Safe parking of a nonholonomic autonomous vehicle by qualitative reasoning

Domen Šoberl[a, b,*], **Jan Lemeire**[c,d,**], **Ruben Spolmink**[d] **and Jure Žabkar**[a]

[a]University of Ljubljana, Faculty of Computer and Information Science
[b]University of Primorska, Faculty of Mathematics, Natural Sciences and Information Technologies
[c]Dept. of Industrial Sciences (INDI), Vrije Universiteit Brussel (VUB), Pleinlaan 2, B-1050 Brussels
[d]Dept. of Electronics and Informatics (ETRO), Vrije Universiteit Brussel (VUB), Pleinlaan 2, B-1050 Brussels

Abstract. Traditional techniques for autonomous driving nonholonomic (car-like) vehicles require precise kinematic models and complex geometric computations of trajectories. Learning such a model through reinforcement learning is highly sample inefficient and thus not always feasible in practice. Moreover, such an approach offers poor explainability. We propose an approach based on qualitative reasoning, where a qualitative model for driving a car-like vehicle is learned over a small set of numerical traces. We define a planning algorithm that is able to interpret the learned qualitative models and quantify the actions to pursue the goal while avoiding collisions. We demonstrate our approach on the problem of reverse parallel car parking. The results show that our qualitative approach is able to deduce an S-shaped trajectory to park the car in one smooth reverse maneuver without the typical backward-forward corrections with negligible error in the final position and orientation.

1 Introduction

Nonholonomic vehicles, which include various types of wheeled robots and autonomous vehicles, are subject to constraints that limit their motion to certain paths. Unlike holonomic systems, which can move freely in any direction, nonholonomic vehicles can only move in specific directions due to their constraints. Parking such vehicles involves finding feasible paths that comply with these motion constraints while achieving precise final positioning.

Traditional techniques for autonomous parking of vehicles rely on a combination of sensors (e.g. ultrasonic sensors, cameras, lidar, and radar) and algorithms for path planning and trajectory generation, along with control systems (e.g. PID controllers and Model Predictive Control) to ensure precise vehicle movement and obstacle avoidance. In practice, autonomous parking systems should also take extra care, when dynamicity is present in the environment (e.g. other moving subjects and/or objects nearby). Such methods require a precise kinematic model and are often computationally complex [13]. Recently, Boyali and Thompson [1] proposed a method for optimal path generation in parking maneuvers using a kinematic car model. Their approach integrates Successive Convexification (SCvx) algorithms and state-triggered constraints to ensure path feasibility and constraint satisfaction in constrained environments. Shahi and Lee [14] introduced a method for autonomous rear parking using Rapidly Exploring Random Trees (RRT) and Model Predictive Control (MPC).

Fundamental geometric methods for generating paths in obstacle-free environments were first studied by Dubins [4]; his paper provides early insights into nonholonomic path planning by studying the shortest paths for car-like vehicles, which can only move forward. Reeds and Shepp [12] addressed also the backward motion of a vehicle. These two studies form the basis for many modern path-planning algorithms used in autonomous vehicles. A basic understanding of motion planning for nonholonomic vehicles is given in Triggs [17].

Alternative approaches use reinforcement learning [19] or fuzzy-based controllers [11] to obtain a good parking strategy, where the vehicle continuously learns from several parking attempts. Reinforcement learning approaches require lots of data and trials, which is not feasible in practice. A recent approach by Moreira [10] explored the application of deep reinforcement learning (DRL) in automated parking. The study focused on training an agent to follow predefined complex paths while avoiding collisions with obstacles.

Commercial autonomous parking systems (APS) can be divided into two types; systems like Bosch's Automated Valet Parking (AVP) also depend on vehicle-to-infrastructure communication to ensure efficiency and safety. For example, Bosch in collaboration with Mercedes-Benz developed an AVP system that enables vehicles to park in predefined parking spots in garages without driver input. These systems exceed the scope of our work. Other brands don't rely on the outside infrastructure: most notably BMW, Audi and Tesla have incorporated APS that use a combination of cameras and ultrasonic sensors to guide the vehicle autonomously. While most of them work well in structured environments like parking garages, real-world scenarios with unpredictable elements (e.g., pedestrians or dynamic obstacles) still present a significant challenge. Vision-based systems, e.g. like those used by Tesla, struggle with low-light conditions, bad weather conditions, and occlusions (e.g., objects blocking sensors). The removal of ultrasonic sensors in some models has also led to inconsistent performance in tight parking spaces.

