Information, Computing and Intelligent Systems, 2025, No. 7, 96 – 109

UDC 004.8 https://doi.org/10.20535/2786-8729.7.2025.341723

DEEP Q-LEARNING POLICY OPTIMIZATION METHOD FOR ENHANCING GENERALIZATION IN AUTONOMOUS VEHICLE CONTROL

Mykhailo Drahan*
https://orcid.org/0009-0002-5583-2907

Andrii Pysarenko
https://orcid.org/0000-0001-7947-218X

National Technical University of Ukraine
“Igor Sikorsky Kyiv Polytechnic Institute”, Kyiv, Ukrainee

*Corresponding author: mykhailodrahan@gmail.com

The development of autonomous vehicle control policies based on deep reinforcement learning is a principal technical problem for cyber-physical systems, fundamentally constrained by the high dimensionality of state spaces, inherent algorithmic instability, and a pervasive risk of policy over-specialization that severely limits generalization to real-world scenarios. The object of this investigation is the iterative process of forming a robust control policy within a simulated environment, while the subject focuses on the influence of specialized reward structures and initial training conditions on policy convergence and generalization capability. The study’s aim is to develop and empirically evaluate a deep Q-learning policy optimization method that utilizes dynamic initial conditions to mitigate over-specialization and achieve stable, globally optimal adaptive control. The developed method formalizes two optimization criteria. First, the adaptive reward function serves as the safety and convergence criterion, defined hierarchically with major penalties for collision, intermediate incentives for passing checkpoints and a continuous minor penalty for elapsed time to drive efficiency. Second, the mechanism of dynamic initial conditions acts as the policy generalization criterion, designed to inject necessary stochasticity into the state distribution. The agent is modeled as a vehicle equipped with an eight-sensor system providing 360° coverage, making decisions from a discrete action space of seven options. Its ten-dimensional state vector integrates normalized sensor distance readings with normalized dynamic characteristics, including speed and angular error. Empirical testing confirmed the policy’s vulnerability under baseline fixed-start conditions, where the agent demonstrated over-specialization and stagnated at a traveled distance of approximately 960 conventional units after 40,000 episodes. The subsequent application of the dynamic initial conditions criterion successfully addressed this failure. By forcing the agent to rely on its generalized state mapping instead of trajectory memory, this approach successfully overcame the learning plateau, enabling the agent to achieve full, collision-free track traversal between 53,000 and 54,000 episodes. Final optimization, driven by penalty, reduced the total track completion time by nearly half. This verification confirms the method’s value in producing robust, stable, and efficient control policies suitable for integration into autonomous transport cyber-physical systems.

Keywords: deep Q-learning, autonomous vehicle, policy generalization, reward function, dynamic initial conditions, cyber-physical systems.

1. Introduction

Modern advancements in computation place Cyber-Physical Systems (CPS) as a primary focus for technological progress in areas including transport and infrastructure. CPS effectively integrate physical components with computing algorithms, enabling autonomous operation, adaptation to changing conditions, and real-time decision-making. Autonomous transportation is a compelling example of this technology, driven by the desire for improved road safety and resource efficiency.

Training Autonomous Vehicles (AVs) is a sophisticated, interdisciplinary process that spans control theory, AI, and computer modeling. Building artificial intelligence into physical vehicles without extensive prior validation in a simulated environment is both costly and high-risk. Consequently, the initial phase of AV development requires creating high-fidelity simulation models. These models must accurately reflect vehicle dynamics, sensor function, and interaction with the environment. These virtual environments offer a scalable venue for developing control policies using machine learning techniques, such as Reinforcement Learning (RL).

Deep Reinforcement Learning (DRL), specifically the Deep Q-learning (DQN) algorithm, is a central technique in this domain. DQN combines Q-learning with deep neural networks to approximate the optimal value function, allowing the agent to manage the extensive state spaces produced by vehicle sensors. This capability allows agents to acquire complex, self-learned maneuvering strategies in simulated environments.

The direct application of DRL to autonomous navigation, however, introduces several complexities that define the general scientific problem.

1. Scale and unstable behavior. DRL implementation faces complications due to the extensive size of the state and action spaces, which makes the search for optimal strategies difficult. Furthermore, training the underlying deep neural networks frequently leads to unstable behavior and sensitivity to stochastic elements and hyperparameter settings.

2. Sparse reward. Many navigation tasks feature sparse and delayed rewards, where positive feedback is only given upon successfully reaching the final goal. This hindered learning progression necessitates the application of reward shaping – providing auxiliary feedback to guide the agent toward the intended policy and hasten convergence.

3. Policy over-specialization. The primary hindrance involves the risk of overfitting, where the agent optimizes its policy for a specific, memorized environmental configuration or set of initial conditions. Performance degradation of DRL models can reach up to 35% when exposed to unseen weather or urban configurations, confirming a generalization gap between simulation and real-world performance. Over-specialization also causes agents to fail up to 60% of the time when faced with dynamic obstacles not present during training.

Current research confirms DRL's promise but maintains that several aspects remain open scientific questions. In particular, universal solutions for counteracting over-specialization and ensuring policy resilience in varied environments are lacking. Insufficient experimental investigation has been dedicated to the influence of varying initial conditions and trajectory diversity on the policy formation process.

The scientific problem is defined as method to stabilize the deep Q-learning process and enhance its policy's generalization capability by directly mitigating the effects of over-specialization. Addressing this problem – improving the generalization ability of DRL agents – is an important scientific-practical problem that represents a necessary prerequisite for integration of DRL systems into complex, real-world autonomous transport CPS.

2. Literature review and problem statement

The study of AV control policies represents a convergence point for several scientific disciplines, with DRL techniques taking a leading role in research [1, 2]. Recent works have demonstrated the efficacy of DRL in tasks such as motion stabilization, obstacle avoidance, and trajectory optimization [3, 4]. For instance, certain implementations show the power of combining convolutional neural networks (CNNs) with DQN agents to manage navigation in geometrically challenging environments [5].

