Military Adaptation of Self-driving System using Reinforcement Learning

[1]Priyesh Patel, [2]Meet Nandu, [3]Saifil Momin, [4]Dr. Neepa Shah
[1,2,3]Student, [4]Head of Department
Department of Information Technology,
Dwarkadas J. Sanghvi College of Engineering, Mumbai, India

***Abstract*—People all around the world are enthusiastic about the advent of autonomous vehicles for the public. There are many paradigm shifts taking place in this sector due to information explosion and the advancements in Artificial Intelligence. Although self-driving technology has permeated the commercial sector, its potential is not being fully utilized in the military sector where such autonomy can save millions of lives and significantly reduce the training costs. Existing autonomy in the military is limited to the use of remotely controlled automatic vehicles. Self-navigating autonomous vehicles can optimize this functioning by inducing the power of decision making that enables self-learning. In this paper, we extensively studied the existing Unmanned aerial Vehicle (UAV), Unmanned ground vehicle (UGV), and Unmanned underwater (UUV) vehicle systems and their drawbacks where we compared the performances of the existing algorithms based on vital parameters. We further state scope of Reinforcement Learning (RL) alongside with Convolutional Neural Network in autonomous military vehicles, which overcomes the drawbacks on existing algorithms. Consequently, we propose a reward-based learning model using RL to train the agent to navigate in a dynamic simulation environment by means of obstacle detection and global path optimization.**

***Index Terms*—Autonomous vehicles, self-driving vehicles, UAV, UGV, UUV, Reinforcement Learning, Convolutional Neural Network**

I. INTRODUCTION

Every year countries spend a tremendous amount of money in military, which keeps on increasing annually. In 2017, the global military expenditure was $1739 billion [1]. Moreover, approximately 100,000 military-related deaths occur every year and a significant amount of the military budget is spent on the training of the troops, where the US alone spent $572 million on training in 2017 [2]. Thus, there is no doubt that revision of the technologies and methodologies used by the military can save millions of lives as well as significantly reduce the expenditures.

Even though the self-driving concept is being extensively developed in the commercial sector [3], the systems currently being used by the military lacks autonomy as they are remotely operated and require human intervention, thus fail to eliminate the direct human interaction in combat. These systems solely operated by informed search methods lack the ability of self-navigation and adapting to the dynamic environment, thus, limiting their usage in actual combat. By introducing self-governed decision-making systems, powered by more adaptive algorithms we can achieve such autonomy.

In recent years, reinforcement learning (RL) has proved to be a tested method for efficiently navigating the agent in gaming simulation, such as Atari [4]. The Q-learning method used by RL enables a reward-based decision-making system, where the most optimal path is calculated. Such a dynamic navigation implementation can benefit the autonomous military vehicle in calculating the next move based on the reward formulated on each point.

In this paper, we study the suitability of RL for navigation and decision making in an autonomous military vehicular system and its advantages over the traditionally used algorithms. We also studied the role of Convolutional Neural Network (CNN) for classifying the obstacles infested in the simulation [5], thus, we consequently proposed our system architecture which predicts the agent's next move based on a reward system.

Following is the organization of the paper: section II describes the existing systems and the related algorithms, section III gives our proposed approach and the role of CNN in the system, section IV describes the detailed system architecture with the working, assumptions, and constraints. In section V, we have stated the conclusion of this study.

II. EXISTING SYSTEMS

This section describes the algorithms and techniques currently being used by military vehicles and commercial vehicles to navigate and take autonomous decisions if any. Currently, the military uses Unmanned Aerial Vehicles (UAVs), Unmanned Ground Vehicles (UGVs) and Unmanned Underwater Vehicles (UUVs) to reduce casualties involving humans.

- An Unmanned Aerial Vehicle (UAV) is an aircraft without a human pilot aboard. UAVs are a component of an Unmanned Aircraft System (UAS); which include a UAV, a ground-based controller, and a system of communications between the two.
- An Unmanned Ground Vehicle (UGV) is a vehicle which operates without any human operator on different terrains. The vehicle will have a set of sensors to observe environment and then based on those inputs a decision is made.
- An Unmanned Underwater Vehicle (UUV) is any vehicle that can operate underwater without a human occupant.