In this paper, we address the problem of parking a nonholonomic vehicle using qualitative models in combination with qualitative reactive planning [21]. Qualitative models [5, 7, 2, 6] describe the dynamics of a system in qualitative terms such as the directions of change of state variables (increasing, steady or decreasing). These qualitative models can be used in planning and control [15, 9]. We obtain the qualitative model from a small set of numerical traces and

* Corresponding Author. Email: domen.soberl@famnit.upr.si.
** Corresponding Author. Email: jan.lemeire@vub.be.

then use a reactive planning approach to pursue the goal.

This work is part of our endeavor to develop a global learning and planning architecture that can adapt to novel situations in a way that is close to how humans learn. Reverse parallel parking is an interesting challenge to steer this development.

2 The parallel parking challenge

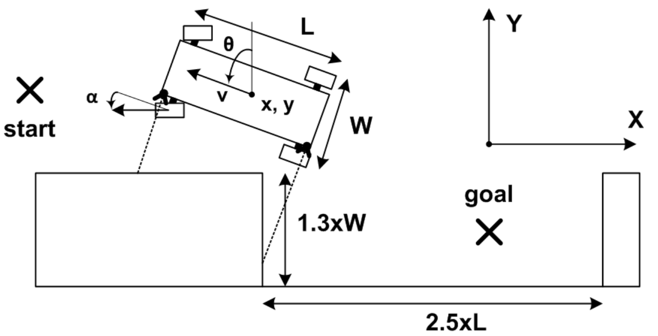

Figure 1. Problem Specification. Here, α is positive and θ is negative. The car width W is 3/5 of the car length L. The car has 2 distance sensors on its left. The distance they measure is shown with dashed lines.

The experimental setup that we used in this paper is shown in Fig. 1. The car starts at the left of the goal position and drives backward towards the parking spot. The initial and the goal positions are marked in the figure with the cross symbol, which depict the center of the car (x, y). The car's orientation θ at the start is 90 degrees (facing left) and should also be 90 degrees when parked. In this experiment, we constrain the speed of the car v to backward driving at a constant speed ($v < 0$), so the car must be parked in one smooth trajectory, with no back-and-forth maneuvers. This way, the actions are simplified only to turning of the steering wheels within $\alpha \in [-30, 30]$. We also place obstacles (other parked cars or walls) in front and behind the parking spot. The obstacle in front is placed at a large enough distance so that the car can perform an S-shaped trajectory in a single maneuver.

There are two distance sensors mounted on the car, one at the front and the other at the back, both on the left side of the car, so that they measure the distance to the nearest wall in the direction perpendicular to the car's orientation (see Fig. 1). The sensor is triggered if the obstacle is closer than the length of the car.

We conducted the experiments in a simulator with a time step of $\Delta t = 40$ milliseconds. The car starts driving backward immediately at a constant speed so that one length of the car is traversed in 20 steps. The speed of turning the steering wheels is 100 degrees per second. Actions therefore only define the direction of turning the steering wheel α, which can be either 0 (no turning), 1 (turning left), or -1 (turning right). The episode stops when the x-position of the car reaches or surpasses the x-position of the parking spot, or when an obstacle is being hit.

3 The numerical model

To simulate the motion of the car, we use a mathematical model similar to the Dubins car model [3], which is often represented as a bicycle model. For an ordinary car, the pairs of parallel wheels are depicted as a single wheel. The car cannot move sideways, and its forward motion is constrained to geometric arcs, as shown in Fig. 2. The future position of the front wheels is determined by the car's current orientation θ, the distance l between the front and the rear wheel, the current forward velocity v of the car, and the steering angle α, which we constrain to $\alpha \in [-30°, 30°]$. We use the following differential equations to model the dynamics of the front wheel:

$$
\begin{aligned}
\dot{\theta} &= v \cdot \frac{\sin(\alpha)}{l/2} \\
\dot{x} &= -v \cdot \sin(\theta + \alpha) \\
\dot{y} &= v \cdot \cos(\theta + \alpha)
\end{aligned}
\tag{1}
$$

After the position of the front wheel is calculated for the next time step, the position of the rare wheel is deduced from the new orientation θ and the length l. The midpoint of the segment l is taken as the current car's position (x, y).

