ERURJ **2025**, *4*, 1, 2126-2137
Article

Projectile trajectory optimization based on reinforcement Q-learning algorithm

Amany A. Khalifa [1], Amira R. Abdel-Malek [1]

[1]*Department of Mathematical and Natural Sciences, Faculty of Engineering, Egyptian Russian University, Badr, Egypt.*

*Corresponding author: Amira R. Abdel-Malek**,** E-mail: amira-ragab@eru.edu.eg, Amany A. Khalifa E-mail: amany-ahmed@eru.edu.eg.

Received 3rd March 2024 Revised 2nd June 2024 Accepted 31st January 2025
DOI:10.21608/erurj.2025.273676.1128

ABSTRACT

In this paper, projectile trajectory optimization is investigated. The main objective is to determine the optimal launch angle that maximizes the projectile achieved total distance (arc length). We investigated the process of determining the most effective launch angle, the angle at which an object is projected to achieve the greatest horizontal distance covered. One of the key determinants of this angle is the initial velocity of the projectile, the impact of air resistance, and the nature of the landing surface. The application of the reinforcement Q-learning to optimize the projectile total traveled distance is explored whereas, traditional methods of optimizing projectile trajectories often rely on mathematical models and iterative approaches. However, in this study, we leverage the flexibility and adaptability of Q-learning to optimize projectile trajectory by learning optimal actions through interaction with the environment. The result is proven graphically, furthermore, the performance of the achieved range and the maximum height at the optimum angle is investigated.

Keywords: reinforcement learning, Q-learning, launch angle, artificial intelligence, parameter optimization.

1- Introduction

Scientists have long been fascinated by projectile motion, a cornerstone of physics and engineering. Optimizing projectile trajectories is crucial for diverse applications in sports, space exploration, and the military. The launch angle is a critical factor in projectile motion, influenced by initial velocity, air resistance, and the nature of the landing surface. Understanding these physical mechanisms is essential for achieving optimal trajectories. This study utilizes reinforcement Q-learning to find the optimum launch angle that maximizes the projectile's total distance traveled (arc length) to reduce undesirable effects like energy loss from air resistance and trajectory deviations. Unlike traditional methods that rely on static mathematical models and iterative calculations, Q-learning provides a dynamic approach by learning optimal actions through environmental interactions. This adaptability enables continuous optimization of the launch angle, resulting in more effective trajectory control. Besides, pure rolling, the motion of an object along a surface without slipping, is crucial in mechanics. It occurs when the object's translational motion matches its rotational motion about its center of mass. This concept is significant in analyzing its relation with projectile motion and optimizing the trajectory and efficiency of launched objects. Determining the optimum launch angle for projectiles relies on this understanding, impacting the range, accuracy, and efficiency of various applications in engineering and physics. Optimizing the launch angle and initial velocity is crucial for improving projectile performance [1-2]. Karadag [3] investigated the intricate relationship between the projectile launch angle and the features of the final trajectory to identify the launch angle that would maximize the projectile's overall distance traveled during flight (the projectile arc length). He derived an analytical solution and illustrated the results graphically. In addition, he determined the launch angle to be about 56.5°.