Unmanned Vehicles currently used by military vehicles are teleoperated which requires human intervention. This makes the system automatic but not autonomous. Unmanned Aerial Vehicles (UAVs) have more sophisticated systems to achieve autonomy in terms of missile locking and landing assists.

Unmanned Aerial Vehicle (UAV)

Defined as a powered, aerial vehicle, a UAV does not require a human operator onboard. It uses aerodynamic forces to provide vehicle lift and can be piloted remotely or fly autonomously. A UAV can be recoverable or expandable, and can carry a non lethal or lethal payload.

Military drones or UAVs can be separated into three major categories: midsize military drones, which also have commercial use, large size military-specific drones and stealth combat drones [6].

Unmanned Ground Vehicle (UGV)

It is a vehicle which operates without any human operator and with ground support. The vehicle will have a set of sensors to observe the environment and will make decision-based on its current position in the environment. There are different techniques of controlling the unmanned ground vehicle [7]. They are:

- Command control mode: In this mode, a human takes a decision and provides navigation commands, from a remote location, based on the live video signal received from a camera mounted on the UGV. The main aim of this mode is to ease the operation of UGV. This is done by giving inputs to the UGV, which could vary from a simple computer keyboard to other self-designed input devices.These commands from the user, are sent over to the UGV with the help of different wireless communication technologies like ZigBee or internet, while at the same time, the UGV transfers live video feedback to the user.
- Gesture control mode: In this mode, the UGV is controlled with the help of commands sent using hand gesture movement and then mapped by the Inertial Measurement Unit (IMU). The hand gestures are given by a human from a remote location and these gestures are then transferred to the UGV wirelessly using ZigBee or internet.
- Self-control mode: Enabling autonomous functioning of the UGV without any human supervision, is the main aim of this mode. Technologies such as GPS and magnetic compass for navigation of the UGV are used to accomplish this operation. The data provided by these technologies is enough for the system to operate as a self-navigated system. Obstacles are detected using infrared sensors. This way the UGV takes dynamic decision and tracks the motion of the vehicle.

The algorithm design for self-control mode is mainly achieved by two important factors called path planning and obstacle detection algorithms for the UGV to navigate accordingly. The user from the command station obtains the current GPS coordinates of the UGV and the destination coordinates to where UGV must navigate. The directions are obtained by the magnetic compass by trigonometric functions by calculating the specific angle required by the UGV to move accordingly. Thus, path planning algorithm decides the path taken by the UGV considering all these measurements. Path planning algorithm is done using the A* algorithm.

Now, obstacle detection is done by the IR sensors that detect the obstacles like mountains, rivers and thus in parallel update the path planning measurements accordingly. It also detects metals like bombs, metal objects those are buried in the ground and send this information to the command control station. Obstacle detection is not done in A* algorithm. Hence, a new modified D* algorithm is used for obstacle detection.

Thus, in self-control mode, path planning and obstacle detection algorithms must work in parallel in the most efficient manner based on all the measurements. On the other side command control station obtains all the measurements and signals from the UGV and coordinates the robot continuously. Based on the path planning and obstacle detection algorithms the UGV navigates to the destination given by the user itself automatically.

Unmanned Underwater Vehicle (UUV)

An Unmanned Underwater Vehicle is a vehicle that travels underwater without requiring input from an operator [8]. The Unmanned Underwater Vehicle can be divided into two main categories – Remotely Operated Vehicles (ROVs) and Autonomous Underwater Vehicles (AUVs).

1. **Remotely Operated Vehicles (ROVs)**

As the name suggests, these vehicles are operated remotely by a crew on board a nearby vessel or on proximate land. ROVs were developed to beat the constraints of human divers and human-occupied diving vehicles. They are used extensively by many navies, primarily for mine hunting and mine breaking.

2. Autonomous Underwater Vehicles (AUVs)

Unlike ROVs, that need commands from a human operator all the time, AUVs are capable of functioning without real-time control from a human operator. They are given the starting position and the goal position and then based on the different route finding algorithms, AUVs decide on a path and then complete the mission. On their path, they are programmed to collect data and send it to the base station.

