against the world champion of Go Lee Sedol back in 2016 [14]. From that, researchers believed that the reinforcement learning field is a promising subject. Following that success, several researchers have developed a novel control algorithm that is either purely based on reinforcement learning, for example, the Deep Deterministic Policy Gradient (DDPG) [15] or combined with existing control algorithms such as PID controller. Reinforcement Learning (RL) algorithms have mainly relied on Markov Decision Processes (MDP), which is shown in Fig. 2.

Q-Learning is one example of value-based, and off-policy RL algorithm, off-policy means that instead of following current policy the Q-Learning algorithm will immediately estimate the action-value function (Q) which is optimal by using following update rule as denoted in (10) [16],

$$Q(S_t, A_t) \leftarrow Q(S_t, A_t) + \vartheta\left[R_{t+1} + \gamma \max_a Q\,(S_{t+1}, a) - Q(S_t, A_t)\right] \quad (10)$$

Where Q is the learned action-value function, ϑ is the learning rate, S_t the agent state, A_t is the agent action at instance t, γ the discount factor, and R_{t+1} the observed reward.

In this paper, the Q-Learning algorithm was used to autotune the PID controller gain over sampling time, so the gain of the PID controller will change and adapt over time. This controller was based on Shi et al. [17]. However, in [17], the proposed Q-PID algorithm was applied to a cart pole problem, which was the regulator problem. Several modifications on the block diagram and algorithm (reward scheme) as well on several parameters constant are needed to make the Q-PID control algorithm precisely can be used on tracking problems such as ball and beam position control. The block diagram of the controller (Q-PID) for controlling

TABLE II. Q-PID CONTROLLER PARAMETERS

Notation	**System Variables**
	Values or Range
K_P Variation	[0, 200]
K_I Variation	[0, 2]
K_D Variation	[0, 190]
Max Episodes	500
Max Simulation Time	10 s
Discount Factor γ	0.99
Learning Rate ϑ	0.02
Number Actions for Q-Tables	55
Discretization Bucket N	25
Sampling Time	0.02 s
Initial Ball Position State	0 m

the ball and beam system is shown in Fig. 3. The Q-PID controller training parameters are shown in Table II.

The Q-PID controller was trained on one fixed position with the reference point of step signal which had the position reference $r(t)$ at 0.38 m. The learning rate of the Q-PID controller was updated using the Delta-Bar-Delta rule [18]. The Q-Learning reward scheme (R_t) was modified as (11).

$$R_t = \begin{cases} 2, & if\ |e_{k+1}| < 0.001 \\ 1, & if\ |e_{k+1}| < |e_k|\ and\ |e_{k+1}| > 0.05 \\ 0, & otherwise \end{cases} \quad (11)$$

Where e_k is denoting the position error at the time step k. In this problem, there are three trained Q-tables, and each Q-table represent one PID controller gain that controls the ball position, as well for every Q-table takes discretized ball position as a state $n_1(k)$. The following is the learning process of Q-PID algorithm on ball and beam system:

Step 1: Initialize all of the $Q_w(s,a)$ value to zero, episode (eps) to 1, learning rate (ϑ), and the exploration rate (ε). Then, set the maximum episode and the maximum time of the simulation.
Step 2: Set the time step $t = 0$, next initialize the $S_t(r(t),\ \dot{r}(t),\ \alpha(t), \dot{\alpha}(t))$. Decay the exploration rate (ε) as in Equation (12).

$$\epsilon_{eps} = \begin{cases} \dfrac{1}{1 + e^{eps}} & eps < 0.6 * maxepisode \\ 0 & otherwise \end{cases} \quad (12)$$

Step 3: Increment the time step (t) by 1, discretize the state S_t to acquire $n_1(t)$.
Step 4: From the discretized state $n_1(t)$ iterate $w = 1$ to $w = 3$, later at every iteration, select the action (A_w) which following the epsilon-greedy policy.
Step 5: Acquire the total controller output $u(t)$ according to Equation (8).
Step 6: Observe the new state $S_{t+1}(r(t+1),\ \dot{r}(t+1),\ \alpha(t+1), \dot{\alpha}(t+1))$, receive the reward ($R_t$) for $Q_1(s,a)$, $Q_2(s,a)$, $Q_3(s,a)$ corresponding to Equation (11).
Step 7: Discretize the state S_{t+1}, obtain the $n_1(t+1)$.
Step 8: Update the learning rate (ϑ) to $Q_1(s,a)$, $Q_2(s,a)$, $Q_3(s,a)$.
Step 9: Update $Q_1(s,a)$, $Q_2(s,a)$, $Q_3(s,a)$ using the obtained reward (R_t) and learning rate (ϑ).
Step 10: Set the S_t equal to S_{t+1}, if the time step (t) is equal to the maximum simulation time, then back to Step 2, if not

