J. Electrical Systems 20-7s (2024): 2765-2778

[1]Deyong Jiang

Pingli Lyu[2*]

Ziming Duan[3]

Autonomous Robot Navigation Based on Depth Deterministic Policy Gradient

Abstract: - Generally speaking, robot navigation can take the place of people in hazardous work contexts including military settings and fires. Furthermore, these robots are frequently assigned to carry out repetitive motions such as delivering things inside like places and the limited observable environment of autonomous robots causes the training efficiency of route planning method to be poor and convergence speed to be slow and is applied to navigation robot path planning. In this manuscript, Autonomous Robot Navigation Based on Depth Deterministic Policy Gradient (DDPG-FFOA-ARN-RL) is proposed. Initially input data are gathered from Indoor Robot Navigation Dataset (IRND). To execute this, input data is pre-processed using Switching Hierarchical Gaussian Filter (SHGF) and it removes the noise from collected data; then the Pre-processed data are fed to Depth Deterministic Policy Gradient (DDPG) for effectively categorize the autonomous robot navigation. Then using the Reinforcement Learning the robot navigation is identified. Generally, DDPG doesn't express adapting optimization approaches to determine optimal parameters to ensure accurate autonomous robot navigation based depth deterministic policy gradient. Hence, the Fennec Fox Optimization Algorithm (FFOA) to optimize Depth Deterministic Policy Gradient; which accurately navigates the autonomous robot navigation. Then the proposed DDPG-FFOA-ARN-RL is implemented and the performance metrics like Success Rate, Navigation Time, Path Efficiency, Collision Rate and Energy Consumption around are analyzed. Performance of the DDPG-FFOA-ARN approach attains 18.75%, 26.89% and 32.57% higher Success Rate; 17.02%, 23.26% and 32.42% lower Navigation Time and 18.43%, 25.64% and 31.40% lower Energy Consumption when analyzed through existing techniques like Deep Deterministic Policy Gradient-depend Autonomous Driving for Mobile Robots in Sparse Reward Environments (DDPG-ADMR-SRE), automatic driving control technique depend on deep deterministic policy gradient (ADC-DDPG), multilayer decision-based fuzzy logic method to navigate mobile robot in unknown dynamic environments (MD-FLM-NMR) methods respectively.

Keywords: Deep Deterministic Policy Gradient, Fennec Fox Optimization Algorithm, Indoor Robot Navigation Dataset and Switching Hierarchical Gaussian Filter.

I. INTRODUCTION

Robots that can move are extensively employed across several domains, including enterprises, hospitals, and military settings. They are skilled in multiple tasks, such as personnel rescue, heavy object transportation, surveillance, and reconnaissance, among other uses [1]. Generally speaking, mobile robots can take the place of people in hazardous work contexts including military settings and fires [2]. Furthermore, these robots are frequently assigned to transport commodities in indoor environments that need repeated motions, including restaurants and hospitals [3]. Notably, mobile robots' capacity to drive themselves has given them the capacity to carry out all of these tasks. But in order to carry out these tasks properly, sophisticated autonomous driving technology react to robots' environment is needed [4]. Generally speaking, there are three main categories of autonomous driving: destination path planning, robot location determination, and surrounding environment detection [5-7].Light detection and ranging (LiDAR) sensors, radar, and cameras are common pieces of equipment used to collect environmental data during the recognition process [8]. As a result, path-planning research has been conducted using simultaneous localization and mapping technology, simultaneously executes dual processes: the position of robot and the creation of a map of surrounding environment, depend on such information data [9].As of right now, SLAM approach depend on LiDAR sensors is thought more precise for estimating location; yet, one drawback is the high cost of LiDAR sensors [10]. In this sense, using cameras or sonar sensors for the SLAM approach results in less precision than using LiDAR, but the cost of the sensors is reduced [11]. Therefore, complex search processes or AI technologies are usually utilized in path-planning process to

[1] [1]Yancheng Institute of Technology, Yancheng, Jiangsu 224007, China
[2*]Suzhou Vocational University, Suzhou, Jiangsu 215104, China
[3]Suzhou University of Science and Technology,Suzhou, Jiangsu 215009, China
[1]Email: jdy5207@ycit.cn

[2]*Corresponding author e-mail: lvpl@mail.xzcit.cn

assure low cost and increased performance [12]. As a result, a number of studies have employed sophisticated search algorithms or produced SLAM-based maps for path planning by gathering information from sonar and camera sensors. Global and local path planning are the two general categories into which the path-planning process falls [13]. Among these, global path planning makes use of all the information that is available about the surrounding environment, and a robot travels in accordance with the plan that is produced [14]. Additionally, static landscapes with defined impediments are the main uses for global path planning [15].

Local route planning, on the other hand, regulates a robot's movement by using only a portion of the available knowledge about the environment, likes dynamic environment by moving impediments [16]. There has been a lot of interest in the study of search processes utilized for local path planning since the real environment might be either dynamic or static [17].The algorithm, which finds shortest path between starting point and destination while avoiding obstacles, served as the foundation for the original proposal of the path-planning approach [18]. In a static environment with immobile barriers, the algorithm often finds the best path with the least distance. The heuristic function has an impact on the algorithm's performance, and it finds it difficult to identify the best course in a dynamic environment [19]. Time and memory usability must also taken into account when determining best course of action [20]. To minimise time complexity, entail storing sizable amount of data in large memory bank or utilizing small amount of memory, waiting for process to finish even if it takes longer.

A. Problem statement and Motivation behind this Research work

The existing methods for autonomous robot navigation based on depth deterministic policy Gradient have not provide an accurate output. Such techniques consume more time, lake of flexibility, few environmental conditions, sparse reward problem in mobile robot autonomous driving**,** the detonation of data may not appear correctly, took more process for accurate result. The proposed technique has overcome all of above disadvantages, achieved the predictable outcome for autonomous robot navigation based on depth deterministic policy Gradient [21-27].

In this paper, the autonomous robot navigation used to reward systems that are split into two categories as navigating path and eatables the best autonomous path. The autonomous robot states are laser distance sensor values, the distances to destination, preceding action.

B. Contribution

- Using the aid of the enhanced DDPG-FFOA-ARN application approach, it is possible to resolve the robotic navigation system of the most basic system when it comes to autonomous robot driving.
- The proposed DDPG application approach can be utilized in conjunction with incentive engineering to enhance mobile robots' autonomous driving capabilities; this approach can be substituted for developing the ideal robot navigation system.

Remaining manuscripts arranged as below: section 2 defines literature review; section 3 portrays proposed method, section 4 displays results with discussions, section 5 conclusion.