Regodić et al. [4] used integration methods to calculate the projectile's movement over time while accounting for the wind, Coriolis inertial force caused by the reactive force, rotation of the Earth, and acceleration due to gravity. These studies have improved our comprehension of the many subtleties associated with projectile motion optimization. In addition, modern developments in resources of optimization and optimization methods have greatly broadened the field of projectile motion study. In (Kahrizi et al., 2020) [5], the authors proposed a global optimization of the projectile trajectory using a novel metaheuristic technique. Similarly, machine learning methods were utilized to optimize the projectile trajectory using the Long Short-Term Memory [6]. In the context of using optimization algorithms for optimizing the projectile firing angle and velocity, Alridha utilized the five different techniques, Nelder-Mead, Powell, Limited-memory Broyden-Fletcher-Goldfarb-Shanno with Box constraints LBFGS-B, Truncated Newton Conjugate-Gradient TNC, and Sequential Least-Squares Quadratic Programming SLSQP to find the optimum launch parameters which maximize the achieved range [2] The author investigated the resulting trajectory for each algorithm and compared the results. In (Amer et al., 2024) [7], the authors transformed the stability analysis of an asymmetric rigid body under constant torque and gyrostatic moment from a two-dimensional phase plane to a three-dimensional phase space. It examines stationary torques on different axes, revealing new analytical and simulation results on equilibrium manifolds and periodic solutions. The findings have significant applications, particularly in gyro theory. The dynamic motion of a two degrees-of-freedom auto parametric system with a rolling cylinder and damped spring under excitation, deriving solutions using Lagrange’s equations and multiple scales method was examined in (Amer, 2022) [8]. Stability and resonance are analyzed, revealing novel insights into the system's behavior. explores the rotational motion of a symmetric gyrostat under magnetic and Newtonian fields were studied in (Amer, 2021) [9], simplifying the system to find asymptotic solutions and analyze stability using Poincaré's method. The results, relevant to submarines, aircraft, and satellites, are illustrated with graphical plots and phase planes. In (Escobar et al., 2022) [10] the authors investigated the distance between the projectile and the object. The results show the for launch angles greater than approximately 70.53°, the distance between the projectile and the object temporarily decreases between two points of maximum and minimum distance. Assuming a ground launch without air friction, the authors used dimensionless coordinates.

In (Abu-Bakr et al., 2019) [11], the authors theoretically examined heat production in a system of magnetically interacting ferromagnetic particles subjected to both rotating and linearly polarized magnetic fields. Using a mathematically regular approximation for pair interactions, results show that interparticle interaction significantly enhances heat production, with a greater thermal effect in rotating fields. magnetic hyperthermia in single-domain ferromagnetic particles within an oscillating magnetic field modeled in (Abu-Bakr et al., 2020) [12], showing that particle clusterization weakens the thermal effect. These findings are crucial for optimizing practical applications of magnetic hyperthermia.

Even with the projectile's motion field advancements, there are still gaps and untapped prospects in the topic of projectile motion optimization. Projectile launch angle optimization using a reinforcement machine learning algorithm called Q-learning is the subject of this article.

2- Problem Statement

To solve the challenging task of optimizing the projectile trajectory, a methodological approach is implemented where the projectile starts its motion by velocity v_o and a positive launch angle δ An optimization problem is formulated to find the optimal launch angle δ aiming to maximize the achieved total distance (arc length) at a fixed positive initial velocity v_o. D (δ) is the total distance traveled by the projectile which is a function of the launch angle δ and can be represented as follows

$$\mathrm{D}\,(\delta) = \frac{{v_o}^2}{g}\left[\sin(\delta) + \cos^2(\delta)\ln\left(\frac{1+\sin(\delta)}{\cos(\delta)}\right)\right]. \tag{1}$$

where g is the gravitational acceleration (g equals 9.81 meters per second squared).

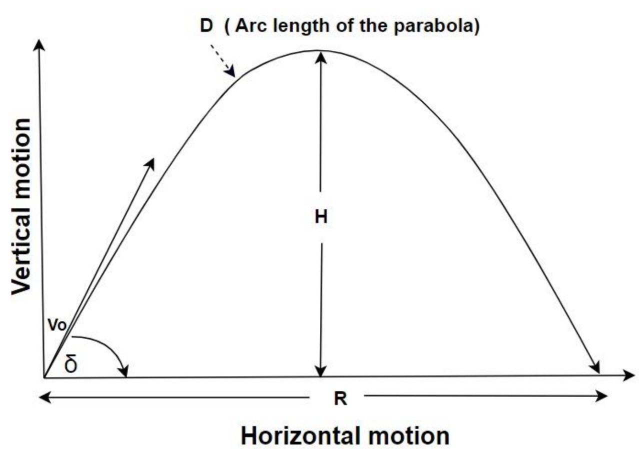

Figure 1: The trajectory of a projectile motion.

In Figure 1, H is the maximum height achieved during the projectile motion and R is the maximum horizontal distance achieved by the projectile.

