DOI: 30.6251/ijearst.43.75.1453 https://ijearst.co.in/

ENGINEERING IN ADVANCED RESEARCH SCIENCE AND TECHNOLOGY

ISSN 2386-7876
Vol.01, Issue.03
March-2026
Pages: 345-355

NARMAX Self-Tuning Controller for Line-of-Sight-Based Waypoint Tracking for an Autonomous Underwater Vehicle

[1] **Pechetti Girish**, [2] **Bhogisetti Adithya**, [3] **Gundumenu Manasa**, [4] **Mamidikuduru Sravani Durga**, [5] **Chukkana Satya Venkata Padmaraju**

[1]M.Tech(Ph.d),Associate professor, Dept of E.C.E, BVCITS, Batlapalem, Amalapuram, AP

[2,3,4,5]B. Tech, Dept of E.C.E, BVCITS, Batlapalem, Amalapuram, AP

Abstract: Autonomous Underwater Vehicles (AUVs) require precise and robust control strategies to navigate complex and uncertain underwater environments. This work presents a NARMAX (Nonlinear Auto Regressive Moving Average with eXogenous inputs) self-tuning controller integrated with a Line-of-Sight (LOS) guidance scheme for accurate waypoint tracking. The NARMAX model effectively captures the nonlinear dynamics of the AUV, including hydrodynamic effects and environmental disturbances such as ocean currents. The proposed self-tuning controller continuously updates its parameters based on real-time system behavior, ensuring adaptability to changing conditions. The LOS-based guidance system generates smooth reference trajectories, reducing tracking errors and ensuring stable navigation between predefined waypoints. Simulation results demonstrate that the proposed approach achieves improved tracking accuracy, reduced steady-state error, and enhanced robustness compared to conventional PID and model-based controllers. This makes it highly suitable for applications such as underwater exploration, surveillance, and environmental monitoring.

Keywords: NARMAX Model, Self-Tuning Control, Autonomous Underwater Vehicle (AUV), Line-of-Sight (LOS) Guidance, Waypoint Tracking, Nonlinear System Identification, Adaptive Control, Underwater Navigation, Hydrodynamic Modeling, Disturbance Rejection, Trajectory Tracking, Real-Time Control, Marine Robotics, Intelligent Control Systems, Ocean Environment Monitoring

Introduction: Autonomous driving is the current research hotspot, which mainly includes trajectory planning and trajectory tracking [1]. Specifically, the trajectory planning of autonomous driving is to design a trajectory suitable for driving environments. Besides, the trajectory tracking is the part to execute the autonomous driving which ensures that the vehicle follows the planned trajectory. Given that the first principle of autonomous driving is safety, the trajectory planning and trajectory parts should be capable of bypassing the obstacles and moving within the road boundaries. In order to achieve user acceptance, we should have the autonomous vehicles which are not only safe and reliable but also comfortable in terms of user experience. However, the individual perception of comfort may vary considerably among various vehicle users. For example some users might prefer sporty driving with high accelerations, while others might prefer a soft style. Typically, a human driver's style is characterized by a large number of parameters representing acceleration profiles, distances to surrounding vehicles, speed during lane-changing, etc. The rapid development in autonomous driving technology has provided many conveniences to people's lives and reduced traffic accidents [1]. Among them, path planning is one of the core research areas of autonomous driving technology, aiming to plan a path for the vehicle to reach the destination safely and ensure the avoidance of obstacles [2]. Although autonomous driving technology has been commercialized in some scenarios, it still faces greater challenges in path planning. In particular, there are avoidance of high-density obstacles under restricted and narrow working conditions, precise kinematic constraint handling, and real-time requirements. Path planning methods can be classified according to different technical means, including graph search-based methods, optimization methods, sampling-based methods, and machine learning methods. Among them, Dijkstra's algorithm [3] and A-star algorithms [4,5], as the classical algorithms of a graph search, need to preform a path search on a discrete map to find the optimal path. Optimization methods utilize mathematical optimization techniques [6,7], such as linear programming and nonlinear programming [8], that are capable of handling complex dynamic environments and multi-objective constraints. Sampling-based methods such as RRT (Rapid Exploration Random Tree) and its variants [9,10] generate path candidates by random sampling and filter the optimal paths to be suitable for high-dimensional spaces and complex obstacle environments. Dmitri Dolgov first proposed the hybrid A-star method, Sebastian Thrun et al., at Stanford University in 2008 [11]. Hybrid A-star combines the A-star algorithm, which “takes obstacles into account without considering motion constraints”, with the Reeds–Shepp curve, which “takes obstacles into account without considering motion constraints”.

The hybrid A-star combines the A-star algorithm “considering obstacles without motion constraints” with the Reeds–Shepp curve [12] “considering motion constraints without obstacles”. Therefore, compared with other algorithms, the hybrid A-star is more suitable for trajectory planning at low speeds in restricted spaces. After years of development, hybrid A-star is improved to be applied in different working scenarios. Meng et al. [13] enhance the performance of hybrid A-star algorithms through a safety-enhanced design and an efficiency-enhanced design. The safety-enhanced design integrates the Voronoi field potential function in the path search phase to better consider the safety of the path. The efficiency-enhanced design proposes a multi-stage dynamic optimization strategy that divides the path planning into multiple stages and performs dynamic optimization at each stage. The problem that the output paths of the hybrid A-star algorithm often contain unnecessary steering maneuvers, and the paths are close to obstacles, is addressed. Tang et al. [14] propose a method that applies the concept of an artificial potential field to optimize the hybrid A-star algorithm. The generated paths not only satisfy the vehicle’s non-integrity constraints, but also smooth and maintain a comfortable distance from obstacles.