II. LITERATURE SURVEY

Numerous investigation works were suggested in literature related to for Autonomous Robot Navigation Based on Depth Deterministic Policy Gradient; few current works are reviewed here;

Park et al. [21] have presented DDPG-depend ADMR in SRE. Here, an application of the hindsight experience replay methodology to mobile robots' DDPG-depend path-planning process in order to mitigate the performance deterioration brought on by sparse reward issues in autonomous driving robots that move. Experimental setting was virtual simulation built on the Gazebo platform, mobile robot in analysis was TurtleBot3 robot running on a robot operating system. It provides higher Success Rate and it provides lower Path Efficiency.

Zhang et al. [22] have presented an ADC technique depend on DDPG. Conventional automatic driving behaviour decision algorithms require the manual establishment of intricate rules, leading to extended vehicle decision-making times, subpar decision-making outcomes, and limited adaptation to novel surroundings. Reinforcement learning was a prominent technique in machine learning and intelligent control that has gained popularity recently. It is limited to learning fair and efficient policies through interactions with the environment. The present state of research on automated driving technology, prevalent automatic driving control techniques are presented. It provides lower Navigation Time and it provides higher Collision Rate.

Kamil and Moghrabiah [23] have presented MD-depend FLM to NMR in unknown dynamic environments. It is crucial to look at how mobile robots navigate in dynamic, uncertain surroundings. This work aims to address the following current issues: failure in complex scenarios; optimality; difficulty in planning in unpredictable, ever-changing surroundings; and difficulty in estimating obstacle velocity vector. Here, aims to comprehend non-collision movement of mobile robots in unknown dynamic environment and to offer multilayer decision-depend fuzzy logic method to identify answer for robot navigation through safe path though preventing all forms of barriers. It provides lower Energy Consumption and it provides higher Navigation Time.

He and Lv [24] have presented Robotic Control in Adversarial and Sparse Reward Environments: A Robust Goal-Conditioned Reinforcement Learning Approach. Here, an innovative, robust goal-conditioned reinforcement learning method for robotic control from E2E in situations with limited rewards and adversaries. To be more precise, a mixed adversarial attack technique that combines white-box, black-box attacks was provided to produce a variety of adversarial perturbations on observations. In the meantime, to transform a bad experience into good one, produce the policy trajectories disturbed by mixed adversarial attacks, hindsight experience replay technique taking observation perturbations into account is devised. It provides higher path efficiency and it provides higher Energy Consumption.

Luong and Pham [25] have presented incremental learning for autonomous navigation of mobile robots depend on DRL. Here a system for incremental learning for autonomous robot navigation. The navigation strategy was generated by using range finder laser sensor, online deep reinforcement learning. This works well for both robot's trajectory avoidance and destination navigation. An empirical study was carried out in both simulated, real-world environments. The outcomes demonstrate that, in simulation setting, suggested strategy may produce a navigation policy with over 90% accuracy after only 150k training rounds. It provides lower Navigation Time and it provides higher Path Efficiency.

Zhou et al. [26] has presented safe reinforcement learning method for autonomous navigation of mobile robots in dynamic environments. Here, the crucial factor is that the high dynamics and unpredictability of pedestrians significantly exacerbate the discrepancy between motion safety and navigation efficiency. This study addresses the difficulty by introducing Conflict-Averse Safe Reinforcement Learning (CASRL), a safe deep reinforcement learning system, enabling autonomous robot navigation in dynamic situations. In particular, initially distinguishes collision avoidance subtask from navigation task as a whole, keeps a safety critic to assess risk and safety of various operations. In order to remove their mutual interference, it then builds dual task-exact but method-agnostic policy gradients for the goal-reaching, collision avoidance subtasks. It provides lower Navigation Time and it provides lower Success Rate.

Ye et al. [27] have presented adaptive road configurations for enhanced autonomous vehicle-pedestrian interactions utilizing reinforcement learning. Here, the composition of road space in Right-of-Way has the potential to be refreshed in light of this disruptive development. To design strategies, intellectual control methods have been put forth; however, and it do not currently have operational framework that dynamically produce ROW plans for pedestrians, AVs in response to demand that changes in real time. This work investigates Reinforcement Learning (RL) techniques for ROW composition evolution based on microscopic traffic simulation. It provides higher path efficiency and it provides higher Collision Rate.

III. PROPOSED METHODOLOGY

In this proposed methodology, Autonomous Robot Navigation Based on Depth Deterministic Policy Gradient (DDPG-FFOA-ARN) is discussed for Autonomous Robot Navigation via Depth Deterministic Policy Gradient System. In general, the autonomous robot navigation using DDPG can be assigned to predict the best path tacking navigation for autonomous robot. Gathering by accurately identifying the presence, location, and characteristics of a navigation robot and then it is sent for further processing. These phases endure major two processes likes preprocessing and autonomous robot navigation in succeeding sectors. Block diagram of DDPG-FFOA-ARN is represented by Figure 1,