3- Problem Formulation

Considering the motion of the projectile, where the objective is to fire the projectile from a starting point to achieve the maximum arc length of the projectile's trajectory $D(\delta)$, the goal is to find the optimal launch angle δ that achieves the maximum arc length. This issue may be presented as an optimization problem which can be expressed as:

$$\underset{\delta}{maximize} \; D(\delta). \qquad (2)$$

Subject to

C: $0 \leq \delta \leq 90$

By constraint C, the launch angle is guaranteed to be within the feasible launch angle range of 0° to 90°. The main goal of this research paper is to methodically investigate the Q-learning algorithm to effectively and accurately solve this optimization problem. Finding the ideal launch angle (δ) that produces the largest achievable traveled distance for the proposed projectile motion scenario.

4- The Proposed Reinforcement Q-Learning Algorithm

To the best of our knowledge, this article marks the pioneering effort in enhancing projectile trajectory optimization through the application of the Q-learning algorithm. To tackle the challenging problem of parameter optimization in the projectile motion domain, a Q-learning algorithm is proposed which was proposed by Dayan as one of the reinforcement learning algorithms in [13-14]. The agent is enabled to learn optimal actions in the environment by associating each pair of action instances with a value Q, which represents the expected cumulative reward. It explores the environment, updating Q-values based on received rewards and transitioning to states with higher Q-values. Through repeated iterations, Q-learning converges to an optimal policy, guiding the agent to make informed decisions and maximize cumulative rewards [15].

The agent in reinforcement learning is modeled as a finite Markov Decision Process (MDP), if the states and actions spaces are finite. The MDP can be expressed as the tuple ($\mathscr{R}$, $\mathcal{A}$, $\mathcal{S}$), where $\mathscr{R}$ denotes the reward function for the agent, $\mathcal{A}$ defines the agent's action space and $\mathcal{S}$ denotes the discrete environment states. At each time step t, the agent is at the state $S_t \in \mathcal{S}$, and performs an action $A_t \in \mathcal{A}$, which will change the system state to $S_{t+1} \in \mathcal{S}$, and receiving a reward $r_t \in \mathscr{R}$. In this manner, the set of discrete environment states $\mathcal{S}$ represents all the possible projectile launch angels.

While the set of actions $\mathcal{A}$ represents the two possible actions that are increasing or decreasing the angel δ by 1 degree, which is $\mathcal{A}$= { $\delta + 1$ or δ -1}, an action is chosen based on an epsilon-greedy strategy, where a random action is chosen from the action space with probability ξ, was set to 0.1. Otherwise, the action returns the current state highest Q-value is selected, exploiting the learned information. On the other hand, the reward r_t is calculated to represent the total achieved traveled distance by the projectile as follows:

$$r = D\ (\delta). \tag{3}$$

The agent's goal in reinforcement learning is to learn an optimal policy $Q^{\pi}(\mathrm{A}, S)$ that denotes the action A selection probability at state S to get maximum rewards over time (i.e., it is a mapping from states to actions).

The algorithm depends on the value of the action-state function $Q^{\pi}(\mathrm{A}, S)$ that encapsulates the agent's anticipated return starting from state S∈ $\mathcal{S}$ and performing action A∈ $\mathcal{A}$ and following the policy $Q^{\pi}(\mathrm{A}, S)$ (i.e. Q-function).

$$Q^{\pi}_{t+1}(A_t, S_t) = (1 - \alpha_L) Q^{\pi}_t(A_t, S_t) + \alpha_L \left[r_t + \gamma \underset{A_{t+1}}{max} Q^{\pi}_t(A_{t+1}, S_{t+1}) \right]. \tag{4}$$

Where $S_t, S_{t+1} \in \mathcal{S}, A_t, A_{t+1} \in \mathcal{A}$, and α_L is the learning rate which was set to 0.1, γ is the discount factor set to 0.9. Algorithm 1 illustrates in detail the steps of the Q-learning algorithm used in the projectile arc length optimization. After initialization (line 1), and at each episode of the predefined number of episodes $N_{episodes}$ which was set to 10000, the initial state (i.e., initial launch angle) is determined randomly. Then, an action is selected and executed based on the ξ-greedy policy at each state [16], which represents the random action selection (i.e., the exploration) and action with the maximum Q-value selection (i.e., the exploitation) trade-off (lines 4,5). After that, r_t is determined according to (3) based on the state s_{t+1}, and the table of action-state pairs is updated (lines 6-8). The algorithm terminates when either the Q-table converges or when it reaches the predetermined number of iterations.

