Parallel, Angular and Perpendicular Parking for Self-Driving Cars using Deep Reinforcement Learning

Bruno Sousa
Industrial Electronics Dept.
University of Minho
Guimarães, Portugal
a76402@alunos.uminho.pt

Tiago Ribeiro
Industrial Electronics Dept.
ALGORITMI Center
University of Minho
Guimarães, Portugal
id9402@alunos.uminho.pt

Joana Coelho
Mechanical Engineering Dept.
CMEMS UMinho
University of Minho
Guimarães, Portugal
id8667@alunos.uminho.pt

Gil Lopes
Communication Sciences and
Information Technologies Dept
University of Maia
Maia, Portugal
alopes@ismai.pt

A. Fernando Ribeiro
Industrial Electronics Dept.
ALGORITMI Center
University of Minho
Guimarães, Portugal
fernando@dei.uminho.pt

Abstract—**The progress in creating a fully autonomous self-driving car has steadily increased in recent decades. Consequently, autonomous parking has been a well-researched field since every driving trip must end with a parking manoeuvre. In recent years, with the current successes in reinforcement learning, the concept of applying it to solve the autonomous parking problem has been more and more explored. A vehicle equipped with a complete autonomous parking system must perform three types of parking: perpendicular, angular and parallel parking. Autonomous parking systems control the steering angle and the vehicle speed by considering the surrounding space conditions to ensure collision-free motion within the available space. This paper presents an approach to the problem of autonomous parking using Reinforcement Learning, more precisely, Deep Deterministic Policy Gradient. This approach proved to be capable of parking in a variety of different environments for the three parking manoeuvres.**

Keywords—Artificial Intelligence, Machine Learning, Reinforcement Learning, Autonomous parking, DDPG

I. Introduction

On the final stage of every driving trip, the vehicle must be parked in a suitable location. There are three different types of parking that self-driving vehicles must perform, parallel, angular and perpendicular parking. Initially, the autonomous car must know or perceive the kind of parking spot. For each type of parking, the vehicle must be aware of the parking spot position. However, it must prioritize safety by dynamically avoiding any obstacles that may appear. It must perform all the necessary manoeuvres to properly place the car between the parking lines to correctly park. In recent years, Reinforcement Learning has been explored to solve the autonomous parking problem, because it uses a trial and error strategy to create a more versatile system that can adapt to a great number of scenarios that traditional controlles can not foresee. In this paper, a Reinforcement Learning method is used to learn a policy that allows the agent to park itself in three different parking spots: parallel parking spot, perpendicular parking spot and angular parking spot. The proposed method uses the Deep Deterministic Policy Gradient (DDPG), consisting of an Actor-Critic, off-policy and model-free method [1]. Reinforcement Learning methods learn the optimal behaviour through a trial-and-error strategy that allows the agent to experience a great variety of scenarios. Thus, the implementation of Reinforcement Learning to autonomous vehicle parking processes makes it possible to generalize parking in various situations, abstracting from external factors that may surprise more traditional methods, creating a more complete and versatile system.

II. Related Work

In recent years, machine learning strategies have been used to solve autonomous driving problems.

Autonomous parking is an essential technology for creating fully autonomous vehicles as any journey ends with a parking manoeuvre. Still, only in recent years, Reinforcement Learning has been used to solve the autonomous parking problem. Most papers that approach this problem using Reinforcement Learning are divided into autonomous perpendicular parking and autonomous parallel parking. Regarding the autonomous perpendicular parking, in Zhang et al. [2] and Junzuo and Qiang [3], a Deep Reinforcement Learning method is used to teach an agent to autonomously park itself. The Deep Reinforcement Learning algorithm implemented in these papers was the Deep Deterministic Policy Gradient. Both methods were able to adapt the movements to various new initial positions and orientations. In [2], a pre-trained with a dataset of many complet parking maneuvers was conducted. Regarding autonomous parallel parking, two papers were published that proposed a model-based Deep Reinforcement Learning algorithm as a motion planner for parallel autonomous parking [4] and [5]. Both papers use a truncated Monte Carlo tree search to evaluate parking states and action selection. Regarding the obstacle avoidance problem, Reinforcement Learning has also been used to solve this problem. Ribeiro et al. [6] proposed two different Q–learning approaches to the problem of autonomous mobile robot obstacle avoidance and in [8] was combined Supervised and Reinforcemente Learning methods to detecte and avoid roadworks and temporary signs. Since the DDPG demonstrated promising results in [2] and [3], it was implemented in this project. Since

971-1-6654-8217-2/22/$31.00 ©2022 IEEE

traffic signs detection is a fundamental part of all autonomous driving problems, in [7] was applied YOLOv3 and Yolov3_tiny to traffic signs and lights detectionand and recognition.