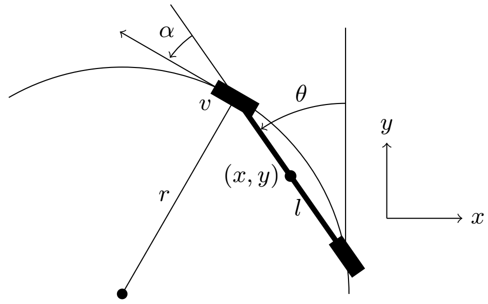

Figure 2. Our mathematical model of a car, which is similar to the Dubins car model.

Differential equations (1) or similar are typically used to model the dynamics of the Dubins car (see, e.g. [8]). However, the Dubins model is constrained to forward motion ($v \geq 0$). The dynamics of moving backward is considerably more complicated. Consider the situation depicted in Fig. 3. The car is oriented towards the left ($\theta = 90°$), drives backwards ($v < 0$), and the steering angle is positive ($\alpha > 0$). The front wheel follows the dynamics of the model (1), which predicts $\dot{y} > 0$, but the rear wheel exhibits the opposite dynamics $\dot{y} < 0$. If such motion is observed long enough, the front wheel will eventually, due to the change in θ, also assume $\dot{y} < 0$. When driving backward, we consider two types of effects: *short-term* effects that describe the immediate dynamics of the front wheel, and *long-term* effects that describe the motion of the back of the car. When parking the car backward, we are interested in the latter.

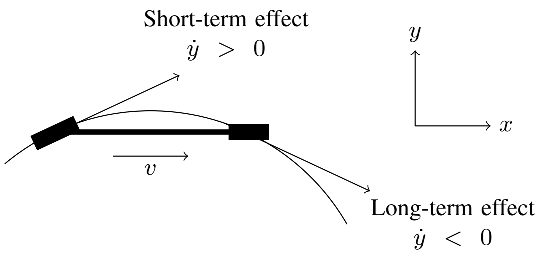

Figure 3. The difference between the *short-term* and the *long-term* action effect.

4 The qualitative model

4.1 A short-term model

The short-term qualitative model can be derived directly from differential equations (1). In this paper we use Q-constraints as defined in

[18]:

$$y = Q^+(x) \quad \text{means} \quad \frac{\partial y}{\partial x} > 0,$$
$$y = Q^-(x) \quad \text{means} \quad \frac{\partial y}{\partial x} < 0.$$

Other functional dependencies may exist, but they are not presumed with the above Q-constraints. We presume that v is constant and $\alpha \in [-30°, 30°]$. Our *short-term* qualitative model for forward driving ($v > 0$) is therefore:

$$\begin{aligned} \dot{\theta} &= Q^+(\alpha), \\ \dot{x} &= \begin{cases} Q^-(\theta+\alpha) & \text{if } -90° \leq (\theta+\alpha) < 90°, \\ Q^+(\theta+\alpha) & \text{otherwise.} \end{cases} \\ \dot{y} &= \begin{cases} Q^-(\theta+\alpha) & \text{if } 0° \leq (\theta+\alpha) < 180°, \\ Q^+(\theta+\alpha) & \text{otherwise.} \end{cases} \end{aligned} \tag{2}$$

And for backward driving ($v < 0$):

$$\begin{aligned} \dot{\theta} &= Q^-(\alpha), \\ \dot{x} &= \begin{cases} Q^+(\theta+\alpha) & \text{if } -90° \leq (\theta+\alpha) < 90°, \\ Q^-(\theta+\alpha) & \text{otherwise.} \end{cases} \\ \dot{y} &= \begin{cases} Q^+(\theta+\alpha) & \text{if } 0° \leq (\theta+\alpha) < 180°, \\ Q^-(\theta+\alpha) & \text{otherwise.} \end{cases} \end{aligned} \tag{3}$$

The interpretation of the above models is as follows. Consider again the short-term effect in scenario from Fig. 3. The orientation of the car is $\theta = 90°$ and $\alpha \in [-30°, 30°]$. Since $(\theta + \alpha) \in [60°, 120°]$, it applies $\dot{y} = Q^+(\theta + \alpha)$. If we are driving slow, so that v approaches 0, it follows from (1) that $\dot{\theta}$ also approaches 0, hence with slow driving, our Q-constraint approximates $\dot{y} = Q^+(\alpha)$, which we interpret as:

> *If the steering angle α increases/decreases and everything else remains constant, the speed $\dot{y}$ also increases/decreases.*

In our scenario, this means that turning the steering wheel *left* increases $\dot{y}$, and turning it *right* decreases $\dot{y}$.