Q-learning algorithm for projectile parameters optimization
1. Initialization: update the environment states $\mathcal{S}$ considering the boundaries. Initialize the action-state table $Q_t^{\pi}(\mathrm{A}_t, S_t)$ with random values within the state space. Initialize the algorithm hyperparameters such as the discount factor γ, the learning rate α_L, the exploration rate, the number of episodes, and the number of steps for each episode N_{steps}. 2. For $episode$ = 1 to $N_{episodes}$ do. 3. Get an initial state S_1. 4. For steps: = 1 to N_{steps} do 5. $A_t = \begin{cases} random\ ,\ with\ \varepsilon\ probability \\ max,\ Q^{\pi}(\mathrm{A}_t, S_t)\ ,\quad otherwise \end{cases}$ 6. Execute action A_t, and obtain S_{t+1}. 7. Determine r_t using (3). 8. An action A_{t+1} is selected based on the state S_{t+1}, and update the action-state table $Q_t^{\pi}(A, S)$ based on (4). 9. Replace $S_t \leftarrow S_{t+1}$ 10. End for 11. End for 12. Result: Optimal state (lunch angle) with maximum traveled total distance (r_{max}).

5- Results and Discussion

The application of the Q-learning algorithm in solving the optimization problem has produced insightful results that clarify the effectiveness of the algorithm for optimizing projectile motion parameters. The findings from the simulation in this area are investigated.

In this study, we employed MATLAB to conduct simulations and implement the Q-learning algorithm for optimizing the launch angle of a projectile to maximize its total arc length. The simulation setup involved discretizing the state space into a number of bins representing different values of the launch angle ($\boldsymbol{\delta}$). Each action in the Q-learning algorithm corresponds to adjusting the launch angle by either increasing or decreasing it by one degree. The simulation environment provided by MATLAB facilitated the exploration and exploitation of the action-state space, allowing the algorithm to learn the optimal launch angle through repeated trials. The simulation showed an optimum launch angle of 56.3964° which produces the maximum total achieved traveled distance $D(\delta)$. However, the exact optimum value of the launch angle can be obtained graphically from the graph of the relation between the arc length D and the angle δ as shown in Figure 1 where the optimum value of the angle δ at which the total traveled distance (arc length) has maximum value is $56.4865°$. This figure explains the effectiveness of the proposed algorithm that obtained a near value with a relative true error of 0.16%.

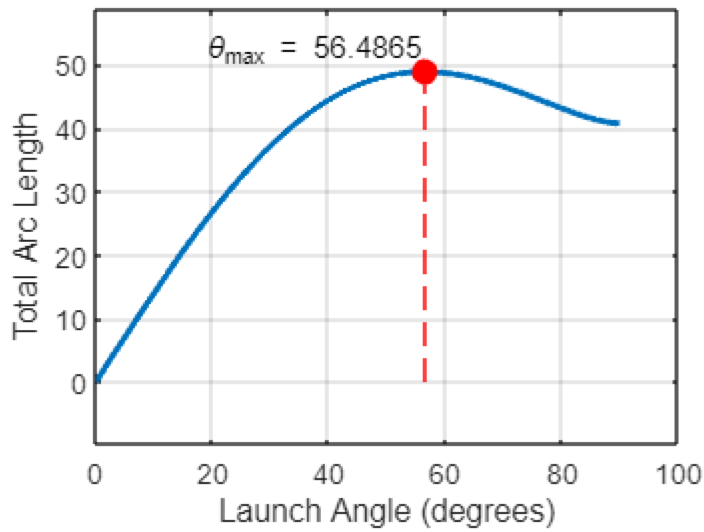

Figure 1: The relation between the total distance traveled by the projectile D and the launch angle δ, $v_o = 20\ m/s$.