The international research community recognizes DRL as a powerful framework but identifies persistent shortcomings that must be addressed before mass deployment in real-world CPS [6, 7]. Analyzing contemporary publications reveals three fundamental, interrelated areas of concern.

A major difficulty in applying DRL is the inherent instability of the learning process. DRL models are often highly sensitive to the initial selection of hyperparameters and stochastic elements within the environment, where even minor changes can trigger large fluctuations in agent performance [8]. Works [9, 10] emphasize that meticulous design of the experience replay buffer and careful sampling techniques are necessary to prevent policy degradation.

To address these limitations, researchers have explored advancements beyond the basic DQN framework. For example, some studies propose integrating algorithms like double DQN with specialized hardware, such as Field Programmable Gate Arrays (FPGAs), to enhance both algorithmic stability and computational efficiency in the context of vehicular CPS [11]. FPGAs provide the necessary parallel processing and reconfigurability to handle the computational load imposed by complex, real-time DRL decision-making. This trend shows that world-leading research is tackling stability not only algorithmically but also through hardware acceleration.

The effectiveness of any RL agent is dictated by the quality of the feedback it receives. In navigation, the sparse reward problem – where the agent receives feedback only upon reaching a distant goal – is widely documented. This sparsity causes reward oscillations, severely limiting the agent's generalization capability and promoting focus on locally beneficial actions over the globally optimal strategy.

The standard approach to mitigating this is reward shaping, which introduces supplementary rewards or penalties to guide the agent. While effective, reward shaping requires substantial domain expertise and is often problem-specific. More advanced approaches are now under investigation to bypass the limitations of manually designed rewards (reward misspecification), which is reported to be the cause of over 20% of reported failures in simulated environments [12]. Recent studies, for example, have introduced Active Preference Learning (APL), which utilizes human preferences to derive reward functions that more accurately reflect human intentions. This approach has demonstrated the ability to significantly improve navigation success rates, overcoming the problem of designing reward functions for complex continuous state spaces [13].

The most persistent obstacle to safely deploying autonomous systems is the generalization gap – the drop in performance when a policy trained in a simulated environment is transferred to an unseen or real-world scenario. Over-specialization (or overfitting) to the training environment's geometry or static conditions is the direct cause.

Research conducted using high-fidelity platforms like the CARLA simulator confirms the severity of this issue:

– end-to-end RL models can experience performance degradation of up to 35% when exposed to unforeseen weather or unfamiliar urban layouts;

– agents frequently fail, sometimes up to 60% of the time, when encountering dynamic obstacles, they did not experience during training [14].

These findings demonstrate that agents typically optimize their policy for the specific training route, relying on memory of the sequence of actions rather than developing a universal control representation. Solutions being explored include architectural innovations, such as using Vision Transformers (ViTs) to replace traditional CNNs, which have shown superior performance in visually complex environments by better capturing global spatial patterns. However, even these advancements do not inherently address the root cause of overfitting tied to static initial conditions.

The analysis of previous studies confirms that while DRL is highly successful for core AV control tasks, the primary scientific obstacles remain policy stability, reward system design, and the generalization gap. The majority of global research efforts focus on algorithmic improvements (double DQN, PPO) or better sensory processing (ViT, CNN) to enhance robustness.

However, insufficiently studied aspect is the direct influence of the environment's starting configuration on policy generalization. The literature identifies the lack of experimental investigation into the impact of initial conditions and trajectory variations on the DRL policy formation process as an open problem.

Therefore, the unresolved scientific problem is the empirical determination of an effective mechanism to stabilize the DQN process and enhance policy generalization by directly counteracting the effects of over-specialization through the modulation of environmental variables. Specifically, this study aims to investigate whether the dynamic alteration of the agent's starting position provides the necessary stochasticity and state diversity required to compel the agent to learn a globally optimal, generalized control strategy. This investigation provides the justification for the experimental study presented in the subsequent sections.

3. The aim and objectives of the study

The aim of this study is enhancing policy generalization in DRL for autonomous vehicle control by developing and validating policy optimization method.

To achieve this aim, the following objectives are established:

1. To formalize and implement the DQN policy optimization method, which combines an adaptive, multi-component reward function (as a convergence and safety criterion) and the mechanism of dynamic initial conditions (as a policy generalization criterion). This implementation will be realized within an autonomous vehicle control simulation environment, necessitating the definition of a comprehensive state vector and action space.
2. To empirically evaluate the effectiveness of the developed method by comparing the learning dynamics of the DQN agent across two scenarios: training with static (fixed) versus stochastic (dynamic) initial conditions. The evaluation will use predefined metrics to quantify policy stability, generalization capability, and final navigation efficiency.

4. The study materials and methods of enhancing policy generalization

4.1. Object, subject, and hypothesis of the study

Object of the study is the process of forming a control policy for an autonomous vehicle operating within a virtual environment, which integrates physics-based elements with computational algorithms to enable adaptive behavior and operational decision-making.

Subject of the study is the influence of the formalized reward function structure, the exploration coefficient schedule, and the initial starting conditions on the convergence, stability, and generalization capability of the DQN policy.

We hypothesize that introducing dynamic variation in the agent's starting position across training episodes will effectively counteract the policy over-specialization observed under fixed starting conditions. This stochastic modulation of initial states will compel the DRL agent to develop a generalized state representation, leading to greater stability in learning, the ability to pass the entire complex trajectory, and subsequent optimization of movement efficiency.

4.2. Formalization of the policy optimization method

The method for enhancing policy generalization is defined by its ability to optimize the DQN agent's control strategy based on safety, convergence and generalization criteria. The optimization strategy involves a structured approach to defining the reward function and training schedule.

4.2.1. Safety and convergence optimization criterion

The method is implemented using the DQN algorithm, which relies on a DNN to approximate the optimal action-value function $Q^*(s,a)$. This function represents the maximum expected discounted return achievable by taking action a in state s. The learning objective is defined by the Bellman optimality equation, which governs the estimation of the optimal Q-values:

$$Q^*(s,a) = \mathbb{E}_{s',r}\left[r + \gamma \max_{a'} Q^*(s',a')\right], \tag{1}$$

where r is the immediate reward, γ is the discount factor, and $Q^*(s',a')$ is the optimal action-value for the subsequent state s'.