4.2 *A long-term model*

When driving backward, we use the long-term qualitative model. It is easy to see that short-term and long-term effects on $\dot{\theta}$ are the same, hence $\dot{\theta} = Q^-(\alpha)$ for $v < 0$. However, the long-term effects on $\dot{x}$ and $\dot{y}$ are not directly deducible from the mathematical model (1) without considering some additional geometric properties of the car. We therefore decided to learn the long-term model instead of deducing it. We used the method called Padé [18] that learns Q-constraints from numerical samples.

We collected 330 samples that uniformly cover the domain $\theta \times \alpha$, as seen in Fig. 5. For each configuration (θ, α), we measured the changes Δx and Δy, while driving backward ($v < 0$) for long enough to observe the long-term effects. Taking into account the duration Δt of each action, we translated the observed values to $\dot{x}$ and $\dot{y}$. The two outputs from Padé — The first one for $\dot{x}$ and the second one for $\dot{y}$ — are shown in Figure 5. Padé labels each sample with the '+' or the '-' sign, which respectively denote $Q^+(\theta + \alpha)$ and $Q^-(\theta + \alpha)$.

Revisiting again the scenario from Fig. 3, we first identify the qualitative sign belonging to the car's configuration $\theta = 90°$, for some $\alpha > 0$. It is clear from the plots that the long-term effect on $\dot{y}$ of driving backward in this configuration is determined by constraint $\dot{y} = Q^-(\theta + \alpha)$, which means that turning the steering wheel *left* (increasing α) results in decreasing the speed $\dot{y}$ while turning it *right* (decreasing α) results in increasing the speed $\dot{y}$.

Fig. 4 gives and interpretation of short-term and long-term qualitative effects on variables y in different states (α, θ). By turning the wheel, we change the value of alpha either in positive (right arrow) or negative (left arrow) direction. This affects the speed with which the orientation of the car (θ) is changing while driving backward. For a short while (shorter arrow), the sign of y is preserved, but after some time (longer arrow) the sign of y may change.

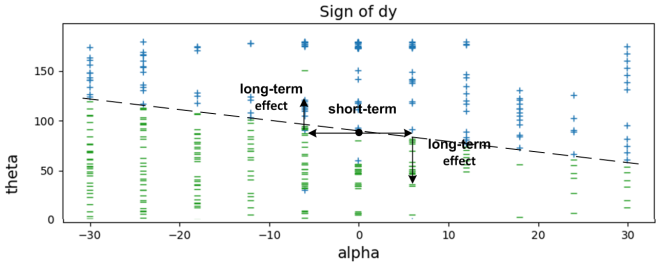

Figure 4. Short-term and long-term action effects observed with the signs of variable y in different states (α, θ).

5 The planning algorithm

To solve a numerical problem using qualitative models, some form of quantification is necessary. A qualitative model can predict which actions will work in the direction towards a goal state, but cannot assert the quantities of actions or decide on their duration. Using a qualitative model we may decide that the value of some output should be increased or decreased, but cannot directly assert the actual rate of change. In our planning algorithm, we tackle this problem by a reactive approach, where the current numerical state is observed multiple times per second, and each time a qualitative action is decided and executed using a small fixed numerical step. By fast consecutive execution of such short actions, the state of the system is controlled dynamically and steered towards the goal direction. In our car parking domain, the speed of turning the front wheels is fixed, so an action merely decides whether — according to the currently observed state — the driver should be turning the steering wheel left or right.

To decide which action should be executed in some specific moment, we consider the current intention of the driver, which could be one of the following two:

- *The goal pursuit mode*. The collision sensors are off and the goal is to park the car to the designated parking spot.
- *The safety mode*. One or more of the collision sensors got triggered. Avoid colliding with the obstacle/wall.