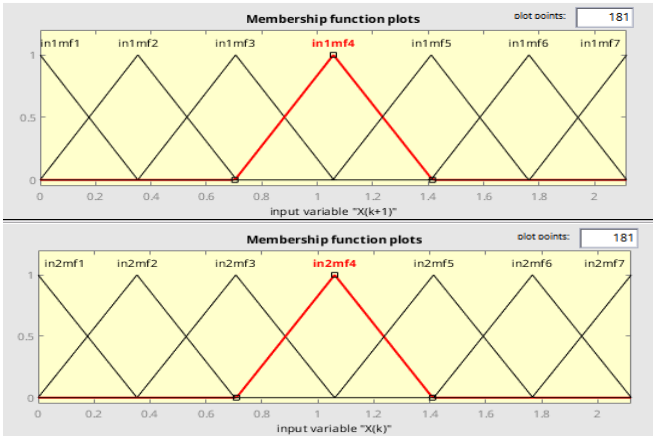

Fig. 4. The ANFIS Membership Functions

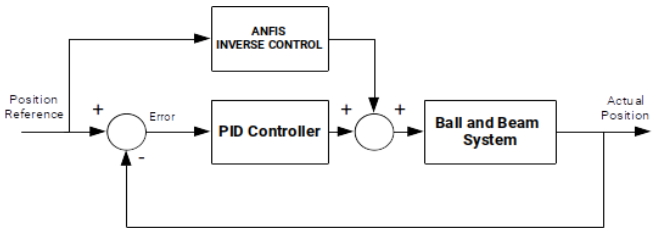

Fig. 5. The ANFIS-PID Controller Block Diagram

back to Step 3. When the maximum episode is reached; thus, the learning process is done.

C. Performance Analysis

To make sure that the proposed controller is working effectively, the response of the Q-PID controller was compared with the PID controller, which was tuned using the Ziegler Nichols method [19] and an Adaptive Neuro-Fuzzy Inference System inverse control combined with PID (ANFIS-PID) controller [20]. The comparison was using several time-domain metrics such as rise-time, settling-time, steady-state error, and the overshoot percentage. As well, for reference signal tracking, the mean square error (MSE) and the mean absolute deviation (MAD) were used.

Furthermore, after the Ziegler Nichols tuning method applied to the ball and beam system, the controller gains were obtained as follows, $K_P = 6$, $K_I = 1.565$, and $K_D = 5.859$ also $\Delta_k = 0.02s$.

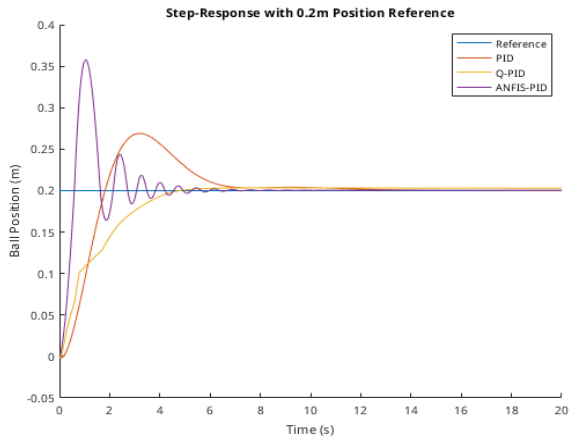

Fig. 6. Step Response with Position Reference 0.2 m

Alongside that, the ANFIS-PID controller was trained using 10 epochs and the membership function for the ANFIS, as shown in Fig. 4. The triangle membership functions were used with a 7x7 rule on it, so the total rule was 49. The ANFIS training process resulted in 0.43 on the training set error. Also, the PID controller gains which are applied to ANFIS-PID are equivalent to the aforementioned Ziegler Nichols tuned PID controller. Fig. 5 shows the ANFIS-PID inverse control block diagram for controlling the ball and beam system.