LITERATURE SURVEY: SLRs are bound to observe a review protocol. The protocol for this study is borrowed from PRISMA (Preferred Reporting Items for Systematic Reviews and Meta-Analyses). PRISMA is a reporting guideline for systematic reviews and meta-analyses. It provides a standardized, evidence-based minimum set of items for reporting in this type of study, aimed at improving the transparency, completeness, and accuracy of their reporting, making them more helpful to readers and easier to compare between studies [2]. Moreover PRISMA recommends following a checklist adopted and outlined under the SLR methodology. It comprises the following steps and subsections: employed database for the selection of primary studies, inclusion criteria, exclusion criteria to filter the primary studies search results, and quality assurance to identify and overcome any bias in the selection of primary studies. This SLR was not registered anywhere, and the protocol for this study has lot been reported as well, as is how it is practiced in health and social science research. However, the adopted methodology followed the recommended guidelines published under [1]. More- over, SLRs have just started to gain acceptance in the engineering disciplines. Many path-planning solutions studied in the literature combine heuristics and deterministic or stochastic methods. As these computing modes are fundamentally different, it is suggested that an algorithms computing nature should be assessed based on the nature of the dominating component of an algorithm. For example, the well-known particle swarm optimization is initially based on randomness and starts to converge at a specific point; however, it is regarded as a meta-heuristic approach because of the conditions that help the method to converge. Hence, it is graded as a meta-heuristic approach rather than stochastic alone. However, this may also be regarded as a hybrid algorithm

EXISTING METHOD:

To illustrate the main concept behind this paper, Fig. 2 illustrates a simple example in which a vehicle (shown in yellow) needs to travel from a source point (located in the coverage area of base station colour in black) to a destination point (located in the coverage area of base station colour in red). Suppose the vehicle uses the straight route that connects the source with the destination. In that case, there will be a region with poor quality signal (shaded in blue) due to strong co-channel interferences coming from other neighbouring base stations that are in Line-of-Sight (LoS); this will notably impact the minimum data rate of the vehicle during the whole trip.

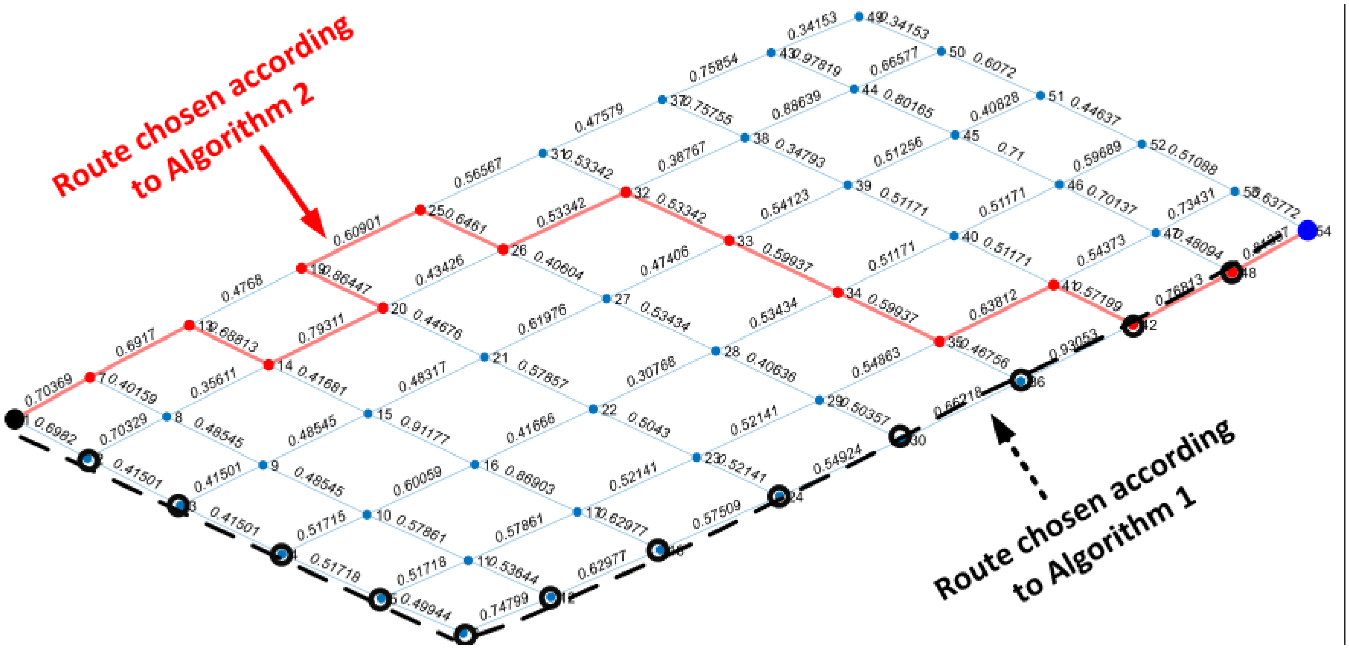


FIGURE 1. Graph-based representation of the mobile network coverage map in which the nodes and edges represent the crossroads and streets connecting crossroads, respectively.