When avoiding collision, the algorithm temporarily ignores the primary goal of parking the car, until the danger of colliding is over.

5.1 *The goal pursuit mode*

When pursuing the goal, the planner decides on the next action based on the direction and the distance of the parking spot. There are three spatial variables to consider: x, y, and θ, each with its own goal value. Using a qualitative model, the planner may, for example, deduce that turning the steering wheel left may work in favor of variable y, but unfavorably for variable θ. The priority is then given to variables that

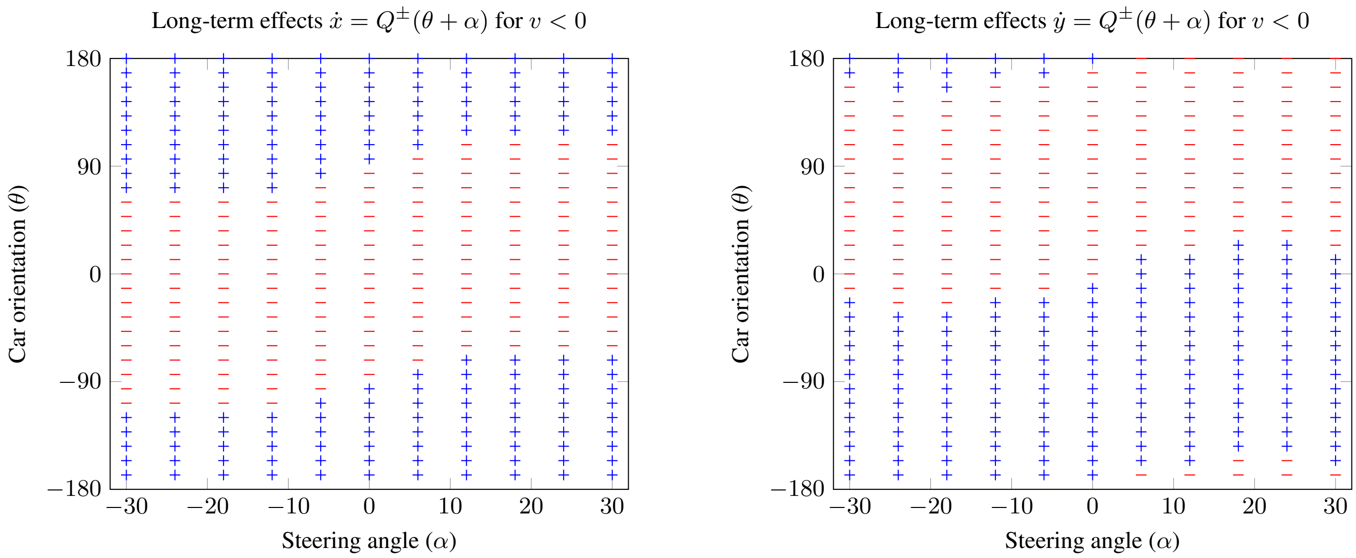

Figure 5. The learned long-term qualitative effects of using different steering angles (α) in different car orientations (θ) while driving backward ($v < 0$). The '+' and the '-' signs respectively denote the Q^+ and the $\dot{y} = Q^-$ constraint.

are farther away from their goal values. This is done in the following way:

1. Observe and store the current speed and acceleration for each variable separately.
2. Using the history of observation, compute theoretically the fastest possible time η_i for each variable x_i to reach its goal value, taking into account the highest observed acceleration $\pm a^i_{\text{max}}$ for each direction separately, the current velocity v^i_0 and the observed terminal velocity v^i_{max}.
3. Use the qualitative model (long-term model for backward driving) to determine how each action would affect the direction of change of each variable.
4. Let each variable x_i use its η_i as a voting score for each action. If the action moves the value x_i towards its goal, the $+\eta_i$ is cast for the action, and if the action moves it away from the goal, $-\eta_i$ is cast.
5. Sum up all the votes for each action and execute the one with the highest score.