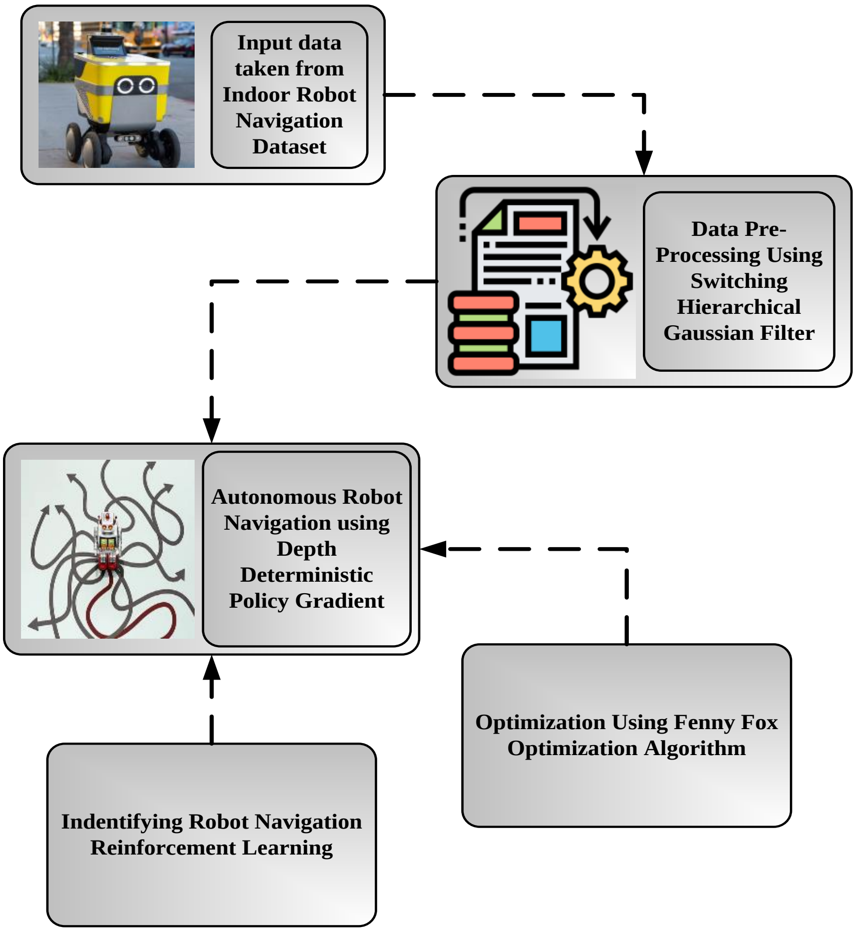

Figure1: Block Diagram for Proposed DDPG-FFOA-ARN-RL Method

A. Data acquisition

The input data is collected from Indoor Robot Navigation Dataset (IRND) [28].The 276 total files in the collection are arranged into 2 folders, each of which holds the data gathered from a single surface. The data from a smooth surface is in the outputs_2 folder, and the data from a rough surface is in the outputs folder. Every dataset file is a JavaScript Object Notation (JSON) file containing a series of data recorded from a single robotic control episode, i.e., data gathered from robotic actuators and sensors during the process of moving robot from its starting, resting position to desired position.

B. Pre-Processing using Switching Hierarchical Gaussian Filter

In this section, Switching Hierarchical Gaussian Filter (SHGF) [29] technique is utilized to remove the noise from the collected input data. SHGF is a part of Gaussian filter, because of its open-source modeling toolbox in readily, which helps to reduce the noise from the collected data. The SHGF is a multi-process nonlinear state space method in which states at higher process to regulate the noise variance of state transitions at given equation (1).

$$R^*\left(y_T^{(i)}, y_{T-1}^{(i)}\right) - \frac{1}{X^{(i)}_{a_{T,T-1}}} Exp\left(\mathrm{E}_{\left\{y_T^{(T)}, y_{T-1}^{(T)}\right\}}\left[\log(z,x)\right]\right) \tag{1}$$

Where, R^* indicates the input data; $y_T^{(i)}$ represents the accuracy of the noise emission distribution; $X^{(i)}_{a_{T,T-1}}$ denotes the normalization constant; $y_{T-1}^{(i)}$ indicates the uniformly distributed data; E represented as an observation data and $\log(z,x)$ indicates the over-sampling ratios from the collected data. Nonetheless, the measured signal may exhibit switching behaviour in numerous real robot applications. When parameter regime shifts govern the underlying dynamics of a time series, the SHGF model will not be able to adequately represent it. Then the basic noise; in data switching behavior in generative method inference for states, parameter of identification of noises given as equation (2).

$$X_{y_T}^{(i)} = \int_{\infty}^{\infty} \uparrow \upsilon\left(y_T^{(i)}\right) \mathrm{N}\left(y_T^{(i)} \middle| \tilde{n}_T^{(i)}, \tilde{u}_T^{(i)}\right) f y_T^{(i)} \quad (2)$$

Where, $X_{y_T}^{(i)}$ indicates the observation points in each dataset; $fy_T^{(i)}$ indicates the corresponding weights; υ indicates the nonlinear transformation; $y_T^{(i)}$ denotes accuracy of noise emission distribution; $\tilde{n}_T^{(i)}$ *and* $\tilde{u}_T^{(i)}$ denotes associated Gaussian statistics that come from the multiplication and N indicates the ground-truth parameters from the data. The SHGF is sophisticated generative method with non-linear couplings between the collected data and hierarchical regime switching dynamics. SHGF interested in emerging closed-form inference updates for states, both slowly time variable, regime-switching parameters which reduces the noise and it is given as equation (3).

$$\mathrm{E}\left[\left(y_T^{(i)}\right)^n\right] = \frac{\sum_L \uparrow \upsilon\left(\psi_L^{(i)}\sqrt{2\tilde{u}_T^{(i)}} + \tilde{n}_T^{(i)}\right)\left(\psi_L^{(i)}\sqrt{2\tilde{u}_T^{(i)}} + \tilde{n}_T^{(i)}\right)^n}{\sqrt{\pi}\, X_{y_T}^{(i)}} \quad (3)$$

Where, $y_T^{(i)}$ represents the accuracy of the noise emission distribution; $\sqrt{\pi}X_{y_T}^{(i)}$ indicates the noise vector that is randomly selected from the noise distribution; $\sqrt{2\tilde{u}_T^{(i)}}$ indicates the bounding noise variance of state transitions; $\tilde{n}_T^{(i)}$ *and* $\tilde{u}_T^{(i)}$ denotes the associated Gaussian statistics that come from the multiplication; $\psi_L^{(i)}$ denotes the noise locations that serve as collected data of robot navigation polynomial root; L denotes the time index and E represented as a observation data. Finally, by using SHGF the noise is removed from the collected data. Then the pre-processed data are fed to Autonomous Robot Navigation phase.

C. Autonomous Robot Navigation using Depth Deterministic Policy Gradient System

In this section, autonomous robot navigation using DDPG [30] is discussed, for effectively categorize the autonomous robot navigation and path. When actions selected through continuous action space, DDPG can manage the situation. Collisions between any individuals in the scenario can be effectively avoided by modifying the individual's pace. The method can more realistically simulate multiple path collision avoidance in crowded and complicated environments, minimise the degree of speed variations, and guarantee the avoidance of local barriers without jitter across groups for navigating robot using equation (4).

$$I = \mathrm{E}_{p_i, q_i \sim F, x_i \sim \pi}\left[P_1\right] \quad (4)$$

Where, I symbolizes the data generated by the pre-processing data; E ; $\left[P_1\right]$; P_t indicates the distance path's length expressed in meters; q_i indicates the state space of robot; π denoted as the policy of DDPG; F and x_i represents the likelihood of the information provided from the autonomous robot. An actor and a critic are the two components that make up the actor-critic. The current state of the environment, the actor predicts navigation path by applying a policy. Along with the path from the environment, that state the critic also receives the state using equation (5).

$$P_t = \sum_{T}^{i=t} \gamma^{i-t} p\left(q_i, x_i\right) \quad (5)$$

Where, P_t indicates the distance path's length expressed in meters; γ^{i-t} represented as the sate factor of the navigating the path diversion; q_i indicates state space of robot; x_i represents likelihood of information provided from the autonomous robot and p indicates the gradient descent as the activation loss function. To calculate the anticipated return following policy-compliant action in the present states to which provides the autonomous robot navigation using equation (6).

