

Article

Intelligent Multi-Objective Path Planning for Unmanned Surface Vehicles via Deep and Fuzzy Reinforcement Learning

Ioannis A. Bartsiokas ^{1,*} , Charis Ntakolia ² , George Avdikos ^{3,*} and Dimitris Lyridis ³ ¹ School of Electrical and Computer Engineering, National Technical University of Athens, Zografos, 15780 Athens, Greece² Department of Aeronautical Studies, Sector of Materials Engineering, Machining Technology and Production Management, Hellenic Air Force Academy, Dekeleia Base, 13672 Acharnes, Greece; charis.nt@gmail.com³ Laboratory for Maritime Transport, School of Naval Architecture and Marine Engineering, National Technical University of Athens, 15773 Athens, Greece; dsvlr@mail.ntua.gr

* Correspondence: giannismpartsioakas@mail.ntua.gr (I.A.B.); gavdikos@mail.ntua.gr (G.A.)

Abstract

Unmanned Surface Vehicles (USVs) are increasingly employed in maritime operations requiring high levels of autonomy, safety, and energy efficiency. However, traditional path planning techniques struggle to simultaneously address multiple conflicting objectives such as fuel consumption, trajectory smoothness, and obstacle avoidance in dynamic maritime environments. To overcome these limitations, this paper introduces a Deep Q-Learning (DQN) framework and a novel Fuzzy Deep Q-Learning (F-DQN) algorithm that integrates Mamdani-type fuzzy reasoning into the reinforcement-learning (RL) reward model. The key contribution of the proposed approach lies in combining fuzzy inference with deep reinforcement learning (DRL) to achieve adaptive, interpretable, and multi-objective USV navigation—overcoming the fixed-weight reward limitations of existing DRL methods. The study develops a multi-objective reward formulation that jointly considers path deviation, curvature smoothness, and fuel consumption, and evaluates both algorithms in a simulation environment with varying obstacle densities. The results demonstrate that the proposed F-DQN model significantly improves trajectory optimality, convergence stability, and energy efficiency, achieving over 35% reduction in path length and approximately 70–80% lower fuel consumption compared with the baseline DQN, while maintaining comparable success rates. Overall, the findings highlight the effectiveness of fuzzy-augmented reinforcement learning in enabling efficient and interpretable autonomous maritime navigation.

Keywords: Unmanned Surface Vehicle (USV); path planning; Deep Q-Learning (DQN); fuzzy reinforcement learning; multi-objective optimization; fuel efficiency; maritime autonomy; autonomous navigation



Academic Editors: Hossein Nejatbakhsh Esfahani, Arash Bahari Kordabad and David Moreno-Salinas

Received: 3 November 2025

Revised: 24 November 2025

Accepted: 28 November 2025

Published: 30 November 2025

Citation: Bartsiokas, I.A.; Ntakolia, C.; Avdikos, G.; Lyridis, D. Intelligent Multi-Objective Path Planning for Unmanned Surface Vehicles via Deep and Fuzzy Reinforcement Learning. *J. Mar. Sci. Eng.* **2025**, *13*, 2285. <https://doi.org/10.3390/jmse13122285>

Copyright: © 2025 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

The field of autonomous robotics has witnessed significant advancements in recent years, largely due to the development of artificial intelligence (AI) methodologies that enable real-time applications. A key focus of research has been the creation of fully autonomous vehicles capable of interpreting their environment and interacting with it independently of human control. Among these autonomous vehicles, Unmanned Ground Vehicles (UGVs) and Unmanned Aircraft Vehicles (UAVs) have seen widespread use in applications ranging from the industrial and agricultural sectors to military and disaster recovery operations. However, a more recent addition to this domain is the Unmanned

Surface Vehicle (USV), which has been attracting growing attention for its potential in ocean sampling, monitoring, and rescue operations. Despite not being as widely implemented as UGVs and UAVs, USVs are becoming an essential tool for applications that require navigating complex, dynamic environments, especially those involving ocean currents and other marine-specific challenges.

The autonomy of USVs hinges on their ability to make real-time decisions, including finding optimal routes and avoiding obstacles while navigating unpredictable environmental conditions. To address these challenges, several path planning algorithms have been developed, many of which focus on optimizing specific objectives such as distance, energy efficiency, or safety. Traditional methods like the A* algorithm [1,2], Dijkstra's algorithm [3], and Ant Colony Optimization (ACO) [4,5] have been widely employed in maritime path planning [6–9]. Moreover, the Voronoi diagram [10] and Particle Swarm Optimization algorithm [11] have also been widely used in path planning applications in the case of USVs. For instance, the Ant Colony Optimization (ACO) algorithm has been refined by dynamically adjusting the pheromone volatility coefficient, thereby improving its search capabilities [12]. Similarly, the Genetic Algorithm (GA) has been augmented through multidomain inversion techniques, which increase offspring diversity and subsequently accelerate convergence rates [13], and in conjunction with other heuristic algorithms for task allocation and obstacle avoidance, particularly in scenarios involving swarms of USVs [14]. Additionally, hybrid strategies have been developed, such as combining the A* algorithm with smoothing processes to ensure safer routes [2] or integrating it with the Artificial Potential Field (APF) algorithm for obstacle avoidance [1,15,16]. Other techniques, including the rapidly-exploring random tree (RRT) algorithm, have been adapted with dynamic step sizes and target-attractive forces to optimize pathfinding in complex environments [17]. Moreover, Particle Swarm Optimization (PSO) has been enhanced through orientation-angle-based grouping to improve convergence speed [18,19]. Other notable methods include the use of B-splines to facilitate collision avoidance, where an initial shortest path is iteratively adjusted to bypass obstacles [20]. However, these approaches are often limited by their focus on single objectives and their computational demands, especially when applied to complex, large-scale environments.

While path planning for Unmanned Surface Vehicles (USVs) has garnered considerable attention, most of the existing literature tends to focus on single-objective optimization approaches, such as minimizing travel distance, time, or energy consumption. These methods, while effective for specific tasks, fail to address the complexity of real-world scenarios where multiple conflicting objectives must be considered simultaneously. Multi-objective path planning, which is crucial for tasks involving dynamic environments and various operational constraints, has received far less attention. Few studies have tackled multi-objective optimization in USV path planning, with notable efforts including dynamic particle swarm optimization algorithms that address the effects of ocean currents [21] and A* or ACO algorithms for optimal route planning under dynamic obstacles and ocean currents [8,22]. Approaches incorporating fuzzy logic or Pareto optimality have also been explored, but these still face challenges in practical applications. For instance, the weighted sum method (WSM), often used in multi-objective optimization, requires careful parameter selection and may struggle to find global optimal solutions, particularly in non-linear environments [23]. Pareto solutions offer multiple trade-offs, requiring human intervention to select the most suitable option, thus limiting their applicability in real-time decision-making [24,25]. Additionally, integrating fuzzy inference systems into heuristic algorithms introduces high computational costs, further complicating their use in time-sensitive applications [26]. These limitations underscore the need for more innovative and computationally efficient approaches in the domain of multi-objective USV path planning.

Indeed, the development of multi-objective path planning techniques has emerged as a promising solution to these limitations, allowing USVs to balance competing criteria such as travel time, energy consumption, and safety, as demonstrated in prior studies on USV and autonomous vehicle navigation [8,21–25]. These works highlight the importance of simultaneously optimizing path deviation, smoothness, and fuel-related metrics in dynamic maritime environments. Reinforcement learning (RL) has emerged as a promising approach for optimizing path planning in USVs, enabling them to autonomously learn optimal navigation strategies through trial and error. By continuously interacting with their environment, USVs can adapt to dynamic obstacles, ocean currents, and other environmental factors in real-time. This method allows USVs to improve their decision-making process over time, ultimately enhancing their ability to navigate complex, unpredictable maritime environments without human intervention. Furthermore, the integration of fuzzy logic into optimization algorithms is gaining traction, offering a way to automate decision-making without the need for human intervention in real-time applications. This approach enhances the ability of USVs to find optimal solutions in dynamic and challenging environments, a critical feature for their successful deployment in a variety of fields, from oceanography to military operations.

Considering the above, this paper proposes and compares two deep reinforcement learning frameworks for intelligent, multi-objective path planning of USVs. The first framework employs a Deep Q-Learning (DQN) algorithm, serving as the baseline for autonomous navigation in dynamic maritime environments. The second framework introduces an enhanced Fuzzy Deep Q-Learning (F-DQN) approach that integrates fuzzy inference mechanisms into the DQN architecture to handle multiple conflicting objectives—such as path deviation, trajectory smoothness, and fuel efficiency—through adaptive reward shaping. By combining the learning capability of DQN with the human-like reasoning of fuzzy logic, the proposed F-DQN improves interpretability, decision stability, and convergence speed.

The key contributions of this paper are summarized as follows:

- A baseline Deep Q-Learning (DQN) framework is developed for USV path planning, forming a foundation for evaluating advanced fuzzy-enhanced techniques.
- A Fuzzy Deep Q-Learning (F-DQN) algorithm is proposed, combining reinforcement learning with fuzzy logic to achieve multi-objective optimization in terms of smoothness, energy efficiency, and safety.
- A multi-objective reward model is designed, integrating path deviation (PC), environmental fuel consumption (EE), and velocity deviation-based energy cost (ED), subject to an overall fuel constraint.
- A simulation environment is implemented, allowing both DQN and F-DQN agents to be tested across multiple scenarios to assess adaptability and robustness.
- Comparative results demonstrate that the proposed F-DQN significantly outperforms the baseline DQN in terms of trajectory smoothness, convergence stability, and total fuel consumption, validating its effectiveness for autonomous USV navigation.

