Grenze International Journal of Engineering and Technology, June Issue

Reinforcement Learning for Fuel Efficient Driving on Hilly Terrain - Maximizing Horizontal Distance Travelled

Parimi Anudheer[1], Ved Krishna Padakanti[2] and Mrs. Sheena Mohamed[3]
[1-3]Artificial Intelligence and Data Science, Chaitanya Bharathi Institute of Technology, Hyderabad, India.
Email: anudheerparimi@gmail.com, pvedkrishna@gmail.com, sheena_aids@cbit.ac.in

***Abstract*— Reinforcement learning is a crucial technical achievement in the present era, playing a key role in generating substantial progress in artificial intelligence and machine learning applications across several areas. The application of this dynamic methodology empowers machines and software agents to independently acquire knowledge and enhance their performance through their interactions with their surroundings, hence improving problem-solving and decision-making activities. This study delves into the application of reinforcement learning techniques to train an autonomous agent within a customized Mountain Car climbing environment. The primary objective revolves around optimizing fuel usage to propel the car uphill, strategically balancing the need for ascent against the subsequent horizontal distance covered during the descent. The constrained resource, finite fuel, necessitates the agent's adaptation through a training process to discern an optimal fuel consumption strategy. The research underscores the iterative nature of this training process, wherein the agent learns to navigate the trade-off between fuel utilization and distance traveled. In essence, this investigation contributes to the realm of reinforcement learning by elucidating the agent's capacity to make informed decisions.**

***Index Terms*— Rewards, Agent, Fuel Optimization, Constrained Resource, Finite Fuel, Trade-off, Decision-Making, Optimal Strategy.**

I. INTRODUCTION

In the real-world challenge domain, efficient resource utilization and strategic decision-making are paramount, especially in scenarios with limited resources [1]. This research addresses a practical problem that reflects the dynamics of resource-constrained scenarios in everyday life. Specifically, it involves training an autonomous agent to climb a hill in a hilly environment while contending with the limitation of finite fuel resources.
Finding a balance between resource consumption and intended outcomes is a persistent difficulty in numerous scenarios, from budget management to energy conservation. Conventional methods for addressing these difficulties typically rely on established principles or heuristics, which are intended to streamline intricate decision-making processes. Nevertheless, these approaches may prove inadequate in tackling the ever-changing and adaptable requirements of real-life scenarios. The objective of this study is to close this disparity by utilizing reinforcement learning, enabling an autonomous agent to acquire knowledge and refine its strategy autonomously

Grenze ID: 01.GIJET.10.2.162_1

as time progresses. The emphasis lies not only on reaching the summit but also on effectively managing the limited fuel reserves and optimizing the horizontal distance covered during the descent phase. This research is focused on a practical challenge and aims to go beyond academic bounds by providing insights that could result in more adaptable and pertinent solutions for optimizing resources in everyday situations.

II. BACKGROUND

The emergence of Reinforcement Learning (RL) [2] can be related to the convergence of several disciplines, such as psychology, neuroscience, and artificial intelligence. The notion of RL originated from the behavioral psychology experiments conducted by B.F. Skinner [3]. The paper focuses on the creation of open-ended algorithms capable of generating a continuous stream of novel and suitably challenging learning opportunities, aimed at automating and accelerating progress in machine learning. The Enhanced POET algorithm continually creates new and more complex environments, along with agents specifically trained to solve them. The origins of RL can also be attributed to the contributions of cybernetics pioneers like Norbert Wiener and Claude Shannon, who established the foundations for feedback control systems [4]. The development of actor-critic architecture, a foundational concept in reinforcement learning given in the paper "Looking Back on the Actor–Critic Architecture" by Andrew G. Barto, Richard S. Sutton, and Charles W. Anderson, published in IEEE Transactions on Systems, Man, and Cybernetics [5]. The researchers proposed the notion of temporal difference learning and devised algorithms such as Q-learning and SARSA, which laid the foundation for contemporary reinforcement learning.