To guide the agent toward safe and efficient policy convergence, an adaptive, multi-component reward function R is formalized as the primary optimization criterion. This structure is designed to mitigate the problem of sparse reward by providing dense, hierarchical feedback. The total reward R at each step t is calculated as:

$$R = r_{goal} + r_{col} + r_{cp} + r_{rev} + r_{time}, \tag{2}$$

where the function components implement explicit reward shaping:
r_{goal} – large positive reward upon reaching the final goal (global incentive);
r_{col} – large negative penalty upon collision, terminating the episode (safety constraint);
r_{cp} – intermediate positive reward for crossing a new control point (mitigates sparsity, encourages progress);
r_{rev} – negative penalty for moving backward or returning to an already passed control point (promotes forward momentum);
r_{time} – small, constant penalty applied at every time step (stimulates time minimization and speed optimization).

The hierarchy of reward magnitudes is established to prioritize collision avoidance and goal attainment over local incentives, ensuring global policy optimization: $|r_{col}|$ is set comparable to r_{goal} to impose a strong safety constraint, while the r_{time} penalty acts as a continuous incentive for speed optimization upon achieving policy generalization.

4.2.2. Policy generalization optimization criterion

To formally address the problem of over-specialization (generalization gap), the core of the developed method is the application of a stochastic training protocol via dynamic initial conditions. This mechanism serves as the direct criterion for achieving a generalized policy:

– baseline protocol. Episodes start from a fixed coordinate used to empirically demonstrate policy overfitting (local optimum);

– generalization protocol. The agent's starting position is periodically varied along the track. This manipulation increases the diversity of initial states, compelling the agent to rely solely on the instantaneous state vector S_t for decision-making rather than memorizing a trajectory. This action forces the deep neural network to learn a true, generalized state-to-action mapping.

4.3. Simulation environment and agent architecture implementation

4.3.1. Environmental configuration

The simulation utilized a closed-loop track geometry with map dimensions of 300×200 conventional units, designed to thoroughly test control policy stability. The full trajectory spans 6845 conventional units and is delineated by a sequence of control points.

The default (fixed) starting position (marked with green in Fig. 1) was established at coordinates (270, 200), with the final target located at (105, 200) (marked with red in Fig. 1). These points define the complete path for full traversal evaluation.

4.3.2. Vehicle agent and action space

The agent is modeled as a simulated car, featuring a specific sensory configuration and a discrete action space suitable for DQN implementation.

The vehicle is equipped with eight distance sensors, uniformly distributed around the vehicle's center with an angular interval of 45°. This setup ensures a full 360° coverage, allowing the agent to detect obstacles (walls) in all directions, which is paramount for forming a safe navigation policy.

The set of available control actions is deliberately limited to seven discrete options to maintain tractability and computational efficiency, which aligns well with the value-based DQN algorithm:

$$A = \{a_1, a_2, a_3, a_4, a_5, a_6, a_7\}, \tag{3}$$

where the actions correspond to specific combinations of angular and translational velocity components: a_1 (move straight), a_2 (slow down/brake), a_3 (straight movement with left turn), a_4

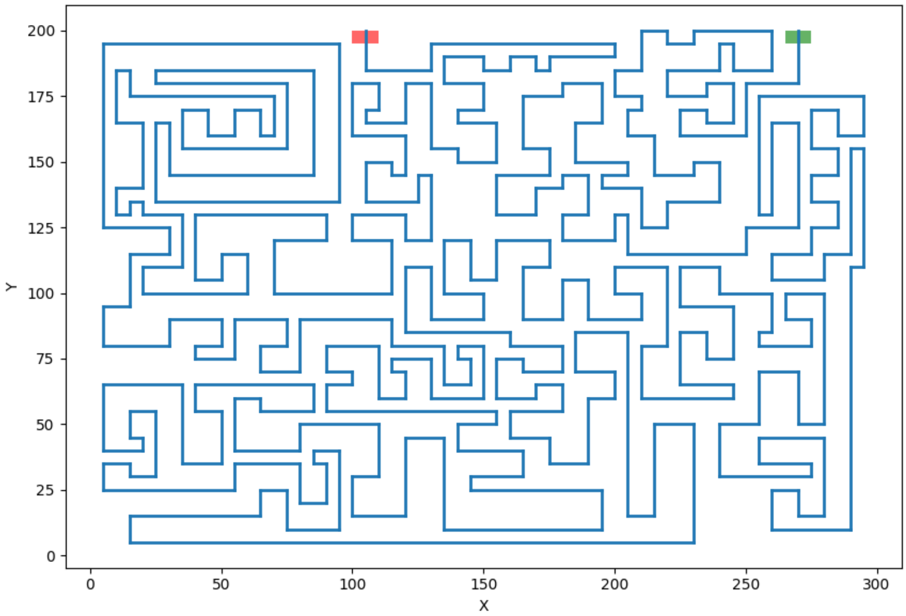

Fig. 1. Route diagram for traffic simulation

(straight movement with right turn), a_5 (turn only left), a_6 (turn only right), and a_7 (reverse movement).

4.4. State vector construction and learning parameters

4.4.1. State vector implementation

To ensure stable network learning, the raw sensor readings and vehicle dynamics were transformed and normalized to construct the final state vector, S_t.

The agent's state vector S_t provides a holistic assessment by integrating geometric observations, speed dynamics, and goal orientation.

Sensor readings (d_i) are against the maximum visibility distance (d_{max}) to scale inputs consistently for the neural network:

$$d_{norm}^{(i)} = \frac{d_i}{d_{max}} \in [0, 1]\,, \;\; i = \overline{1, 8}. \tag{4}$$

The vector of spatial observations is thus $\mathbf{d}_{norm} = [d_{norm}^{(1)}, \ldots, d_{norm}^{(8)}]$.