The rest of this paper is organized as follows: Section 2 presents relevant literature on the field of DRL-based path planning in UGV and USV scenarios. Section 3 depicts the system model and problem formulation, including the mathematical representations of the performance metrics. Section 4 describes the design of the DQN and F-DQN frameworks and the fuzzy inference mechanism. Section 5 discusses the simulation setup, evaluation metrics, and comparative results. Section 6 provides concluding remarks and future research directions.

2. Related Works

Path planning is a core challenge in autonomous vehicle navigation, encompassing UAVs, UGVs, and USVs. Over the past decade, DRL has emerged as a powerful framework for enabling agents to make sequential navigation decisions in complex and dynamic environments. Unlike classical deterministic algorithms such as A*, D* Lite, and potential-field methods [11–18], DRL-based approaches allow vehicles to learn optimal motion strategies directly from interactions with the environment, making them suitable for partially known or stochastic settings [27–34]. Early applications of DRL in UAVs focused on obstacle avoidance and trajectory optimization under energy or communication constraints, while UGV-oriented studies explored path planning on uneven terrains and cooperative navigation. More recently, USV-focused research has adopted DRL to address dynamic maritime challenges, including moving obstacles, wind and current disturbances, and limited maneuverability [27–34]. However, most existing studies prioritize either safety or energy efficiency, rarely addressing these objectives jointly through a multi-objective optimization framework, as depicted also in Table 1.

Table 1. DRL-aided UxV path planning related works.

Work	Vehicle Domain	Method	Objectives	Environment	Key Findings
[27]	USV (single & formations)	Deep RL for cooperative path planning (DQN)	Collision avoidance, path length, formation maintenance, kinematic constraints	Simulation + real harbor data	DRL can compute shortest collision-avoiding paths and maintain formations; applicability to USV formation control and obstacle-dense areas.
[28]	USV	Dynamic Window Approach + Safe RL	Obstacle avoidance, safety, path feasibility	Simulation	Combines local DWA motion generation with safe-RL policy for higher-level steering; improved safety in marine simulations.
[29]	USV	RL	Coverage completeness, path length, energy proxy	Simulation	Presents RL-based complete coverage planner for USVs; focuses on mission coverage metrics
[30]	USV	Cross-domain DRL + NN-MPC hybrid	Dynamic obstacle avoidance, tracking accuracy, collision avoidance	Simulated dynamic obstacles	Hybrid NN-MPC + DRL improves dynamic obstacle avoidance and tracking vs baseline controllers.
[31]	Multi-USV	Multi-agent RL	Cooperative navigation, target capture, multi-agent coordination	Simulation	Presents MARL strategies for coordinated capture/navigation; emphasizes cooperation and role assignment among USVs.
[32]	USV	Hybrid DRL	Obstacle avoidance, safety, maybe hybrid methods	Simulation	Effective scheduling and path planning under energy constraints
[33]	Multi-USV cluster	Value-decomposition network	Task allocation + collision avoidance	Simulated multi-USV ocean observation tasks	Decomposes planning into allocation and avoidance; improved TD error and transfer learning accelerate convergence and safety.

Table 1. Cont.

Work	Vehicle Domain	Method	Objectives	Environment	Key Findings
[34]	USV	Hybrid-priority Twin-Delayed DDPG	Trajectory tracking, robustness to currents and winds, convergence speed	Realistic marine data with currents and winds	Uses LOS-guided MDP and hybrid-priority replay to enhance robustness and accuracy under time-varying disturbances.
This paper	USV	Deep Q-Learning (DQN) and Fuzzy DQN (F-DQN)	Multi-objective path planning under fuel various constraints	Simulation	F-DQN achieves smoother paths, lower fuel use, and faster convergence than baseline DQN; interpretable and energy-aware navigation framework.

In this context, a growing body of research explores approaches where DRL is augmented with auxiliary optimization or reasoning modules to improve convergence and interpretability. The existing works largely focus on UAV or ground robotics applications, with few studies extending fuzzy DRL principles to hydrodynamic or marine environments characterized by currents, drag, and nonlinear dynamics. These works are depicted in Table 1.

Most existing approaches focus on single or limited objectives, such as collision avoidance, path smoothness, or cooperative task execution. For instance, Zhou et al. [27] and Qu et al. [28] employed DQN and Safe-RL variants to achieve collision-free path planning and improved safety performance, while other works [29,30] extended DRL to mission coverage and dynamic obstacle avoidance through hybrid neural and model-predictive control frameworks. Multi-agent RL formulations [31,33] have further enabled cooperative behaviors among fleets of USVs, addressing task allocation and formation maintenance. In addition, recent advances such as hybrid-priority agents [34] and hierarchical DRL architectures have enhanced convergence and robustness under environmental uncertainty. Collectively, these studies confirm that DRL can produce effective policies for complex maritime navigation scenarios while maintaining adaptability to dynamic conditions.

Nevertheless, most of the above works are designed around single-criterion optimization or heuristic reward shaping, without explicitly addressing the multi-objective trade-offs that arise in real-world USV operations. In particular, factors such as fuel consumption, path deviation, and trajectory smoothness are interdependent yet rarely optimized jointly. Furthermore, the majority of DRL-based USV planners rely solely on neural approximations, which often obscure interpretability and limit policy transparency, which is especially critical for safety-certified maritime systems. To overcome these limitations, the present work introduces a Fuzzy Deep Q-Learning (F-DQN) framework that embeds fuzzy reasoning within the DRL reward function. This design allows the agent to balance competing objectives in a continuous and interpretable manner, providing both explainable decision-making and enhanced navigation efficiency. By comparing the proposed F-DQN against a baseline DQN across multiple obstacle scenarios, this study directly addresses the gap between single-objective DRL path planning and the need for energy-aware, multi-objective, and interpretable USV navigation solutions.

3. System Model and Problem Formulation

The path planning of a USV can be formulated as a multi-objective optimization problem, where the vehicle must autonomously determine a feasible route from a start point to a goal point while minimizing navigation cost functions and respecting environmental and physical constraints. The optimization criteria adopted in this work include the

traveled distance, path deviation, and fuel consumption, all of which jointly influence the overall navigation efficiency and energy performance of the vehicle. The corresponding mathematical definitions are presented below.

3.1. Overall System Topology

The operational topology considered in this work, which is depicted in Figure 1, represents a two-dimensional maritime environment in which a USV must navigate autonomously from a starting point S to a destination point G . The area of operation is defined as a bounded workspace containing both navigable regions and obstacle zones, which correspond to static obstacles. The objective of the USV is to determine an optimal collision-free trajectory that minimizes distance, path deviation, and fuel consumption while respecting its dynamic and energy constraints.

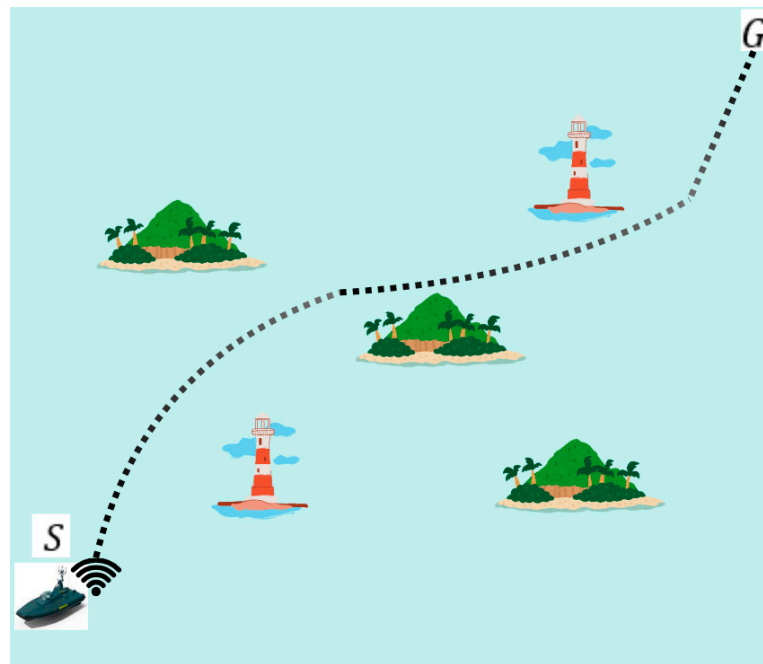


Figure 1. Overall topology of the USV navigation environment, illustrating the start and goal points and obstacle regions.

To model the environment, the workspace is discretized into a finite set of nodes $N = \{n_1, n_2, \dots, n_K\}$ representing candidate waypoints and a corresponding set of edges $E \subseteq N \times N$ that connect navigable node pairs. Each edge $(m, n) \in E$ is characterized by its Euclidean distance d_{mn} , the direction of motion, and the local environmental conditions, such as current velocity vectors $\vec{v}_{(m,n)}^c$. Obstacles are defined as forbidden regions that remove corresponding nodes or edges from the feasible navigation graph, thus enforcing collision avoidance constraints. The USV motion between nodes is assumed to follow kinematic feasibility conditions based on its velocity range and turning radius. The number and placement of obstacles can be varied to represent scenarios of increasing complexity (e.g., 4, 6, and 8 obstacles) for comparative evaluation.