RL has experienced significant progress and practical implementations throughout the years. Notably, in 2013, Volodymyr Mnih et al. introduced a groundbreaking technique called Deep Q-Networks (DQN). This approach coupled deep neural networks with Q-learning and achieved exceptional performance beyond human capabilities in different Atari games. The incorporation of policy gradient techniques such as Proximal Policy Optimization (PPO) and Actor-Critic algorithms has significantly enhanced the capabilities of Reinforcement Learning (RL) [7]. These advancements have facilitated more reliable and effective learning in intricate contexts. Currently, RL is being used in several fields such as robotics, autonomous systems, economics, healthcare, and gaming, demonstrating its adaptability and significant influence on solving real-world problems [8]. Furthermore, the use of reinforcement learning (RL) in fuel-efficient driving has gained significant relevance in recent years [9]. By integrating fuel consumption into the reward-punishment framework, RL agents can be trained to improve vehicle energy utilization, resulting in decreased environmental impact and financial savings. The use of reinforcement learning (RL) in autonomous cars shows great potential in boosting sustainable transportation by enabling agents to negotiate difficult terrains while minimizing fuel usage. The combination of RL with fuel-efficient driving technologies has the potential to change the automotive industry, promoting the development of environmentally friendly and intelligent transportation systems [10]. The ongoing advancement and fusion of Reinforcement Learning (RL) with other Artificial Intelligence (AI) methodologies hold great potential for the development of intelligent agents that can acquire knowledge and adjust to many complex situations, particularly those that prioritize energy conservation.

III. RELATED WORKS

The study conducted by Ardalan Vahidi and Antonio Sciarretta, as outlined in their work "Energy Saving Potentials of Connected and Automated Vehicles," [11] provides a comprehensive examination of the significant role that connected and automated vehicles (CAVs) can play in decreasing energy usage. This study, showcased in Transportation Research Part C: Emerging Technologies, explores the capacity of CAVs to augment energy efficiency, potentially resulting in energy conservation ranging from 3% to 20% using enhanced driving tactics. The study highlights the distinct advantages of CAVs in terms of enhanced safety, comfort, and efficiency, enabled by their sophisticated information processing, heightened computational capacity, and accurate control mechanisms. The study demonstrates how Connected and Autonomous Vehicles (CAVs) could contribute to energy conservation in the automotive industry by utilizing fundamental motion concepts, optimum control theory, and a comprehensive analysis of existing eco-driving research.

The study conducted by Kristen E. Brown and Rebecca Dodder in 2019 [12], investigates the potential transformative effects of vehicle automation on the transportation industry in the United States. The research investigates in depth the environmental and energy consumption repercussions of automation. This study examines the uncertainties surrounding the potential ramifications of automation on the demand for travel and the efficacy of vehicles. To methodically examine these variables, the researchers utilize the MARKAL (MARKet ALlocation) energy system model. They examine a range of scenarios that project varied trajectories for the integration of autonomous vehicles. Every scenario aims to illustrate the varied potential consequences of vehicle automation on the energy system, thereby offering valuable perspectives on the complex interplay between emerging automotive technologies, patterns of energy consumption, and environmental considerations.

The paper, "Deep Reinforcement Learning based Energy-efficient Decision-making for Autonomous Electric Vehicle in Dynamic Traffic Environments" by Wu, Song, and Lv, published in IEEE Transactions on Transportation Electrification [13], presents a novel method for improving the energy efficiency of autonomous electric vehicles (EVs) in dynamic traffic conditions. The study centers on autonomous driving techniques, which are seen as a viable approach for enhancing the energy efficiency of electric vehicles (EVs). These strategies encompass modifying driving choices and optimizing energy demands in different traffic scenarios. An innovative feature of the article is the introduction of a distinctive decision-making technique for autonomous electric vehicles. This technique seeks to optimize energy efficiency by considering both lane-changing and car-following behaviors, emphasizing the delicate equilibrium between driving behavior and energy economy in dynamic traffic situations.

The paper, "Safe and Energy-Saving Vehicle-Following Driving Decision-Making Framework of Autonomous Vehicles," authored by Y. Zhang, T. You, J. Chen, C. Du, Z. Ai, and X. Qu, and published in the December 2022 issue of IEEE Transactions on Industrial Electronics [14], aims to create a decision-making framework for autonomous vehicles specifically designed for vehicle-following situations. The main goal of the framework is to improve safety and increase energy efficiency. This technology tackles the formidable task of mitigating rear-end collisions, which are a prevalent hazard in scenarios where vehicles are following each other. Furthermore, the framework aims to enhance the overall energy efficiency of self-driving vehicles in these situations. The article highlights the crucial significance of the decision-making process in guaranteeing the safety and energy efficiency of autonomous vehicles, particularly when they are behind other vehicles on the road. This study enhances the field of autonomous vehicle technology by offering a systematic strategy to managing particular driving situations in a way that gives priority to both safety and energy efficiency.