III. Problem Definition

The proposed task is part of the RoboCup Portuguese Open "Autonomous Driving" competition [9]. In this competition, there is a set known as "Parking Challenges" consisting of different variations of parking the robot must address. This competition has two different types of parking challenges: the parallel parking challenge and the perpendicular parking challenge. The proposed task adds angular parking to the already mentioned competition challenges. The track of this competition was implemented in the simulator CoppeliaSim.

For the three environments, the agent must park itself in the three parking spots, perpendicular, parallel and angular, with and without obstacles. The agent only knows the data from the sensors, its orientation in the current time step, its desired final orientation and its position relative to the parking spot (X and Y coordinates). Figure 1 represents the part of the track where the three parking spots are located. As presented in Figure 1 the agent is a four wheels robot with six proximity sensors.

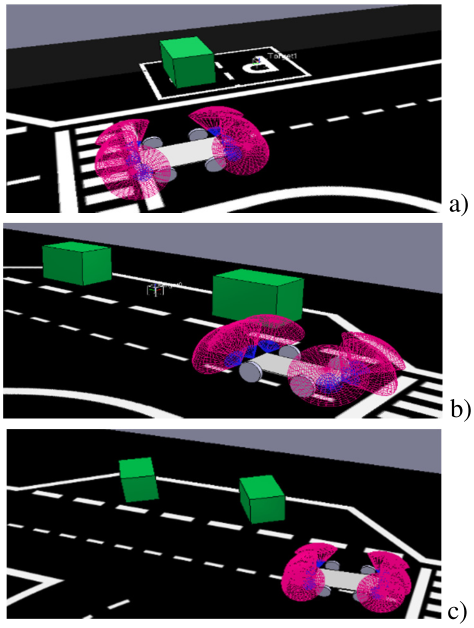

Figure 1- Perpendicular (a), parallel (b) and angular (c) parking spots in CoppeliaSim

IV. Methodologies

A. Deep Deterministic Policy Gradients

DDPG [1] is a model-free, off-policy Reinforcement Learning algorithm that learns by mini-batches of experiences collected by the agent's interaction with the environment stored in the replay buffer. The algorithm comprises four networks, Actor, Critic, Target Actor and Target Critic. The Target Actor is a copy of the Actor, and Target Critic is a copy of the Critic. Each of the neural networks has different functions. The Actor function defines the agent's action and the Critic function judges how good the action was. The target Neural Networks determine the target value (Q-target) used in the temporal difference error equation. The deterministic policy gradient theorem provides the weight update rules for the Actor, and is expressed as:

$$\nabla_{\theta^\mu} J \approx E_{s_t \sim \rho^\beta}[\nabla_{\theta^\mu} Q(s, a|\theta^Q)|_{s=s_t, a=\mu(s_t|\theta^\mu)}] \qquad (1)$$

The Critic network is updated regarding the gradients obtained from the TD error , through the following expression:

$$L(\theta^Q) = E_{s_t \sim \rho^\beta, a_t \sim \beta, r_t \sim E}[(Q(s_t, a_t|\theta^Q) - y_t)^2] \qquad (2)$$

where:

$$y_t = r(s_t, a_t) + \gamma Q(s_{t+1}, \mu(s_{t+1})|\theta^Q) \qquad (3)$$

The target Neural Networks are updated via soft-update , which is expressed as:

$$\theta' = \tau\theta + (1 - \tau)\theta' \qquad (4)$$

B. State-Space

The state-space consists of the six proximity sensors, the robot X and Y coordinates relative to the target parking spot, the robot's current orientation, and the desired final orientation, which is the posture that the robot should have in the end of the manoeuvre. The final orientation is a fixed value set at the beginning of the episode, and it depends on the type of parking spots. The orientation and the final orientation are limited to the range of -180º to 180º.

C. Action–Space

The action is the linear velocity and the steering angle of the vehicle. The steering angle was limited to the range -40º to 40º. Since parking manoeuvres are carried out at low speeds, the maximum speed is limited to 1 m/s.

D. Reward Function

The reward function returns the reward obtained for each action carried out by the agent on the current time step. The reward function is divided into five parts, distance reward, collision reward, orientation reward, incentive reward and success reward. These rewards are presented in table I.