Extensively used for surveying by the oil and gas industry and in scientific research, AUVs are also used for naval defense applications such as mine detection, payload delivery and surveillance. Military AUVs have been networked with UAVs to provide a cross-domain Intelligence, Surveillance and Reconnaissance (ISR) solution for the battlefield [8].

Common Algorithms

Route planning is an important part for any autonomous vehicle. Nowadays, the application of these vehicles is more and more extensive. There are many uncertain factors in the environment in which these vehicles have to run, or these vehicles need to be operated in a completely unknown environment. Hence, route planning is an important part while building these systems. The route planning involves many algorithms for different scenarios. Some of the currently used algorithms in these vehicles are:

- Genetic Algorithm
- Ant-Colony Algorithm
- A* Algorithm
- D* Algorithm

1. Genetic Algorithm

Predicated on the idea of natural selection and genetics, Genetic Algorithm (GA) was formed. Genetic algorithm is an adaptive heuristic search algorithm that can be used to find a path between a source node and goal node. The process of genetic algorithm commences with a set of individuals which is known as Population. Everyone in this population is a possible solution to the problem. The fitness function then calculates how fit an individual is by giving a fitness score to everyone. Based on this fitness score, the fitter individuals undergo crossover and mutation (like in natural genetics), to yield fitter individuals. This theory goes with the Darwinian Theory of “Survival of the fittest”. In this way, we keep “evolving” better individuals or solutions over generations, till we reach a stopping criterion. Fig. 1 shows how the genetic algorithm works in a simpler manner.

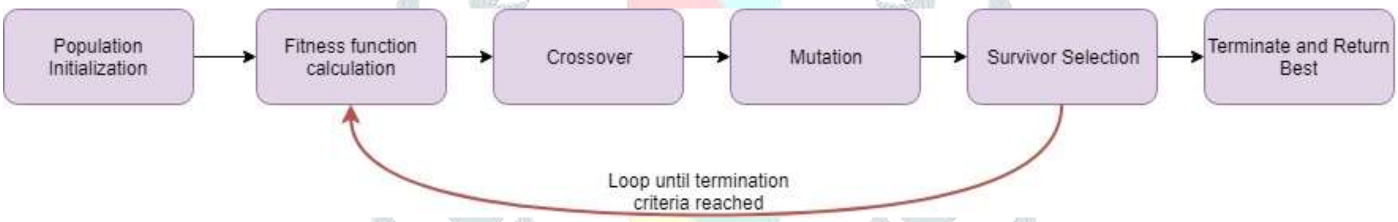

Figure 1: Flow of Genetic Algorithm

2. Ant Colony Algorithm

Predicated on the natural behavior of ants probing for food, the ant colony algorithm was developed for finding optimal paths between two given nodes. Ant colony algorithm is a probabilistic search algorithm that uses bioinformatics hormones as a basis for the follow-up behavior of ants. As shown in Fig. 2(a), the ants wander in a haphazard manner at first. When an ant finds a source of food, it ambulates back to the colony leaving "markers" (pheromones) that show that this path will lead to food, as shown in Fig. 2(b). Then when other ants come upon these markers, they are liable to follow the trail with a likelihood. If they do follow the same trail, they populate the trail with their own markers as they bring the food back. As additional ants realize this trail, it gets stronger till there are a few streams of ants traveling to varied food sources proximate to the colony. Because the ants drop pheromones every time they bring food, shorter paths are more likely to be stronger, as shown in Fig. 2(c) hence optimizing the "solution."