IV. Proposed Methodology

Our proposed technique entails a systematic approach to project development, with a specific emphasis on the convergence of machine learning and environmental interaction. The initial stage is developing a custom gym environment, a simulated context that offers a regulated yet adaptable platform for our agents to engage with and acquire knowledge from. The customized setting will be created to replicate the precise circumstances and difficulties that are pertinent to our research investigation, guaranteeing that the educational encounter is as authentic and relevant as feasible.

Once the gym environment is set up, we will use the Q-Learning Algorithm as the fundamental paradigm for our agent's learning process. Q-learning, a type of reinforcement learning and a form of model-free reinforcement learning is selected for its efficacy in dealing with issues that involve significant randomness and for its capacity to learn the best behaviors by considering the total reward. This algorithm will enable our agent to make decisions and adapt its strategy depending on prior experiences and learning rewards inside the environment.

Finally, an essential element of our approach is the creation of a clearly defined reward function. This function will measure the effectiveness of the agent's actions by assigning a numerical value, which will serve as a guide for the learning process. The reward function will be meticulously designed to correspond with the aims of our project, guaranteeing that it precisely mirrors the desired outcomes and motivates the agent to cultivate efficient, goal-driven behaviors. By integrating a personalized gym setting, the Q-Learning Algorithm, and a customized reward system, our approach seeks to establish a strong foundation for the agent to acquire knowledge and adjust proficiently to the obstacles encountered in the simulated environment.

A. Creating a Custom Gym Environment

Creating a custom environment in OpenAI Gym involves defining the specific characteristics and dynamics of the simulated scenario. This is done by implementing a Python class that inherits from the Gym Env class. The class encapsulates the logic and behavior of the environment, defining methods for resetting the environment, executing actions, and calculating rewards. Once the custom environment class is defined, it needs to be registered with the Gym using the register function, making it accessible for reinforcement learning experiments. This involves creating a registration entry with a unique identifier. After registration, the custom environment can be instantiated and used like any other Gym environment in Python code. This process allows developers to tailor simulations to problem domains, providing a versatile platform for the development and evaluation of reinforcement learning algorithms in specific scenarios.

Steps for Creating the Environment

1. The environment can now be created by using the *gym.make()* function into which *'CarEnvironment'* is passed as the argument along with the *render_mode* of 'human' if the user wishes to render the results.
2. *metadata{}* is a necessary attribute that holds the supported *render_modes* and framerate.
3. The different object variable that needs to be defined is *action_space*, which holds the different actions that can be performed by the agent, and *observation_space* which defines the structure of the observations the environment will be returning.
4. Learning agents usually need to know these before they start running, to set up the policy function.
5. The structure of the environment is defined in the *render()* function which creates a *pygame* window and renders the environment and agent.
6. The position of the car, its angle, the position of the wheels, and the distance between them are all defined in this render function.
7. In the *step()* function the velocity of the car is calculated for different position ranges and positions respectively depending on the action taken.
8. This is the most crucial part of the environment as it decides whether an episode is complete or not by returning a Boolean variable and the current state of the car.
9. The reward function for the model is also defined and returned in this function itself, which is the most crucial part of the algorithm.
10. The *reset()* function is called after an episode is done and this function resets the fuel level of the car and the position of the car is set to the beginning of the path.
11. The car is given 3 actions, 0,1 and 2. 0 corresponds to reverse, 1 to neutral, and 2 to forward.
12. Then the boundaries of the environment are defined. It ranges from -2.4 to 3.6 and the values between these are used to determine the current position of the car.
13. The car is given a fuel level of 60.

B. Q-Learning Algorithm as a model for the agent

After setting up the Gym environment, the next step is to select an appropriate algorithm tailored to our specific environment. There are several algorithms to consider, including Q-Learning, Deep Q-Learning (DQN), SARSA, A2C, and Proximal Policy Optimization (PPO). For our project, we have chosen the Q-learning algorithm. This decision is based on its relative ease of implementation compared to neural network-based options, and its suitability for projects requiring straightforward yet effective solutions. Q-Learning offers a balance of simplicity and efficiency, making it an ideal choice for our environment.