Computing the time estimates η_i instead of using actual spatial distances bridges the gap between different units (e.g., meters for x, y, and degrees for θ), while also accounting for different kinematic properties (e.g., rotations could be slow in comparison to forward/backward motion). This way the planner dynamically adapts to the numerical properties of the system. Moreover, by considering this temporal component, the planner aims to bring all variables to their goal values simultaneously. The experiments with a similar approach in [22, 20] show that the action selection algorithm first brings all the η_i values approximately to the same value and then simultaneously lowers them to $\eta_i = 0$ (meaning that the goal state has been reached), if such a behavior is possible. This applies to our parking problem as the capability to park the car in one smooth trajectory without the need for additional corrections, if such a trajectory is possible with the given steering constraints. However, there is no guarantee that the obtained trajectories are optimal.

5.2 The safety mode

When one or more sensors are triggered, the algorithm switches to the *safety mode*, where the goal stops being pursued and the aim is to avoid collision. In some of the previous work (e.g., [21, 22]), collision avoidance has been successfully executed while simultaneously pursuing the goal. However, it was only shown to work with point obstacles and a sensory input that exhibits continuous changes in the input values. In our parking domain, sensory input is typically not continuous — a wall may come to an end, and the input may instantly jump from, e.g., 0.5 meters to infinity. This confuses the η_i computations with erroneous observations of velocities and accelerations, so the sensory variables cannot be compared with the pursuit variables when voting for individual actions. We therefore introduce the *safety mode*, where the actual values of the distance sensors are used instead of η_i, to prioritize actions.

In *safety mode*, actions are decided in the following way:

1. Observe the current values x_i of active sensors (the distance from the obstacle).
2. Use the qualitative model to predict whether an action increases or decreases the sensor's distance to the obstacle.
3. Vote by $1/x_i$ for an action, if the action increases the distance from the obstacle, and by $-1/x_i$ if it decreases it.
4. Sum up all the votes for each action and execute the one with the highest score.

In our parking domain, we use the short-term qualitative model for the front sensor and the long-term qualitative model for the rear sensor. The reasons are obvious from Fig. 3.

6 Experimental Results

The proposed planning approach succeeded in parking the car without a collision. Fig. 6 shows the result of a simulation at 25 FPS, which took 80 steps (3.2 seconds).

In the beginning, both sensors turn on because of the proximity of the left-side obstacle. The car therefore drives straight back until the rear sensor turns off. Still in safety mode, a slight clockwise turn is made to increase the distance of the front sensor from the obstacle, and soon after the front sensor also turns off. The car continues with pursuing the goal and makes an S-curved trajectory towards the goal position. The parking finishes with the goal orientation error of 2.9 degrees (final θ was $87.1°$).

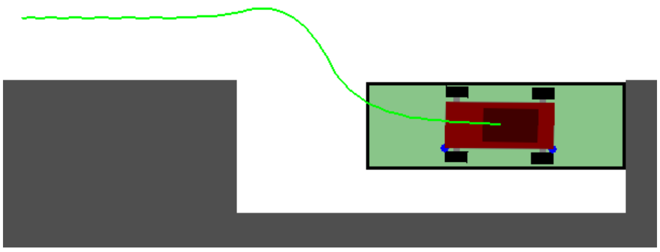

Figure 6. Successful parking maneuver with the proposed approach. The green rectangle denotes where we want the car to be parked. The yellow points mark the points where the distance sensors touch the wall.

Next, we performed 100 experiments to test the efficacy of our method. Initial positions were randomly chosen. The x-position was varied with a maximum deviance of twice the length of the car and chosen so that the back of the car was not past the first corner. The y-position was varied with a maximum deviance of the length of the car with a minimal distance to the wall of $\frac{1}{4}$ of the car's width. The initial orientation θ was chosen within $[60°, 120°]$, thus with a maximal deviation of $30°$ from a perfect parallel orientation. The results are shown in Fig.7. The arrows show the initial positions, such as the arrow in Fig. 1. Green arrows indicate a successful parking maneuver, while red indicates failures. 89 out of the 100 experiments were successful.

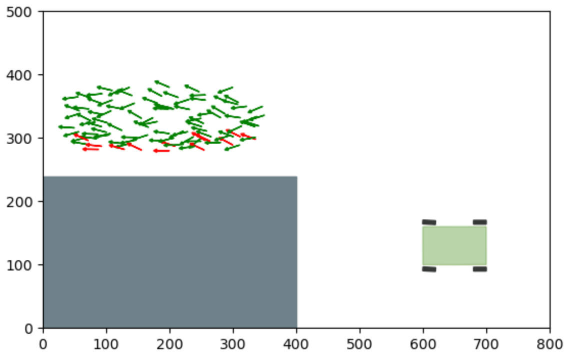

Figure 7. The results of 100 experiments with different initial states. Green and red arrows respectively indicate initial positions and orientations of successful and unsuccessful parking attempts.