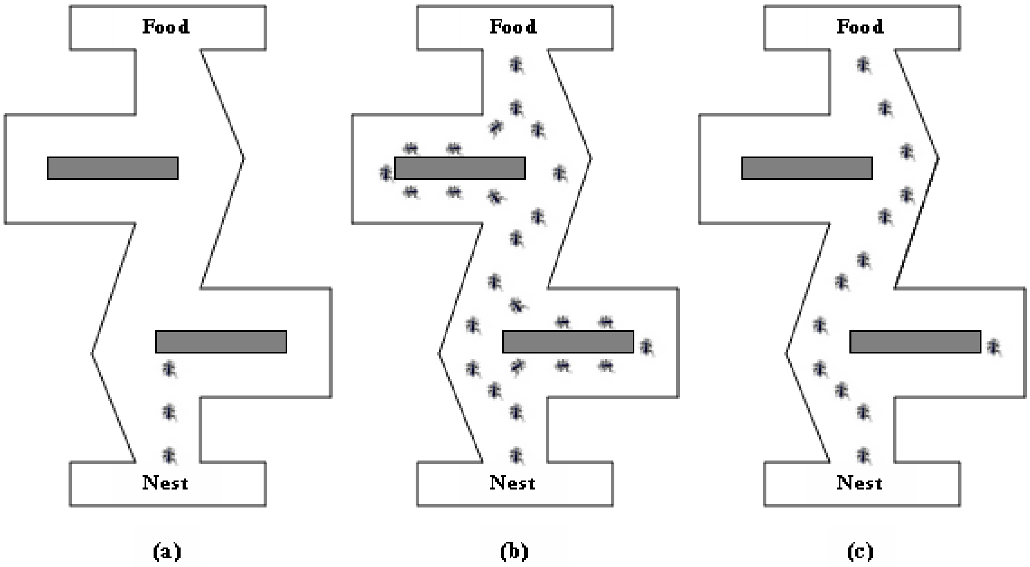

Figure 2: Path map of Ants [9]

3. A* algorithm

One of the most popular algorithms for finding the shortest path between two locations in a mapped area is A*. A* was developed to combine heuristic approaches like Best-First-Search (BFS) and formal approaches like Dijkstra's algorithm. In this algorithm, all nodes have a cost associated with themselves. This cost is given by f(n) = g(n) + h(n), where g(n) is the cost of the path from the start node to that node n and h(n) is the heuristic value or the approximate cost of the path from the node n till the goal node. At each point in the map or at every node in the graph, the next node with the lowest f(n) value is chosen for further expansion of the path. If there is a tie among two nodes of equal f values, the node with lower h value should be chosen. Whenever we reach the goal node, the algorithm terminates. The best thing about this algorithm is that if the heuristic function is admissible, meaning it never overestimates the actual cost, then the algorithm will guide us to an optimal path.

4. D* Algorithm

The D* algorithm solves the path planning problem, where the system must navigate to a given location or goal node in an unknown terrain. First, the algorithm will assume that the terrain it has to travel in contains no obstacles and finds the shortest path from its current location to the goal location under these assumptions. This trail is then followed by the system. As and when another hindrance comes up, it adds the information about this obstacle to its map and, re-plans a new shortest path from its current location to the given goal location. This process is reiterated unless the system reaches the goal position.

Table 1: Comparison of algorithms

Parameter	Genetic Algorithm	Ant-Colony	A*	D*
Will find a path?	Maybe	Yes	Yes	Yes
Will find the optimal path?	No	Yes	Yes	Yes
How fast will it find the optimal path?	-	Very Slow	Slow	Fast
Obstacle detection?	No	Yes	No	Yes

As per our research, we concluded D* algorithm is the best algorithm to implement any obstacle detection and routing protocol in autonomous vehicles from table 1. But while traversing unknown terrain, which is quite often in the military domain, new obstacles may be discovered frequently, so this replanning needs to be fast. Thus, D* might be a bit slow if new obstacles come up quite frequently. So upon further research, we came across Reinforcement Learning algorithms.

Reinforcement Learning

Supervised learning algorithms have a target label to which it tries to classify the data. On the other hand, unsupervised learning algorithms have no labels to target. Reinforcement algorithms fall between unsupervised and supervised learning. The reinforcement learning has a future reward system based on sparse labels and they are delayed in time. Depending on these reward system, the agent adapts to the environment. The goal of reinforcement learning algorithms is developing an efficient learning algorithm which can take into consideration all the limitations and merits.

