Indian Institute of Technology, Ropar

Bachelor of Technology in Chemical Engineering

CP303: Capstone Project

Design of Reinforcement Learning Based Controller for Dynamic Nonlinear Chemical Processes

May 5, 2024

Author
Kirti Sharma

Student ID
2020CHB1043

Contents

1 Introduction

In process industries, the main goal is to operate the process variables like temperature, pressure, flow rates and composition near or close to the set points. Controllers are used to achieve this objective. Controllers can be divided into three main categories - Classical controller, Advanced controller and Intelligent Controller. Classical controllers include Proportional (P), Proportional Integral (PI) and Proportional Integral Derivative (PID) controllers whereas advanced controllers are model predictive controllers.

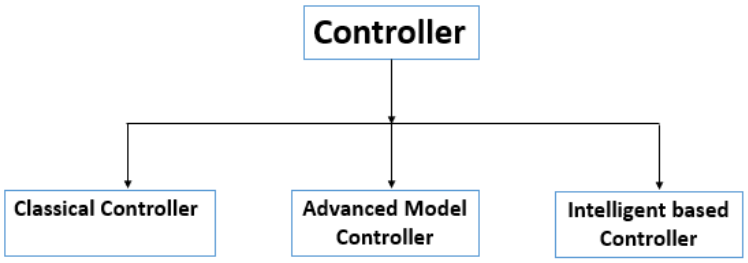

Figure 1: Classification of Controllers

There are two main types of system - Single Input Single Output (SISO) and Multiple Input Multiple Output (MIMO). Most of the systems found in the industries are complex and involves multiple inputs and multiple outputs i.e MIMO system. An important feature of this system is that there is a significant process interaction occurring among the process variables i.e. each manipulated variable affects each controlled variable [1]. The Fig. 2 shows a MIMO system where there are multiple variables that can be manipulated and 'n' corresponds to the number of variables that can be controlled.

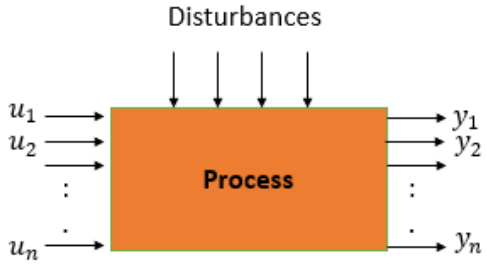

Figure 2: MIMO process

Complexity in controlling the system is caused due to the significant process interactions. A change in one input variable will affect all other controlled variables. If a classical controller like PID is used, then we need to consider the best pairing of input and output variables. Traditional control design for MIMO systems becomes complex as the number of input and output grows. It requires domain expertise and covering the entire operating space by one design is also not feasible. For these situations, advanced control techniques like Model Predictive Control (MPC) can be used to handle the process interactions. MPC strategy is widely used in process industries because of its ability to handle both process interactions and inequality constraints. But there are some challenges associated with the use of MPC techniques. One such is that its performance is highly dependent on the accuracy of the model. Also for large scale systems, the process becomes computationally intensive as solving the dynamic model inside the controller takes longer time and gives delayed response. The third category of controllers are the Intelligent control systems that involve different artificial intelligence approaches like machine learning, neural networks, reinforcement learning etc. In this project, the idea of intelligent based controller tries to address the shortcomings of conventional controller.

2 Literature Survey

In the project, the focus is given on designing the Reinforcement Learning (RL) Controller of the Intelligent Control sytems. Reinforcement learning is the science of decision making which sits in the intersection of many fields of science. It can be differentiated from other machine learning algorithms like supervised and unsupervised learning. It is a trial and error approach, there is no supervisor which predicts the output. Reinforcement learning uses reward signals which provide the feedback for the actions taken by the algorithm. Whereas in unsupervised learning, hidden patterns inside the unlabeled data are identified. It is a technique which enables a computer to learn a policy for making a series of decisions to perform a task with the goal of maximizing the cumulative reward [2].