There are two patterns of failures. First, if the car is close to the wall and oriented toward the wall with its back side, it cannot make s smooth backward trajectory without colliding with the wall, due to the constraints on the steering angle (Fig. 8). To resolve the situation, a forward maneuver should be made, which is not allowed by our current experimental setup. The second failing scenario occurs when in safety mode, the car is brought into a position of the first type. In our experimental setup, that would happen after successfully passing the wall on the left and taking a sharp turn left while still in safety mode. When switching back to the goal pursuit mode, the car is positioned too close to the wall to be able to make a smooth trajectory without collision (Fig.9). In this case also, forward driving to correct the position would resolve the situation.

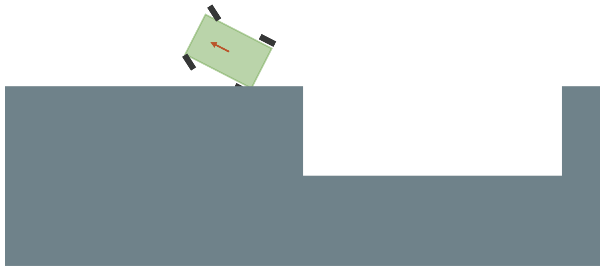

Figure 8. Parking fails if the car is initially too close to the wall and oriented in such a way, that even with the maximum steering angle, collision with the wall in unavoidable.

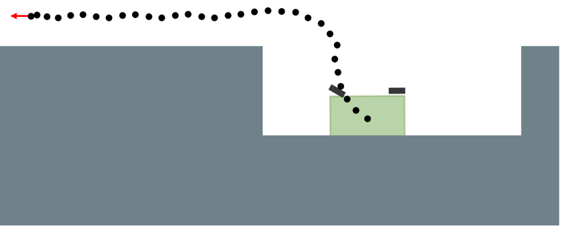

Figure 9. Parking fails, if in safety mode, the car is brought into a position, from which a smooth trajectory is not possible, after switching back to goal pursuit mode.

7 Conclusion

In this paper we showed that the problem of learning how to park a nonholonomic autonomous vehicle can be approached qualitatively. The main advantage of our qualitative approach is significantly higher samples efficiency and the speed of learning a model than with the traditional reinforcement learning methods. We employed a reactive planning method that has already been successfully used with qualitative models for differential drive, quadcopter control, and a cart-pole system [21, 22, 16]. We proposed a novel addition to this type of planning, which is a separation of the *safety model* from the *goal pursuit* model, which solved the problem of discontinuous input from the sensors, as well as previously unaddressed problem of detecting multiple obstacles simultaneously or through multiple sensors. The results showed the ability of our method to park the car with high accuracy in a single maneuver.

The experiment demonstrated in this paper was simplified by keeping the speed constant at all times, which simplified car actions to only turning the wheel. It would be more realistic to also employ speed regulation with the possibility to also move forwards and stop the car at any time. This would also address the problem of non-determinism of constraints of type $Q^{\pm}(\theta + \alpha)$, where the actual outcome depends on the rate of change of both, θ and α. By stopping the car, θ can be considered as a constant, hence $Q^{\pm}(\theta + \alpha)$ becomes equivalent with $Q^{\pm}(\alpha)$. This also complies with the way humans usually park a car — often stopping the car while turning the steering wheel, so to make for the moment the steering wheel the only deciding factor of the next driving direction. By keeping the speed constant, certain trajectories were not possible that would otherwise be feasible, which also includes collisions that could otherwise be avoided.

Acknowledgements

This work was partially supported by the Slovenian Research Agency (ARRS), grant L2-4436: Deep Reinforcement Learning for optimisation of LV distribution network operation with Integrated Flexibility in real-Time (DRIFT).

References

[1] A. Boyali and S. Thompson. Autonomous parking by successive convexification and compound state triggers. In *2020 IEEE International Conference on Robotics and Automation (ICRA)*, pages 1204–1210. IEEE, 2020.