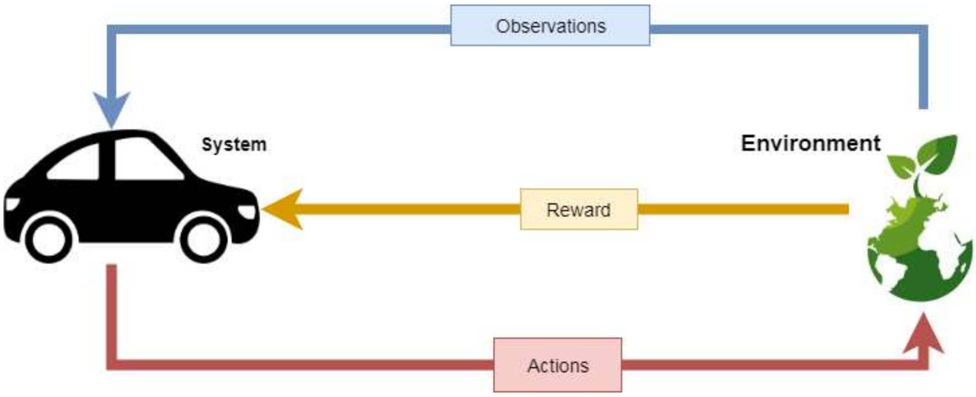

Figure 3: Flow of Reinforcement Learning

But all the algorithms we researched in RL, are currently used in games like Mario, AlphaGo, etc. No applications of RL was found in any autonomous vehicle and hence we decided that RL could provide better results than the other algorithms in an autonomous vehicle.

III. PROPOSED ALGORITHMS AND APPROACH

Convolutional Neural Networks

Convolutional neural network (CNN) is currently being used by commercial self-driving cars for image classification using lane detection and obstacle detection. It is combined with localization for this purpose. Such a system can take input from the camera sensors and navigate through local roads with or without lane markings, parking spaces and highways. With minimal human training, it learns to classify the road signs and signals [10].

- We will use CNN for image classification taken by the sensors of the system and use the localization concept for object detection.
- Reinforcement Learning algorithms will help us find the optimum path and navigate our system through a smooth path on a difficult terrain.

Proposed Approach

The final system that we aim to make is a simulation involving a gaming engine which will take inputs from virtual sensors and grab future frames in RGB format for path calculation from the environment. Once an agent receives the coordinates of the destination, it makes dynamic decisions based on the calculated parameters from the environment. If the frames that are used as an input in the CNN model detects obstacles, it uses localization to bound the object by a boundary.

The reinforcement learning algorithm will help to find the optimum global path for the agent to reach the destination. If the sensors detect any obstacle, a local path to avoid the obstacle is found and hence the global path is updated with it. The vehicle actuators would hence take dynamic decisions to move the agent according to the path specified by the algorithm.