$$bt = p\left(q_t, x_t\right) + \gamma\, O\left(q_{t+1}, \mu\left(q_{t+1}\right)\middle|\theta^O\right) \tag{6}$$

Where, θ^O denotes the reference distance at a recognized place from which a reading is obtained; bt denotes the input to actor and critic net; O denotes likelihood that discriminator which determine whether the data produced by the generator are accurate data or not for robot navigation; γ represented as the sate factor of the navigating the path diversion; μ denotes the total number of iteration; and p specifies gradient descent as activation loss function; q_t indicates path loss exponent; x_t represents the idea of being perfect at every stage, which is utterly unrealistic and q_{t+1} symbolizes the route loss at the distance in path navigation. Then the autonomous robot navigation path is calculated using equation (7)

$$M = \frac{1}{n}\sum_i \left(b_i - O\left(q_i, x_i \middle|\theta^O\right)\right)^2 \tag{7}$$

Where, θ^O denotes the reference distance at a recognized place from which a reading is obtained; q_i indicates state space of robot; x_i represents likelihood of information provided from the autonomous robot; O denotes the likelihood that the discriminator which determine whether the data produced by the generator are accurate data or not for robot navigation; M denotes the actual data that was taken from the distribution of data and b_i denotes to find the best navigation path value of a robot. Finally, DDPG accurately analyze the Autonomous Robot Navigation. Due to its convenience, pertinence, AI-depend Optimization approach is taken into account in DDPG network. Here, Fennec Fox Optimization Algorithm (FFOA) is employed to optimize the DDPG. Here, FFOA is employed for tuning weight, bias parameter of DDPG.

D. Optimization Using Fennec Fox Optimization Algorithm

The FFOA is utilized to enhance weights parameters θ^O of proposed DDPG [31]. The smallest canid species, the fennec fox, is easily recognized thanks to huge ears. The fur of the fennec fox is straw in colour. Its tapering tail has a black tip. Its huge ears are so hairy inside that external auditory meat us is covered, and its back has longitudinal reddish stripes. On back, the ear margins are darker than the white ones. Because of its dense fur, its paw pads make walking in hot, sandy dirt easier. There is a picture of a fennec fox. Here, step by step procedure for obtaining appropriate DDPG values using FFOA is described here. To creates a uniformly distributed population for optimizing the ideal DDPG parameters. The entire step method is then presented in below,

Step 1: Initialization

Initial population of FFOA is, initially generated by randomness. Then the initialization is derived in equation (8).

$$A = \begin{bmatrix} a_{1,1} \cdots a_{1,j} \cdots a_{1,m} \\ \vdots \quad \ddots \quad \vdots \quad \iddots \quad \vdots \\ a_{i,j} \ \cdots a_{i,j} \ \cdots a_{i,m} \\ \vdots \quad \iddots \quad \vdots \quad \ddots \quad \vdots \\ a_{N,1} \cdots a_{N,j} \ \cdots a_{N,m} \end{bmatrix}_{N\times M} \tag{8}$$

Where, x denotes the total population of sand cat in the tracks; n indicates the n^{thh} number of FFOA while attacking towards its prey, D signifies distance among hunting prey, FFOA.

Step 2:Random Generation

Input weight parameter θ^O developed randomness via FFOA method.

Step 3:Fitness Function

It creates random solution from initialized values. It is calculated by optimizing parameter. It is given in equation (9)

$$Fitness\,Function = optimizing\ \ [\theta^{O}] \tag{9}$$

Step4:Exploration Phase

Wild predators like striped hyenas, caracals, and Pharaoh eagle-owls can attack fennec foxes. But due to its incredible speed and ability to evade predators, it made a sudden change in direction of motion. This fennec fox's escape tactic serves as foundation for worldwide scanning of search space in our mathematical method. The FFOA has more exploratory power when this escape plan is simulated. Finding the ideal global location is aided by avoiding becoming mired in the optimum local ones and it is given as equation (10).

$$a_{i,j}^{R2} = \begin{cases} a_{i,j} + Rand\ .\ \left(a_{i,j}^{Rand} - I.a_{i,j}\right), & G_i^{Rand} < G_i; \\ a_{i,j} + Rand\ .\ \left(a_{i,j} - a_{i,j}^{Rand}\right), & else, \end{cases} \tag{10}$$

Where, $a_{i,j}^{Rand}$ denotes j^{th} dimension of FFOA; G_i^{Rand} represented as value of its objective function; $a_{i,j}^{R2}$ denoted as the j^{th} dimension of target in FFOA; I represented a random number drawn from range $\{1,2\}$; $Rand$ represented as a random integer between $0\ and\ 1$; $a_{i,j}$ is the dimension of fenny fox in its j^{th} target and G_i denoted value of goal function that was found for fennec fox i^{th}. Finding ideal global location is aided by avoiding becoming mired in the optimum local ones. Thus, each candidate solution's random location within search space thought of as method for fennec fox's behaviour through its escape using equation (11).

$$A_i = \begin{cases} A_i^{R2}, & G_i^{R2} < G_i; \\ A_i, & else \end{cases} \tag{11}$$

Where, A_i^{R2} denoted as the second phase's new proposed status for the i^{th} fennec foxes; G_i^{R2} denoted as the objective function's value; G_i denoted value of goal function that was found for fennec fox i^{th} and A_i symbolizes range's i^{th} fennec fox in population. And Figure 2 shows the corresponding flowchart.