This topological abstraction allows for a general mathematical formulation of the path-planning problem, in which the vehicle's trajectory corresponds to an ordered sequence of nodes connecting S and G . The subsequent subsections define the cost functions and constraints governing this multi-objective optimization problem.

3.2. Problem Formulation

The multi-objective USV path-planning problem addressed in this study aims to minimize three primary cost components: the traveled distance, the path deviations, and the fuel consumption. The overall formulation is expressed through a weighted combination of these objectives, with constraints imposed by the USV's fuel capacity and operational range. To this end, we adopt the mathematical problem of minimizing the following objective terms that can also be found in similar studies [8,9,21,22,35–37].

3.2.1. Traveled Distance

Given the set N of nodes and the set of edges $E \subseteq N \times N$, the total path length can be computed as the sum of Euclidean distances between all consecutive nodes:

$$D = \sum_{n \in N} \sum_{\substack{m \in N \\ (n,m) \in E}} d_{nm} = \sum_{n \in N} \sum_{\substack{m \in N \\ (n,m) \in E}} \sqrt{(m_x - n_x)^2 + (m_y - n_y)^2} \quad (1)$$

where d_{nm} denotes the distance between nodes n and m . Minimizing D ensures geometric efficiency and reduced travel time.

3.2.2. Path Deviations

Trajectory smoothness is quantified through angular deviations between consecutive path segments. Let ϕ_{lmn} denote the turning angle formed by three consecutive nodes l , m , and n .

The cumulative deviation across the trajectory is expressed as:

$$PC = \sum_{l \in N} \sum_{\substack{m \in N \\ (l,m) \in E}} \sum_{\substack{n \in N \\ (m,n) \in E}} \phi_{lmn} \quad (2)$$

Lower values of PC indicate smoother paths, reducing mechanical stress and energy losses associated with frequent heading adjustments.

3.2.3. Fuel Consumption

Fuel consumption is modeled as the sum of two main components:

- (i) fuel consumption due to environmental conditions, and
- (ii) fuel consumption due to velocity deviations between consecutive path segments.

Fuel consumption due to environmental conditions

Given the nominal fuel rate $F(\text{kg/h})$, the USV's velocity vector $\vec{V}_{(m,n)}$ along each edge, and the environmental current vector $\vec{v}_{(m,n)}^c$, the fuel consumed to overcome hydrodynamic effects is:

$$EE = \sum_{m \in N} \sum_{\substack{n \in N \\ (m,n) \in E}} \frac{d_{mn}}{|\vec{V}_{(m,n)} + \vec{v}_{(m,n)}^c|} F \quad (3)$$

Fuel consumption due to velocity deviations

For a USV with mass M , the energy cost of velocity changes between successive path segments is expressed as:

$$ED = \sum_{m \in N} \sum_{\substack{n \in N \\ (m,n) \in E}} \frac{1}{2} M |\vec{V}_{(j,k)} - \vec{V}_{(i,j)}| \quad (4)$$

The total fuel consumption is then given by:

$$EC = EE + ED \quad (5)$$

subject to the operational fuel constraint:

$$EC \leq F_{\max} \quad (6)$$

where $F_{\max}$ denotes the maximum fuel capacity of the USV.

3.2.4. Overall Optimization Objective

Combining the above components, the USV path-planning problem can be expressed as a multi-objective optimization task:

$$\min_{\pi} J(\pi) = w_1 D + w_2 PC + w_3 EC \quad (7)$$

where π represents the control policy or trajectory plan, and w_1, w_2, w_3 are weighting coefficients that balance the importance of distance, path smoothness, and energy efficiency.

In subsequent sections, this multi-objective framework serves as the foundation for developing the DQN and F-DQN algorithms, which aim to adaptively minimize $J(\pi)$ under varying environmental conditions and obstacle densities.

4. Deep Q-Learning and Fuzzy Deep Q-Learning Frameworks

This section presents the design of the proposed learning frameworks for multi-objective USV path planning: the baseline DQN algorithm and the enhanced F-DQN. Both methods are developed and trained to operate under identical environmental conditions defined in Section 3.

4.1. Reinforcement-Learning Formulation

The USV navigation process is modeled as a Markov Decision Process (MDP) represented by the tuple (S, A, R, P, γ) , where S denotes the set of observable states, A the set of discrete control actions, R the reward function, $P(s' | s, a)$ the transition probabilities, and $\gamma \in [0, 1)$ the discount factor.

At each time step t , the USV observes its current state $s_t \in S$, executes an action $a_t \in A$, receives a scalar reward r_t , and transitions to a new state s_{t+1} . The optimal policy $\pi^*(a | s)$ maximizes the expected cumulative discounted reward:

$$Q^*(s, a) = \mathbb{E} \left[\sum_{t=0}^{\infty} \gamma^t r_t \mid s_0 = s, a_0 = a, \pi^* \right]. \quad (8)$$

4.2. State and Action Representations

The state vector represents the navigation status and environmental context of the USV:

$$s_t = [x_t, y_t, \theta_t, d_{goal}, d_{obs}, v_t, v_t^c], \quad (9)$$

where x_t, y_t are the current coordinates, θ_t is the heading angle, d_{goal} is the distance to the target, d_{obs} is the distance to the closest obstacle, v_t is the USV's velocity, and v_t^c is the magnitude of the local current vector.

The action space consists of discrete steering and velocity commands:

$$A = \{\Delta\theta, \Delta v\} = \{-\Delta\theta, 0, +\Delta\theta\} \times \{-\Delta v, 0, +\Delta v\}, \quad (10)$$

allowing the USV to adjust its heading and speed within defined limits, consistent with its kinematic feasibility.

4.3. Reward Function Design

The reward function encodes the multi-objective optimization defined in Section 3 and penalizes undesirable behavior while encouraging efficient, smooth, and fuel-conscious motion. At each time step, the incremental reward is expressed as:

$$r_t = -(w_1 \Delta D_t + w_2 \Delta PC_t + w_3 \Delta EC_t), \quad (11)$$

where ΔD_t , ΔPC_t , and ΔEC_t denote the incremental variations of traveled distance, path deviation, and energy consumption, respectively.

Terminal conditions apply as follows:

$$r_t = \begin{cases} +R_{goal}, & \text{if } s_t \in G, \\ -R_{collision}, & \text{if a collision occurs,} \\ -R_{fuel}, & \text{if } EC > F_{max}. \end{cases} \quad (12)$$

In the DQN baseline, the weights w_1 , w_2 , and w_3 remain constant throughout training. In contrast, in the F-DQN algorithm, these weights are dynamically adapted using a fuzzy inference mechanism, enabling the agent to emphasize different objectives depending on the navigation context.

The dimensioning of the fuzzy linguistic variables followed standard fuzzy control design principles. Each of the three fuzzy inputs—trajectory deviation, curvature smoothness, and energy-related cost—were partitioned into three linguistic terms ('Low', 'Medium', and 'High') to balance interpretability with sufficient granularity for reward shaping. Preliminary tuning experiments showed that increasing the number of fuzzy sets did not yield additional performance benefits and resulted in unnecessary rule-based complexity. Similarly, the fuzzy output variable (shaping reward) was dimensioned using three output linguistic terms ('Small', 'Moderate', 'Large'), which provided smooth transitions between reward levels without introducing instability in the Q-learning update. This configuration offered the best trade-off between expressiveness and computational efficiency.

4.4. Baseline Deep Q-Learning (DQN)

The DQN model employs a feedforward neural network to approximate the optimal action-value function $Q(s, a; \theta)$, parameterized by the network weights θ . The training objective is to minimize the mean-squared Bellman error:

$$L(\theta) = \mathbb{E}_{(s,a,r,s') \sim \mathcal{D}} [(r + \gamma \max_{a'} Q(s', a'; \theta^-) - Q(s, a; \theta))^2], \quad (13)$$

where θ^- are the parameters of a target network updated periodically and $\mathcal{D}$ denotes the experience replay buffer. The DQN is trained using stochastic gradient descent with an Adam optimizer, learning rate 10^{-3} , and discount factor $\gamma = 0.95$. Exploration follows an ϵ -greedy policy, where ϵ decays linearly from 1.0 to 0.05 over training episodes.

This DQN implementation serves as the baseline for performance comparison with the fuzzy-enhanced algorithm.

4.5. Fuzzy Deep Q-Learning (F-DQN)

To address the limitations of fixed-weight reward shaping, the F-DQN algorithm introduces fuzzy logic into the reward evaluation process. Fuzzy inference allows the

agent to reason over uncertain or conflicting objectives and to assess the quality of each state-action pair in a gradual, interpretable manner.

Fuzzy logic enhances the DQN in two complementary ways:

- Action evaluation: the FIS assesses the feasibility or quality of possible actions based on fuzzy state variables, rather than discrete thresholds
- Reward evaluation: the FIS dynamically computes the composite reward by considering multiple uncertain or interdependent factors (distance, path deviation, and energy consumption).