Q-Learning works by updating the agent's knowledge of the best actions to take in different states, based on the feedback received from the environment. It consists of an exploration-exploitation policy where the model either chooses to explore, i.e, take random actions or exploit the environment from previous actions. Through repeated interactions, the agent learns which actions lead to higher rewards and builds a table of "Q-values" that represent the expected rewards for each action-state pair. The Q-values can be calculated using the equation (1) shown. By selecting actions with the highest Q-values, the agent gradually improves its decision-making abilities and learns to navigate the environment more effectively to achieve its goals. The Q-Learning algorithms workflow for updating Q-values is shown the Algorithm 1.

$$Q(s, a) = Q(s, a) + \alpha * (r + \gamma * \max(Q(s', \text{all } a') - Q(s, a))) \quad (1)$$

Algorithm *Q-Learning(s,a)*
{
//Assign arbitrary values to the Q-values table during initialization.
initialize Q(s, a) *for all state-action pairs (s, a)*
for each episode
{
  // Perform a system reset and carefully observe the initial state
   s = reset_environment()
   while *(episode not ended)*
     {

//Select an action according to the exploration-exploitation policy, such as the ε-greedy approach.

a = choose_action(s, policy="ε-greedy")

// Execute the selected action, see the subsequent condition, and receive a reward.

s′, r = take_action(a)

// Apply the Q-learning update rule to modify the Q-value of the current state-action pair.

Q(s, a) = Q(s, a) + α * (r + γ * max(Q(s′, all a′)) - Q(s, a))

// The variables 'r', 'α', and 'γ' represent the reward received, the learning rate, and the discount factor, respectively.

// Proceed to the subsequent state

s = s′

}

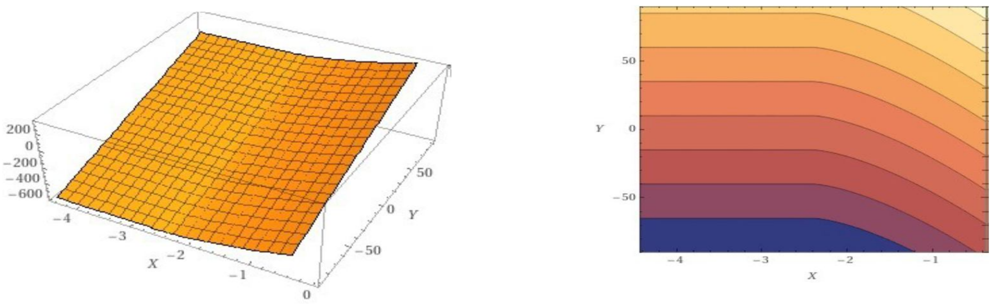
Algorithm 1. Workflow for Q-values

C. Reward Function

The heart of the algorithm lies in the reward function. A small change in the function can cause the model to not learn anything as opposed to a good one which can easily be used to train the model. The reward function shown in equation (2) we used is

$$\text{reward} = 75*((\text{position}+2.4)^{2.5}) - 4*(60-\text{fuel\_level})*a \quad (2)$$

The position is the position of the car at the end of the episode, by giving it a huge coefficient, it is made sure that the position is given just as much preference as the fuel level remaining in the car at the end of the episode. ‘a’ is a variable which changes based on the actions taken and hence the reward is higher if a good action is taken as opposed to a bad one. The algorithm’s exploration-exploitation policy, coupled with the carefully designed reward function, allows the agent to gradually learn to navigate the environment effectively and achieve its goals. Figure 1(a) represents the reward function 3D-graph where the x-axis represents position, the y-axis represents the fuel level values and the z-axis represents reward values. Figure 1(b) shows the 2D graph plotted for Position and fuel values correspondingly. The negative position values give an uphill direction and positive position values give a downhill direction.