TABLE I. Reward function table

Name	***Reward*** **Equantion**	Decription
Distance Reward	$R_d(t) = e^{-(k_x X^2 + k_y Y^2)}$	Reward calculated at each timestep. It depends on the robot's distance to the parking spot.
Orientation Reward	$R_\theta = e^{-k_\theta(\theta_{target} - \theta)^2}$	Reward calculated at each timestep. It depends on the difference between the agent's current orientation and the final desired orientation.
Incentive Reward	−1	Reward calculated at each timestep to force the agent to finish the manoeuvre as fast as possible.
Collision Reward	-100	Reward given in case of collision.
Success Reward	200	Reward given when the manoeuvre is correctly completed.

On equations of table I, X and Y are the agent coordinates relative to the target, K_X and K_Y are constants that prioritize the movement in one of the axis, it were obtained experimentally and set to 0.06, and 0.04 respectively, θ represents the robot's orientation in the current time step,

θ_{target} represents the final orientation and k_θ is a constant, which was obtained experimentally and set to 40.

The final reward function is a combination of all the previously mentioned rewards and is given by the following equation:

$$R = (1 - w) * R_d + w * R_\theta + R_i + R_c + R_s \quad (5)$$

where w is the prioritize coefficient between the orientation and distance rewards. If w=0, the orientation reward is ignored. If w=1, the distance reward is ignored. With this reward function, every positive average episodic reward would mean that the agent completed the episode. Since some rewards are given at each time step, certain factors, for example, the agent's initial distance to the parking spot, could influence what is considered an acceptable reward for that episode. Thus, up to a distance of 1.5 meters for any scenario where the agent consistently obtained an average reward higher than 190 would be considered acceptable.

E. Neural Network

All neural networks, Actor, Critic, Target Actor and Target Critic, are constituted by two fully connected layers. The first layer has 400 artificial neurons, and the second layer has 300. The activation function of the two hidden layers is ReLU. The Actor and Target Actor have one input layer with 12 neurons, which must have the same numbers as the state space. The output layer has two neurons with the activation function *tanh* to bind the actions. The Critic and Target Critic have two input layers, one for the actions and the other for the state, but the action was only included in the second hidden layer. The output layer has one neuron whose output is the action value.

V. Tests

For all three parking manoeuvres, multiple sections of training were carried out in CoppeliaSim, in a trial-and-error strategy, in the search for the optimal hyperparameters. Table II presentes the set of hyperparameters and the training specifications that obtained the best results and were used in the final training.

TABLE II. HYPERPARAMETERS AND THE TRAINING SPECIFICATIONS

Parameters	*Value*	Parameters	*Value*
Actor Learning Rate	0.0001	Critic Learning Rate	0.0002
Replay Buffer size	30000	Training batch size	512
Reward Decay	0.995	Soft-update parameters	0.01
Nº of training rounds	3000	Max step per rounds	300

A. Perpendicular Parking

To create a more versatile system to learn how to perpendicular park, the agent's initial position would randomly change between both sides of the parking spot. In both sides, a small variation was also added to the initial position of the agent. The agent's initial orientation was set to 0 degrees at the left of the parking spot and 180 degrees at the right side. This orientation would suffer a variation of -10º to 10º. At the beginning of the episode, the agent's distance to the parking spot could vary between 1.2 to 2 meters. The training lasted 3000 episodes, and Figure 2 presents its progress in terms of average reward.

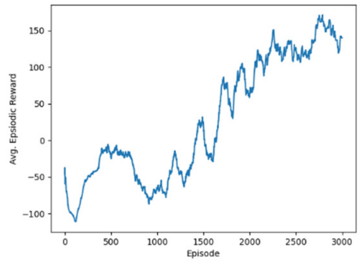

Figure 2–Agent's training progress for the final training of the perpendicular parking challenge

As shown in Figure 2, the agent reaches its best performance around episode number 2800 with an average episodic reward of 180. After this episode, the agent's performance decreased, but it could still park. Four tests were conducted to prove the system's versatility in this manoeuvre. The set of weights at best performance are used in the remaining tests.

1) First test: The agent was tested in two environments, one without obstacles and other with obstacles. The agent's initial position would change between the left and the right side of the parking spot. This test lasted 50 episodes, and the progress is presented in *Figure 3*.