This adaptive mechanism allows the agent to focus on distance minimization in open waters, smoothness near obstacles, and energy conservation under adverse currents.

The Fuzzy Inference System FIS is implemented using the Mamdani inference methodology [38], commonly employed in multi-objective navigation and decision support systems [8,9,26,39,40]. Its advantages can be summarized as: (i) expressive power; (ii) easy formalization and interpretability; (iii) reasonable results with relatively simple structure; (iv) suitable and widely used for decision support applications due to the intuitive and interpretable nature of the rule base; (v) can be used for Multiple Input Single Output and Multiple Input Multiple Output systems; and (vi) the output value can be either crisp or fuzzy [41–43].

The fuzzy universes of discourse are defined as:

- U_{y1} : traveled distance
- U_{y2} : path curvature
- U_{y3} : energy consumption,
- U_q : overall path quality.

Each crisp input is represented by overlapping fuzzy sets:

$$\tilde{y}_{1,v} = \{ \langle x, \mu_{y_{1,v}}(x) \rangle \mid x \in U_{y_1} \}, v = \{\text{short, moderate, long}\}, \quad (14)$$

$$\tilde{y}_{2,v} = \{ \langle x, \mu_{y_{2,v}}(x) \rangle \mid x \in U_{y_2} \}, v = \{\text{smooth, adequate, brut}\}, \quad (15)$$

$$\tilde{y}_{3,v} = \{ \langle x, \mu_{y_{3,v}}(x) \rangle \mid x \in U_{y_3} \}, v = \{\text{low, medium, high}\}, \quad (16)$$

$$\tilde{q}_v = \{ \langle x, \mu_{q_v}(x) \rangle \mid x \in U_q \}, v = \{\text{very low, low, medium, high, very high}\}. \quad (17)$$

Figure 2 depicts the membership functions corresponding to each universe (normalized distance, path deviation, fuel consumption, and path optimality).

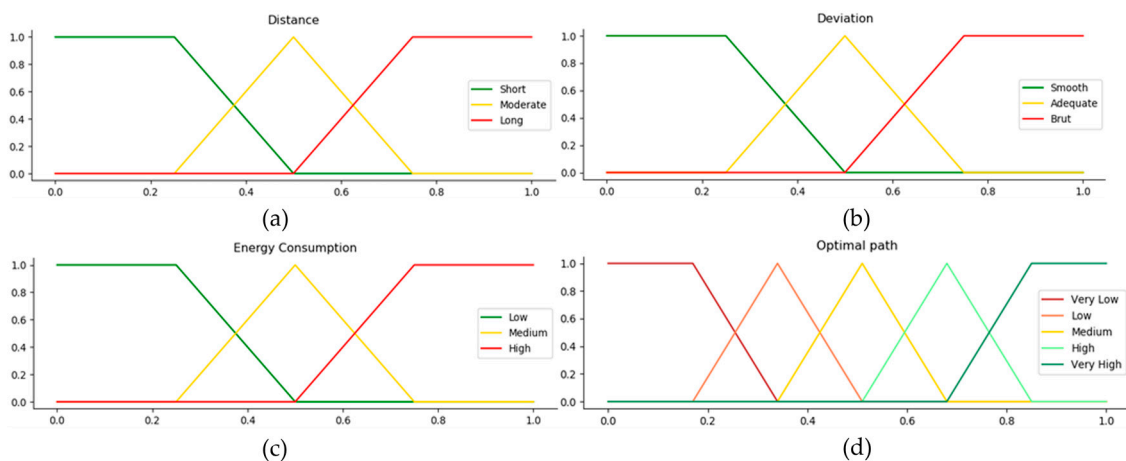


Figure 2. Membership functions of (a) normalized distance, (b) deviations among the path, (c) fuel consumption, and (d) path optimality.

The fuzzy rule base (Table 2) consists of 27 rules that model the relationship among distance, smoothness, and energy consumption in determining overall path quality.

Table 2. Fuzzy rules for the fuzzy evaluation process of DQN.

Fuzzy Rules	Distance	Smoothness	Energy Consumption	Path Quality
Rule 1	Short	Smooth	Low	Very High
Rule 2	Short	Smooth	Medium	Very High
Rule 3	Short	Adequate	Low	Very High
Rule 4	Moderate	Smooth	Low	Very High
Rule 5	Short	Smooth	High	High
Rule 6	Short	Adequate	Medium	High
Rule 7	Short	Brut	Low	High
Rule 8	Moderate	Smooth	Medium	High
Rule 9	Moderate	Adequate	Low	High
Rule 10	Long	Smooth	Low	High
Rule 11	Short	Adequate	High	Medium
Rule 12	Short	Brut	Medium	Medium
Rule 13	Short	Brut	High	Medium
Rule 14	Moderate	Smooth	High	Medium
Rule 15	Moderate	Adequate	Medium	Medium
Rule 16	Moderate	Brut	Low	Medium
Rule 17	Long	Smooth	Medium	Medium
Rule 18	Long	Smooth	High	Medium
Rule 19	Long	Adequate	Low	Medium
Rule 20	Long	Brut	Low	Medium
Rule 21	Moderate	Adequate	High	Low
Rule 22	Moderate	Brut	Medium	Low
Rule 23	Moderate	Brut	High	Low
Rule 24	Long	Adequate	Medium	Low
Rule 25	Long	Adequate	High	Low
Rule 26	Long	Brut	Medium	Low
Rule 27	Long	Brut	High	Very Low

Each rule follows the general format:

$$\text{IF } (y_1 \text{ is } A_i) \text{ AND } (y_2 \text{ is } B_i) \text{ AND } (y_3 \text{ is } C_i) \text{ THEN } (q \text{ is } D_i),$$

where A_i, B_i, C_i, D_i correspond to the linguistic variables defined in the universes above. The full set of 27 rules—covering combinations such as Short–Smooth–Low $\rightarrow$ Very High and Long–Brut–High $\rightarrow$ Very Low—is shown in Table 2 and governs the FIS reasoning process.

At each time step, the FIS evaluates the normalized inputs (y_1, y_2, y_3) to produce a fuzzy output $q_v \in [0, 1]$, representing the instantaneous path-quality index. The resulting index adaptively modifies the reward weights according to:

$$[w_1, w_2, w_3] = f_{\text{FIS}}(y_1, y_2, y_3), \quad (18)$$

and the shaped reward $r_t^{(F)}$ replaces r_t in the Bellman update of the DQN. This hybridization allows F-DQN to maintain the exploration and generalization power of deep reinforcement learning while benefiting from the interpretability and contextual awareness of fuzzy logic.

4.6. Neural-Network Configuration

Both DQN and F-DQN employ identical neural architectures to ensure fair comparison, as depicted in Figure 3. Both employ a feed-forward deep neural network (DNN) to

approximate the action–value function $Q(s, a; \theta)$, where s denotes the observed state and a denotes the discrete action. The architecture is identical for both algorithms to ensure a fair comparison, and it consists of the following layers:

- Input layer: Receives the environment’s state vector, which includes the USV’s current position, goal coordinates, velocity, and relative distances to the nearest obstacles.
- Fully connected layer 1 (128 neurons): Applies a linear transformation followed by a ReLU activation function.
- Fully connected layer 2 (128 neurons): Also activated by ReLU; reduces dimensionality while capturing higher-order correlations between motion and obstacle data. Serves as a transitional latent representation linking the abstracted features to the action space.
- Output layer: Produces one Q-value per available action (e.g., turn-left, turn-right, accelerate, decelerate, or maintain speed).
- For F-DQN: The same network is used, but the output Q-values are refined by the Fuzzy Inference System (FIS), which adjusts them based on the fuzzy evaluation of distance, path smoothness, and energy consumption.

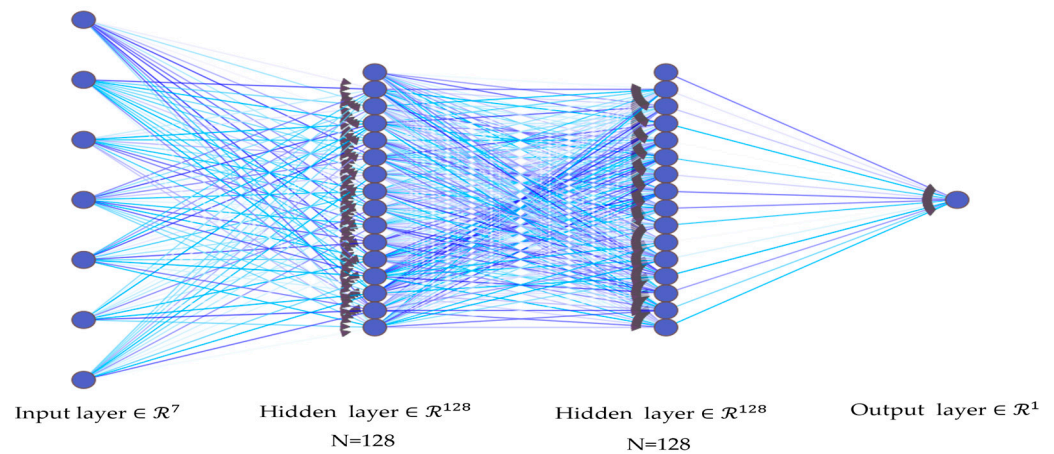


Figure 3. DNN structure both for DQN and FDQN.