(a) 3D– reward function graph (b) 2D - reward function graph

Figure 1. Reward Function Graphs

V. RESULTS

The results of our implemented Q-learning algorithm in the custom Gym environment for the car agent have been successful and promising. The agent learned to navigate the complex environment efficiently, optimizing its fuel usage to achieve its goal. The utilization of the hill’s inertia on the downhill portion of the track showcased the agent’s adaptability to various scenarios. The agent exhibited intelligent decision-making by strategically balancing its actions to reach the goal while conserving fuel.

The visualizations of the reward function further supported the agent's behavior and highlighted its ability to exploit the environment's features. The agent's performance, characterized by its reduction in fuel consumption while maintaining goal achievement demonstrates the effectiveness of the Q-learning algorithm for this task. Table 1 shows the reduction in fuel consumption on a Hilly Terrain, where negative position values give the uphill and positive position values give the downhill. The figures 2, 3, 4 5, and 6 show the fuel levels at various positions and can observe the fuel consumption is high in the uphill direction and the fuel consumption is reduced in the downhill direction.

Our work not only successfully implemented the Q-learning algorithm but also demonstrated its ability to learn and adapt in a dynamic environment. The outcomes of this endeavor stand as a testament to the power of reinforcement learning algorithms in solving real-world problems.

Sl.No	**Figure Number**	**Fuel Level**	**Position values**	**Fuel consumption (difference in fuel level from the previous position)**
1	Figure 2	60	-2.27	Initial
2	Figure 3	53	-0.95	7
3	Figure 4	38	-0.09	15
4	Figure 5	33	0.91	5
5	Figure 6	33	3.30	0