Where the mathematical formula of the achieved range and the maximum height can be expressed as follows [2-3].

$$Range\ (\delta) = \frac{{v_o}^2 \sin(2\delta)}{g}. \tag{5}$$

$$Height = \frac{{v_o}^2 \sin^2(\delta)}{2\ g}\ . \tag{6}$$

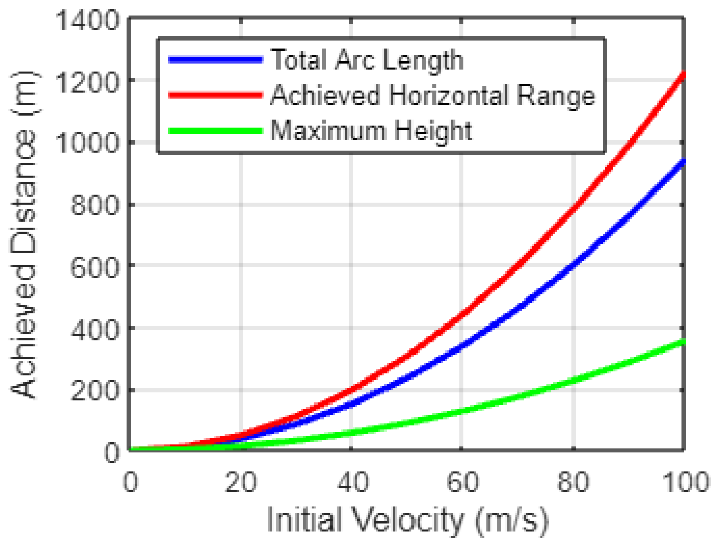

Figure 2: Comparison between the achieved total distance, range, and maximum height at $\delta = 65.3964^{\circ}$.

Figure 2 shows that when launching the projectile at 56.3964° only the total distance traveled by the projectile is maximized, whereas the achieved maximum height and range are not in their maximum values. Also, it can be noticed that the achieved distance generally increased by increasing the initial velocity.

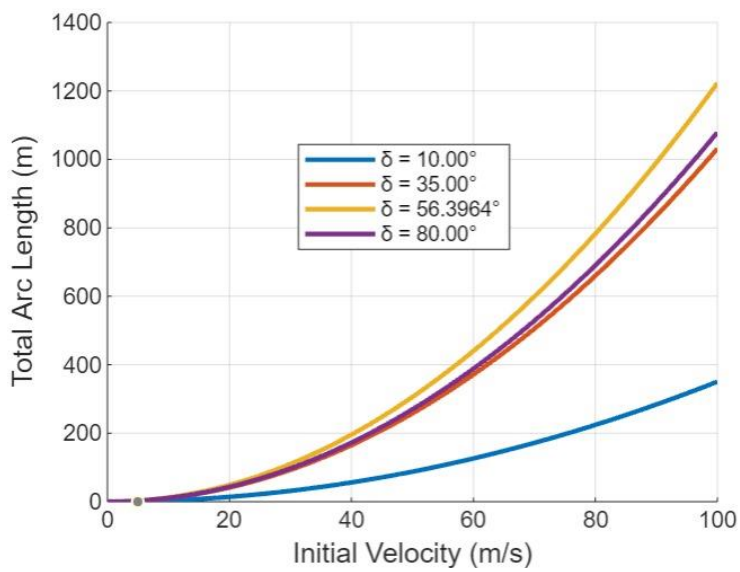

Figure 4: The total distance traveled (arc length) at different values of angle δ.

Figure 4 illustrates the relation between the total distance traveled (arc length) and the initial velocity at different values of angle δ. It can be noticed that in general, the projectile's arc length increases as the initial velocity increases. Also, it can be noted clearly that the maximum total distance traveled (arc length) occurs at $\delta = 56.3964^{\circ}$ which is the optimal value obtained by the Q-learning algorithm.

Table 1: Comparison between the proposed work and the literature.

	The optimal launch angle which maximizes the projectile arc length
Mathematical solution [3]	56.5°
The proposed graphical solution	56.4865°
The proposed Q-learning algorithm	56.3964°

Table 1 provides a detailed comparison of the results obtained from the proposed Q-learning algorithm, the graphical solution depicted in Figure 2, and the study conducted by [3] The table highlights the close agreement between the Q-learning algorithm's output and the other two methods. Specifically, the angles derived from the Q-learning algorithm are nearly identical to those obtained through the graphical solution and the methodology presented by Karadag. This comparison underscores the accuracy and reliability of the Q-learning algorithm in producing results that align well with established methods.