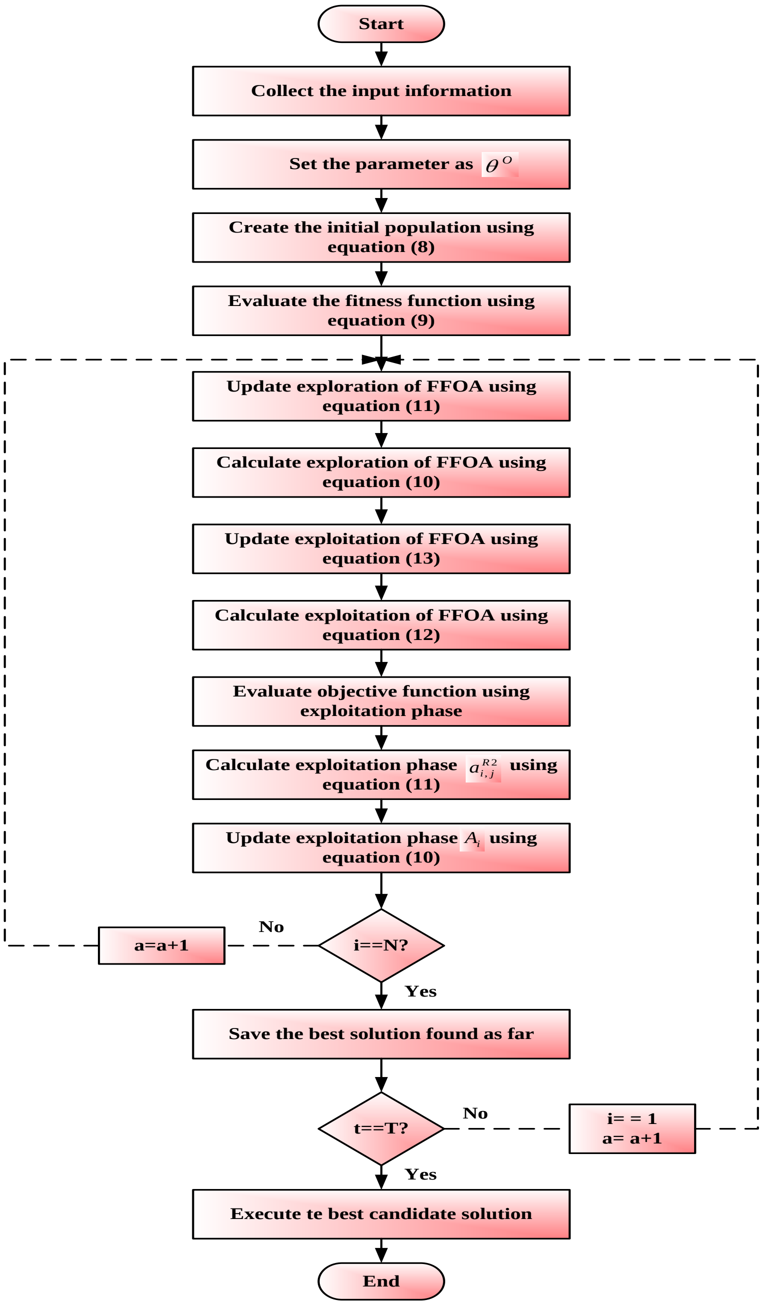

Figure2: Flow Chart of FFOA for Optimizing DDPG

Step5: Exploitation phase for optimizing θ^O

The fennec fox hunts by itself at night. It locates its prey by using its huge ears to discover sounds beneath the sand, it uses feet to dig a hole to reveal and pursue its meal. The exploitation potential of the FFOA is increased when it simulates the local search behaviour of fennec foxes, which leads to a solution that is closer to the global optima. A neighborhood with a radius R surrounding the fennec fox's real location is considered in order to simulate its behaviour while digging and it is given as equation (12).

$$S_{i,j} = \alpha.\left(1 - \frac{T}{t}\right).a_{i,j} \tag{12}$$

Where, $a_{i,j}$ is the dimension of fenny fox in its j^{th} target; α is represented as a constant set of fenny fox; $S_{i,j}$ represented as neighborhood radius for $a_{i,j}$; T denotes single iteration counter in FFOA and t indicates total iteration in FFOA. A local search in this area by the fennec fox can lead to a more optimal solution using equation (13).

$$A_i = \begin{cases} A_i^{R1}, & G_i^{R1} < G_i; \\ A_i, & else, \end{cases} \tag{13}$$

Where, G_i^{R1} denoted as value of its objective function; A_i symbolizes range's i^{th} fennec fox in the population; G_i denoted value of the goal function that was found for fennec fox i^{th} and A_i^{R1} indicates the revised status for ith fennec fox, as proposed by exploitation phase.

Step 6: Termination

The weight parameter value of generator θ^O from Depth Deterministic Policy Gradient System is optimized by utilizing FFOA; and it will repeat step 3 until it obtains its halting criteria $a = a + 1$.Then DDPG-FFOA-ARN accurately assesses for Autonomous Robot Navigation based on Depth Deterministic Policy Gradient byhigher Success Rate, lessening Navigation Time, error.

IV. RESULT WITH DISCUSSION

Experimental results of DDPG-FFOA-ARN are discussed. The simulation is implemented in PYTHON using PC through Intel core i5, 2.50 GHz CPU, 8GB RAM, windows 7 using Indoor Robot Navigation Dataset. Attained result of DDPG-FFOA-ARN method is analyzed with existing techniques likes DDPG-ADMR-SRE, ADC-DDPG and MD-FLM-NMR systems.

A. Performance measures

Performance of proposed method is examined utilizing performance metrics such as Success Rate, Navigation Time, Path Efficiency, Collision Rate and Energy Consumption.

1) Success Rate

The number of successful outcomes divided by total attempts or opportunities is a common formula for calculating success rate. The result is then multiplied by 100 to get the percentage using equation (14).

$$Success\ Rate(\%) = \frac{Number\ of\ Successful\ Outcomes}{Total\ Number\ of\ Attempts\ or\ Opportunities} \times 100 \tag{14}$$

2) Navigation Time

The particulars of the navigation system or context you're referring to may affect Navigation Time. Generally speaking, the time required to get from one place to another constitutes the way Navigation Time is determined using equation (15).

$$Navigation\ Time = Time\ of\ Arrival - Time\ of\ Depature \tag{15}$$

3) Path Efficiency

Path efficiency is a measurement of the degree to which a path or route is followed, frequently taking time, distance, and resource usage into account. Then path efficiency is given as equation (16).

$$Path\ Efficiency = \frac{Distance\ or\ Resource\ Expected}{Distance\ or\ Resource\ Actual} \times 100 \tag{16}$$

Where, $Distance\ or\ Resource\ Expected$ denotes the optimal or anticipated travel time or resource usage for the specified route and $Distance\ or\ Resource\ Actual$ indicates the real distance travelled or the amount of resources used when traversing.

4) Collision Rate

A metric called collision rate is utilized in many different areas, including computer networks, workplace safety, and traffic safety. Then the collision rate is given as equation (17).

$$Collision\ Rate = \frac{Number\ of\ Collisions}{Total\ Exposure\ or\ Usage} \times Multiplier \tag{17}$$

Where, $Number\ of\ Collisions$ indicate the total number of observed collisions rate in robot navigation; $Total\ Exposure\ or\ Usage$ reflects the entire amount of time, space, or use that the collisions took place and $Multiplier$ denoted as scale the rate in robot navigation.