IV. Results and Discussion

Subsequent to the simulation, which was executed using MATLAB software, the first simulation was testing the controller response using a unit step signal with the ball reference position $r(t)$ of 0.2 m. Fig. 6 shows the step response of each controller tested with unit step reference position of 0.2 m.

Then, the step response of each controller was calculated to found the time domain response. Table III shows the controller time-domain response for all controllers. The step response results that were gotten shown the proposed controller or Q-PID was having the fastest settling time and the lowest overshoot percentage. But the Q-PID controller steady-state error was greater than the PID controller, which was tuned using the Ziegler Nichols method, and also, the Q-PID controller rise-time was the slowest among the other controllers. However, the rise-time and the steady-state error of Q-PID were not the best performed compared to the two other controllers. Nevertheless, the steady-state error percentage of Q-PID was smaller than the ANFIS-PID controller. Then after the controllers were tested on the unit step signal, the controllers as well examined on the step-wise signal with three different ball position references $r(t)$ (0.2 m, 0 m, and 0.1 m).

TABLE III. Controller Time-Domain Response

Controller	**Step Response Properties**			
	Rise-Time (s)	***Settling-Time (s)***	***Steady State Error***	***Overshoot***
Q-PID	2.99	4.24	1.44%	1.65%
PID	1.2	6.97	0.05%	34.4%
ANFIS-PID	0.42	4.82	2%	78.78%

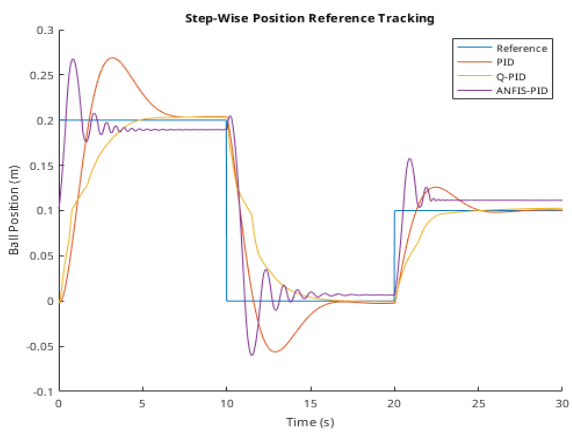

Fig. 7. Step Wise Position Reference Tracking

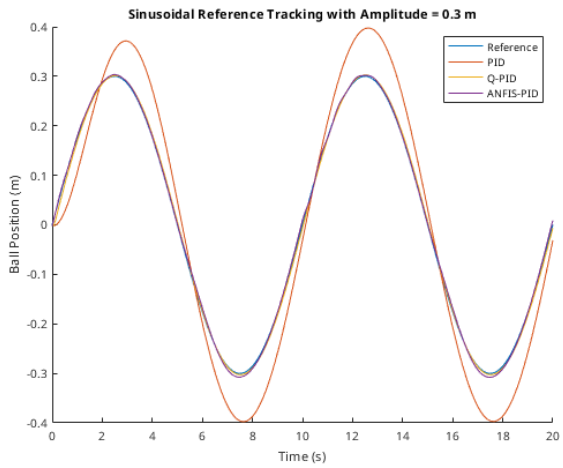

Fig. 8. Sinusoidal Position Reference Tracking

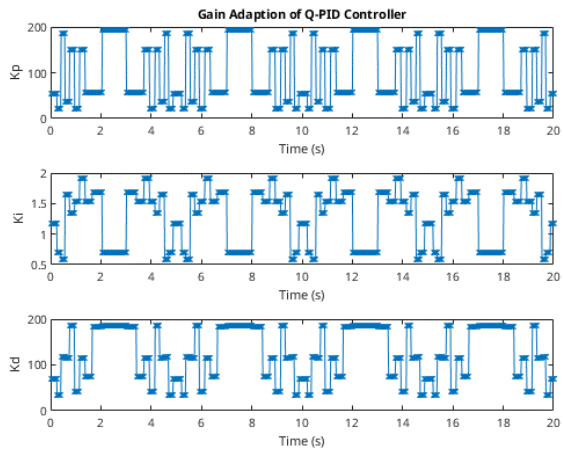

Fig. 9. Q-PID Gain Adaption on Sinusoidal Reference Tracking

TABLE IV. POSITION TRACKING ERROR

Controller	**Tracking Error Measurement**	
	Mean Square Error (MSE)	***Mean Absolute Deviation (MAD)***
Q-PID	2.7406 x 10^{-5}	0.0045
PID	0.005	0.0641
ANFIS-PID	1.8949 x 10^{-5}	0.0036

The outcomes of the examination shown in Fig. 7. The last assessment for the controllers was sinusoidal position reference tracking with an amplitude of 0.3. The outcomes of the tracking were shown in Fig. 8. Q-PID controller gains adaption for the sinusoidal reference tracking test, as well as shown in Fig. 9.