[2] I. Bratko and D. Suc. Learning qualitative models. *AI Magazine*, 24: 107–119, 01 2004.

[3] L. E. Dubins. On curves of minimal length with a constraint on average curvature, and with prescribed initial and terminal positions and tangents. *American Journal of Mathematics*, 79(3):497–516, 1957.

[4] L. E. Dubins. On curves of minimal length with a constraint on average curvature, and with prescribed initial and terminal positions and tangents. *American Journal of Mathematics*, 79(3):497–516, 1957.

[5] K. D. Forbus. Qualitative process theory. *Artificial intelligence*, 24: 85–168, 1984.

[6] K. D. Forbus. *Qualitative representations: How People Reason and Learn about the Continuous World*. MIT Press, 2018.

[7] B. Kuipers. Qualitative simulation. *Artificial Intelligence*, 29(3):289–338, 1986.

[8] S. M. LaValle. *Planning Algorithms*. Cambridge University Press, USA, 2006. ISBN 0521862051.

[9] J. Lemeire, N. Wouters, M. Van Cleemput, and A. Heirman. Contextual qualitative deterministic models for self-learning embodied agents. In C. L. Buckley, D. Cialfi, P. Lanillos, M. Ramstead, N. Sajid, H. Shimazaki, T. Verbelen, and M. Wisse, editors, *Active Inference*, pages 3–13, Cham, 2024. Springer Nature Switzerland. ISBN 978-3-031-47958-8.

[10] D. Moreira. Deep reinforcement learning for automated parking. *Applied Intelligence*, 51(2):397–416, 2021.

[11] N. Nakrani and M. Joshi. A human-like decision intelligence for obstacle avoidance in autonomous vehicle parking. *Appl Intell*, 52:3728–3747, 2022.

[12] J. A. Reeds and L. A. Shepp. Optimal paths for a car that goes both forwards and backwards. *Pacific Journal of Mathematics*, 145(2):367–393, 1990.

[13] M. Rosenfelder, H. Ebel, J. Krauspenhaar, and P. Eberhard. Model predictive control of non-holonomic vehicles: Beyond differential-drive. *arXiv preprint arXiv:2205.11400*, 2022.

[14] S. Shahi and H. Lee. Autonomous rear parking via rapidly exploring random-tree-based reinforcement learning. *arXiv preprint arXiv:2205.11400*, 2022.

[15] D. Šoberl and I. Bratko. Unified approach to qualitative motion planning in dynamic environments. In *Proceedings of the 29th International Workshop on Qualitative Reasoning*, 2016.

[16] D. Šoberl and I. Bratko. Transferring a learned qualitative cart-pole control model to uneven terrains. In A. Bifet, A. C. Lorena, R. P. Ribeiro, J. Gama, and P. H. Abreu, editors, *Discovery Science*, pages 446–459, Cham, 2023. Springer Nature Switzerland. ISBN 978-3-031-45275-8.

[17] B. Triggs. Motion planning for nonholonomic vehicles: An introduction. In *Proceedings of the 1993 IEEE International Conference on Robotics and Automation*, pages 2605–2610. IEEE, 1993.

[18] J. Žabkar, I. Bratko, and J. Demšar. Learning qualitative models through partial derivatives by Padé. In *Proceedings of the 21st Annual Workshop on Qualitative Reasoning*, pages 193–202, 2007.

[19] P. Zhang, L. Xiong, Z. Yu, P. Fang, S. Yan, J. Yao, and Y. Zhou. Reinforcement learning-based end-to-end parking for automatic parking system. *Sensors*, 19(18), 2019. ISSN 1424-8220. doi: 10.3390/s19183996. URL https://www.mdpi.com/1424-8220/19/18/3996.

[20] D. Šoberl. *Automated planning with induced qualitative models in dynamic robotic domains*. PhD thesis, University of Ljubljana, 2021.

[21] D. Šoberl and I. Bratko. Reactive motion planning with qualitative constraints. In *Advances in Artificial Intelligence: From Theory to Practice*, pages 41–50. Springer International Publishing, 2017. doi: 10.1007/978-3-319-60042-0_5.

[22] D. Šoberl and I. Bratko. Learning to control a quadcopter qualitatively. *Journal of Intelligent & Robotic Systems*, 2020. doi: 10.1007/s10846-020-01228-7.