5) Energy Consumption

The result of a certain amount of energy being spent is energy consumption. While the commercial unit of energy consumption is the kilowatt-hour, the SI unit of energy is the joule. Then the Energy Consumption is given as equation (18).

$$Tansmit - Energy - Consume = \frac{Transmit - current * voltage}{Timr \ for\, nodeTransmitPackets} \tag{18}$$

B. Performance analysis

Figure 3 to 7 shows simulation result of DDPG-FFOA-ARN method. The performance metrics are analyzed with existing DDPG-ADMR-SRE, ADC-DDPG and MD-FLM-NMR methods.

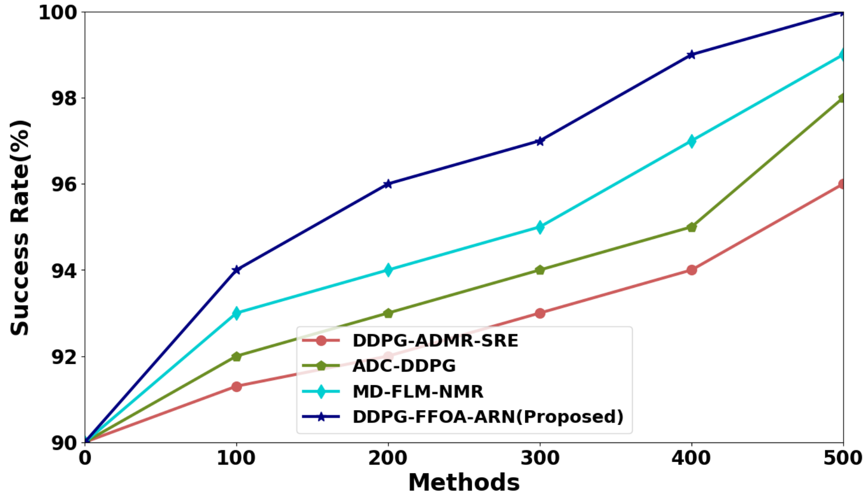

Figure 3: Performance analysis of Success Rate

Figure 3 depicts Success Rate analysis. The DDPG-FFOA-ARNattains 16.57%, 24.89% and 32.75% higher success rate at number of methods at 100; 17.12%, 21.92% and 35.61% higher success rate at number of methods at 300; 15.05%, 24.63% and 31.84% higher success rate at number of methods at 500; which are analyzed with DDPG-ADMR-SRE, ADC-DDPG and MD-FLM-NMR methods respectively.

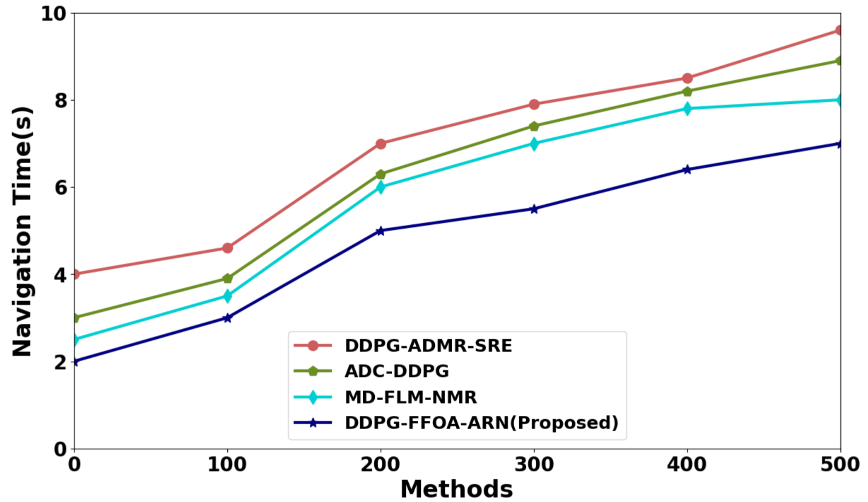

Figure 4: Performance analysis of Navigation Time

Figure 4 depicts Navigation Timeanalysis. The DDPG-FFOA-ARNattains 15.99%, 23.61% and 31.10% lower navigation timeat number of methods at 100; 18.10%, 25.77% and 31.60% lower navigation time at number of methods at 300; 16.21%, 22.08% and 34.75% lower navigation time at number of methods at 500; which is analyzed with DDPG-ADMR-SRE, ADC-DDPG and MD-FLM-NMR methods respectively.

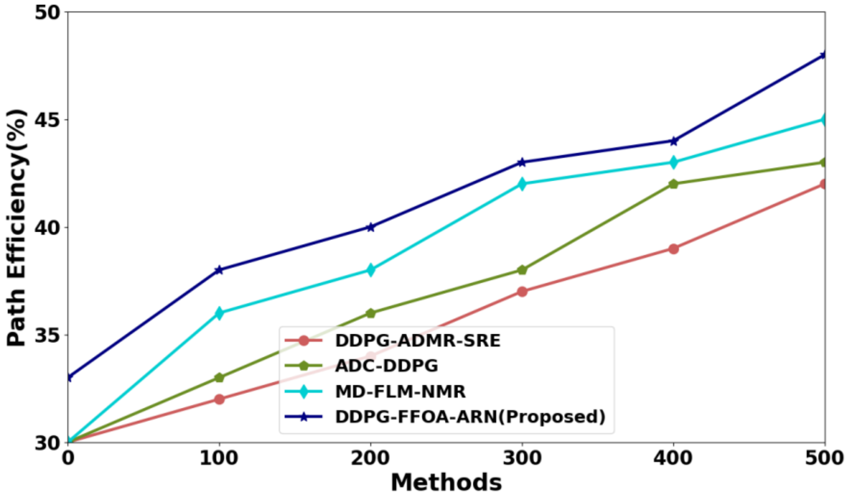

Figure 5: Performance analysis of Path Efficiency

Figure 5 depicts Path Efficiency analysis. The DDPG-FFOA-ARNattains 15.67%, 24.42% and 32.93% higher path efficiency at number of methods at 100; 17.31%, 25.61% and 32.30% higher path efficiency at number of methods at 300; 18.14%, 23.75% and 32.81% higher path efficiency at number of methods at 500; which are analyzed with DDPG-ADMR-SRE, ADC-DDPG and MD-FLM-NMR methods respectively.