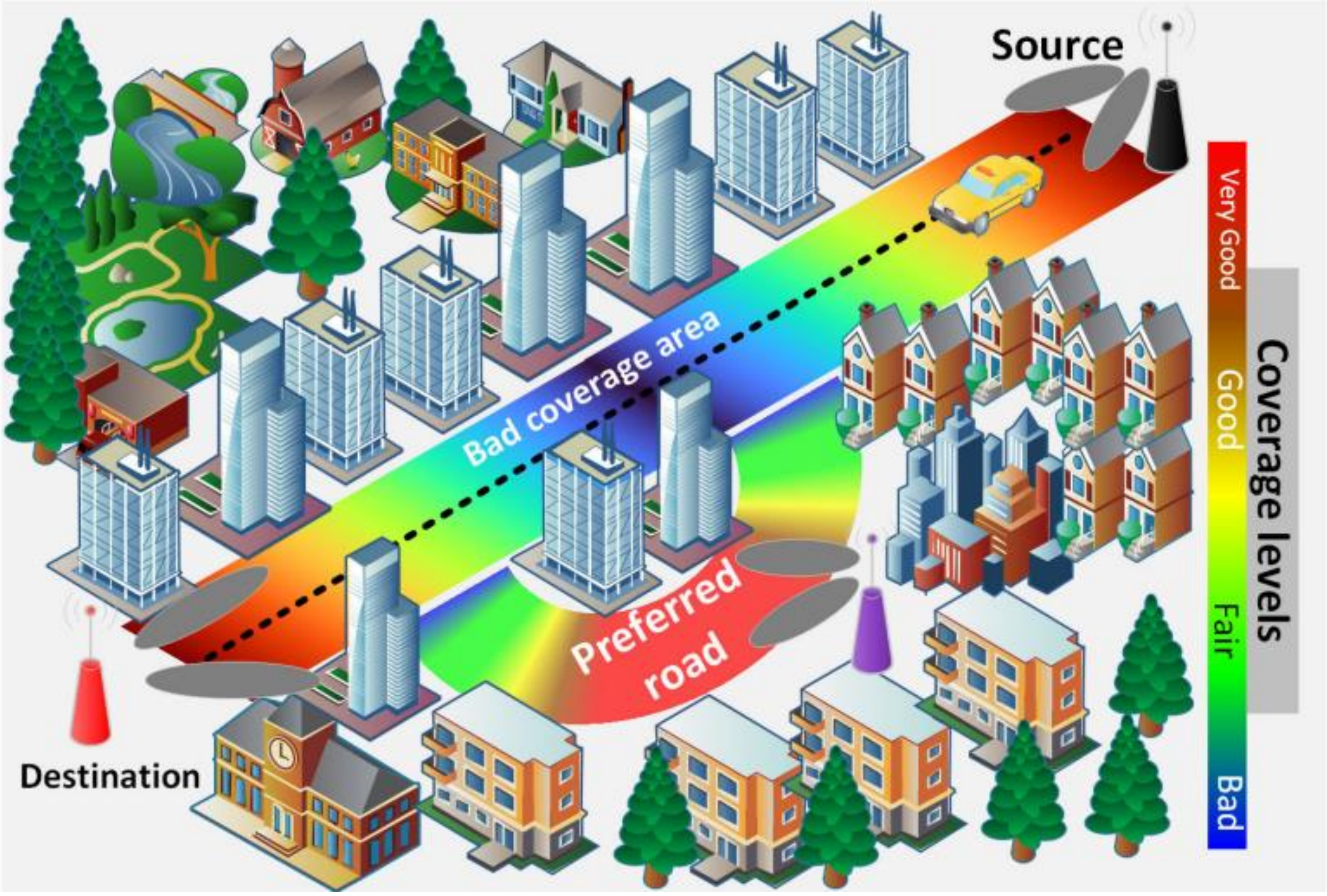


FIGURE 2. Illustration of a mobile network coverage in a densely populated urban area. The minimum data rate experienced by a vehicle can be notably improved by avoiding the streets in which the quality of the received signal is poor (bluish part of the road) due to strong interference and/or line-of-sight blockage.

However, suppose the vehicle knows about this condition in advance, it may plan the trajectory to take the diverted route (by taking a left turn toward the elliptical road, shaded in red) in which the quality of the received signal will be much better. It is noted that the reddish part of the route represents the parts of the road with good cellular coverage. This way the quality of the received signal during the entire trip could be much better, thus, ensuring improved QoS for mobile AVs. Without loss of generality, in this paper, we assume that the urban environment under analysis is part of the major streets of the Manhattan borough of the city of New York, which was modelled with the aid of a 3D map that was imported in Matlab from an open-source website named OpenStreetMap [25], which is maintained by a community of contributors who provide data on geographic locations worldwide. This location is chosen because the actual Manhattan has been historically used as a representative example of dense urban areas when modelling the propagation of radio signals. And the densely packed buildings, streets and crossroads in such urban settings significantly impact the quality of the received radio signal due to the multiple propagation mechanisms that simultaneously take place, such as reflection, scattering, diffraction and absorption. Therefore, by using Manhattan as a sample location, more precise information about the received signal strength and mobile network coverage that is expected in dense urban environments will be obtained. In addition, it will also enhance the understanding of how urban mobility patterns, like Manhattan grid-type layouts, influence network optimization and planning tasks to be performed by mobile operators. More precisely, this paper focuses on the area of Manhattan that is illustrated in Fig. 3, which spans from East River Walk to Central Park (from West-to-East) and from the Lenox Hill to Yorkville area (from North-to-South). To provide mobile network coverage in this service area, multiple base stations have been deployed by designating the positions for mmWave base stations (red labels) according to a predefined pattern. That is, base stations are strategically placed at every second crossroads in which the streets cross in Manhattan, under what is specified in 3GPP Technical Reports [26]. The network deployment that is shown in Fig. 3 was generated in run-time using the built-in Site Viewer function of Matlab. From the perspective of the square grid layout of the streets in Manhattan, each base station is equipped with four sectorized transmitting antennas placed at a height of 10 m, each of them pointed to serve a specific direction