IV. PROPOSED SYSTEM ARCHITECTURE

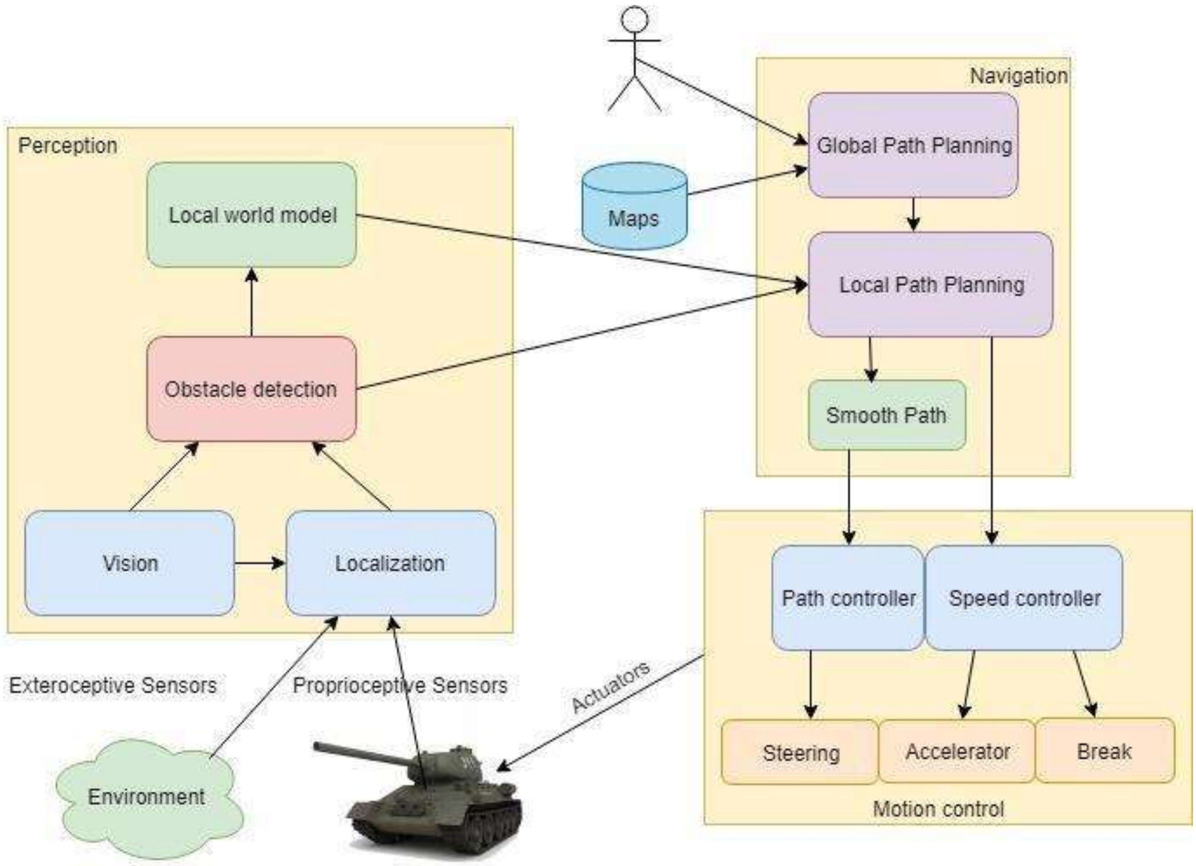

Figure 4: System Architecture

As shown in figure 4, the three main modules proposed in the system are:

Perception:

Analyzing the environment by taking inputs from various sensors and providing meaningful information from it is an important part of autonomous systems. Sensors give vision to the system and are classified based on two functional axes: Passive or Active, Proprioceptive or Exteroceptive[11].

- **Proprioceptive** sensors measure the systems internal phenomena; e.g. axle angle, battery voltage, pressure on the wheel, etc.
- **Exteroceptive** sensors extract information from the systems environment.; e.g. intensity of light, the amplitude of sound and distance measurements.
- **Passive** sensors do not try to alter the environment, they just sense the ambiance and calculate the results based on the energy. Examples of passive sensors include CMOS cameras for obstacle detection, infrared sensors.
- **Active** sensors measure the environment's response to the energy emitted by the sensors. Active sensors may provide better results because of controlled interactions, but it possesses a risk of altering, destroying or changing the characteristics of the environment or phenomenon that it is currently sensing.
- **Localization** helps to identify the object with a boundary box surrounding it. This is often referred to as classification with localization.

Navigation

- **Local Path planning:** On detecting an obstacle, the system finds an immediate path to avoid the obstacle. This technique utilizes the concept of local path planning and finds a smooth path that alters the global path in a minimal way.
- **Global Path planning:** Once the system receives the coordinates of the destination from the user, it utilizes the proposed algorithms to find the optimal path to reach the destination.

Motion Control

The actuators like steering, accelerator, brake, etc. take input from the Navigation module which has a Path controller and a Speed controller to move the vehicle. With every move, the navigation module makes dynamic decisions and adapts as per the requirements.

Assumptions and Constraints

The certain assumptions associated with this system are listed below:

1. **Technical Assumptions**
   - The training environment independently developed will be well integrated with the ML algorithm designed.
   - The training environment will provide an option to modify the map design.
   - Reinforcement Learning will provide the same accuracy in real life scenario as it does on the simulation.

2. **Logical Assumptions**
   - The global path devised by the system will be the shortest path.
   - Quick adaptation to new coursework.

3. **Constraints**

There are certain constraints which hinder the technical implementation of the system. These constraints are identified below:

- Technical limitation in the implementation of the training environment.
- Physical constraints which do not allow the actual on-field implementation of the algorithm.
- Lack of hardware planning and knowledge to implement a ready-to-go autonomous system.
- Lack of multiple environments for training and testing the algorithm.