The state representation used by the Q-network is seven-dimensional and is given by

$$s_t = [x_t, y_t, x_{goal}, y_{goal}, v_t, d_{min}, \Delta\theta] \quad (19)$$

where (x_t, y_t) denote the current coordinates of the USV, (x_{goal}, y_{goal}) represent the target waypoint, v_t is the instantaneous vessel velocity, d_{min} corresponds to the minimum distance to the nearest obstacle, and $\Delta\theta$ is the heading-angle deviation between the current orientation and the desired direction of travel. Accordingly, the input layer of the deep neural network contains seven neurons, each corresponding to one component of the state vector extracted from the simulation environment.

The F-DQN incorporates an external fuzzy module that modifies the reward signal before backpropagation; thus, the Q-network itself remains unchanged.

Training is conducted using mini-batch updates from an experience replay buffer of 10^5 samples, with target network synchronization every 100 steps and an ϵ -greedy exploration strategy.

In each decision cycle of the proposed framework, the processing order is as follows. The environment first produces the current state vector, which is passed to the Q-network to generate a set of action-value estimates $Q(s_t, a)$. These values represent the expected long-term reward associated with each discrete manoeuvre (e.g., turning, accelerating, or maintaining speed) and are used to select the agent’s next action under the ϵ -greedy explo-

ration policy. After the action is executed, the fuzzy inference system evaluates trajectory deviation, path smoothness, and instantaneous energy-related metrics and computes a shaping reward term. This fuzzy-derived reward is then combined with the environment reward to form the final reinforcement signal used during the Q-learning update. Thus, the neural network and the fuzzy block do not operate simultaneously; the Q-network determines the action, while the fuzzy module refines the reward before backpropagation.

The fuzzy inference block operates in parallel with the environment reward and outputs a shaping reward term r_f , computed from trajectory deviation, curvature smoothness, and instantaneous fuel-related cost. The total reward used during training is defined as

$$r_t = r_e + r_f \quad (20)$$

where r_e is the environment-generated reinforcement signal. This fused reward formulation stabilizes the Q-learning update and enables multi-objective optimization by guiding the agent toward solutions that are simultaneously energy-efficient, smooth, and collision-free. For completeness, Figure 4 illustrates the systemic flow of information between the state extraction module, the fuzzy inference system, and the Q-learning agent.

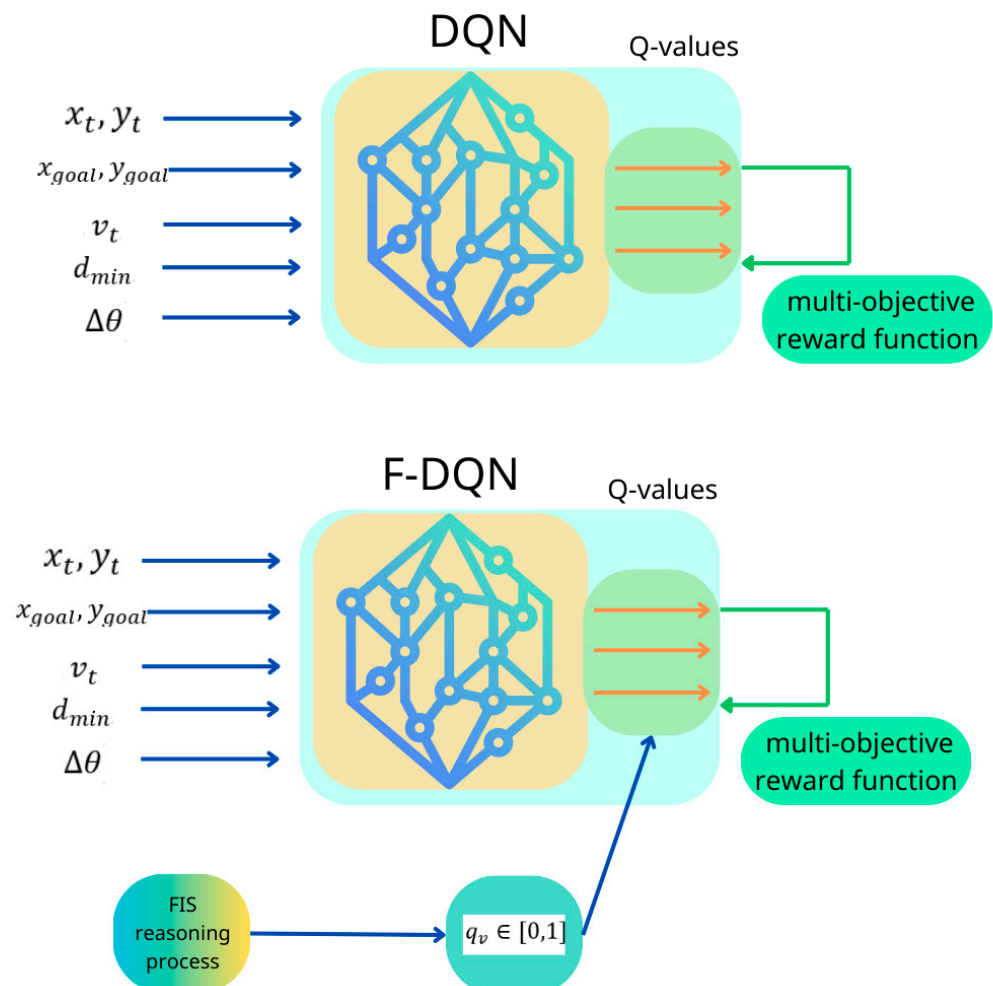


Figure 4. Proposed DQN and F-DQN frameworks.

5. Performance Evaluation

5.1. Simulation Environment

The simulation experiments were conducted in MATLAB R2025b using the Reinforcement Learning Toolbox. The navigation environment emulated a two-dimensional

maritime operational area of $100\text{ m} \times 100\text{ m}$, containing circular obstacles of varying diameters randomly distributed throughout the workspace.

Each obstacle was characterized by its center coordinates and radius, forming a static map configuration that changed across different experimental scenarios. The USV operated under kinematic constraints reflecting realistic surface mobility, defined by maximum linear velocity, limited turning rate, and fuel availability.

To comprehensively assess algorithm performance, three increasingly complex obstacle configurations were defined:

- Scenario 1: four obstacles (low-density environment),
- Scenario 2: six obstacles (medium-density environment),
- Scenario 3: eight obstacles (high-density environment).

In all cases, the USV began at a fixed start point and aimed to reach the fixed goal point. The vehicle's instantaneous velocity V was sampled uniformly from the interval $[1.5, 2.5]\text{ m/s}$ to simulate environmental variability. The nominal fuel consumption rate was fixed at 2 kg/h , while the total fuel capacity constraint was set to 6 kg .

For the training process, approximately $10,510^{5105}$ state–transition pairs were collected from multiple simulated navigation episodes covering all obstacle density scenarios. These transitions were stored in an experience replay buffer and partitioned into an 80% training subset and a 20% validation subset. The validation portion was used solely to monitor learning stability and detect potential overfitting during training. Mini-batch sampling, target network synchronization, and ϵ -greedy exploration were applied consistently across all experiments. All hyperparameters used in the learning algorithms are summarized in Table 3.

Table 3. Simulation setup and hyperparameters.

Parameter	Symbol	Value/Range	Description
USV velocity	V	$[1.5, 2.5]\text{ m/s}$	Randomized cruising speed
Fuel consumption rate	F	2 kg/h	Nominal energy usage
Fuel capacity	F_{max}	6 kg	Maximum fuel reserve
Number of obstacles	-	$[4, 6, 8]$	Scenario-dependent
Learning rate	α	0.001	Adam optimizer
Discount factor	γ	0.99	Future reward discount
Replay buffer	-	10^5 experiences	Experience memory size
Mini-batch size	-	64	Samples per training step
Exploration rate	ϵ	$0.9 \geq \epsilon \geq 0.05$	Linear decay schedule
Episodes per scenario	-	500	Training episodes per environment

5.2. Evaluation Metrics

To evaluate the comparative performance of DQN and Fuzzy-DQN, four multi-objective performance indicators were adopted, aligned with the mathematical formulation of Section 3:

- Path Length (D): Total distance traveled by the USV between start and goal points.
- Path Smoothness (PC): Cumulative angular deviation between consecutive trajectory segments (Equation (2)), representing the curvature of the path.
- Energy Consumption (EC): Combined metric incorporating environmental and velocity-dependent components (Equations (3)–(5)).
- Success Rate (SR): Ratio of episodes where the USV successfully reached the goal without collisions.

All metrics were normalized for visualization in comparative plots.

5.3. Comparative Results

To quantitatively assess the effectiveness of the proposed Fuzzy DQN framework against the baseline DQN agent, both models were evaluated across three scenarios with increasing environmental complexity, comprising four, six, and eight obstacles. Each experiment consisted of 500 episodes, and the results were averaged over successful runs. Table 4 depicts the overall performance of both algorithms in all three scenarios under test. Moreover, Figure 5 visualizes the normalized behavior of the proposed approaches (DQN and F-DQN) in all scenarios, and Figure 6 depicts a direct comparison of the two methods in the best joint episode.