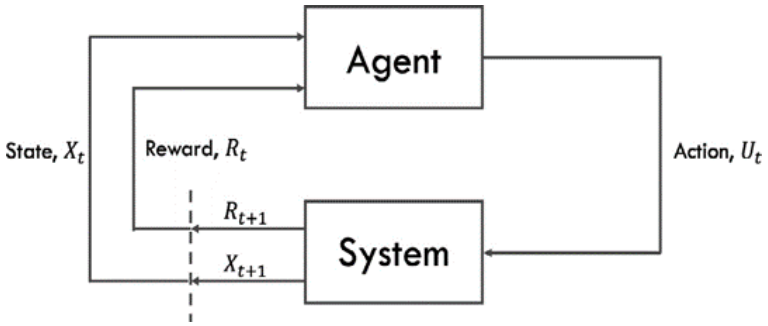

Figure 3: Reinforcement learning agent [3]

An agent takes the observations from the environment (system) and takes the decision (action U_t). A system consists of everything which is not in the controller i.e. agent here. The markov decision processes (MDPs) describe the environment for reinforcement learning. The markov property states that the future is independent of the past given the present which means the state captures all relevant information from the history [4]. There are different terms used in the RL like agent, policy, reward and value function. A policy represented by π which is basically the mapping from state to action. Based on the policy, the agent takes action which changes the state of the system (from X_t to X_{t+1}). The reward (R_t) is the feedback which the agent gets for each selected action (U_t). The goal of the agent or the reinforcement learning algorithm is to maximize the cumulative reward. There is a term called value function which denotes the long term value of being in a particular state. Reinforcement learning algorithm can be divided into three categories – Policy based, Value based and Actor critic algorithm.

Q-learning is a value based model free approach which uses action value function and tries to maximize the total sum of rewards over all time steps in future. A matrix Q [S, A] is maintained called Q-table in which S is the set of states and A is the set of actions. In Q-learning algorithm process , the Q-table is initialized which has dimensions of $m * n$ where m represents no. of states and n represents no. of actions. An action is chosen in state 's', performed and reward is measured. The Bellman equation is used to calculate the Q value given by the equation (1) [3].

$$Q(s_t, a_t) = \alpha(r + \gamma * maxQ(s_{t+1}, a_{t+1})) \tag{1}$$

$$Q^{new}(s_t, a_t) = (1 - \alpha)Q^{old}(s_t, a_t) + \alpha(r + \gamma * maxQ(s_{t+1}, a_{t+1})) \tag{2}$$

In equation (1), r represent the reward, α is the learning rate and γ is the discount factor. The process of finding optimal action for a given state based on maximum value function is continued until the learning is stopped. The Q-value in the matrix is updated iteratively using equation (2). One of the drawbacks of Q-learning is its inability to handle large state spaces. It becomes inefficient when number of states and actions increases as the size of Q-table also grows. The method is suitable for discrete action spaces and gives poor performance if the action space is continuous. The chemical systems deal with continuous action

spaces and using Q-learning for training is not a reliable option. There is another algorithm named deep deterministic policy gradient (DDPG) which is an online, off-policy RL method. It has an actor-critic RL agent that are designed for continuous action spaces and is well suited for problems like robot control. It uses Q value function critic $Q(S, A)$ and deterministic policy actor $\pi(s)$. The exploration in the DDPG is achieved by adding random noise to the actions. The direct mapping from observations to action values is achieved by the deterministic actor. The observations and actions are sent as input to the Q-value critic which returns the value (of taking those action in that state) as output. In short, the agent in the controller observes the state of the system and passed the information to the actor. The actions determined by the actor along with the observations of the system are sent to the critic which evaluates the value of the performing action in that particular state. In this project, DDPG agent is used for training the Mixer system.

3 Methodology

3.1 System Description

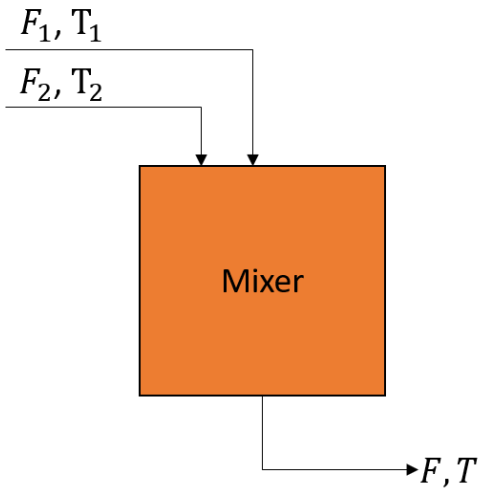

Figure 4: Mixer System

The case study which is considered for the design of control system is mixer model (Fig. 4). The system has two inlet streams - one hot ($T_2 = 1$) and other cold ($T_1 = 0$). The flow rates of stream 1 and stream 2 are represented by F_1 and F_2 respectively. The outlet stream is represented by flow rate F ans temperature T. The height of the tank is allowed to vary. F_1 and F_2 are considered as manipulated variables and F and T are the controlled variables. The objective is to control the flow rates of two inlet streams to achieve the outlet flow rate and temperature at the desired set point. The system is described by the following equation:

$$\frac{dF}{dt} = \frac{K(F_1 + F_2 - F)}{F} \tag{3}$$

$$\frac{dT}{dt} = \frac{2K(F_1T_1 + F_2T_2 - (F_1 + F_2)T)}{F * F} \tag{4}$$

where $F = \beta h^{1/2}$ and $K = \beta^2/2A\rho$. For simplicity, the variables in above equations are non-dimensionalized. The value of K is taken as 1. The model is implemented in Simulink environment (see Fig. 5).

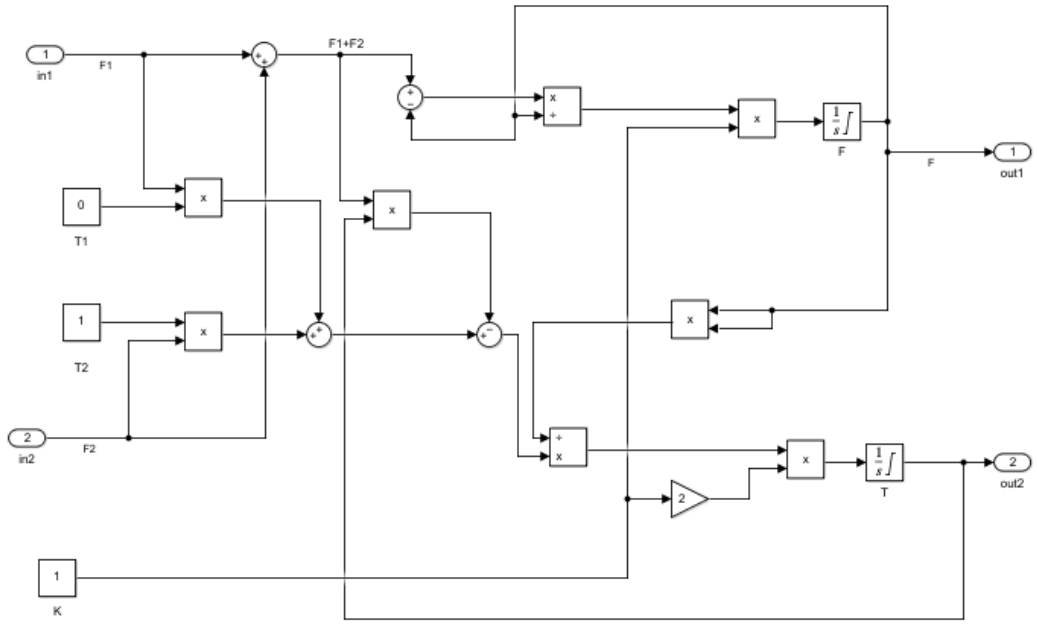

Figure 5: Mixer model in Simulink

3.2 Workflow

To achieve the objective, the reinforcement learning based controller is designed in the simulink. The RL agent considered was DDPG. The reward function is defined as

$$Reward = 10(|e_F| < 0.1) - 1(|e_F| > 0.1) + 10(|e_T| < 0.1) - 1(|e_T| > 0.1)$$

where e_F and e_T are the corresponding errors in the outlet flow rate (F) and outlet temperature (T) measurements. The reward is positive when the flow rate error and temperature error is below 0.1 and negative otherwise. The objective is to maximize the cumulative reward and minimize the negative reward.

The entire workflow involves the building of process model (Fig. 5), creating reward function, choosing the appropriate RL agent, training it to accumulate reward and writing the training script to make required changes to correct the wrong actions. The entire RL workflow is built in the Simulink (see Fig. 6).