The vector is augmented with two normalized dynamic characteristics: normalized speed v_{norm} and angular error ψ_{err}.

Normalized speed v_{norm} is calculated as the distance traveled between time steps divided by the time interval (5), and normalized against the vehicle's maximal speed, v_{max} (6).

$$v_t = \frac{\|p_t - p_{t-1}\|}{\Delta t}, \;\; p_t = (x_t, y_t)\,. \tag{5}$$

$$v_{norm} = \frac{v_t}{v_{max}} \in [0, 1]\,. \tag{6}$$

Angular error ψ_{err} is defined as the difference between the current vehicle course ψ_t and the desired direction ψ_{goal} toward the next checkpoint $c_t = \left(x_{cp}, y_{cp}\right)$, with the value constrained to the interval

$(-\pi, \pi]$ using the wrap$(\cdot)$ function:

$$\psi_{err} = \text{wrap}(\psi_t - \psi_{goal}), \tag{7}$$

where $\psi_{goal} = \text{atan2}\left(y_{cp} - y_t, x_{cp} - x_t\right)$.

The final state vector presented to the DQN agent at time t is:

$$\mathbf{S}_t = [d_{norm}^{(1)}, \ldots, d_{norm}^{(8)}, v_{norm}, \psi_{err}]. \tag{8}$$

This construction allows the agent to comprehensively assess the situation on the track – simultaneously taking into account the geometry of the environment, its own speed, and orientation relative to the target. This provides more stable and effective learning, especially in environments with a large number of turns and variable trajectory characteristics.

4.4.2. Exploration-exploitation strategy

Training follows an iterative episodic format with an ϵ-greedy strategy to balance exploration of unknown states with exploitation of acquired knowledge.

Initial phase (0–1000 episodes). Full exploration is enforced to populate the experience buffer with a diverse set of state-action transitions.

Decay phase (from 1001st episode). The exploration coefficient ϵ starts at 95% and gradually decreases using a power-law function to favor exploitation as knowledge accumulates. This non-linear decay ensures rapid reduction in randomness early on, followed by slower stabilization:

$$\epsilon(k) = 0.95 \cdot \left(1 - \frac{k - 1000}{k_{\max} - 1000}\right)^p, \tag{9}$$

where k is the current episode number, k_{max} is the maximum episode number, and p is the power-law exponent. By 40,000 episodes, ϵ reaches a minimal value of 5%, signifying a full transition to policy exploitation.

The dependence of the exploration coefficient on the episode number is shown in Fig. 2.

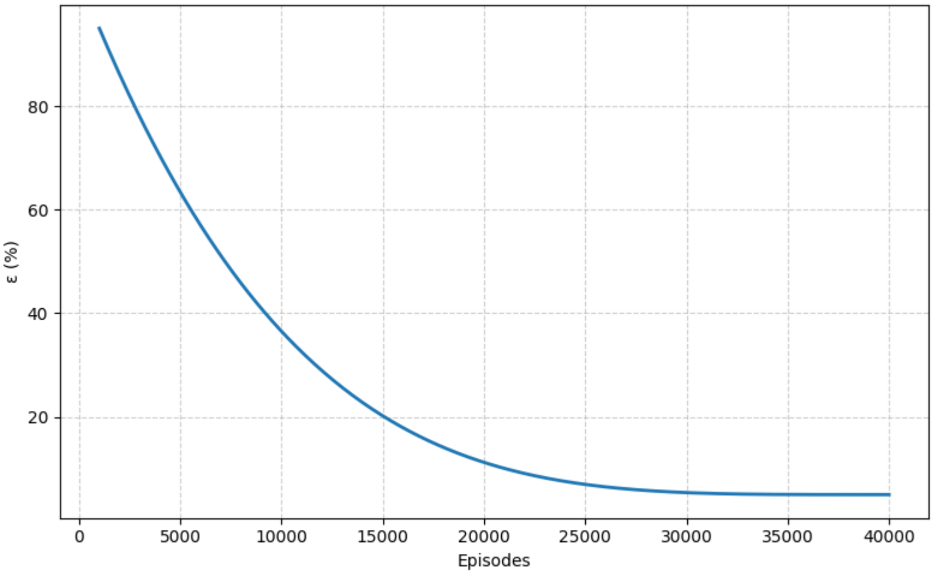

Fig. 2. Nature of the decline in the exploration coefficient

As can be seen from the Fig. 2, at the initial stages there is a rapid decrease in ϵ, after which the rate of change gradually slows down, forming an exponentially similar curve, optimal for combining exploration and use of experience. Such a dependence ensures a rapid decrease in exploration at the initial stages of learning – when the agent actively explores the environment – and a slower decrease after the accumulation of experience, which helps stabilize the process of exploitation of the acquired knowledge.

5. Results of investigating of the enhancing policy generalization

This section presents the quantitative and qualitative results that verify the effectiveness of the proposed policy optimization method: first, identifying the generalization failure under baseline conditions; and second, evaluating the performance of the method under dynamic conditions, followed by the final efficiency assessment.

5.1. Policy over-specialization under fixed conditions

The initial phase of the experiment, where the agent consistently started from the fixed position (270, 200), served as the baseline to establish the typical learning dynamics and identify the limitations imposed by a static environment setup. The training process followed three characteristic phases.

Exploration phase (0–1000 episodes). During this phase, the agent operated under high randomness ($\epsilon \geq 95\%$). Without relying on prior knowledge, behavior was chaotic, leading to immediate collisions. Observations confirmed that the agent could only traverse a short distance, averaging 20–40 conventional units, before terminating the episode. This phase was necessary to populate the experience replay buffer with diverse, fundamental interactions between actions and state changes.