Table 4. Performance evaluation of all scenarios.

Scenario (# obst.)	Algorithm	Success Rate (%)	Mean Path Length (m)	Mean Smoothness (rad)	Energy Proxy
4	DQN	88.0	214.10	0.199	30.246
	F-DQN	81.3	141.65	0.065	6.637
6	DQN	82.7	208.52	0.198	29.499
	F-DQN	81.7	135.02	0.062	5.957
8	DQN	84.7	203.21	0.192	27.614
	F-DQN	82	130.75	0.064	5.794

The obtained results confirm the efficiency of the F-DQN agent in generating shorter and smoother trajectories, while maintaining comparable success rates to the baseline DQN, even as the number of obstacles increases. Specifically, the F-DQN achieved a reduction of more than 35% in path length and an approximate 70–80% decrease in energy consumption across all tested scenarios. This significant improvement stems from the fuzzy reward-shaping mechanism, which enables more nuanced evaluation of competing objectives—namely, distance minimization, trajectory smoothness, and energy efficiency.

While the baseline DQN demonstrated slightly higher goal-reaching success (within 2–6%), its longer trajectories and higher energy expenditure reveal its less efficient motion planning behavior. Conversely, the F-DQN agent achieves more stable and energy-aware decisions, emphasizing trajectory optimality over aggressive exploration. It is worth noting that the small difference in success rate between DQN and F-DQN reported in Table 4 is not statistically significant and falls within the normal stochastic fluctuation observed across repeated reinforcement-learning runs. Both models consistently converge to robust navigation policies, and extensive testing did not indicate any dependence of this variation on obstacle density, dataset size ($\approx 10^5$ transitions), or network architecture. Therefore, the F-DQN preserves the safety and reliability of the baseline DQN while delivering substantial improvements in energy efficiency, trajectory smoothness, and path optimality.

As environmental complexity increases (from four to eight obstacles), both agents maintain relatively stable success rates, indicating strong generalization capabilities. However, the relative advantage of the F-DQN in smoothness and energy metrics remains consistent, confirming that the integration of fuzzy logic improves multi-objective balance without compromising task completion reliability.

Figure 6 illustrates the trade-off relationship between the average path length and the mean smoothness achieved by the two agents across all tested scenarios. Each point corresponds to a specific obstacle configuration (4, 6, and 8 obstacles), annotated accordingly. As observed, the F-DQN agent consistently occupies the lower-left region of the plot, indicating shorter and smoother trajectories, whereas the baseline DQN solutions

are positioned toward the upper-right region, corresponding to longer and less stable paths. This demonstrates the effectiveness of the fuzzy reward mechanism in balancing competing objectives—distance minimization and trajectory smoothness—without explicit multi-objective optimization. The F-DQN’s ability to generalize across different levels of environmental complexity also reflects its capacity for context-sensitive decision-making, as the fuzzy inference system dynamically adjusts reward contributions from progress, energy use, and deviation. By contrast, the baseline DQN often overemphasizes goal convergence, leading to sharper turns and redundant maneuvers. Overall, the trade-off scatter confirms that the Fuzzy DQN yields more energy-efficient and kinematically feasible paths, which are particularly desirable for real-world autonomous surface vehicle navigation, where stability and efficiency are critical.

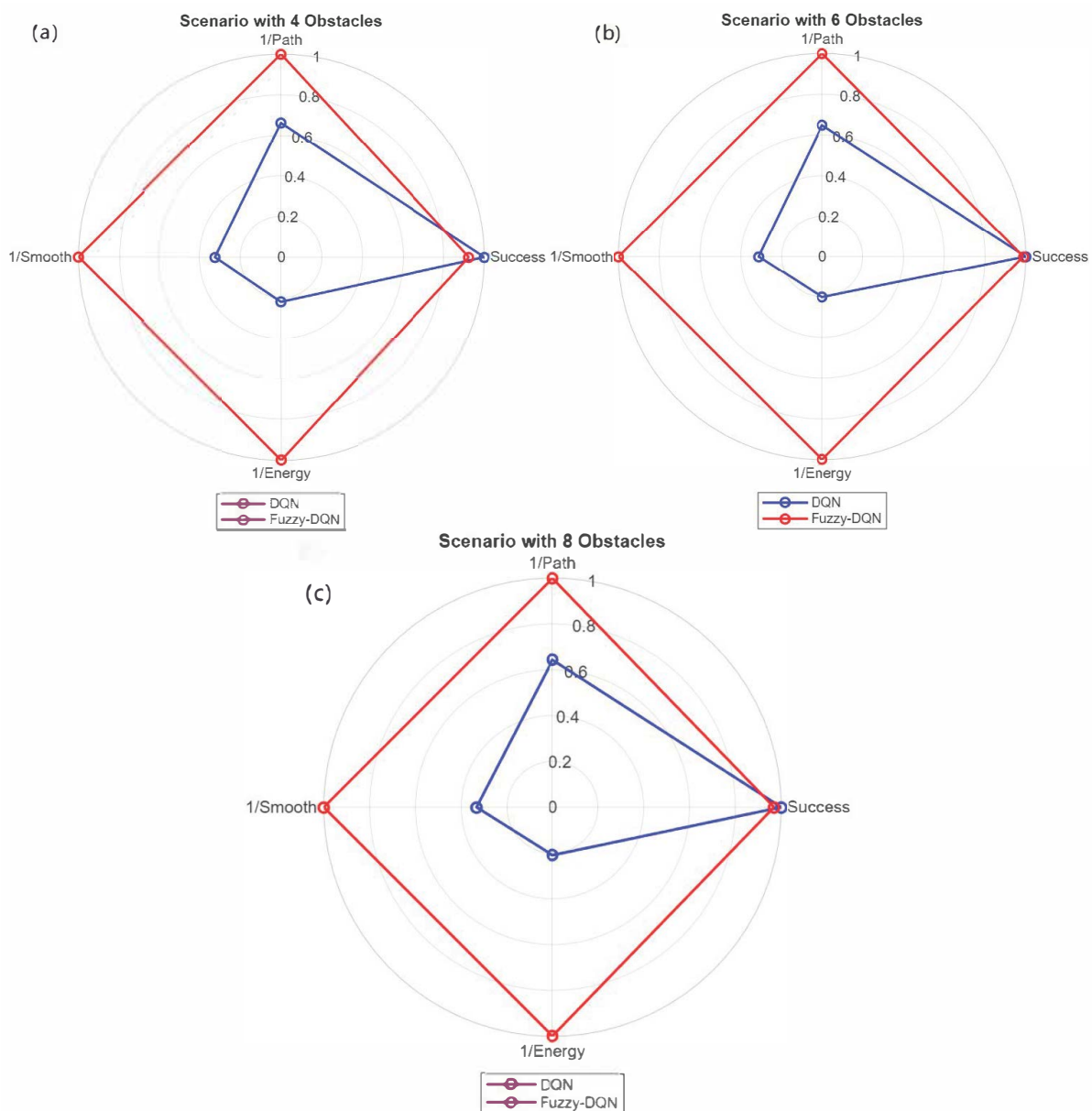


Figure 5. (a) DQN and F-DQN overall comparison for 4 obstacle topology, (b) DQN and F-DQN overall comparison for 6 obstacle topology, (c) DQN and F-DQN overall comparison for 8 obstacle topology.

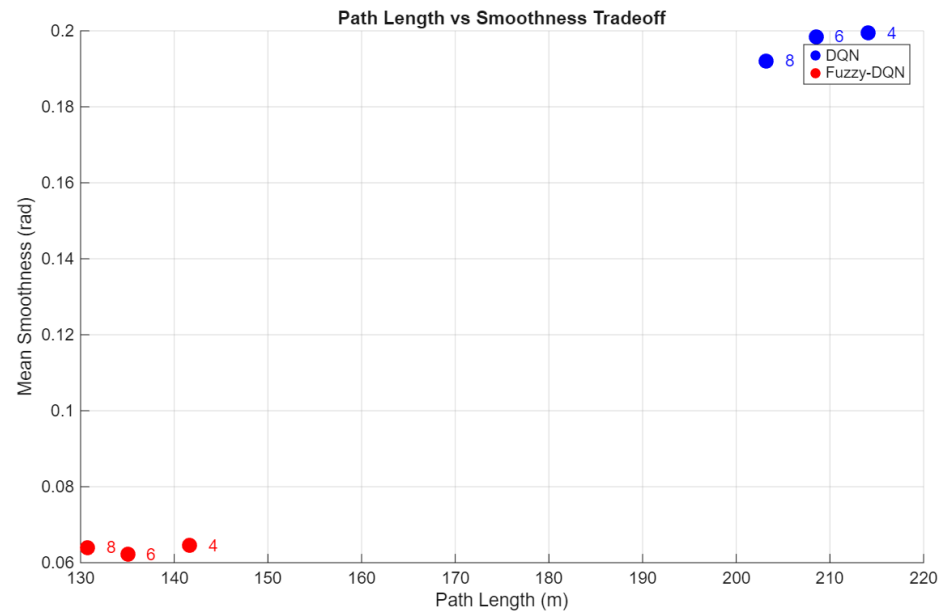


Figure 6. Trade-off Analysis between Path Length and Smoothness.