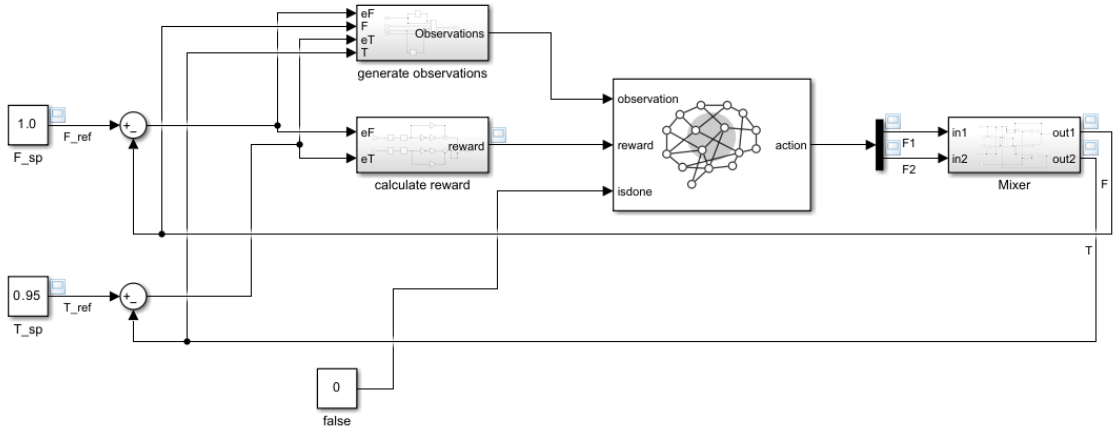

Figure 6: RL model in Simulink

4 Results and Discussion

The RL agent is trained for around 2500 episodes and average reward obtained after training is 43.6. The training plot can be seen in Fig. 7. The figure shows the plot between episode reward and episode number where the dark blue curve represents the cumulative reward and light blue curve represents each episode reward.

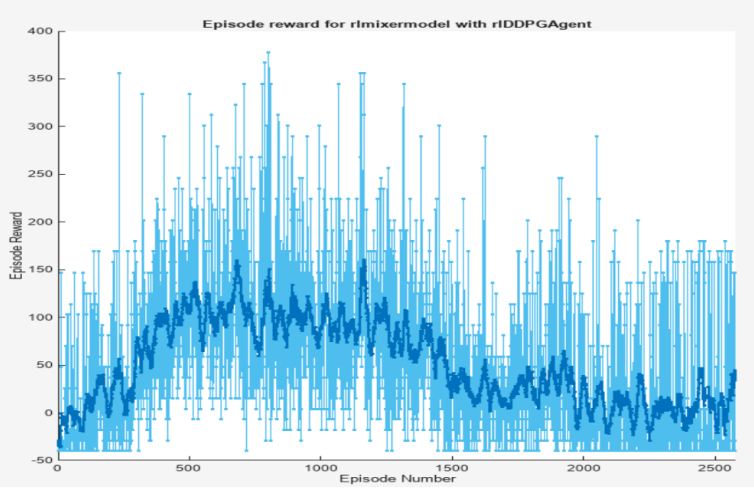

Figure 7: Training

The model was saved and then loaded to test the working of the trained agent. The trained RL model performance is checked on different set points. The two cases are considered. In first case, the outlet flow rate F is set at 1 and set point of outlet temperature T is taken as 0.05. The Fig. 8 and Fig. 9 shows the profile of input and output variables respectively. From Fig. 8, it can be seen that when the value of T is set low (0.05), inlet flow rate of cold stream, F_1 starts increasing and inlet flow rate of hot stream, F_2 starts dropping. The Fig. 9 shows that trained agent is able to track the set points by manipulating the inlet variables.