Active exploitation phase (20,000–30,000 episodes). Following the power-law decay of ϵ (Section 4.4.2.), the agent began to actively exploit accumulated knowledge. This interval showed a substantive increase in performance, marked by more stable trajectories and a reduction in collisions. The agent successfully mastered the initial, simpler sections of the track (referred to as the "red segment" in Fig. 3), reaching approximately 820 conventional units by the 29,000th episode. This confirmed the effectiveness of the multi-component reward function R and the DQN methodology in establishing basic orientation skills.

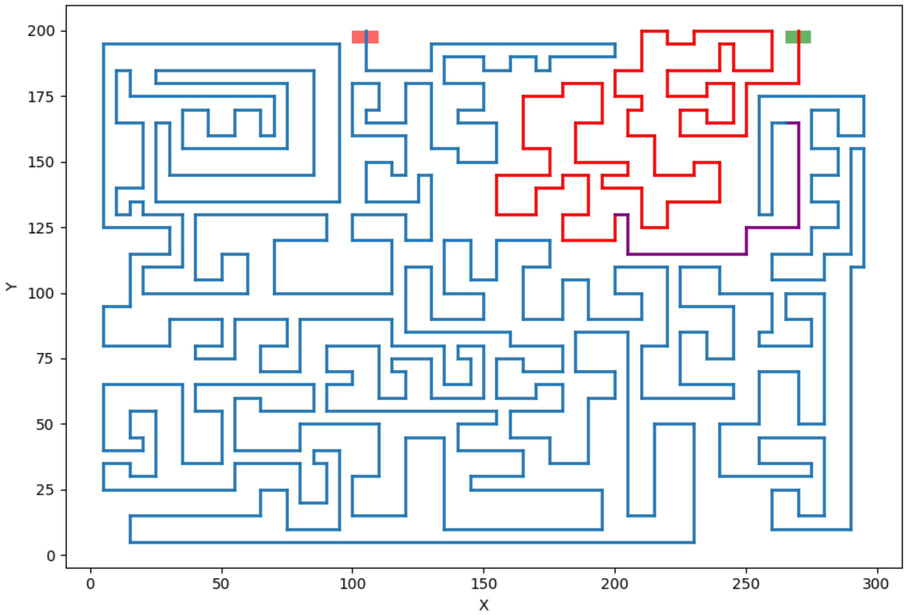

Fig. 3. The environment passed by the agent within 40,000 training episodes

Local optimum and stagnation (40,000 episodes). Despite continued training and the dominance of exploitation (ϵ decreased to 5%), the agent's performance halted. As illustrated in Fig. 3 (showing the environment passed by the agent within 40,000 training episodes), the agent reached a maximum distance of 960 conventional units, completing its movement at the coordinates (265, 165). This stagnation is clearly visible in the learning curve presented in Fig. 4. The agent failed to successfully navigate the subsequent, more geometrically complex ”purple segment” of the track. After 37,000 episodes, no substantial progress was recorded, indicating that the policy had reached a local optimum – a characteristic manifestation of over-specialization. This behavior is typical when the environment has a static structure, causing the agent to optimize its policy for a specific, memorize route rather than developing a generalized navigation strategy.

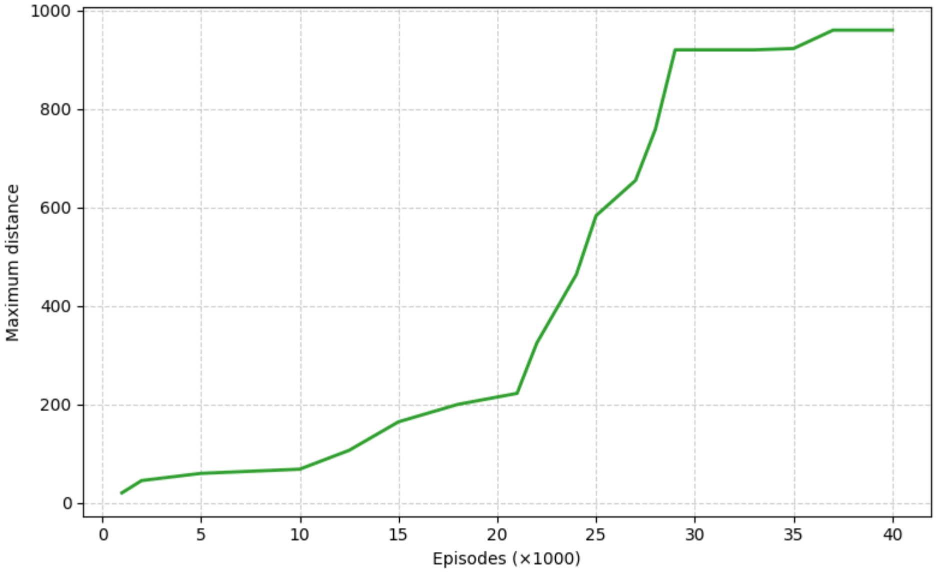

Fig. 4. Agent learning curve

5.2. Verification of policy generalization via dynamic initial conditions

To compel the agent to rely on its general state vector representation S_t, the initial starting position was dynamically varied throughout the experimental protocol.

The results of the modified approach are presented in Fig. 5, which provides a comparative graph of the learning dynamics.

Avoidance of the plateau. In direct contrast to the fixed-start baseline (green curve), the agent trained with dynamic initial conditions (blue curve) did not exhibit a learning plateau in the 20,000–40,000 episode range. The blue curve showed continuous, stable improvement in performance, gradually expanding its effective navigation area.

Quantified generalization. The increase in state diversity forced the agent to learn a generalized control strategy. This success was quantified by the agent achieving the complete, collision-free traversal of the entire complex trajectory between 53,000 and 54,000 episodes. This achievement represents a successful solution to the generalization problem under the defined constraints, validating the application of dynamic initial conditions as an effective method for enhancing DRL policy robustness in complex environments.

Improved robustness. The stability of the blue curve in Fig. 5 confirms that the use of different starting points significantly increased the robustness of the learning process, supporting the hypothesis that modulating the environment’s initial state is a powerful tool for generalization.