Overall, these findings demonstrate that fuzzy reward modeling provides a powerful extension to deep reinforcement learning, enabling more interpretable and physically consistent navigation behavior for unmanned surface or aerial vehicles operating in cluttered and uncertain environments.

5.4. Comparison with Existing Works

A comparison of the obtained results with the existing literature further highlights the effectiveness of the proposed F-DQN framework. Classical and metaheuristic-based USV path-planning approaches, such as A*, smoothed A*, and ant-colony or particle-swarm variants [1–4,8,9,13,18], typically achieve improvements in path efficiency in the range of 10–25%, with performance strongly dependent on obstacle layout and environmental uncertainty. Energy-aware or multi-layer planning methods [11,20] generally report 15–30% reductions in energy use, but often require strong assumptions regarding currents or thrust modeling. Even hybrid marine navigation planners and fuzzy-search methods [12,16,19] achieve smoother trajectories but limited overall efficiency gains.

Machine-learning-based approaches cited in this work also tend to demonstrate accuracy levels or success rates that are similar to or lower than those achieved in our study, particularly under dynamic or cluttered environments. For example, DRL- or DQN-based maritime navigation methods focusing on collision avoidance or local decision making [21,33,35] generally report success rates in the range of 80–90% and modest improvements in trajectory quality, with performance degrading as obstacle density increases. Related learning-based frameworks for UAV/UGV navigation [30,37] show similar behavior, often achieving stable convergence but limited optimization in energy or curvature metrics. In contrast, the proposed F-DQN delivers a more substantial and robust performance benefit—exceeding 35% reduction in total path length, 70–80% decrease in energy consumption, and a three-fold improvement in path smoothness—while maintaining success rates within 2–6% of the DQN baseline across all scenarios. These findings demonstrate that integrating Mamdani-type fuzzy reasoning into the RL reward structure not only aligns with the positive trends observed in prior ML-based navigation studies but, importantly, achieves equal or higher success rates along with significantly greater efficiency and stability.

6. Conclusions

This paper proposed a DQN and an F-DQN framework for the multi-objective USV path planning problem, addressing the simultaneous optimization of path length, smoothness, and fuel consumption. The approach combines the adaptability of Deep Q-Learning with the interpretability of fuzzy inference, enabling the agent to balance competing objectives through linguistic reasoning. The conventional DQN served as the baseline for comparison, while the F-DQN incorporated Mamdani-type fuzzy rules to refine reward evaluation and action selection based on multiple performance indicators.

Simulation results across different obstacle density scenarios demonstrated that the proposed F-DQN consistently outperformed the standard DQN. Specifically, the F-DQN achieved more than 35% reduction in total path length, an approximate 70–80% decrease in energy consumption, and a three-fold improvement in trajectory smoothness, while maintaining comparable success rates (differences within 2–6%). These quantitative findings confirm that integrating Mamdani-type fuzzy logic into deep reinforcement learning enhances navigation efficiency, stability, and interpretability, offering a robust solution for energy-aware autonomous USV motion planning. These results confirm that the integration of fuzzy logic into deep reinforcement learning enhances both performance and transparency in decision-making, offering a robust solution for energy-aware and interpretable USV navigation.

Future research will focus on extending the proposed model to dynamic environments with time-varying disturbances, as well as to cooperative multi-agent settings where multiple USVs operate jointly. Further investigation into transfer learning and real-time hardware deployment will also be pursued to validate the scalability and practical feasibility of the framework. Overall, the presented F-DQN approach constitutes a promising step toward explainable and efficient reinforcement learning solutions for intelligent maritime autonomy.

Author Contributions: Conceptualization, I.A.B., C.N., and G.A.; methodology, I.A.B., C.N., and G.A.; software, I.A.B.; validation, I.A.B., C.N., G.A., and D.L.; formal analysis, C.N., I.A.B., G.A., and D.L.; investigation, I.A.B., C.N., G.A., and D.L.; resources I.A.B., C.N., G.A., and D.L.; data curation, I.A.B.; writing—original draft preparation, I.A.B. and C.N.; writing—review and editing, I.A.B., C.N., G.A., and D.L.; visualization, I.A.B.; supervision, D.L.; project administration, G.A. and D.L. All authors have read and agreed to the published version of the manuscript.

Funding: This research has not received funding.

Data Availability Statement: Dataset available upon request from the authors.

Conflicts of Interest: The authors declare no conflicts of interest.

Abbreviations

Acronym	Definition
ACO	Ant Colony Optimization
AI	Artificial Intelligence
APF	Artificial Potential Field
DNN	Deep Neural Network
DOAJ	Directory of Open Access Journals
DRL	Deep Reinforcement Learning
DQN	Deep Q-Learning
EE	Energy Consumption due to Environmental Conditions
EC	Total Energy (Fuel) Consumption
ED	Energy Consumption due to Velocity Deviations
F-DQN	Fuzzy Deep Q-Learning

FIS	Fuzzy Inference System
GA	Genetic Algorithm
MDPI	Multidisciplinary Digital Publishing Institute
MDP	Markov Decision Process
NN-MPC	Neural Network Model Predictive Controller
PC	Path Curvature (Path Deviation)
PSO	Particle Swarm Optimization
RL	Reinforcement Learning
RRT	Rapidly-Exploring Random Tree
SR	Success Rate
TLA	Three-Letter Acronym
UGV	Unmanned Ground Vehicle
UAV	Unmanned Aerial Vehicle
USV	Unmanned Surface Vehicle
WSM	Weighted Sum Method

References

1. Singh, Y.; Sharma, S.; Sutton, R.; Hatton, D.; Khan, A. A Constrained A* Approach towards Optimal Path Planning for an Unmanned Surface Vehicle in a Maritime Environment Containing Dynamic Obstacles and Ocean Currents. *Ocean Eng.* **2018**, *169*, 187–201. [\[CrossRef\]](#)
2. Song, R.; Liu, Y.; Bucknall, R. Smoothed A* Algorithm for Practical Unmanned Surface Vehicle Path Planning. *Appl. Ocean Res.* **2019**, *83*, 9–20. [\[CrossRef\]](#)
3. Singh, Y.; Sharma, S.; Sutton, R.; Hatton, D. Optimal Path Planning of an Unmanned Surface Vehicle in a Real-Time Marine Environment Using a Dijkstra Algorithm. In *Marine Navigation*; CRC Press: Boca Raton, FL, USA, 2017; ISBN 978-1-315-09913-2.
4. Mirjalili, S.; Song Dong, J.; Lewis, A. Ant Colony Optimizer: Theory, Literature Review, and Application in AUV Path Planning. In *Nature-Inspired Optimizers: Theories, Literature Reviews and Applications*; Mirjalili, S., Song Dong, J., Lewis, A., Eds.; Studies in Computational Intelligence; Springer International Publishing: Cham, Switzerland, 2020; pp. 7–21; ISBN 978-3-030-12127-3.
5. Wang, H.; Guo, F.; Yao, H.; He, S.; Xu, X. Collision Avoidance Planning Method of USV Based on Improved Ant Colony Optimization Algorithm. *IEEE Access* **2019**, *7*, 52964–52975. [\[CrossRef\]](#)
6. Li, D.; Wang, P.; Du, L. Path Planning Technologies for Autonomous Underwater Vehicles-A Review. *IEEE Access* **2019**, *7*, 9745–9768. [\[CrossRef\]](#)
7. Zhang, H.; Lin, W.; Chen, A. Path Planning for the Mobile Robot: A Review. *Symmetry* **2018**, *10*, 450. [\[CrossRef\]](#)
8. Ntakolia, C.; Lyridis, D.V. A Swarm Intelligence Graph-Based Pathfinding Algorithm Based on Fuzzy Logic (SIGPAF): A Case Study on Unmanned Surface Vehicle Multi-Objective Path Planning. *J. Mar. Sci. Eng.* **2021**, *9*, 1243. [\[CrossRef\]](#)
9. Ntakolia, C.; Lyridis, D.V. A Comparative Study on Ant Colony Optimization Algorithm Approaches for Solving Multi-Objective Path Planning Problems in Case of Unmanned Surface Vehicles. *Ocean Eng.* **2022**, *255*, 111418. [\[CrossRef\]](#)
10. Niu, H.; Savvaris, A.; Tsourdos, A.; Ji, Z. Voronoi-Visibility Roadmap-Based Path Planning Algorithm for Unmanned Surface Vehicles. *J. Navig.* **2019**, *72*, 850–874. [\[CrossRef\]](#)
11. Ding, F.; Zhang, Z.; Fu, M.; Wang, Y.; Wang, C. Energy-Efficient Path Planning and Control Approach of USV Based on Particle Swarm Optimization. In Proceedings of the OCEANS 2018 MTS/IEEE Charleston, Charleston, SC, USA, 22–25 October 2018; pp. 1–6.
12. Deng, H.; Zhu, J. Optimal Path Planning for Unmanned Vehicles Using Improved Ant Colony Optimization Algorithm. In *Proceedings of the Neural Computing for Advanced Applications*; Zhang, H., Yang, Z., Zhang, Z., Wu, Z., Hao, T., Eds.; Springer: Singapore, 2021; pp. 701–714.
13. Xin, J.; Zhong, J.; Yang, F.; Cui, Y.; Sheng, J. An Improved Genetic Algorithm for Path-Planning of Unmanned Surface Vehicle. *Sensors* **2019**, *19*, 2640. [\[CrossRef\]](#) [\[PubMed\]](#)
14. Xia, G.; Sun, X.; Xia, X. Multiple Task Assignment and Path Planning of a Multiple Unmanned Surface Vehicles System Based on Improved Self-Organizing Mapping and Improved Genetic Algorithm. *J. Mar. Sci. Eng.* **2021**, *9*, 556. [\[CrossRef\]](#)
15. Zhang, J.; Zhang, F.; Liu, Z.; Li, Y. Efficient Path Planning Method of USV for Intelligent Target Search. *J. Geovis Spat. Anal.* **2019**, *3*, 13. [\[CrossRef\]](#)
16. Sang, H.; You, Y.; Sun, X.; Zhou, Y.; Liu, F. The Hybrid Path Planning Algorithm Based on Improved A* and Artificial Potential Field for Unmanned Surface Vehicle Formations. *Ocean Eng.* **2021**, *223*, 108709. [\[CrossRef\]](#)