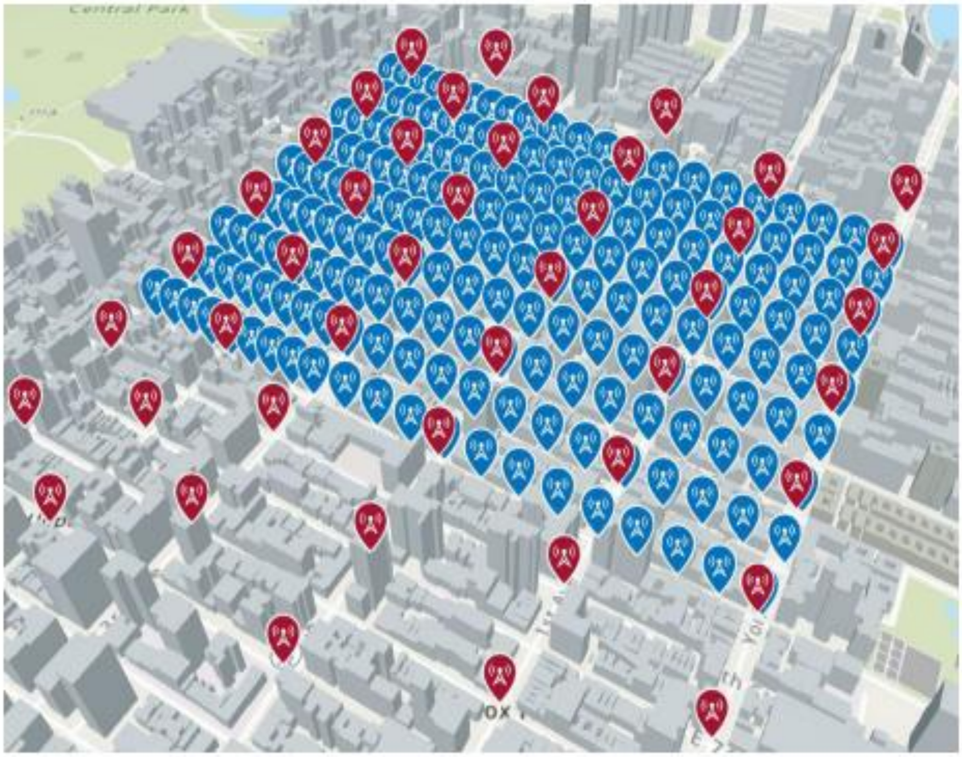


FIGURE 3. Positions of the mmWAve base stations (red labels) that were deployed to give service in the sample positions of the user terminals placed on the moving vehicles (blue labels).

for ensuring maximum mobile coverage. A custom antenna element pattern is also considered for horizontal and vertical radiation as specified in ITU-R reports [27]. That is, the horizontal radiation pattern of each directional antenna is modelled according to

$$A_{\mathrm{E\,H}}(\varphi) = -\min\left\{12 \times \left(\frac{\varphi}{\varphi_{3\,\mathrm{dB}}}\right)^2, \mathrm{SLA}\right\},$$

where −180◦ < ϕ ≤ 180◦ is the azimuth angle, ϕ3 dB is the horizontal half-power beamwidth, and SLA denotes the maximum side lobe level attenuation. Similarly, the vertical radiation pattern of the directive transmit antenna is modelled as

$$A_{\mathrm{E\,V}}(\theta) = -\min\left\{12 \times \left(\frac{\theta - \theta_{\mathrm{tilt}}}{\theta_{3\,\mathrm{dB}}}\right)^2, \mathrm{SLA}\right\},$$

where −90◦ < θ ≤ 90◦ is the elevation angle, θ3 dB is the vertical half-power beamwidth, and θtilt is the (down)tilt angle of the transmit antenna, which is usually pointing downwards to improve the quality of the received signal of user terminals placed at the ground level. Moreover, the method that is used to model the 3D antenna radiation pattern is obtained by combining the vertical radiation pattern in (4) with the horizontal radiation pattern in (3) to obtain

$$A(\theta, \varphi) = -\min\left\{-\left\{A_{\mathrm{E,V}}(\theta) + A_{\mathrm{E,H}}(\varphi)\right\}, \mathrm{SLA}\right\},$$

whereas A(θ , ϕ) denotes the relative antenna gain in dB of an antenna element in the direction of angles (θ , ϕ). To model the received signal strength in the possible trajectories that could be followed by an AV that travels from a source point to a destination point, a grid area of the urban environment has been chosen to establish the possible locations of the receiver during the trip. This grid consists of several vertical and horizontal streets that intersect, forming crossroads in which the vehicle may have the chance to change the trajectory. The mobile AVs can move around freely in the streets in any direction. Without loss of generality, the receiving antenna height is assumed 1.5 m. In this paper, we consider that communication between base stations and moving terminals in the AVs takes place on the 28 GHz mmWave, due to its potential to enable high data rates with its enormous bandwidth of the component carriers (i.e., up to 400 MHz). However, this also brings notable challenges to verifying the radio link budget between transmitter and receiver due to the strong path loss attenuation, weak reflections and reduced diffraction in edges in these highfrequency bands, making communication much more challenging without LoS conditions. This is especially serious in urban scenarios, where obstacles between transmitter and receiver are common. In pursuit of this objective, Ray Tracing (RT), a deterministic approach that employs the Shooting and Bouncing Rays (SBR) method to estimate precisely the effect that electromagnetic waves experience when propagating from transmitter to receiver, is used in this paper. Within the SBR methodology, multiple rays are radiated from the base station, which interacts with surfaces such as buildings and terrains, encountering physical phenomena like reflection, scattering and diffraction that affect the strength of the signal that reaches the antenna of the receiver. More precisely, this work employs a Matlab built-in RT method to compute the propagation statistics (e.g., free space loss, reflection loss, diffraction loss) between the transmitter and receiver points in a 3D outdoor environment [28]. The RT tool of Matlab enables a few configuration parameters to consider the number of reflections, and refraction, and account for the material of the reflector to determine the energy loss upon each reflection. It also assesses the strength of rays for continued propagation from transmitter to receiver, providing an option to set the limit for rays to be bending around the obstacles and clutters. RT enables the determination of received signal strength, essential for calculating the performance metrics in network planning and optimization tasks, such as estimating the signal-to-noise ratio and data rate. Moreover, for all the observed streets in the Manhattan grid scenario under analysis, a direct LoS link is assumed when the base station and mobile AVs are located on the same street. Otherwise, the urban infrastructure can generate (strong) reflections that may provide wireless connectivity between a base station and a mobile AV in case of a Non-Lineof-Sight (NLoS) condition; this usually takes place when both extremes of the wireless link are placed in different streets. Thus, the RT method helps to determine the situations in which communication services in NLoS can take place through reflection and diffraction of rays. In this paper, the SINR is determined via a built-in SINR object in Matlab. For SINR calculation, we assume that all base stations use the same transmission power; then, the base station that offers the strongest received signal strength at the given mobile location of the AV is the one that takes the role of serving base station (or cell). Let us assume that k is the index of the serving base station and i ∈ I is the index of the base stations that are generating co-channel interference to the user under analysis, in which I is the set with the indices of the interfering base stations. Then, the SINR experienced at the receiving locations at a given frequency can be written as