Moreover, in the sinusoidal signal reference tracking case, the controllers tracking response were also analyzed more in-depth by calculating the MSE and MAD of the controlled response as opposed to the given sinusoidal reference signal. The yielded results were shown in Table IV. The Q-PID controller MSE and MAD were less than the PID controller but greater if it was compared to ANFIS-PID. However, the tracking error of the Q-PID controller was nonetheless small because it was nearly zero.

Furthermore, after the outcomes were evaluated, it demonstrated that the Q-PID controller response for controlling the ball and beam system was outstanding, even though the controller was not entirely exceptional in every tested aspect. Yet, the controller did not require prior information about the system at all. Conversely, the ANFIS-PID needed the model of the plant or the ball and beam system in the controller design.

Although the ANFIS-PID had prior information about the system model, the controller accomplishment was not too significant compared to the Q-PID controller. Compared to the other RL-based controller (Q-Learning controller and Q-Learning with policy function approximation controller) response for controlling the ball and beam system in [2]. The Q-PID response in this experiment is far smoother and also superior in every tested time-domain criteria (settling time and rise time). Hence, from that, the proposed Q-PID controller successfully applied to control the position of the ball on the ball and beam system and outperformed the other tested controller in several metrics.

V. CONCLUSIONS AND FUTURE WORK

To sum up this paper, the ball and beam system was effectively controlled with the Q-PID controller. Even though it was initially trained on one fixed position reference point, the Q-PID controller can be used on every position reference point. The response of the Q-PID controller outperforms the conventional PID controller and heuristic method controller (ANFIS-PID) in several aspects, with detail the overshoot percentage and the settling time. In addition, though the ANFIS-PID controller tracking error (MSE and MAD) was smaller than the Q-PID, the overshoot of the ANFIS-PID controller was quite significant, and a response like this is undesired.

In forth work, our RL-based controller for the ball and beam system will be developed. With intention, the response result of our controller will be compared with the other RL-based controller, such as Deep Deterministic Policy Gradient (DDPG), Q-PID, and the other heuristics control method so our RL based controller effectiveness can be compared and benchmarked.

REFERENCES

[1] C. Aguilar-Ibañez, M. S. Suarez-Castanon, and J. de J. Rubio, "Stabilization of the Ball on the Beam System by Means of the Inverse Lyapunov Approach," *Math. Probl. Eng.*, vol. 2012, pp. 1–13, 2012, doi: 10.1155/2012/810597.

[2] B. J. Pandian, S. T. Kumar, and M. M. Noel, "Q-LEARNING WITH POLICY FUNCTION APPROXIMATION FOR A BENCHMARK BALL AND BEAM CONTROL PROBLEM," p. 10.

[3] R. Ali, F. M. Malik, M. Liaqat, and M. Shah, "Ball and Beam Tracking Application with Linear Quadratic Control Design," in *2019 8th International Conference on Systems and Control (ICSC)*, Marrakesh, Morocco, Oct. 2019, pp. 153–157, doi: 10.1109/ICSC47195.2019.8950644.

[4] N. S. A. Aziz, R. Adnan, and M. Tajjudin, "Design and evaluation of fuzzy PID controller for ball and beam system," in *2017 IEEE 8th Control and System Graduate Research Colloquium (ICSGRC)*, SHAH ALAM, Malaysia, Aug. 2017, pp. 28–32, doi: 10.1109/ICSGRC.2017.8070562.

[5] A. Kharola and P. P. Patil, "Soft-Computing Control of Ball and Beam System:," *Int. J. Appl. Evol. Comput.*, vol. 9, no. 4, pp. 1–21, Oct. 2018, doi: 10.4018/IJAEC.2018100101.