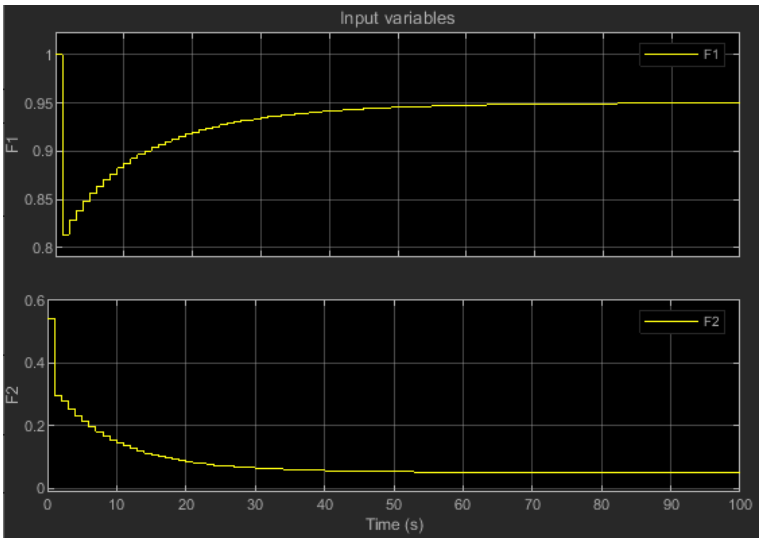

Figure 8: Input Profile (Case 1)

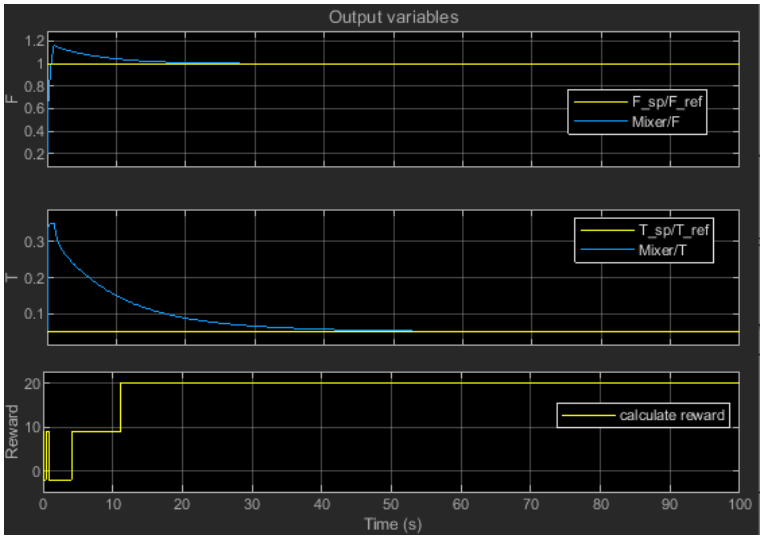

Figure 9: Output Profile and Reward (Case 1)

For the second case, the set point of F is kept constant at 1 and the set point of T is changed to a high value i.e. 0.95. The Fig. 10 and Fig. 11 show the plots of second case. Similar to the first case, as the value of outlet temperature T is set high (0.95), the inlet flow rate of the cold stream F_1 drops (see Fig. 10) and that of hot stream, F_2 rises. It is clearly seen that based on the given reward function and the fed observation signals, the RL agent is successfully able to learn the patterns of manipulating the input variables in order to obtain the desired set point.

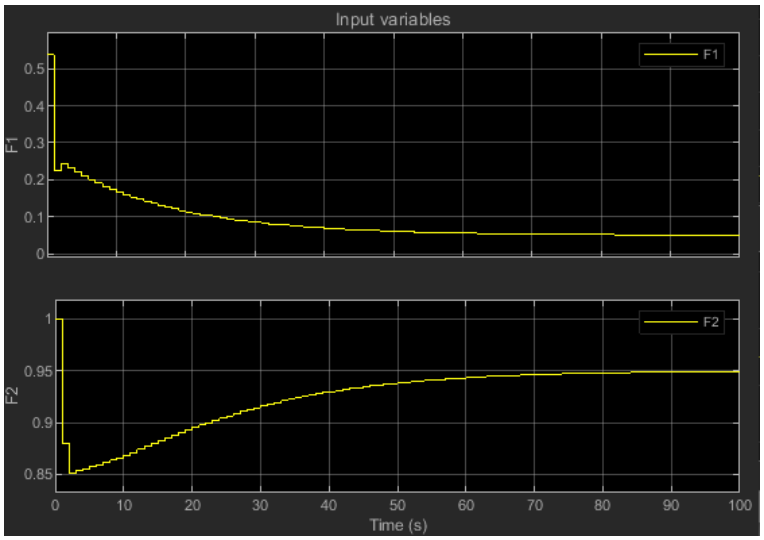

Figure 10: Input Profile (Case 2)