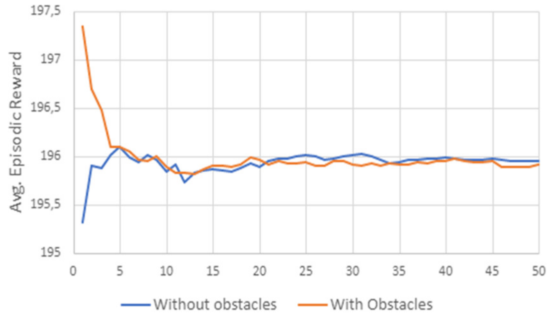

Figure 3–Test with and without obstacles of the perpendicular parking challenge

Figure 3 shows that for all 50 episodes, the average episodic reward is around 196 in both environments. The initial episodes present a higher average reward for the environment with obstacles than for the environment without obstacles. This happened because the initial agent position was randomly changed between 1.2 meters to 2 meters of the parking spot. In those episodes, the agent was further away from the parking spot in the obstacle-free environment than in the obstacle environment, leading to faster completion of the manoeuvre in the obstacle environment and consequently a better reward. It was already expected for the average reward to be similar in both environments because an obstacle should not significantly impact this manoeuvre since the movement is identical for both environment.

2) Second test: The purpose was to check the maximum distance between the agent and the parking spot where the agent could still park. The initial distance between the agent and the parking spot was set to 1 meter, and it was increased 0.5 meters every time the agent completed the test. For every length tested, the initial agent position would randomly change between both sides of the parking spot. For every distance tested, the test lasted 50 episodes, and the progress is presented in *Figure 4*.

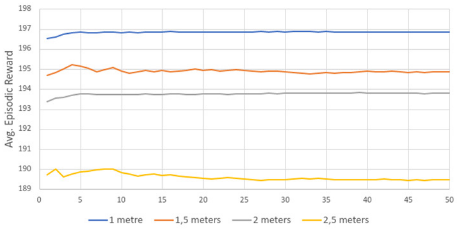

Figure 4–Distance test of the perpendicular parking challenge

Figure 4 presents the average episodic reward for the distances: 1 meter, 1.5 meters, 2 meters and 2.5 meters. The agent successfully parked itself at all distances with an average reward of 197, 194.9, 193.8 and 189.5, respectively. As expected, the average episodic reward decreases with distance since the longer the agent takes to complete the task, the lower the overall reward is. The agent was also tested for the 3 meters distance, but it could not park.

3) *Third Test:* The goal was to determine the maximum angle of the agent's initial orientation the agent would still be able to park. The initial agent position was fixed to 2 meters from the parking spot and from both sides of it. The initial agent orientation was set to 0 degrees or 180 degrees, depending on the side of the parking spot. Every time the agent completed the test, the orientation was increased by 10 degrees. For every orientation tested, the test lasted 50 episodes, and the progress is shown in *Figure 5*. Table III presents the average episodic reward for all the carried out tests.

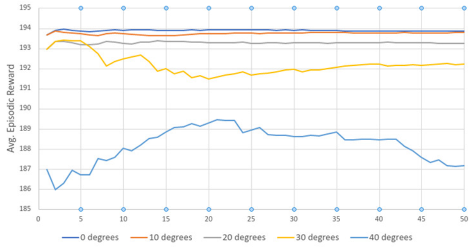

Figure 5–Orientation test of the perpendicular parking challenge

TABLE III. RESULTS OBTAINED IN THE THIRD TEST OF THE PERPENDICULAR PARKING

Orientations	Left Side	Right Side	Avg. Reward
Orientation A	0º	180º	193.9
Orientation B	-10º or 10º	170º or 190º	193.8
Orientation C	-20º or 20º	160º or 200º	193.2
Orientation D	-30º or 30º	150º or 210º	192.2
Orientation E	-40º or 40º	140º or 220º	187.2

As presented in Figure 5, the average reward obtained in the first two orientations was similar. That was expected since, during the training phase, the variation applied to the initial orientation of the agent was ±10 degrees. In other words, the agent already had experienced those situations. Although the test with orientation C was a new scenario for the agent, it could still obtain a similar average reward from the previous two cases, proving that the agent does not memorize movements and can adapt to different variations of the problem. For the tests with the orientations D and E, the agent is still able to park. However, for orientations with a higher variation, the agent could not complete the parking process.

4) *Fourth Test:* To verify whether the agent could park in different parking spots, the parking spot and the obstacle positions would randomly alternate between the two possible positions. For comparison, two more tests were carried out, one always on the left parking spot and another always on the right parking spot. The agent's initial position would vary from left to right for all the three tests. All the three tests lasted 50 episodes, and the progress is presented in *Figure 6*.