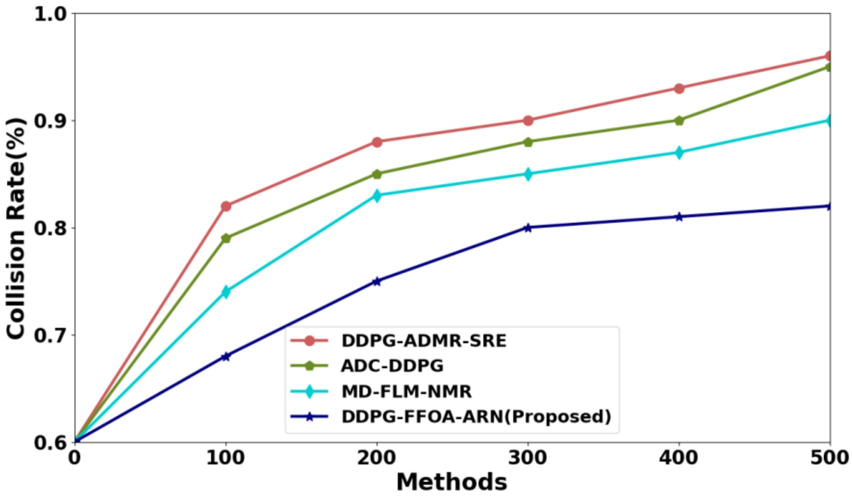

Figure 6: Performance analysis of Collision Rate

Figure 6 depicts Collision Rate analysis. The DDPG-FFOA-ARNattains 18.61%, 25.68% and 35.20% lower collision rate at number of methods at 100; 16.21%, 21.14% and 30.09% lower collision rateat number of methods at 300; 14.44%, 23.63% and 31.30% lower collision rateat number of methods at 500; which is analyzed with DDPG-ADMR-SRE, ADC-DDPG and MD-FLM-NMR methods respectively.

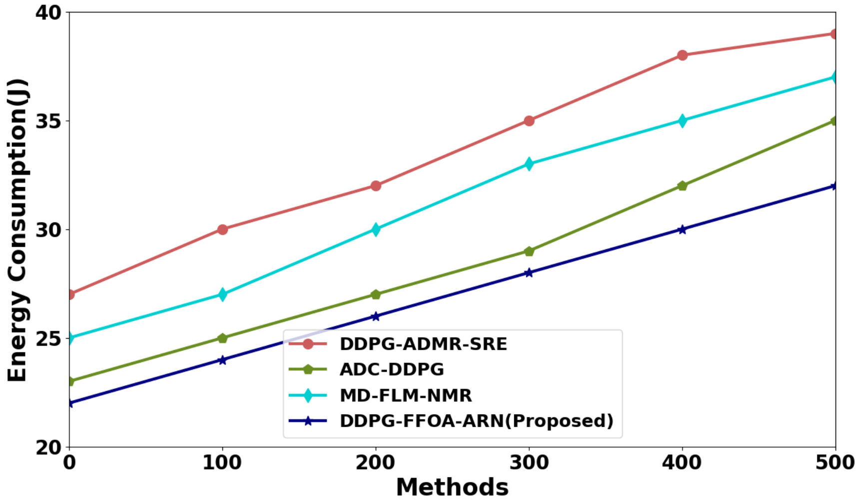

Figure 7: Performance analysis of Energy Consumption

Figure 7 depicts Energy Consumptionanalysis. The DDPG-FFOA-ARNattains 15.61%, 23.20% and 33.98% higher energy consumptionat number of methods at 100; 16.57%, 24.42% and 31.02% higher energy consumptionat number of methods at 300; 14.12%, 25.86% and 32.08% higher energy consumptionat number of methods at 500; which is analyzed with DDPG-ADMR-SRE, ADC-DDPG and MD-FLM-NMR methods respectively**.**

C. Discussion

A novel DDPG-FFOA-ARN model for Autonomous Robot Navigation Based Depth Deterministic Policy Gradient using collected data from Indoor Robot Navigation Dataset is developed in this paper. The DDPG-FFOA-ARN model involves encompasses SHGF based global data on Indoor Robot Navigation Dataset preprocessing and DDPG based path navigation for Autonomous Robot Navigation. Finally, DDPG model utilized for performing for Autonomous Robot Navigation which analyze an unknown environment by measuring distances. The instance of Indoor Robot Navigation dataset, the average highest outcomes of the approach were compared to the average results given in existing methods such as DDPG-ADMR-SRE, ADC-DDPG and MD-FLM-NMR. This is less expensive than comparing to the proposed method. The proposed DDPG-FFOA-ARN method for autonomous robot navigation from robot navigation dataset; however, the proposed method employs a faster DDPG in conjunction with the FFOA algorithm, resulting in a more efficient collection of data and a better ability to deal with the model over-fitting problem. The high selection of autonomous robot navigation values of DDPG-ADMR-SRE, ADC-DDPG and MD-FLM-NMR are 17.57%, 23.89% and 32.96% respectively higher than existing methods such as DDPG-ADMR-SRE, ADC-DDPG and MD-FLM-NMR respectively. Similar to this, the robot navigation path of proposed method is 96.94% analyzed with average path efficiency of comparison techniques of 82.42%. The proposed method DDPG-FFOA-ARN has high time consumption and collision rate than existing methods. Therefore, the comparative methods are expensive than the proposed technique. As a result, the proposed technique analyses the autonomous robot navigation more effectively and efficiently.

V. CONCLUSION

In this section, Autonomous Robot Navigation Based on Depth Deterministic Policy Gradient (DDPG-FFOA-ARN-RL) is successfully executed. Proposed DDPG-FFOA-ARN-RL method is applied in PYTHON with Indoor Robot Navigation Dataset. Then the data are pre-processed from Indoor Robot Navigation Dataset, for autonomous robot navigation based depth deterministic policy gradient analysis. According to the experimental results, DDPG-FFOA-ARN-RL performed better to improve the autonomous driving performance of robot navigation. The performance of DDPG-FFOA-ARN-RL approach attains 18.61%, 25.68% and 35.20% lower Collision Rate and 15.67%, 24.42% and 32.93% higher Path Efficiency when analyzed with existing methods like DDPG-ADMR-SRE, ADC-DDPG and MD-FLM-NMR respectively.

Acknowledgement

Scientific Research Initiation Project for High-level Talents in Xuzhou Institute of Industrial Technology: Research on Deep Reinforcement Learning Methods Based on Value Function Exploration Strategy (Project No. XGY2021EG02)

REFERENCE

[1] Jiang, J., Kong, D., Hou, K., Huang, X., Zhuang, H., & Fang, Z. (2023). Neuro-planner: A 3d visual navigation method for mav with depth camera based on neuromorphic reinforcement learning. *IEEE Transactions on Vehicular Technology*.

[2] Yu, L., Huo, S., Wang, Z., & Li, K. (2023). Hybrid attention-oriented experience replay for deep reinforcement learning and its application to a multi-robot cooperative hunting problem. *Neurocomputing*, *523*, 44-57.

[3] Staroverov, A., Yudin, D.A., Belkin, I., Adeshkin, V., Solomentsev, Y.K., & Panov, A.I. (2020). Real-time object navigation with deep neural networks and hierarchical reinforcement learning. *IEEE Access*, *8*,195608-195621.