$$\mathrm{SINR} = \frac{P_k^{(\mathrm{rx})}}{P_N + \sum_{i=1}^{\mathcal{I}} P_i^{(\mathrm{rx})}},$$

where P (rx) k is the received power from the serving base station with index k, P (rx) i is the received power from the interfering base station with index i, and PN are the received power from base station i and the thermal noise power, respectively. Additionally, the thermal noise power is set to be −107 dBm by default in the whole communication bandwidth. For the sake of simplicity, the obtained SINR values for each receiver location are upper bounded to an achievable SE with the aid of the Shannon formula, i.e.,

$$\mathrm{SE} < \log_2(1 + \mathrm{SINR}),$$

in which the SINR should be input in a linear scale. Note that tighter upper bounds for the SE can be obtained by including a correction factor for the SINR and bandwidth.

Modified Version of Dijkstra’s Algorithm :

Modified versions of **Dijkstra’s algorithm** are used when the standard algorithm needs improvement or adaptation for specific problems. In this case, a modified version of Dijkstra’s algorithm is used to find the most convenient route from source S to destination D by maximizing the minimum edge weight on the route (i.e., the minimum SE when travelling through the segment, wSE(u, v)). It is noted that the edge with minimum weight in a route represents the bottleneck for the whole QoS of the route. Mathematically, this can be expressed as follows: given a weighted graph, G = (V, E), where V are the vertices/nodes and E the edges with non-negative weights wSE(u, v) ∈ E, identify the route R that maximizes the minimum weight in the segments that connect the source to the destination, i.e.,

$$\mathrm{W}_{\mathrm{SE}}(\mathrm{R}) = \max_{R \in \mathcal{R}(\mathrm{S},\mathrm{D})} \left(\min_{(u,v)\in R} w_{\mathrm{SE}}(u,v) \right)$$

where R(S, D) represents the set of all the possible routes connecting the source node S with the destination node D. To provide a comparative analysis between both algorithms, Fig. 1 shows that according to Algorithm 2 (solid red line), the most convenient route from source to destination is the one that maximizes the minimum edge weight along the route. In contrast, Algorithm 1 selects the path that minimizes the sum of weights, which is proportional to the distance from the source to the destination nodes, as shown by the black dashed line.

PROPOSED METHOD:

DEEP REINFORCEMENT LEARNING (DRL) – DEEP Q-NETWORK (DQN)

Reinforcement Learning (RL) is a machine learning paradigm where an **agent** learns to make decisions by interacting with an **environment** to maximize cumulative reward.

Basic Components:

- **Agent** – Learner/Decision maker
- **Environment** – Where agent operates
- **State (S)** – Current situation
- **Action (A)** – Decision taken
- **Reward (R)** – Feedback
- **Policy (π)** – Strategy of agent

Deep Reinforcement Learning combines:

- Reinforcement Learning
- Deep Neural Networks

In classical RL, Q-values are stored in tables.
In DRL, Q-values are approximated using **Deep Neural Networks**.

Q-Learning (Foundation of DQN)

Q-learning updates values using:

$$Q(s,a) = Q(s,a) + \alpha[r + \gamma \max\ Q(s',a') - Q(s,a)]$$

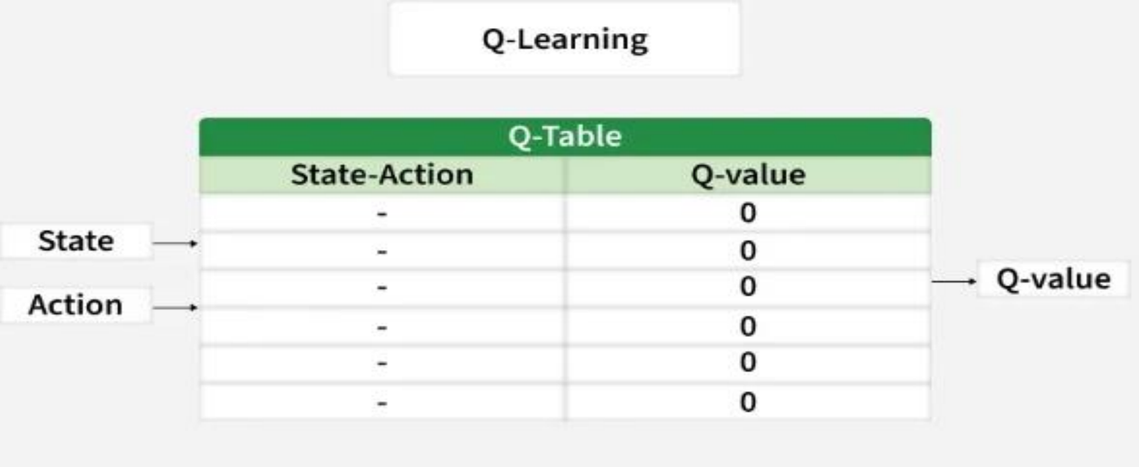
Where:

- α = Learning rate
- γ = Discount factor
- r = Reward
- s' = Next state

Deep Q-Learning in Reinforcement Learning

Deep Q-Learning is a method that uses deep learning to help machines make decisions in complicated situations. It’s especially useful in environments where the number of possible situations called states is very large like in video games or robotics.