6- Conclusion

A framework is proposed in this paper to optimize the trajectory of the projectile using the reinforcement Q-learning algorithm by finding the optimum firing angle that maximizes the total distance traveled by the projectile (projectile's arc length). The results showed an accurate angle of 56.9364 degrees with a 0.16% relative error which proves the effectiveness of the proposed Q-learning algorithm. The range and the maximum height of the projectile are investigated at this optimum angle to conclude that this optimum angle maximizes only the arc length of the projectile. In contrast, the range and the maximum height have their maximum values at some other angles. The total traveled distance (arc length) of the projectile is investigated at different values of the launch angle, the result indicated that the largest total traveled distance occurs at the launch angle obtained from the Q-learning algorithm. To evaluate the effectiveness of the proposed Q-learning algorithm, we compared it with related work in the literature and the graphical solution. This comparison shows that the results of our algorithm are in good agreement with those found in the literature. Consequently, Q-learning is highly adaptable to various environments and conditions without requiring a predefined model. It efficiently explores the solution space to find the optimal firing angle with minimal computational resources. Additionally, it is robust against variations and uncertainties, maintaining accurate results.

7- References

[1] Sari, V. (2023). Effect of Change of Reluctance Launcher Parameters on Projectile Velocity. IEEE Access, 11(August), 90027–90037.

[2] Alridha, A. H. (2023). Optimization Algorithms for Projectile Motion: Maximizing Range and Determining Optimal Launch Angle. Journal of Fundamental Mathematics and Applications (JFMA), 6(2), 176–187.

[3] Karadag, M. (2020). A study for determining the launch angle that maximises the total distance travelled by the projectile during its flight in the projectile motion. Physics Education, 55(3), 1–5.

[4] Regodić, D., Spalević, P., Milić, D., Jović, S., & Regodić, R. (2020). Application Integration Method for the Calculation of Flight Projectile. Journal of the Astronautical Sciences, 67(4), 1189–1205.

[5] Kahrizi, M. R., & Kabudian, S. J. (2020). Projectiles Optimization: A Novel Metaheuristic Algorithm for Global Optimization. 33(10), 1924–1938.

[6] Roux, A., Weber, J., Lauffenburger, J. P., Changey, S. (2022, November). Projectile trajectory estimation: an LSTM approach. In Conference on Artificial Intelligence for Defense.

[7] Amer, W. S., Abady, I. M., & Farag, A. M. (2024). Stability analysis of an acted asymmetric rigid body by a gyrostatic moment and a constant body-fixed torque. Journal of Low Frequency Noise Vibration and Active Control, 43(1), 325–341.

[8] Amer, W. S. (2022). The dynamical motion of a rolling cylinder and its stability analysis: analytical and numerical investigation. Archive of Applied Mechanics, 92(11), 3267–3293.

[9] Amer, W. S. (2021). Modelling and analyzing the rotatory motion of a symmetric gyrostat subjected to a Newtonian and magnetic fields. Results in Physics, 24, 104102.

[10] Escobar, I., Arribas, E., Ramirez-Vazquez, R., & Beléndez, A. (2022). Projectile motion revisited: Does the distance between the launcher and the object always increase? Journal of King Saud University - Science, 34(3), 101842.

[11] Abu-Bakr, A. F., & Zubarev, A. Y. (2019). Hyperthermia in a system of interacting ferromagnetic particles under rotating magnetic field. Journal of Magnetism and Magnetic Materials, 477, 404–407.

[12] Abu-Bakr, Ali F., & Zubarev, A. Y. (2020). Effect of ferromagnetic nanoparticles aggregation on magnetic hyperthermia. European Physical Journal: Special Topics, 229(2–3), 323–329.

[13] Clifton, J., & Laber, E. (2020). Q-learning: Theory and applications. Annual Review of Statistics and Its Application, 7, 279-301.

[14] Watkins, C.J.C.H., Dayan, P. Q-learning. Mach Learn 8, 279–292 (1992).

[15] Kucharski, B., Ziel, A., Hickey, M., & Travers, C. (2018, October). Real-World Projectile Catching with Reinforcement Learning: Empirical Analysis using Discretized Simulations. In 2018 IEEE MIT Undergraduate Research Technology Conference (URTC) (pp. 1-5). IEEE.

[16] Khenak, F. (2010). V-learning. 2010 International Conference on Computer Information Systems and Industrial Management Applications, CISIM 2010, 292, 228–232.