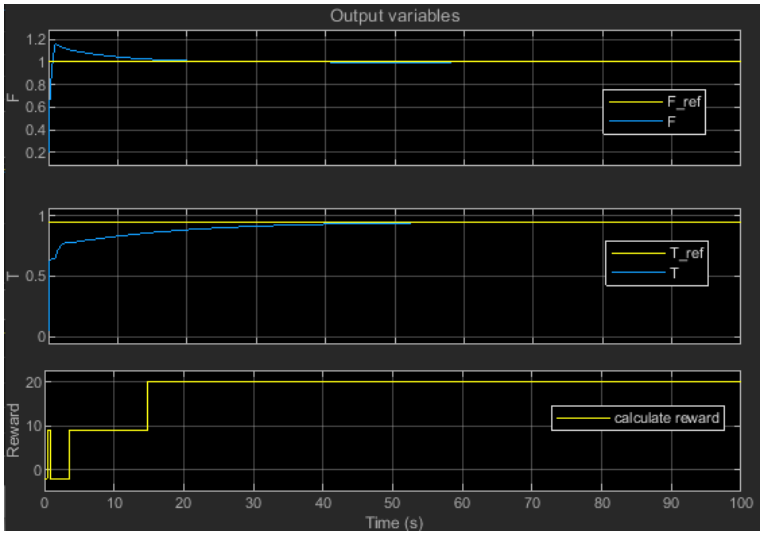

Figure 11: Output Profile and Reward (Case 2)

The trained model was also simulated by varying output temperature set points as 0.05, 0.95, 0.4, 0.1 and 0.45. The outlet flow rate is kept at constant set point. The below figure 12 shows the results. The model is able to track the different values of set point temperature.

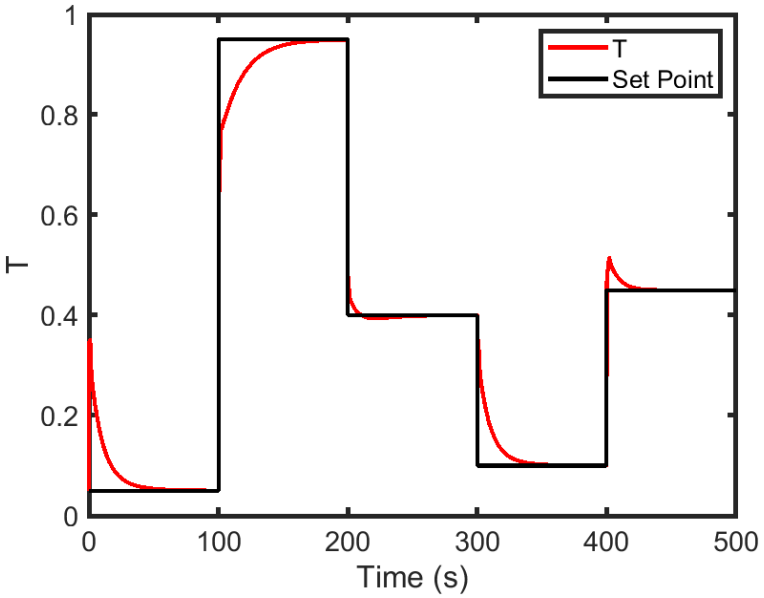

Figure 12: Varying T set point

5 Future Scope

The objective of designing the RL based controller for multiple input and multiple output system is achieved. In future, we can add complexities by training the model on systems with large no. of input and output variables to check the performance of controller. The weather conditions or aging can affect the performance of advanced controller like MPC as it requires the tuning of model parameters. To overcome it, the possibility of training the RL agent by including the effects of process drift can be explored.

6 Conclusion

In the project, the RL based controller is implemented on the mixer system. From the literature study, it was found the traditional controllers like PI or PID show poor performance on the multiple input multiple output systems. Conventional control design provides multiple control loop options and requires high domain expertise. Selecting one design from multiple loops can't cover the entire operating space. To address the shortcomings of conventional controllers, the RL based controller is designed. The results show that designed controller is able to able to achieve the different set points of hot and cold temperature regions. It can be concluded that Reinforcement learning can serve as an alternative to the conventional control design.

References

[1] D. Mellichamp D. Seborg and T. Edgar. *Process Dynamics and Control*.
[2] Mathworks. URL: `https://in.mathworks.com/discovery/reinforcement-learning.html`.
[3] Andrew Barto Richard S. Sutton. *Reinforcement learning: An introduction*.
[4] David Silver. URL: `https://www.davidsilver.uk/teaching/`.