5.3. Evaluation of policy efficiency and optimization achievement

Following successful generalization, training was extended from 53,000 up to 90,000 episodes. During this phase, the exploration coefficient ϵ was set to a minimal level of 1%. This maximized policy

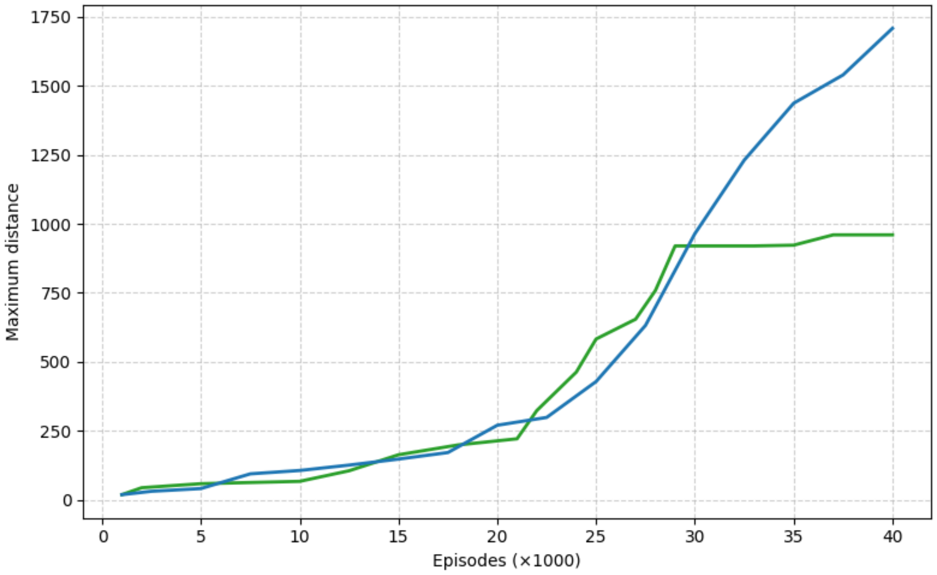

Fig. 5. Comparative graph of the learning dynamics

exploitation, with the continuous time penalty r_{time} component of the reward function driving the primary optimization objective.

The successful traversal path corresponds to the entire route geometry defined in Fig. 3. This complete, optimized trajectory confirms the fulfillment of the generalization criterion achieved in Section 5.2.

The dynamic of this fine-tuning phase is illustrated in Fig. 6, which shows the reduction in traversal completion time as a function of training episodes.

Continued learning led to a substantial improvement in movement efficiency, verifying the optimization goal. At the point of initial successful traversal (approximately 53,000 episodes), the time required to complete the track was approximately 11.0 minutes. Following optimization up to 90,000 episodes, the completion time was reduced to approximately 6.2 minutes (Fig. 6). This reduction of over 43% confirms that the total time required for the agent to complete the trajectory was reduced by almost half.

This time reduction was achieved while maintaining high maneuvering accuracy and stability. The consistency of the average reward and the smoothness of the resulting trajectories confirmed that the policy had reached a state of stable and globally optimal convergence for both safety and speed, justifying the conclusion of the training process.

6. Discussion of results of the enhancing policy generalization

The experimental outcomes validate the scientific problem addressed: the successful mitigation of policy over-specialization in DQN agents for autonomous control.

The baseline experiment, utilizing a fixed starting position, clearly demonstrated the agent's vulnerability to over-specialization, resulting in a performance plateau at 960 conventional units (Fig. 4). This behavior is directly attributable to the environmental setup failing to satisfy the generalization optimization criterion. In static training environments, the DRL agent primarily learns a sequence of high-value actions specific to the single trajectory originating from the fixed start point. The agent, in essence, optimizes a ”rote memory” solution rather than a generalized functional mapping from state space S_t to action values $Q^*(s,a)$. When the agent reached the geometrically complex ”purple segment” (Fig. 3), the pre-learned sequence became invalid. The policy failed because the necessary control actions deviated too sharply from the established

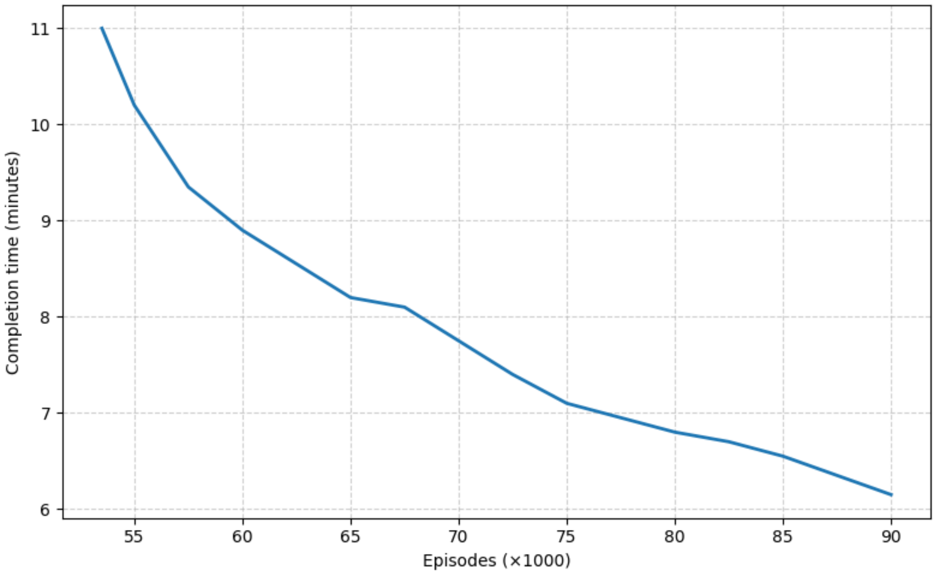

Fig. 6. Reduction in track completion time during the optimization phase

pattern, confirming that the agent was maximizing rewards within a localized, narrow subspace of the environment. This stagnation empirically aligns with the generalization gap problem, which reports performance degradation of DRL agents when exposed to unseen configurations.