V. CONCLUSION

In this paper, a study of existing systems for autonomous military vehicles, namely, UGV, UAV, and UUV has been presented. The study of these systems that make use of algorithms like genetic algorithms, Ant-colony, A*, and D*, elicited the drawbacks of these systems which rely on external human interaction for global path optimization in case of an adaptive environment. Consequently, we study the suitability of reinforcement learning for navigation of the military vehicle, where reinforcement learning proved to overcome the shortcomings of the existing implementations by training the learning agent to adapt to the dynamic environment by using a reward based system.

We further propose our model, where we utilize convolutional neural network trained on image frame input from the environment to identify the obstacles in the environment. The detected obstacles alongside with the environmental parameters are then utilized by the navigation segment to devise an optimal path to the destination where reinforcement learning plays a vital role in the calculation of the agent's moves. Thus, we have presented an improved model which eliminates the shortcomings of the existing unmanned systems by introducing dynamic decision-making by means of environment feedback and reward based training.

REFERENCES

[1] "Trends In World Military Expenditure", Sipri.org. 2017. [Online]
Available: https://www.sipri.org/sites/default/files/2018-04/sipri_fs_1805_milex_2017.pdf [Accessed: Oct, 2018]

[2] "Army Is Spending Half a Billion to Train Soldiers to Fight Underground", Military.com, June 2018. [Online] Available: https://www.military.com/daily-news/2018/06/24/army-spending-half-billion-train-troops-fight-underground.html [Accessed: Sept, 2018]

[3] E. Ohn-Bar and M. M. Trivedi, "Looking at Humans in the Age of Self-Driving and Highly Automated Vehicles," in IEEE Transactions on Intelligent Vehicles, vol. 1, no. 1, pp. 90-104, March 2016. doi: 10.1109/TIV.2016.2571067

[4] Mnih, V., Kavukcuoglu, K., Silver, D., Graves, A., Antonoglou, I., Wierstra, D., and Riedmiller, M. (Dec 2013). Playing Atari with deep reinforcement learning. Technical Report arXiv:1312.5602 [cs.LG], Deepmind Technologies.

[5] M. Bojarski, D. Del Testa, D. Dworakowski, B. Firner, B. Flepp, P. Goyal, L. D. Jackel, M. Monfort, U. Muller, J. Zhang, et al. End to end learning for self-driving cars. arXiv preprint arXiv:1604.07316, 2016.

[6] T. Chen, G. Zhang, X. Hu and J. Xiao, "Unmanned aerial vehicle route planning method based on a star algorithm," 2018 13th IEEE Conference on Industrial Electronics and Applications (ICIEA), Wuhan, 2018, pp. 1510-1514. doi: 10.1109/ICIEA.2018.8397948

[7] Unmanned Ground Vehicle. Sep. 24, 2016. [Online]
Available:http://ijaems.com/upload_images/issue_files/2%20IJAEMS-SEP-2016-24-Unmanned%20Ground%20Vehicle.pdf [Accessed: Oct, 2018]

[8] "Unmanned underwater vehicle", Wikipedia. [Online]
Available: https://en.wikipedia.org/wiki/Unmanned_underwater_vehicle [Accessed: Oct, 2018]

[9] "Multiobjective Optimization of an Operational Amplifier by the Ant Colony Optimisation Algorithm", Electrical and Electronic Engineering, p-ISSN: 2162-9455 e-ISSN: 2162-8459, 2012; 2(4): 230-235, doi: 10.5923/j.eee.20120204.09

[10] "Initialization of Weights in Neural Networks", International Journal of Science & Engineering Development Research (www.ijsdr.org), ISSN:2455-2631, Vol.3, Issue 11, page no.73 - 79, November-2018, Available: http://www.ijsdr.org/papers/IJSDR1811013.pdf

[11] End to End Learning for Self-Driving Cars, arXiv:1604.07316 N. Bernini, M. Bertozzi, L. Castangia, M. Patander and M. Sabbatelli, "Real-time obstacle detection using stereo vision for autonomous ground vehicles: A survey," 17th International IEEE Conference on Intelligent Transportation Systems (ITSC), Qingdao, 2014, pp. 873-878. doi: 10.1109/ITSC.2014.6957799