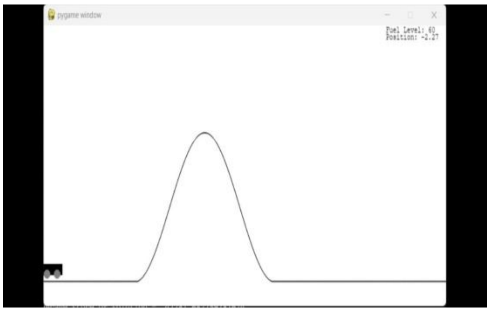


Figure 2 Car moving towards uphill with fuel level 60, position -2.27

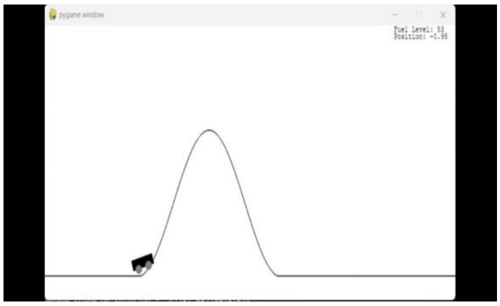


Figure 3 Car moving uphill with fuel level 53, position -0.95

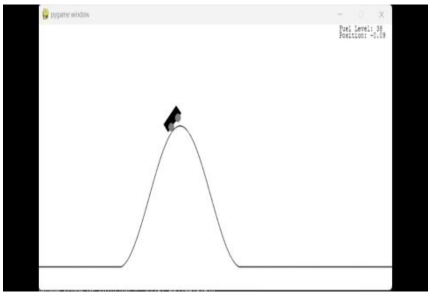


Figure 4 Car moving uphill to the top with fuel level 38, position -0.09

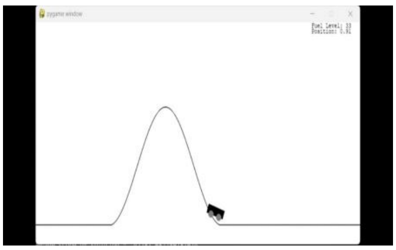


Figure 5 Car moving downhill with fuel level 33, position 0.91

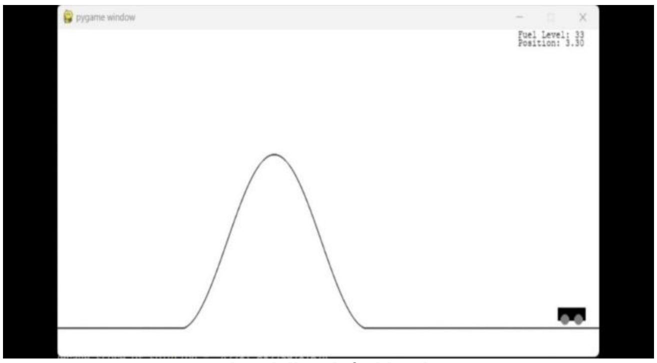


Figure 6 Car moving downhill with fuel level 33, position 3.30

VI. Conclusion And Future Scope

The reinforcement learning for fuel-efficient driving in hilly terrain presents a compelling and valuable endeavor. By employing RL algorithms and strategies, the research aims to minimize fuel consumption during uphill and downhill drives, addressing a crucial aspect of environmental sustainability and cost-effectiveness in transportation. The optimization of fuel usage in challenging terrains is particularly important, as hills often pose significant challenges for vehicles, necessitating effective decision-making and control. Throughout the project, the RL agent exhibited promising performance, consuming a fuel amount close to the calculated optimum required to reach the final goal position. This outcome indicates that the agent successfully learned and adapted its behavior to achieve the project's objectives. The results exemplify the potential of RL to tackle real-world problems, such as fuel-efficient driving, offering scalable and adaptable solutions that can contribute to mitigating environmental impacts and saving valuable resources. Moreover, the project's success serves as a testament to the broader applicability of RL in diverse scenarios, where intelligent agents can learn and optimize behaviors to improve overall efficiency, safety, and sustainability in various industries and domains. As RL continues to advance and find more applications, it holds the promise of revolutionizing the way we address complex challenges and make intelligent decisions in an ever-changing world.

References

[1] Bessedik, Madani, et al. "Strategic Decision-Making in Sustainable Water Management Using Demand Analysis and the Water Evaluation and Planning Model." Sustainability 15.22 (2023): 16083.

[2] Wang, Rui, et al. "Enhanced poet: Open-ended reinforcement learning through unbounded invention of learning challenges and their solutions." International Conference on Machine Learning. PMLR, 2020.

[3] O'Donohue, William, and Kyle E. Ferguson. The psychology of BF Skinner. Sage, 2001.

[4] Wiener, Norbert. "Cybernetics in history." Theorizing in communication: Readings across traditions (1954): 267-273.

[5] Barto, Andrew G., Richard S. Sutton, and Charles W. Anderson. "Looking back on the actor–critic architecture." IEEE Transactions on Systems, Man, and Cybernetics: Systems 51.1 (2020): 40-50.

[6] Mnih, Volodymyr, et al. "Human-level control through deep reinforcement learning." nature 518.7540 (2015): 529-533.

[7] Vahidi, Ardalan, and Antonio Sciarretta. "Energy saving potentials of connected and automated vehicles." Transportation Research Part C: Emerging Technologies 95 (2018): 822-843.

[8] Bartley, Travis D. Hierarchical Reinforcement Learning with Model-Based Planning for Finding Sparse Rewards. University of California, Irvine, 2023.

[9] Vogel, Adam, et al. "Improving hybrid vehicle fuel efficiency using inverse reinforcement learning." Proceedings of the AAAI Conference on Artificial Intelligence. Vol. 26. No. 1. 2012.

[10] Alagha, Ahmed, et al. "Target localization using multi-agent deep reinforcement learning with proximal policy optimization." Future Generation Computer Systems 136 (2022): 342-357.

[11] Raja, Gunasekaran, and Gayathri Saravanan. "Eco-Friendly Disaster Evacuation Framework for 6G Connected and Autonomous Vehicular Networks." IEEE Transactions on Green Communications and Networking 6.3 (2022): 1368-1376.

[12] Brown, Kristen E., and Rebecca Dodder. "Energy and emissions implications of automated vehicles in the US energy system." Transportation Research Part D: Transport and Environment 77 (2019): 132-147.

[13] J. Wu, Z. Song and C. Lv, "Deep Reinforcement Learning based Energy-efficient Decision-making for Autonomous Electric Vehicle in Dynamic Traffic Environments," in IEEE Transactions on Transportation Electrification, doi: 10.1109/TTE.2023.3290069.

[14] Y. Zhang, T. You, J. Chen, C. Du, Z. Ai and X. Qu, "Safe and Energy-Saving Vehicle-Following Driving Decision-Making Framework of Autonomous Vehicles," in IEEE Transactions on Industrial Electronics, vol. 69, no. 12, pp. 13859-13871, Dec. 2022, doi: 10.1109/TIE.2021.3125562.