A Comparison of Algorithms for Solving the CartPole Balancing Problem

CS7IS2 Project 2020/2021

Claire Farrell, Daniel Farrell, Joshua Cassidy, Matteo Bresciani

Trinity College Dublin, Dublin, Ireland
{farrec26, farred18, cassidj3, bresciam} @ tcd.ie

Abstract. The abstract should summarize the contents of the report and should contain at least 70 and at most 150 words. It should be set in 9-point font size and should be inset 1.0 cm from the right and left margins. There should be two blank (10-point) lines before and after the abstract. This document is in the required format. The abstract should give a concise overview of the main points of the report: the motivation behind the work, a very high level description of the problem and how it was solved by the proposed algorithms. The abstract must not include any figures or table.

Keywords: Artificial Intelligence, Reinforcement Learning, Cart-Pole, Genetic Algorithm, PID Controller, Gradient Ascent

This document is a guideline for writing the final report for the CS7IS2 module Artificial Intelligence. You should follow its general structure as shown below. You should not change its format (font, size, margin, space, etc.). Your report should be between 8 and 10 pages. Report that not comply to the format or exceed the maximum length will be penalised (-5 marks). Brevity is desirable in communication, however you should provide all those details necessary for the good understanding of the described methods and algorithms.

The report will be graded on the basis of:

– Originality - 10%;
– Technical soundness - 20%;
– Organisation - 20%;
– Clarity of presentation - 20%;
– Adequacy of bibliography/Results - 10%
– Presentation slides and recording) - 20%

Your report should provide a survey and an experimental comparison of multiple solution approaches to a particular problem. This is a critical review of at least three papers that significantly contributed to advance the state-of-the-art for the problem you are analysing. It should not be a mere summary of the papers. You are expected to conduct an analytical review of the methods under analysis to try to find common aspect and differences, connections between methods, drawbacks and open problems. Unless the faced problem has emerged recently, students should choose their papers by diversifying the range of approaches used to solve the problem. A good guideline could be to choose a paper from a decade or two ago, and a couple of more recent papers. You need to experimentally evaluate approaches in a simulation of a problem, in a range of scenarios, and analyse the pros and cons of each approach.

1 Introduction

The Cart-Pole balancing system is a classic control problem in the field of artificial intelligence. The system has been widely studied in the context of reinforcement learning, but little studies have set out to compare a range of different control techniques to solve the problem. Due to its recent successes, such as with AlphaGo and Alpha Zero (Silver et al., 2016; Silver et al., 2017), reinforcement learning has become extremely popular and is seen as a go to choice when dealing with dynamic environments such as Cart-Pole.

The motivation behind this research is to demonstrate that not only can the Cart-Pole system be solved using standard reinforcement learning techniques, but a whole range of algorithms from different families can perform just as well - if not better. This research will compare and contrast three vastly different algorithms for solving the Cart-Pole problem, namely, a Deep Q-Network with experience replay, a Genetic Algorithm, and a Proportional-Integral-Derivative (PID) controller with gradient ascent with multiple restarts.

Fig. 1. Cart-Pole in a balanced state vs an imbalanced state

The Cart-Pole problem, also known as the inverted pendulum, consists of a cart that moves along a friction-less horizontal track, along with a pole attached to the cart body that has a centre of gravity directly above its axis. This pole, starting in an upright position, can swing freely around its axis on the cart's body. The goal is to apply appropriate forces to the cart (left or right) in order to keep the pole balanced vertically and to prevent it from falling over. In figure 1.1 above, we can see the cart-pole in a balanced state and an imbalanced state.

In this section, you should introduce your work: what are the motivations behind this work? What is the relevant problem that you are investigating? Why is it relevant? Briefly, introduce the background information required to understand the problem and the concepts that you will develop. This section should also contain the link to the recording of your presentation (college OneDrive link – please make sure sharing permissions are such that everyone with tcd email can access it)

2 Related Work

In this section you will discuss possible approaches to solve the problem you are addressing, justifying your choice of the 3 you have selected to evaluate. Also, briefly introduce the approaches you are evaluating with a specific emphasis on differences and similarities to the proposed approach(es).

There is not too much research with Cart-Pole using PID controllers which select the PID parameters using gradient ascent, but we can see from [ref] that a common algorithm for choosing the parameters from a PID controller is gradient ascent, thus gradient ascent with multiple restarts was used. Gradient ascent using 10 random restarts was used as we see that gradient ascent with 10 random restarts is a very common choice thus 10 random restarts were used [ref]

3 Problem Definition and Algorithm

In order to evaluate the chosen algorithms, OpenAI Gym is used to simulate the Cart-Pole problem (OpenAI, 2021). As highlighted in the introduction, the goal of the system is to apply appropriate forces to the cart (left or right) such that the pole will remain balanced in an upright position above the cart without falling over.