The most significant finding is the successful verification of the policy generalization optimization criterion: the introduction of dynamic initial conditions. By compelling the agent to commence episodes from various points along the track, we effectively injected necessary stochasticity into the initial state distribution. This training protocol forced the agent to decouple its decision-making from the specific starting position and rely exclusively on the instantaneous state vector S_t to determine the optimal action. This mechanism forced the deep neural network to learn a universal, state-dependent control policy capable of adapting to varying geometric contexts, irrespective of its history.

The continuous, stable improvement observed in the dynamic learning curve (Fig. 5, blue curve), culminating in complete track traversal between 53,000 and 54,000 episodes, provides quantitative proof of the method's effectiveness. This transition from a limited, specialized policy (stagnation at 960 units) to a universal one (completion of 6845 units) demonstrates a practical solution to a major hurdle in DRL research and establishes the developed method as a viable technique for enhancing generalization.

The final stage of training validated the safety and efficiency criteria embedded in the adaptive reward function (Section 4.2.1.). The multi-component reward system proved effective in guiding policy refinement. The use of the large collision penalty r_{col} and the intermediate checkpoint reward r_{cp} ensured the rapid and stable acquisition of basic navigation and safety skills. This structure addressed the inherent problem of sparse reward, a common challenge in complex navigation tasks [14, 15]. Once generalization was achieved (after 54,000 episodes), the continuous, small penalty r_{time} became the dominant gradient shaping the policy. This fine-tuning incentive drove the agent to reduce the path length and minimize the duration of the episode, leading to a reduction in traversal time by almost half compared to the initial successful run (Section 5.3). This outcome confirms that a carefully balanced, hierarchical reward system is not only necessary for safety and learning stability but is also highly effective for achieving complex efficiency objectives.

The overall stability of the DQN algorithm throughout both phases of learning aligns with research indicating DQN's robustness to reward variations and its strong performance when using discrete action spaces and sparse sensory data.

The successful empirical verification of the generalization method opens clear pathways for future research aimed at deployment within real-world CPS.

The future step is the testing of this generalized policy in high-fidelity environments incorporating domain randomization (e.g., varying traffic, lighting, and friction) to test resilience under dynamic uncertainty. This must be coupled with investigations into integrating the policy onto FPGA-based embedded architectures to ensure the achieved speed and stability are maintained during real-time inference in resource-constrained vehicular CPS [11].

Future work should explore extending the dynamic modulation concept beyond just the starting position to include other environmental parameters, such as dynamically generated, non-repeating obstacle patterns or stochastic sensor noise. This would further enhance the policy's robustness against real-world uncertainties.

The optimized policy could serve as a strong baseline for transfer learning or for integration into hybrid DRL architectures (e.g., combining DQN with policy-gradient methods) to explore benefits in continuous control settings.

Conclusions

This study addressed the problem of policy generalization in DRL for autonomous vehicle control by developing and validating policy optimization method. The findings confirm the effectiveness of modulating environmental initial conditions to counter over-specialization.

DQN policy optimization method was formalized and implemented, defined by two primary criteria: the adaptive reward function for safety and convergence, and the dynamic initial conditions for policy generalization. The hierarchical design of the reward function (integrating collision penalties, checkpoint incentives and a continuous time penalty) maintained learning stability and provided the necessary gradient for subsequent optimization. This structure allowed for the separation of learning phases, ensuring that safety constraints were acquired first, followed by speed optimization.

The effectiveness of the developed method was empirically verified by comparison with a static baseline. The fixed-start training protocol resulted in policy over-specialization, reaching a local optimum that halted performance at approximately 960 conventional units after 40,000 episodes. The implementation of the dynamic initial conditions criterion forcing the agent to learn a generalized control strategy. This achievement was quantitatively demonstrated by the agent attaining stable learning dynamics and achieving full, collision-free track traversal (6845 conventional units) between 53,000 and 54,000 episodes. Subsequent refinement of the generalized policy further increased navigation efficiency, resulting in a reduction of the total traversal time by nearly half.

Acknowledgements

The research was implemented within the National Research Foundation of Ukraine project #2023.04/0077 "Drone for water sampling".

References

[1] E. Figetakis, Y. Bello, A. Refaey, and A. Shami, "Decentralized semantic traffic control in AVs using RL and DQN for dynamic roadblocks," 2024. [Online]. Available: https://doi.org/10.48550/arXiv.2406.18741

[2] K. B. Ravi, S. Ibrahim, T. Victor, M. Patrick, A. A. A. Sallab, Y. Senthil, and P. Patrick, "Deep reinforcement learning for autonomous driving: A survey," *IEEE Transactions on Intelligent Transportation Systems*, vol. 23, no. 6, pp. 4909–4926, 2022. https://doi.org/10.1109/TITS.2021.3054625.

[3] Y. Albrekht and A. Pysarenko, "Exploring the power of heterogeneous uav swarms through reinforcement learning," *Technology audit and production reserves*, vol. 6, no. 2(74), p. 6–10, Dec. 2023. https://doi.org/10.15587/2706-5448.2023.293063.

[4] S. Ibrahim, M. Mostafa, A. Jnadi, H. Salloum, and P. Osinenko, "Comprehensive overview of reward engineering and shaping in advancing reinforcement learning applications," 2024. https://doi.org/10.48550/arXiv.2408.10215.

[5] V. Mnih, K. Kavukcuoglu, D. Silver, A. A. Rusu, J. Veness, M. G. Bellemare, A. Graves, M. Riedmiller, A. K. Fidjeland, G. Ostrovski, S. Petersen, C. Beattie, A. Sadik, I. Antonoglou, H. King, D. Kumaran, D. Wierstra, S. Legg, and D. Hassabis, "Human-level control through deep reinforcement learning," *Nature*, vol. 518, no. 7540, pp. 529–533, feb 2015. https://doi.org/10.1038/nature14236.