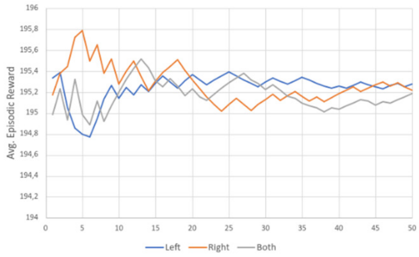

Figure 6–Parking spot variation test of the perpendicular parking challenge

The first test carried out was for the left parking spot. This was the parking spot where the agent was trained, so it was expected to complete all 50 episodes successfully. As shown in Figure 6, the average reward for the 50 episodes obtained by the agent in this test was 195.3.

The second test was intended for the second parking spot, which was a new scenario since the agent was not trained for this parking spot. The agent adapted and completed all 50 episodes with an average episodic reward of 195.2, as shown in Figure 6. For the last test, the agent was tested in both parking spots, and it was able to complete with success all 50 episodes with an average reward of 195.15.

As presented in Figure 6, the average episodic reward obtained by the agent in all the three referred tests was extremely close. It demonstrates that the agent could generalize the movement rather than memorize it and can adapt to new situations.

B. Angular Parking

For the angular parking, the initial agent position was fixed to the left side of the parking spot. The initial agent orientation could vary between 170 degrees and 190 degrees, and two obstacles were placed in both sides of the parking spot, as shown in Figure 1. In every episode, a slight initial position variation was added between 1.2 meters to 2 meters. The training lasted 3000 episodes, and the progress is presented in Figure 7.

As shown in Figure 7, the agent reaches its best performance around the 2000th episode with an average episodic reward of 190 and keeps a similar performance until the end of training. To prove the versatility of this system three tests were carried out.

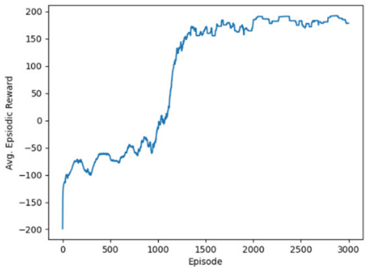

Figure 7–Agent's training progress for the final training of the angular parking challenge

1) First test: The same conditions used in the training phase were set. The agent was placed on the left side of the parking spot, and its initial position would vary between 1.2 meters and 2 meters from the parking spot. This test intends to evaluate the difference in performance from the environment with and without obstacles. This test lasted 50 episodes, and the progress is presented in *Figure 8*.

Figure 8 shows that the average reward for all 50 episodes is 195.6 without obstacles and 195.2 with obstacles, showing that obstacles affect the agent performance.

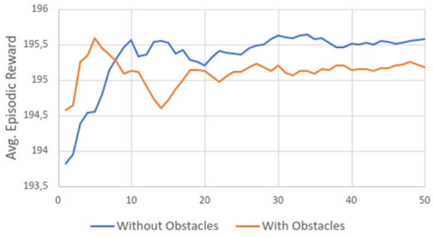

Figure 8– Test with and without obstacles of the angular parking challenge

2) Second Test: The agent was placed in a fixed position for all 50 episodes. This position started from a meter distance from the parking spot and would increase half a meter every test the agent could successfully complete. The test lasted 50 episodes for every distance tested, and the progress is presented in *Figure 9*.

The agent has successfully completed all tests up to 3 meters, and obtained an average episodic reward of 198.4 for 1 meter distance, 196.5 for 1.5 meters distance, 194.2 for 2 meters distance, 190.9 for 2.5 meters distance and 183.3 for 3 meters distance. The agent was also tested for distances superior to 3 meters but could not park.

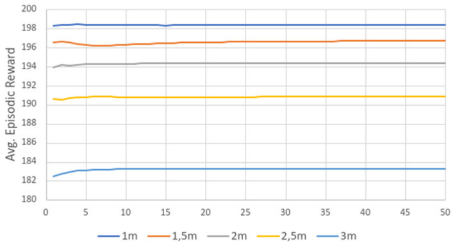

Figure 9–Distance test of the angular parking challenge

3) Third test: The difference in performance for different initial orientations was evaluated. The agent was placed in a fixed position at 1.5 meters from the target. The initial agent orientation was set to 180 degrees, and 10 degrees were randomly added or subtracted for every test the agent completed. For every orientation, the test lasted 50 episodes and the progress is presented in *Figure 10*.

Table IV presents the average episodic reward for all the carried out tests.