[4] Hu, H., Zhang, K., Tan, A.H., Ruan, M., Agia, C. & Nejat, G. (2021). A sim-to-real pipeline for deep reinforcement learning for autonomous robot navigation in cluttered rough terrain. *IEEE Robotics and Automation Letters*, *6*(4), 6569-6576.

[5] Samsani, S.S. & Muhammad, M.S. (2021). Socially compliant robot navigation in crowded environment by human behavior resemblance using deep reinforcement learning. *IEEE Robotics and Automation Letters*, *6*(3), 5223-5230.

[6] Wang, J., Elfwing, S. & Uchibe, E. (2021). Modular deep reinforcement learning from reward and punishment for robot navigation. *Neural Networks*, *135*, 115-126.

[7] Liu, B., Xiao, X. & Stone, P. (2021). A lifelong learning approach to mobile robot navigation. *IEEE Robotics and Automation Letters*, *6*(2), 1090-1096.

[8] Wellhausen, L., Ranftl, R. & Hutter, M., (2020). Safe robot navigation via multi-modal anomaly detection. *IEEE Robotics and Automation Letters*, *5*(2), 1326-1333.

[9] Ran, T., Yuan, L. & Zhang, J.B., (2021). Scene perception based visual navigation of mobile robot in indoor environment. *ISA transactions*, *109*, 389-400.

[10] Devo, A., Mezzetti, G., Costante, G., Fravolini, M.L. & Valigi, P., (2020). Towards generalization in target-driven visual navigation by using deep reinforcement learning. *IEEE Transactions on Robotics*, *36*(5), 1546-1561.

[11] Hodge, V.J., Hawkins, R. & Alexander, R., (2021). Deep reinforcement learning for drone navigation using sensor data. *Neural Computing and Applications*, *33*, 2015-2033.

[12] Back, S., Cho, G., Oh, J., Tran, X.T. & Oh, H., (2020). Autonomous UAV trail navigation with obstacle avoidance using deep neural networks. *Journal of Intelligent & Robotic Systems*, *100*, 1195-1211.

[13] de Queiroz Mendes, R., Ribeiro, E.G., dos Santos Rosa, N. & GrassiJr, V., (2021). On deep learning techniques to boost monocular depth estimation for autonomous navigation. *Robotics and Autonomous Systems*, *136*, 103701.

[14] Kahn, G., Abbeel, P. & Levine, S., (2021). Badgr: An autonomous self-supervised learning-based navigation system. *IEEE Robotics and Automation Letters*, *6*(2), 1312-1319.

[15] Zhang, Z., Kayacan, E., Thompson, B. & Chowdhary, G., (2020). High precision control and deep learning-based corn stand counting algorithms for agricultural robot. *Autonomous Robots*, *44*(7), 1289-1302.

[16] Han, R., Chen, S., Wang, S., Zhang, Z., Gao, R., Hao, Q. & Pan, J., (2022). Reinforcement learned distributed multi-robot navigation with reciprocal velocity obstacle shaped rewards. *IEEE Robotics and Automation Letters*, *7*(3), 5896-5903.

[17] Aguiar, A.S., Dos Santos, F.N., De Sousa, A.J.M., Oliveira, P.M. & Santos, L.C., (2020). Visual trunk detection using transfer learning and a deep learning-based coprocessor. *IEEE Access*, *8*, 77308-77320.

[18] Liu, Y., Zhang, W., Pan, S., Li, Y. & Chen, Y., (2020). Analyzing the robotic behavior in a smart city with deep enforcement and imitation learning using IoRT. *Computer Communications*, *150*, 346-356.

[19] Fang, B., Mei, G., Yuan, X., Wang, L., Wang, Z. & Wang, J., (2021). Visual SLAM for robot navigation in healthcare facility. *Pattern Recognition*, *113*, 107822.

[20] Yang, Y., Bevan, M.A. & Li, B., (2020). Efficient navigation of colloidal robots in an unknown environment via deep reinforcement learning. *Advanced Intelligent Systems*, *2*(1), p.1900106.

[21] Park, M., Lee, S.Y., Hong, J.S. & Kwon, N.K., (2022). Deep Deterministic Policy Gradient-Based Autonomous Driving for Mobile Robots in Sparse Reward Environments. *Sensors*, *22*(24), 9574.

[22] Zhang, H., Xu, J. & Qiu, J., (2022). An automatic driving control method based on deep deterministic policy gradient. *Wireless Communications and Mobile Computing*, *2022*, 1-9.

[23] Kamil, F. & Moghrabiah, M.Y., (2022). Multilayer decision-based fuzzy logic model to navigate mobile robot in unknown dynamic environments. *Fuzzy Information and Engineering*, *14*(1), 51-73.

[24] He, X. and Lv, C., (2023). Robotic Control in Adversarial and Sparse Reward Environments: A Robust Goal-Conditioned Reinforcement Learning Approach. *IEEE Transactions on Artificial Intelligence*.

[25] Luong, M. & Pham, C., (2021). Incremental learning for autonomous navigation of mobile robots based on deep reinforcement learning. *Journal of Intelligent & Robotic Systems*, *101*(1), 1.

[26] Zhou, Z., Ren, J., Zeng, Z., Xiao, J., Zhang, X., Guo, X., Zhou, Z. & Lu, H., (2023). A safe reinforcement learning approach for autonomous navigation of mobile robots in dynamic environments. *CAAI Transactions on Intelligence Technology*.

[27] Ye, Q., Feng, Y., Macias, J.J.E., Stettler, M. & Angeloudis, P., (2022). Adaptive road configurations for improved autonomous vehicle-pedestrian interactions using reinforcement learning. *IEEE transactions on intelligent transportation systems*, *24*(2), 2024-2034.

[28] https://www.kaggle.com/datasets/narayananpp/indoor-robot-navigation-dataset-irnd

[29] Şenöz, İ., Podusenko, A., Akbayrak, S., Mathys, C. & de Vries, B., (2021, July). The Switching Hierarchical Gaussian Filter. In *2021 IEEE International Symposium on Information Theory (ISIT)* (pp. 1373-1378). IEEE.

[30] Grabowsky, D.P., Conrad, J.M. & Browne, A.F., (2021, March). Limited log-distance path loss model path loss exponent estimation using deep deterministic policy gradient. In *SoutheastCon 2021* (pp. 1-6). IEEE.

[31] Trojovská, E., Dehghani, M. & Trojovský, P., (2022). Fennec fox optimization: A new nature-inspired optimization algorithm. *IEEE Access*, *10*, 84417-84443.