Before understanding Deep Q-Learning it’s important to understand the main concept of Q-Learning. It is a model-free method that learns an optimal policy by estimating the Q-value function which tells how good it is to take a certain action in a certain situation. The goal is to find a plan that gives the highest total reward over time.



Q-Table	
State-Action	Q-value
-	0
-	0
-	0
-	0
-	0
-	0

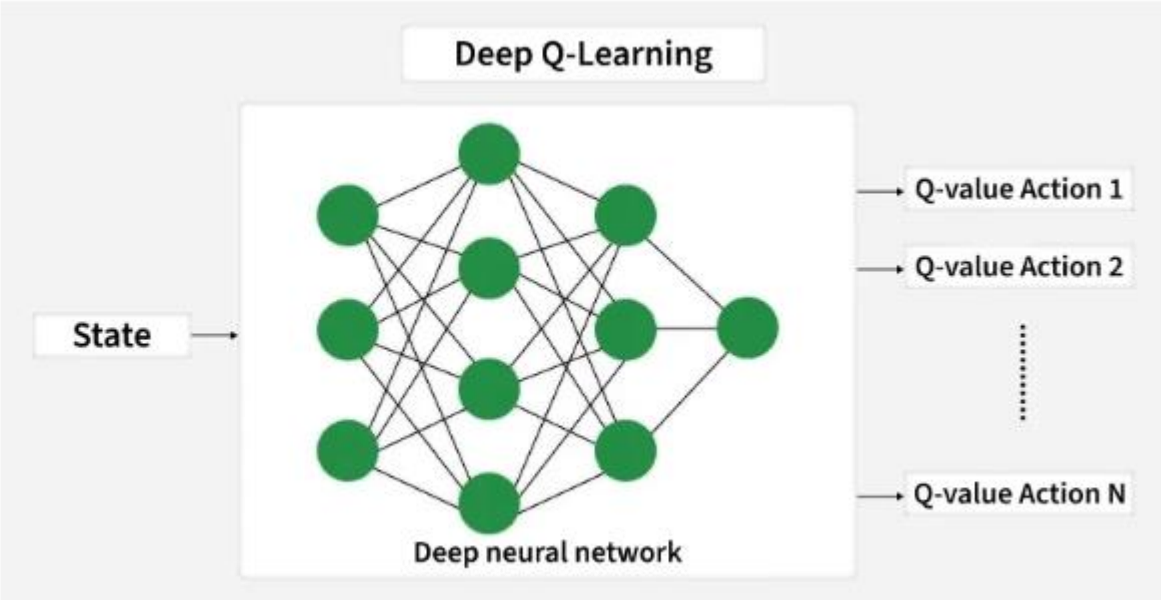
Q-Learning works well for small problems but struggles with complex ones like images or many possible situations. Deep Q-Learning solves this by using a neural network to estimate values instead of a big table.

Key Challenges Addressed by Deep Q-Learning

- **High-Dimensional State Spaces:** Traditional Q-Learning uses a table to store values but this becomes impossible when there are too many situations. Neural networks can understand and work with many different situations at once so they are better for complex problems.
- **Continuous Input Data:** Real-world problems often have continuous data like video images. Neural networks are good at handling this kind of information.
- **Scalability:** Deep learning helps Q-Learning grow and handle bigger, harder tasks that regular Q-Learning couldn't solve before.

Architecture of Deep Q-Networks

A DQN consists of the following components:



Deep Q-Learning

1. Neural Network

- The network approximates the Q-value function $Q(s, a; \theta)$ where θ represents the trainable parameters.
- For example in Atari games the input might be raw pixels from the game screen and the output is a vector of Q-values corresponding to each possible action.

2. Experience Replay

- To stabilize training, DQNs store past experiences (s, a, r, s') in a replay buffer.
- During training, mini-batches of experiences are sampled randomly from the buffer, breaking the correlation between consecutive experiences and improving generalization.

3. Target Network

- A separate target network with parameters θ^- is used to compute the target Q-values during updates. The target network is periodically updated with the weights of the main network to ensure stability.

4. Loss Function :

- The loss function measures the difference between the predicted Q-values and the target Q-values:

$$L(\theta) = E[(r + \gamma \max_{a'} Q(s', a'; \theta^-) - Q(s, a; \theta))^2]$$

Training Process

The training process of a DQN involves the following steps:

1. Initialization :

- Initialize the replay buffer, main network (θ) and target network (θ^-).
- Set hyperparameters such as learning rate (α), discount factor (γ) and exploration rate (ϵ).

2. Exploration vs. Exploitation : Use an ϵ-greedy policy to balance exploration and exploitation:

- With probability ϵ, select a random action to explore.
- Otherwise, choose the action with the highest Q-value according to the current network.

3. Experience Collection : Interact with the environment, collect experiences (s، a، r، s') and store them in the replay buffer.

4. Training Updates :

- Sample a mini-batch of experiences from the replay buffer.
- Compute the target Q-values using the target network.
- Update the main network by minimizing the loss function using gradient descent.