[6] A. A. Ezzabi, K. C. Cheok, and F. A. Alazabi, "A nonlinear backstepping control design for ball and beam system," in *2013 IEEE 56th International Midwest Symposium on Circuits and Systems (MWSCAS)*, Columbus, OH, USA, Aug. 2013, pp. 1318–1321, doi: 10.1109/MWSCAS.2013.6674898.

[7] D. Martinez and F. Ruiz, "Nonlinear model predictive control for a Ball&Beam," in *2012 IEEE 4th Colombian Workshop on Circuits and*

Systems (CWCAS), Barranquilla, Colombia, Nov. 2012, pp. 1–5, doi: 10.1109/CWCAS.2012.6404073.

[8] Gao Liqing and Liu Yongxin, "Design of BP neural network controller for ball-beam system," in *2016 IEEE Advanced Information Management, Communicates, Electronic and Automation Control Conference (IMCEC)*, Xi'an, China, Oct. 2016, pp. 1087–1091, doi: 10.1109/IMCEC.2016.7867379.

[9] R. Soni and Sathans, "Robust Control of a Ball and Beam System Through Sliding Mode Controller," in *2018 International Conference on Emerging Trends and Innovations In Engineering And Technological Research (ICETIETR)*, Ernakulam, Jul. 2018, pp. 1–5, doi: 10.1109/ICETIETR.2018.8529082.

[10] M. Aburakhis and R. Ordonez, "Interaction of Fractional Order Adaptive Law and Fractional Order PID Controller for The Ball and Beam Control System," in *NAECON 2018 - IEEE National Aerospace and Electronics Conference*, Dayton, OH, Jul. 2018, pp. 451–456, doi: 10.1109/NAECON.2018.8556774.

[11] M. Ding, B. Liu, and L. Wang, "Position control for ball and beam system based on active disturbance rejection control," Syst. Sci. Control Eng., vol. 7, no. 1, pp. 97–108, Jan. 2019, doi: 10.1080/21642583.2019.1575297.

[12] Lucas Niro, Marcio Aurelio Furtado Montezuma, Bruno Masaharu Shimada, Fabian Andres Lara-Molina, Edson Hideki Koroishi, and Lucia Valeria Ramos de Arruda, "Design of an eigenstructure assignment control using Genetic Algorithm applied to a Ball and Beam system," presented at the 23rd ABCM International Congress of Mechanical Engineering, Rio de Janeiro, Brazil, 2015, doi: 10.20906/CPS/COB-2015-0668.

[13] D. Silver *et al.*, "Mastering the game of Go without human knowledge," *Nature*, vol. 550, no. 7676, pp. 354–359, Oct. 2017, doi: 10.1038/nature24270.

[14] D. Silver *et al.*, "Mastering the game of Go with deep neural networks and tree search," *Nature*, vol. 529, no. 7587, pp. 484–489, Jan. 2016, doi: 10.1038/nature16961.

[15] T. P. Lillicrap *et al.*, "Continuous control with deep reinforcement learning," *ArXiv150902971 Cs Stat*, Jul. 2019, Accessed: Jul. 18, 2020. [Online]. Available: http://arxiv.org/abs/1509.02971.

[16] R. S. Sutton and A. G. Barto, *Reinforcement learning: an introduction*, Second edition. Cambridge, Massachusetts: The MIT Press, 2018.

[17] Q. Shi, H.-K. Lam, B. Xiao, and S.-H. Tsai, "Adaptive PID controller based on Q-learning algorithm," *CAAI Trans. Intell. Technol.*, vol. 3, no. 4, pp. 235–244, Dec. 2018, doi: 10.1049/trit.2018.1007.

[18] R. A. Jacobs, "Increased rates of convergence through learning rate adaptation," *Neural Netw.*, vol. 1, no. 4, pp. 295–307, Jan. 1988, doi: 10.1016/0893-6080(88)90003-2.

[19] C. C. Hang, K. J. Åström, and W. K. Ho, "Refinements of the Ziegler–Nichols tuning formula," *IEE Proc. Control Theory Appl.*, vol. 138, no. 2, p. 111, 1991, doi: 10.1049/ip-d.1991.0015.

[20] J.-S. R. Jang and Chuen-Tsai Sun, "Neuro-fuzzy modeling and control," *Proc. IEEE*, vol. 83, no. 3, pp. 378–406, Mar. 1995, doi: 10.1109/5.364486.