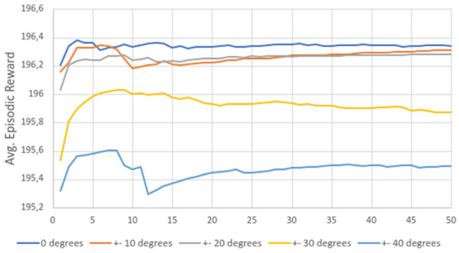

Figure 10–Orientation test of the angular parking challenge

TABLE IV. RESULTS OBTAINED IN THE THIRD TEST OF THE ANGULAR PARKING

Orientations	Right Side	Avg. Reward
Orientation A	180°	196.35
Orientation B	170° or 190°	196.3
Orientation C	160° or 200°	196.28
Orientation D	150° or 210°	195.87
Orientation E	140° or 220°	195.5
Orientation F	130° or 230°	170.4

The agent presented a similar performance for orientations A to E, despite only experiencing orientations A and B in the training phase. This demonstrates the adaptability of the agent for new environments. The agent completed all 50 episodes for orientation F, but the performance was too unstable to be accepted. Lastly, the agent was also tested in environments where it was randomly added and subtracted 60, 70, 80 and 90 degrees. However, the agent was only able to complete the environments where those orientations were subtracted.

C. Parallel Parking

For the parallel parking training, the same setup was used as the angular manoeuvre. The agent was placed on the left side of the parking spot. In every episode, a small variation was added to the agent's initial position, which would vary it between 1.2 meters to 2 meters. The initial agent orientation was set at 180 degrees, and a slight variation of ±10 degrees was added. Lastly, two obstacles were positioned on both sides of the parking spot, as shown in Figure 1. The training lasted 3000 episodes, and the progress is presented in Figure 11.

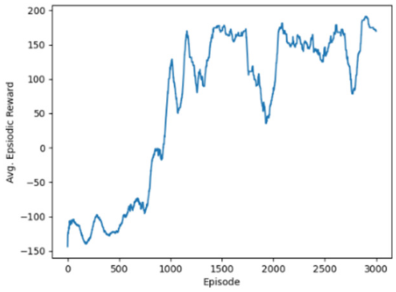

Figure 11–Agent's training progress for the final training of the parallel parking challenge

Figure 11 shows that the agent performs best around episode 2900 with an average episodic reward of 190. Since there is no significant variation between the best performance and the performance at the end of the episode, it was withdrawn the weights of the end of the episode because it experienced more episodes. To verify the versatility of the system, the same three tests as previous parking methods were carried out.

1) First test: The agent was placed on the left side of the parking spot, and its initial position would vary between 1.2 meters and 2 meters from the parking spot. The intention was to evaluate the performance difference with and without obstacles. This test lasted 50 episodes, and the progress is presented in *Figure 12*.

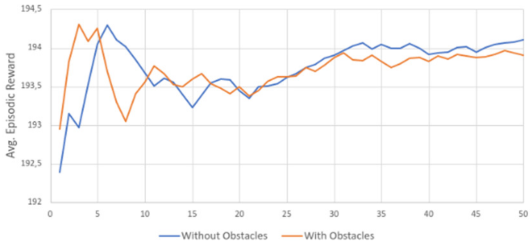

Figure 12– Test with and without obstacles of the parallel parking challenge

Figure 12 shows that the average episodic reward is 194.1 for the environment without obstacles and 193.9 for the environment with obstacles, showing that the obstacles affect agent performance.

2) Second test: This test intends to evaluated the difference in performance for distinct distances form the parking spot. The agent was placed 1 meter to the left of the parking spot. The agent position was increased 0.5 meters every time the environment was successfully completed. For every distance evaluated, the test lasted 50 episodes and the progress is presented in *Figure 13*.

The agent successfully completed all tests up to 3 meters. The agent obtained an average episodic reward of 198 for the 1 meter distance, 196.5 for 1.5 meters distance, 194.1 for 2 meters distance, 189.8 for 2.5 meters distance and 182.3 for 3 meters distance. Lastly, the agent was tested for 3.5 meters distance, but despite completing all 50 episodes, the movement was too unstable to be accepted.

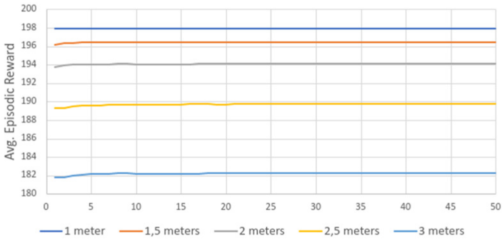

Figure 13- Distance test of the parallel parking challenge

3) Third test: The difference in performance for different orientations was evaluated in this test. The agent was placed 1.5 meters to the left of the parking spot, with an orientation of 180 degrees. For every successful test, 10 degrees were randomly added and subtracted to the initial orientation. For every orientation tested, the test lasted 50 episodes, and the progress is presented in *Figure 14*.