5. Target Network Update: Periodically copy the weights of the main network to the target network to ensure stability.

6. Decay Exploration Rate: Gradually decrease ϵ over time to shift from exploration to exploitation.

RESULTS:

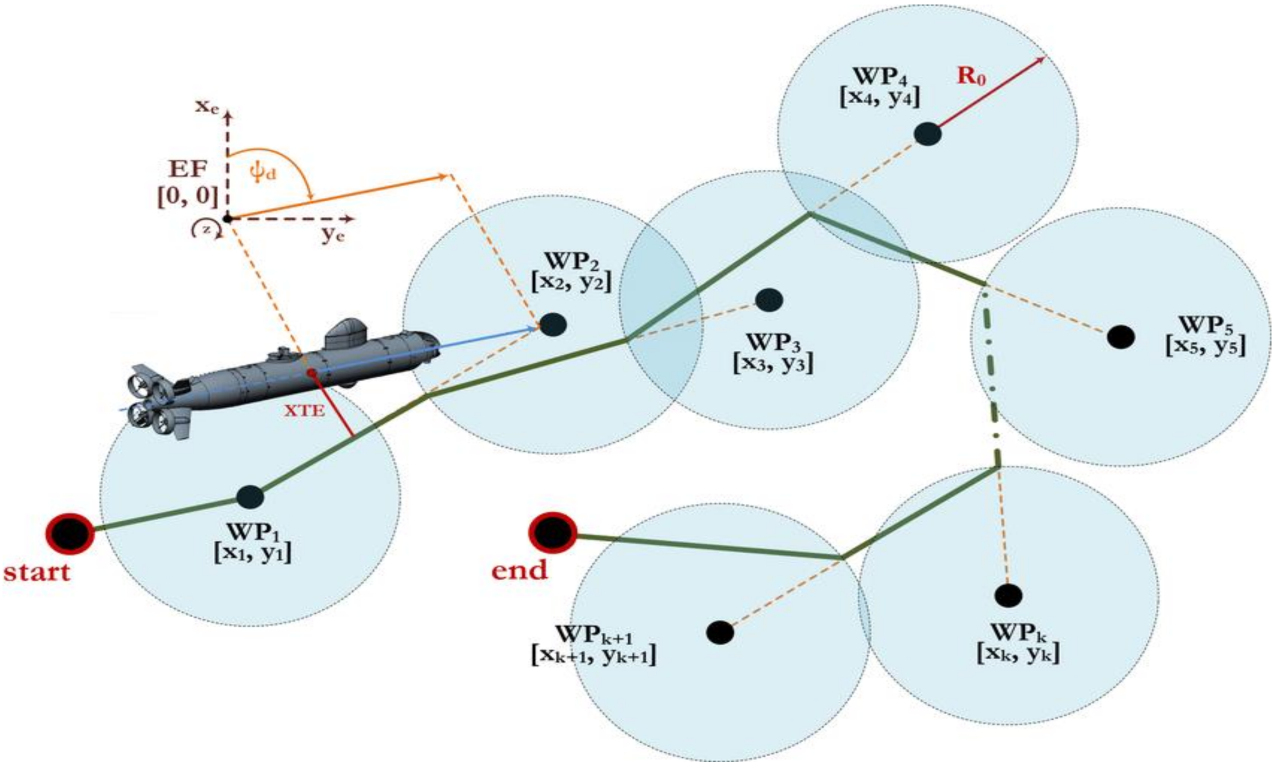


Fig:a Simulation result

ADVANTAGES:

- **Guaranteed Quality of Service (QoS):** By selecting paths that maximize signal strength (e.g., in mmWave bands), AVs avoid areas with poor coverage or high latency, ensuring consistent performance.
- **Enhanced Connectivity and Throughput:** Proactive trajectory planning ensures that data-hungry, high-bandwidth applications (like 4K/8K, and high-resolution sensor data) are supported by, enabling seamless multimedia and immersive experiences.
- **Optimized Resource Allocation:** It reduces data loss and improves communication efficiency, which is vital for real-time traffic management and autonomous decision-making.
- **Improved Safety and Reliability:** By reducing communication failures, it supports safer, more reliable autonomous driving up to Level 5.
- **Energy Efficiency:** Optimized routing can lead to more energy-efficient operation for both the vehicles and the 6G network, reducing environmental impact.

APPLICATIONS:

- **Proactive Network-Aware Routing:** AVs use 6G-enabled predictive modeling to select routes that maintain high signal quality, avoiding cellular dead zones or congested network cells to ensure uninterrupted, real-time communication for navigation and safety.
- **UAV-IRS Assisted Connectivity:** Unmanned Aerial Vehicles (UAVs) acting as Intelligent Reflecting Surfaces (IRS) can dynamically adjust their trajectories to optimize the communication link between base stations and mobile AVs, providing enhanced coverage in challenging environments.
- **AI-Enabled Dynamic Path Planning:** Utilizing machine learning, AVs forecast future traffic, obstacles, and network conditions to make real-time, instantaneous adjustments to their trajectory, enhancing safety and reducing latency.

Conclusion:

This study demonstrates that Deep Reinforcement Learning (DQN) can effectively guide autonomous vehicles to plan trajectories that maintain target QoS in 6G networks. The integration of network-awareness into trajectory planning enables AVs to adapt to dynamic communication conditions, reducing latency, improving throughput, and enhancing overall reliability. Simulation results validate that the DRL-based approach outperforms conventional trajectory planning methods that ignore network constraints. The proposed framework paves the way for intelligent, network-conscious autonomous mobility, offering a foundation for future research in multi-agent coordination, edge-assisted trajectory optimization, and real-time 6G-enabled vehicular networks.

Future scope:

Future trajectory planning for autonomous vehicles (AVs) in 6G networks will shift from solely avoiding physical obstacles to optimizing paths based on real-time network Quality of Service (QoS), such as spectral efficiency and latency, especially over mmWave/THz bands. This enables enhanced V2X, seamless high-bandwidth services, and proactive, AI-driven routing to ensure 99.99999% reliability.