[6] J. Escobar-Naranjo, G. Caiza, P. Ayala, E. Jordan, C. A. Garcia, and M. V. Garcia, "Autonomous navigation of robots: Optimization with dqn," *Applied Sciences*, vol. 13, no. 12, 2023. https://doi.org/10.3390/app13127202.

[7] M. A. Alohali, H. Alqahtani, A. Darem, M. Abdullah, Y. Nam, and M. Abouhawwash, "Integrating cyber-physical systems with embedding technology for controlling autonomous vehicle driving," *PeerJ Comput. Sci.*, vol. 11, Jun. 2025. https://doi.org/10.7717/peerj-cs.2823.

[8] P. Henderson, R. Islam, P. Bachman, J. Pineau, D. Precup, and D. Meger, "Deep reinforcement learning that matters," 2019. [Online]. Available: https://doi.org/10.48550/arXiv.1709.06560

[9] S. Zhang and R. S. Sutton, "A deeper look at experience replay," 2018. [Online]. Available: https://doi.org/10.48550/arXiv.1712.01275

[10] T. Pohlen, B. Piot, T. Hester, M. G. Azar, D. Horgan, D. Budden, G. Barth-Maron, H. van Hasselt, J. Quan, M. Večerík, M. Hessel, R. Munos, and O. Pietquin, "Observe and look further: Achieving consistent performance on atari," 2018. [Online]. Available: https://doi.org/10.48550/arXiv.1805.11593

[11] A. Khlifi, M. Othmani, and M. Kherallah, "A novel approach to autonomous driving using double deep q-network-bsed deep reinforcement learning," *World Electric Vehicle Journal*, vol. 16, no. 3, 2025. https://doi.org/10.3390/wevj16030138.

[12] P. Czechowski, B. Kawa, M. Sakhai, and M. Wielgosz, "Deep reinforcement and il for autonomous driving: A review in the carla simulation environment," *Applied Sciences*, vol. 15, no. 16, 2025. https://doi.org/10.3390/app15168972.

[13] L. Ge, X. Zhou, and Y. Li, "Designing reward functions using active preference learning for reinforcement learning in autonomous driving navigation," *Applied Sciences*, vol. 14, no. 11, 2024. https://doi.org/10.3390/app14114845.

[14] R. Audinys, Z. Slikas, J. Radkevicius, M. Sutas, and A. Ostreika, "Deep reinforcement learning for a self-driving vehicle operating solely on visual information," *Electronics*, vol. 14, no. 5, 2025. https://doi.org/10.3390/electronics14050825.

[15] A. Trott, S. Zheng, C. Xiong, and R. Socher, "Keeping your distance: Solving sparse reward tasks using self-balancing shaped rewards," 2019. [Online]. Available: https://doi.org/10.48550/arXiv.1911.01417

УДК 004.8

МЕТОД ОПТИМІЗАЦІЇ ПОЛІТИКИ ГЛИБОКОГО Q-НАВЧАННЯ ДЛЯ ВДОСКОНАЛЕННЯ УЗАГАЛЬНЕННЯ В КЕРУВАННІ АВТОНОМНИМИ ТРАНСПОРТНИМИ ЗАСОБАМИ

Михайло Драган
https://orcid.org/0009-0002-5583-2907

Андрій Писаренко
https://orcid.org/0000-0001-7947-218X

Національний технічний університет України
«Київський політехнічний інститут імені Ігоря Сікорського», Київ, Україна

Розроблення політик автономного керування транспортними засобами на основі глибокого навчання з підкріпленням є однією з основних технічних задач для кіберфізичних систем, яка суттєво обмежується високою розмірністю простору станів, притаманною алгоритмічною нестабільністю та поширеним ризиком надмірного перенавчання, що обмежує можливість застосування політик узагальнення до реальних сценаріїв. Об'єктом цього дослідження є ітеративний процес формування ефективної політики керування в імітаційному середовищі, тоді як предмет дослідження зосереджується на вивченні впливу спеціалізованих функцій винагороди та початкових умов навчання на збіжність політики та здатність до узагальнення. Метою дослідження є розроблення та емпірична оцінка методу оптимізації політики глибокого Q-навчання, який використовує динамічні початкові умови для пом'якшення надмірної спеціалізації та досягнення стійкого і оптимального адаптивного керування.

Розроблений метод формалізує два критерії оптимізації. По-перше, адаптивна функція винагороди слугує критерієм безпеки та збіжності, яка визначається ієрархічно з великими штрафами за зіткнення, середніми стимулами за проходження контрольних точок та постійними невеликими штрафами за витрачений час для підвищення ефективності руху. По-друге, механізм динамічних початкових умов діє як критерій політики узагальнення, призначений для введення необхідної стохастичності в розподіл станів. Агент моделюється як транспортний засіб, оснащений системою з восьми датчиків, що забезпечують покриття у 360°, який приймає рішення з семи варіантів дискретного простору дій. Його десятивимірний вектор стану інтегрує нормалізовані покази датчиків відстані з нормалізованими динамічними характеристиками, включаючи швидкість і кутову похибку.

Емпіричні дослідження підтвердили вразливість політики в базових умовах фіксованого старту, де агент продемонстрував надмірну спеціалізацію і застряг на відстані приблизно 960 умовних одиниць після 40 000 епізодів. Подальше застосування динамічних початкових умов успішно вирішило цю проблему. Змушуючи агента покладатися на узагальнене відображення стану замість того, щоб покладатися на історію проходження траєкторії, цей підхід успішно подолав плато навчання, дозволивши агенту досягти повного проходження траєкторії без зіткнень у проміжку між 53 000 і 54 000 епізодами. Остаточна оптимізація, зумовлена штрафами, скоротила загальний час проходження траси майже наполовину. Ці експериментальні дослідження підтверджують цінність методу у створенні надійних, стабільних та ефективних політик керування, придатних для інтеграції в автономні транспортні кіберфізичні системи.

Ключові слова: глибоке Q-навчання, автономний транспортний засіб, політика узагальнення, функція винагороди, динамічні початкові умови, кіберфізичні системи.