In table V the average episodic reward for all the carried out tests are presented.

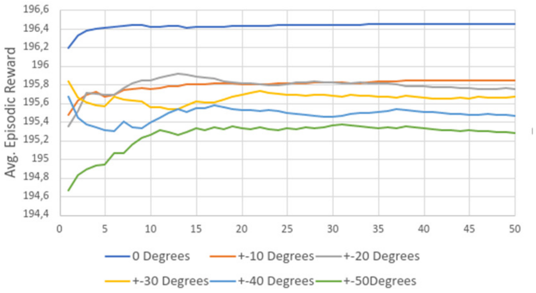

Figure 14–Orientation test of the parallel parking challenge

TABLE V. RESULTS OBTAINED IN THE THIRD TEST OF THE PARALLEL PARKING

Orientations	Right Side	Avg. Reward
Orientation A	180°	196.5
Orientation B	170° or 190°	196.1
Orientation C	160° or 200°	195.75
Orientation D	150° or 210°	195.66
Orientation E	140° or 220°	195.46
Orientation F	130° or 230°	195.27

The agent obtained a similar performance for all six orientations presented in Figure 14 and table V, demonstrating that the agent shows identical performance in new environments. The agent was also tested in scenarios where it was randomly added and subtracted 60 and 70 degrees. Still, the agent was only able to complete the environments where it was subtracted those orientations.

VI. RESULTS DISCUSSION

Regarding the perpendicular parking manoeuvre, the agent can park for all situations presented in table VI. For most environments tested with a starting position at the left of the parking spot, the agent presents an almost ideal parking. For the environments tested with a starting position at the right side of the parking spot, the agent demonstrates more difficulties with higher variations, presenting a final position further off-centre to environments with higher variations added to the initial position. Due to the reward function implemented, the agent shows a lack of concern for the final orientation and position. This happens because, in the reward function implemented, the manoeuvre was only accepted for an orientation error lower than 10 degrees and a distance inferior to 5 cm. Still, there were no discounts implemented in the reward considering these deviations. Consequently, the difference in the reward obtained by the agent for all the scenarios that respect the distance and orientation requirements was extremely small.

TABLE VI. AGENT LIMITATIONS FOR THE PERPENDICULAR PARKING MANOEUVRE

Limitations	*Value*
Side of the parking spot	Left and Right
Range of the initial orientations	-40 degrees to 40 degrees
Maximum distance to the parking spot	2.5 meters

Concerning the angular parking manoeuvre, the agent learns to park for all scenarios presented in table VII, but with the increased variations, the agent's behaviour becomes increasingly unstable with more significant fluctuations in the steering. The model created was only able to park either from the left side or the right side but could not converge into a solution that could park from both sides simultaneously. Similarly to the perpendicular parking manoeuvre, a more off-centre position of the parking spot was verified with higher variations. However, the agent learned a policy that enables it to park for the trained environments and adapt to new environments.

TABLE VII. AGENT LIMITATIONS FOR THE ANGULAR PARKING MANEUVER

Limitations	*Value*
Side of the parking spot	Right

Limitations	***Value***
Range of the initial orientations	130 degrees to 230 degrees
Maximum distance to the parking spot	3 meters
Minimum distance between obstacles	90 centimetres

Regarding the parallel parking manoeuvre, the agent learned a policy that enabled it to parallel park for a great variety of different environments but could not park in a environment where the distance between the obstacles is lower than 1.5 meters. The model created was only able to park from one side of the parking spot but not from both simultaneously. Table VIII presents all the different situations in which the agent can park.

TABLE VIII. AGENT LIMITATIONS FOR THE PARALLEL PARKING MANOEUVRE

Limitations	***Value***
Side of the parking spot	Right
Range of the initial orientations	130 degrees to 230 degrees
Maximum distance to the parking spot	3 meters
Minimum distance between obstacles	1.5 meters

VII. CONCLUSIONS

Regarding the perpendicular parking manoeuvre, the agent learned a policy that enables it to park from both sides of the parking spot with a great variety of different positions and orientations. For the situations in which the agent can park, it performs a stable movement with no unnecessary steering. In most cases, the agent ends the episode close to an optimal final position.

Concerning the angular parking manoeuvre, the agent learned a policy that enables it to park in a great variety of different environments. For the trained environments, the agent performs a stable movement with close to no unnecessary steering, but the greater the distance to the parking spot the more unstable the movement became. The agent is only able to park in a parking spot with a distance between obstacles greater than 90 cm. Still, it could not adapt to parking from both sides of the parking spot.