17. Zhang, Z.; Wu, D.; Gu, J.; Li, F. A Path-Planning Strategy for Unmanned Surface Vehicles Based on an Adaptive Hybrid Dynamic Stepsize and Target Attractive Force-RRT Algorithm. *J. Mar. Sci. Eng.* **2019**, *7*, 132. [\[CrossRef\]](#)
18. Zhong, J.; Li, B.; Li, S.; Yang, F.; Li, P.; Cui, Y. Particle Swarm Optimization with Orientation Angle-Based Grouping for Practical Unmanned Surface Vehicle Path Planning. *Appl. Ocean Res.* **2021**, *111*, 102658. [\[CrossRef\]](#)
19. Xin, J.; Zhong, J.; Li, S.; Sheng, J.; Cui, Y. Greedy Mechanism Based Particle Swarm Optimization for Path Planning Problem of an Unmanned Surface Vehicle. *Sensors* **2019**, *19*, 4620. [\[CrossRef\]](#)
20. Wang, N.; Jin, X.; Er, M.J. A Multilayer Path Planner for a USV under Complex Marine Environments. *Ocean Eng.* **2019**, *184*, 1–10. [\[CrossRef\]](#)
21. Ma, Y.; Hu, M.; Yan, X. Multi-Objective Path Planning for Unmanned Surface Vehicle with Currents Effects. *ISA Trans.* **2018**, *75*, 137–156. [\[CrossRef\]](#) [\[PubMed\]](#)
22. Lyridis, D.V. An Improved Ant Colony Optimization Algorithm for Unmanned Surface Vehicle Local Path Planning with Multi-Modality Constraints. *Ocean Eng.* **2021**, *241*, 109890. [\[CrossRef\]](#)
23. Ntakolia, C.; Iakovidis, D.K. A Swarm Intelligence Graph-Based Pathfinding Algorithm (SIGPA) for Multi-Objective Route Planning. *Comput. Oper. Res.* **2021**, *133*, 105358. [\[CrossRef\]](#)
24. Xiujuan, L.; Zhongke, S. Overview of Multi-Objective Optimization Methods. *J. Syst. Eng. Electron.* **2004**, *15*, 142–146.
25. Gunantara, N. A Review of Multi-Objective Optimization: Methods and Its Applications. *Cogent Eng.* **2018**, *5*, 1502242. [\[CrossRef\]](#)
26. Ntakolia, C.; Platanitis, K.S.; Kladis, G.P.; Skliros, C.; Zagorianos, A.D. A Genetic Algorithm Enhanced with Fuzzy-Logic for Multi-Objective Unmanned Aircraft Vehicle Path Planning Missions*. In Proceedings of the 2022 International Conference on Unmanned Aircraft Systems (ICUAS), Dubrovnik, Croatia, 21–24 June 2022; pp. 114–123.
27. Zhou, X.; Wu, P.; Zhang, H.; Guo, W.; Liu, Y. Learn to Navigate: Cooperative Path Planning for Unmanned Surface Vehicles Using Deep Reinforcement Learning. *IEEE Access* **2019**, *7*, 165262–165278. [\[CrossRef\]](#)
28. Qu, T.; Xiong, G.; Ali, H.; Dong, X.; Han, Y.; Shen, Z. USV Path Planning Under Marine Environment Simulation Using DWA and Safe Reinforcement Learning. In Proceedings of the 2023 IEEE 19th International Conference on Automation Science and Engineering (CASE), Auckland, New Zealand, 26–30 August 2023; pp. 1–6. [\[CrossRef\]](#)
29. Xing, B.; Wang, X.; Yang, L.; Liu, Z.; Wu, Q. An Algorithm of Complete Coverage Path Planning for Unmanned Surface Vehicle Based on Reinforcement Learning. *J. Mar. Sci. Eng.* **2023**, *11*, 645. [\[CrossRef\]](#)
30. Li, J.; Chavez-Galaviz, J.; Azizzadenesheli, K.; Mahmoudian, N. Dynamic Obstacle Avoidance for USVs Using Cross-Domain Deep Reinforcement Learning and Neural Network Model Predictive Controller. *Sensors* **2023**, *23*, 3572. [\[CrossRef\]](#)
31. Nantogma, S.; Zhang, S.; Yu, X.; An, X.; Xu, Y. Multi-USV Dynamic Navigation and Target Capture: A Guided Multi-Agent Reinforcement Learning Approach. *Electronics* **2023**, *12*, 1523. [\[CrossRef\]](#)
32. Dong, P.; Li, J.; Wang, Y.; Zhang, H.; Chen, S.; Liu, M. An Optimized Scheduling Scheme for UAV-USV Cooperative Search via Multi-Agent Reinforcement Learning Approach. In Proceedings of the 2024 20th International Conference on Mobility, Sensing and Networking (MSN), Harbin, China, 12–14 December 2024; pp. 172–179. [\[CrossRef\]](#)
33. Zhang, J.; Ren, J.; Cui, Y.; Fu, D.; Cong, J. Multi-USV Task Planning Method Based on Improved Deep Reinforcement Learning. *IEEE Internet Things J.* **2024**, *11*, 18549–18567. [\[CrossRef\]](#)
34. Zhou, Y.; Gong, C.; Chen, K. Adaptive Control Scheme for USV Trajectory Tracking Under Complex Environmental Disturbances via Deep Reinforcement Learning. *IEEE Internet Things J.* **2025**, *12*, 15181–15196. [\[CrossRef\]](#)
35. Chen, Z.; Zhang, Y.; Zhang, Y.; Nie, Y.; Tang, J.; Zhu, S. A Hybrid Path Planning Algorithm for Unmanned Surface Vehicles in Complex Environment with Dynamic Obstacles. *IEEE Access* **2019**, *7*, 126439–126449. [\[CrossRef\]](#)
36. Song, R.; Liu, Y.; Bucknall, R. A Multi-Layered Fast Marching Method for Unmanned Surface Vehicle Path Planning in a Time-Variant Maritime Environment. *Ocean Eng.* **2017**, *129*, 301–317. [\[CrossRef\]](#)
37. Xia, G.; Han, Z.; Zhao, B.; Liu, C.; Wang, X. Global Path Planning for Unmanned Surface Vehicle Based on Improved Quantum Ant Colony Algorithm. *Math. Probl. Eng.* **2019**, *2019*, e2902170. [\[CrossRef\]](#)
38. Harliana, P.; Rahim, R. Comparative Analysis of Membership Function on Mamdani Fuzzy Inference System for Decision Making. *J. Phys. Conf. Ser.* **2017**, *930*, 012029. [\[CrossRef\]](#)
39. Ntakolia, C.; Lyridis, D.V. A n – D Ant Colony Optimization with Fuzzy Logic for Air Traffic Flow Management. *Oper. Res. Int. J.* **2022**, *22*, 5035–5053. [\[CrossRef\]](#)
40. Ntakolia, C.; Kladis, G.P.; Lyridis, D.V. A Fuzzy Logic Approach of Pareto Optimality for Multi-Objective Path Planning in Case of Unmanned Surface Vehicle. *J. Intell. Robot. Syst.* **2023**, *109*, 21. [\[CrossRef\]](#)
41. Hamam, A.; Georganas, N.D. A Comparison of Mamdani and Sugeno Fuzzy Inference Systems for Evaluating the Quality of Experience of Hapto-Audio-Visual Applications. In Proceedings of the 2008 IEEE International Workshop on Haptic Audio visual Environments and Games, Ottawa, ON, Canada, 26–27 October 2008; IEEE: Ottawa, ON, Canada, 2008; pp. 87–92.

42. Bagis, A.; Konar, M. Comparison of Sugeno and Mamdani Fuzzy Models Optimized by Artificial Bee Colony Algorithm for Nonlinear System Modelling. *Trans. Inst. Meas. Control* **2016**, *38*, 579–592. [[CrossRef](#)]
43. Singla, J. Comparative Study of Mamdani-Type and Sugeno-Type Fuzzy Inference Systems for Diagnosis of Diabetes. In Proceedings of the 2015 International Conference on Advances in Computer Engineering and Applications, Ghaziabad, India, 19–20 March 2015; IEEE: Ghaziabad, India, 2015; pp. 517–522.

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.