At each episode, you can observe the following four states:

- The position of the cart
- The velocity of the cart
- The angle of the pole
- The angular velocity of the pole

At any state, the cart only has two possible actions:

- Apply a force of +1 (move the cart right)
- Apply a force of -1 (move the cart left)

The agent will receive a reward of +1 for every episode that the pole remains upright. And the episode will terminate if the pole is more than 20 degrees away from its vertical position, or if the cart moves more than 2.4 units away from its starting location. In figure 3.1 below we can see the initial state of the environment and the two possible actions that can be taken.

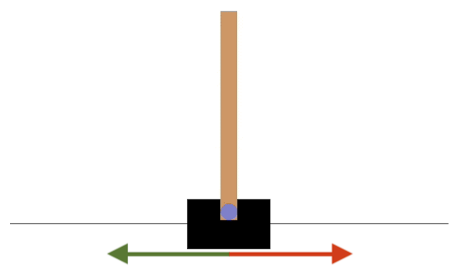

Fig. 2. Cart-Pole initial state and possible actions

3.1 Baseline Model

In order to adequately evaluate the algorithms, a baseline model was implemented using Random Search. The model navigated the search space by randomly choosing an action at each state. This model was evaluated and the results were used to understand the performance of the algorithms implemented.

3.2 Deep Q-Network with Experience Replay

The Deep Q-Network (DQN) algorithm was developed by DeepMind in 2015 and builds on the well-known Q-Learning algorithm by utilising the power of deep neural networks (Mnih et al. 2015).

3.3 Genetic Algorithm

3.4 PID Controller with Gradient Ascent

This section formalises the problem you are addressing and the models used to solve it. This section should provide a technical discussion of the chosen/implemented algorithms. A pseudocode description of the algorithm(s) can also be beneficial to a clear explanation. It is also possible to provide one example that clarifies the way an algorithm works. It is important to highlight in this section the possible parameters involved in the model and their impact, as well as all the implementation choices that can impact the algorithm.

4 Experimental Results

The three algorithms were each implemented in the same environment to ensure they could be evaluated effectively. As aforementioned, the Open-AI gym simulated the Cart-Pole environment and the following metrics were captured to evaluate the algorithms: Final Reward, Number of episodes, Average Reward, and Time Taken in Minutes.

Evaluation Metrics				
Algorithm	Final Reward	Number of episodes	Average Reward	Time Taken (Minutes)
Baseline	40.0	4	34.75	0.03
DQN with Experience Replays	500.0	167	196.06	25.428
Genetic Algorithm				
PID Controller with Gradient Ascent	500.0	20	500.0	1.50

4.1 Results Discussion

This section should provide the details of the evaluation. Specifically:

- Methodology: describe the evaluation criteria, the data used during the evaluation, and the methodology followed to perform the evaluation.
- Results: present the results of the experimental evaluation. Graphical data and tables are two common ways to present the results. Also, a comparison with a baseline should be provided.
- Discussion: discuss the implication of the results of the proposed algorithms/models. What are the weakness/strengths of the method(s) compared with the other methods/baseline?

5 Conclusions

Provide a final discussion of the main results and conclusions of the report. Comment on the lesson learnt and possible improvements.

Further research could be done performed by extending this problem to a 3-dimension space where the cart could move in any direction in a 360 degrees.

Another way how this work could be extended is the enviroment could have different surfaces that the cart would move on i.e. surfaces with different textures and frictions such as sand, concrete or rock. Another way how this work could be extended is introducing wind speed that the agent must take into account when balancing the pole.

A standard and well formatted bibliography of papers cited in the report. For example:

References

1. Silver, D., Huang, A., Maddison, C. et al. Mastering the game of Go with deep neural networks and tree search. Nature 529, 484–489 (2016). doi.org/10.1038/nature16961
2. Silver, D. et al. Mastering Chess and Shogi by Self-Play with a General Reinforcement Learning Algorithm. ArXiv (2017). abs/1712.01815
3. OpenAI Gym. (2021). A toolkit for developing and comparing reinforcement learning algorithms [Online]. Available at: https://gym.openai.com/[8 April 2021]
4. Mnih, V., Kavukcuoglu, K., Silver, D. et al. Human-level control through deep reinforcement learning. Nature 518, 529–533 (2015). doi.org/10.1038/nature14236