Regarding the parallel parking manoeuvre, the agent learned to park from various initial positions and orientations. The agent performed a stable and direct manoeuvre for the trained environments. Although this strategy completes the manoeuvre efficiently and quickly, it limits the parking spot space to which the agent can park. It was verified that the greater the initial distance and the variation added to the orientation, the closest the final position of the agent was from the second obstacle. This limits the agent's ability to park in a parking spot with a distance between obstacles lower than 1.5 meters. However, it was unable to adapt its movement to park from both sides of the parking spot.

For the three parking manoeuvres, a vast number of training pieces were performed in search of the optimal hyperparameters. Still, only a small set of hyperparameters led to a policy capable of completing the parking manoeuvres. This happens because DDPG suffers from instability in its hyperparameters sensitivity, leading to a tendency to converge to solutions that do not converge or stop learning midway.

Since Proximal Policy Optimization (PPO) is considerably more data-efficient and less sensitive to hyperparameters than DDPG, it could be interesting to test this algorithm and compare the results for the autonomous parking problem.

ACKNOWLEDGMENTS

This work has been supported by FCT—Fundação para a Ciência e Tecnologia within the R&D Units Project Scope: UIDB/00319/2020. In addition, this work has also been funded through a doctoral scholarship from the Portuguese Foundation for Science and Technology (Fundação para a Ciência e a Tecnologia) [grant number SFRH/BD/06944/2020], with funds from the Portuguese Ministry of Science, Technology and Higher Education and the European Social Fund through the Programa Operacional do Capital Humano (POCH).

REFERENCES

[1] Timothy P. Lillicrap, Jonathan J. Hunt, Alexander Pritzel, Nicolas Heess, Tom Erez, Yuval Tassa, David Silver and Daan Wierstra, "Continuous control with deep reinforcement learning", 4th International Conference on Learning Representations, ICLR 2016 - Conference Track Proceedings, 2016.

[2] Peizhi Zhang, Lu Xiong, Zhuoping Yu, Peiyuan Fang, Senwei Yan, Jie Yao and Yi Zhou, "Reinforcement learning-based end-to-end parking for automatic parking system", Sensors (Switzerland), vol. 19 ,September 2019.

[3] Li Junzuo and Long Qiang," An Automatic Parking Model Based on Deep Reinforcement Learning", 2021 J. Phys.: Conf. Ser. 1883 012111.

[4] Shaoyu Song, Hui Chen, Hongwei Sun and Meicen Liu School,``Data efficient reinforcement learning for integrated lateral planning and control in automated parking system``, Sensors (Switzerland), vol. 20, pp. 1–24, dec 2020.

[5] J. Zhang, H. Chen, S. Song and F. Hu, "Reinforcement Learning-Based Motion Planning for Automatic Parking System," in IEEE Access, vol. 8, pp. 154485-154501, 2020, doi: 10.1109/ACCESS.2020.3017770.

[6] T. Ribeiro, F. Gonçalves, I. Garcia, G. Lopes and A. F. Ribeiro, "Q-Learning for Autonomous Mobile Robot Obstacle Avoidance," 2019 IEEE International Conference on Autonomous Robot Systems and Competitions (ICARSC), Porto, Portugal, 2019, pp. 1-7. DOI: 10.1109/ICARSC.2019.8733621

[7] Marques, R.; Ribeiro, T.; Lopes, G. and Ribeiro, A. (2022). YOLOv3: Traffic Signs & Lights Detection and Recognition for Autonomous Driving. In Proceedings of the 14th International Conference on Agents and Artificial Intelligence - Volume 3: ICAART, ISBN 978-989-758-547-0, pages 818-826. DOI: 10.5220/0010914100003116.

[8] Andrade, N.; Ribeiro, T.; Coelho, J.; Lopes, G. and Ribeiro, A. (2022). Combining YOLO and Deep Reinforcement Learning for Autonomous Driving in Public Roadworks Scenarios. In Proceedings of the 14th International Conference on Agents and Artificial Intelligence - Volume 3: ICAART, ISBN 978-989-758-547-0, pages 793-800. DOI: 10.5220/0010913600003116.

[9] SPR, "Robótica 2019 - Rules for Autonomous Driving," tech. rep., 2019

APPENDIX

A) Video of all the tests referred in this paper: https://www.youtube.com/watch?v=Ko_f7AEoNDk&t=9s