REFERENCES:

[1] P. R. Singh, V. K. Singh, R. Yadav, and S. N. Chaurasia, “6G networks for artificial intelligence-enabled smart cities applications: A scoping review,” Telematics Informat. Rep., vol. 9, Mar. 2023, Art. no. 100044. [2] R. Rabnawaz, M. K. A. M. K. Abid, D. Naeem Aslam, and F. Bukhari, “Exploring 6G wireless communication: Application technologies, challenges and future direction,” Int. J. Inf. Syst. Comput. Technol., vol. 2, no. 2, pp. 26–43, Jul. 2023. [3] H. Viswanathan and P. E. Mogensen, “Communications in the 6G era,” IEEE Access, vol. 8, pp. 57063–57074, 2020. [4] M. H. C. Garcia, A. Molina-Galan, M. Boban, J. Gozalvez, B. Coll-Perales, T. Sahin, and A. Kousaridas, “A tutorial on 5G NR V2X communications,” IEEE Commun. Surveys Tuts., vol. 23, no. 3, pp. 1972–2026, 3rd Quart., 2021. [5] Study on Enhancement of 3GPP Support for 5G V2X Services, 3GPP, Dec. 2018. [6] M. Boban, A. Kousaridas, K. Manolakis, J. Eichinger, and W. Xu, “Connected roads of the future: Use cases, requirements, and design considerations for vehicle-to-everything communications,” IEEE Veh. Technol. Mag., vol. 13, no. 3, pp. 110–123, Sep. 2018. [7] A. M. Al-Hazemi, A. A. Al-Mekhlafi, and R. Q. Shaddad, “Densification of 5G heterogeneous cellular wireless network based on millimeter-wave communication,” in Proc. 1st Int. Conf.

Intell. Comput. Eng. (ICOICE), Dec. 2019, pp. 1–6. [8] J. Moon, S. Kim, H. Ju, and B. Shim, "Energy-efficient user association in mmWave/THz ultra-dense network via multi-agent deep reinforcement learning," IEEE Trans. Green Commun. Netw., vol. 7, no. 2, pp. 692–706, Jun. 2023. [9] C. Wang, S.-H. Fang, H.-C. Wu, S.-M. Chiou, W.-H. Kuo, and P.-C. Lin, "Novel user-placement ushering mechanism to improve quality-of-service for femtocell networks," IEEE Syst. J., vol. 12, no. 2, pp. 1993–2004, Jun. 2018. [10] Y. Ullah, M. Roslee, S. M. Mitani, M. Sheraz, F. Ali, A. F. Osman, M. H. Jusoh, and C. Sudhamani, "Reinforcement learning-based unmanned aerial vehicle trajectory planning for ground users' mobility management in heterogeneous networks," J. King Saud Univ.-Comput. Inf. Sci., vol. 36, no. 5, Jun. 2024, Art. no. 102052. [11] T. Tanagi and K. Adachi, "Dynamic path planning for QoS improvement in multiple automated guided vehicles," in Proc. VTS Asia–Pacific Wireless Commun. Symp. (APWCS), Aug. 2023, pp. 1–5. [12] E. Zeydan, J. Mangues-Bafalluy, O. Dedeoglu, and Y. Turk, "Performance comparison of QoS deployment strategies for cellular network services," IEEE Access, vol. 8, pp. 176073–176088, 2020. [13] N. P. Kuruvatti, S. B. Mallikarjun, S. C. Kusumapani, and H. D. Schotten, "Proactive allocation of radio resources to enhance service continuity in cellular networks," in Proc. Int. Symp. Electr. Electron. Eng. (ISEE), Apr. 2021, pp. 100–105. [14] M. Beshley, N. Kryvinska, and H. Beshley, "Energy-efficient QoE-driven radio resource management method for 5G and beyond networks," IEEE Access, vol. 10, pp. 131691–131710, 2022. [15] C.-H. Wang, C.-J. Lee, and X. Wu, "A coverage-based location approach and performance evaluation for the deployment of 5G base stations," IEEE Access, vol. 8, pp. 123320–123333, 2020. [16] A. Ghosh and I. S. Misra, "Ultra dense three -tier 5G heterogeneous network model for improved coverage and rate," in Proc. IEEE Calcutta Conf. (CALCON), Dec. 2022, pp. 60–64. [17] R. V. Akhpashev, V. G. Drozdova, and T. M. Leonenko, "The software development for 5G new radio coverage planning," in Proc. Int. Sci.- Tech. Conf. Actual Problems Electron. Instrum. Eng. (APEIE), Nov. 2021, pp. 256–259. [18] A. Mpatziakas, A. Sinanis, I. Hamlatzis, A. Drosou, and D. Tzovaras, "AIbased mechanism for the predictive resource allocation of V2X related network services," in Proc. 18th Int. Conf. Netw. Service Manage. (CNSM), Oct. 2022, pp. 282–288.

[19] M. Raeisi and A. B. Sesay, "Power control of 5G-connected vehicular network using PPO-based deep reinforcement learning algorithm," IEEE Access, vol. 12, pp. 96387–96403, 2024.

[20] W. Yao, R. P. Cabrera, C. Lung, and S. A. Ajila, "Pre-connect handover management for 5G networks," in Proc. IEEE Future Netw. World Forum, Oct. 2022, pp. 556–561.

[21] B. Li, Z. Na, R. Qian, and H. Ding, "QoS and fuzzy logic based communication handover strategy in 5G cellular Internet of Vehicles," in Proc. 26th Int. Conf. Comput. Supported Cooperat. Work Design (CSCWD), May 2023, pp. 1458–1463.

[22] H.-W. Kao, E. H. Wu, and C.-H. Lan, "5G handover analysis in real network," in Proc. IEEE 3rd Int. Conf. Electron. Commun., Internet Things Big Data (ICEIB), Apr. 2023, pp. 7–11.

[23] S. Mathew, E. Oliosi, L. Davoli, G. Ferrari, N. Strozzi, and A. Notari, "On the reduction of unnecessary handovers using 5G small cells in a 5G NR environment," in Proc. Int. Conf. Confluence Advancements Robot., Vis. Interdiscipl. Technol. Manage. (IC-RVITM), Nov. 2023, pp. 1–6.