Autonomous Parking Using A Sensor Based Approach

David Pérez Morales
Salvador Domínguez Quijada
Olivier Kermorgant
Philippe Martinet

2016/10/13

Table of Contents

INTRODUCTION

Introduction

Parking and unparking are often difficult tasks, especially for unexperienced drivers. They involve:

- Finding a suitable parking spot
- Maneuver either into or from the (often tight) parking spot without colliding with anything while trying to avoid disturbing the surrounding traffic.

All of this hassle could be avoided with an ***Autonomous Parking*** system

Different Approaches

The different control approaches available in the literature can be divided into two categories:

- Based on stabilizing the vehicle to a target point
- Based on path planning.

Typically, highly dependent on the localization performance.

⇒ Sensor based control, an interesting possibility.

Modeling and control

Car-Like Robot Rear-Wheel Driving

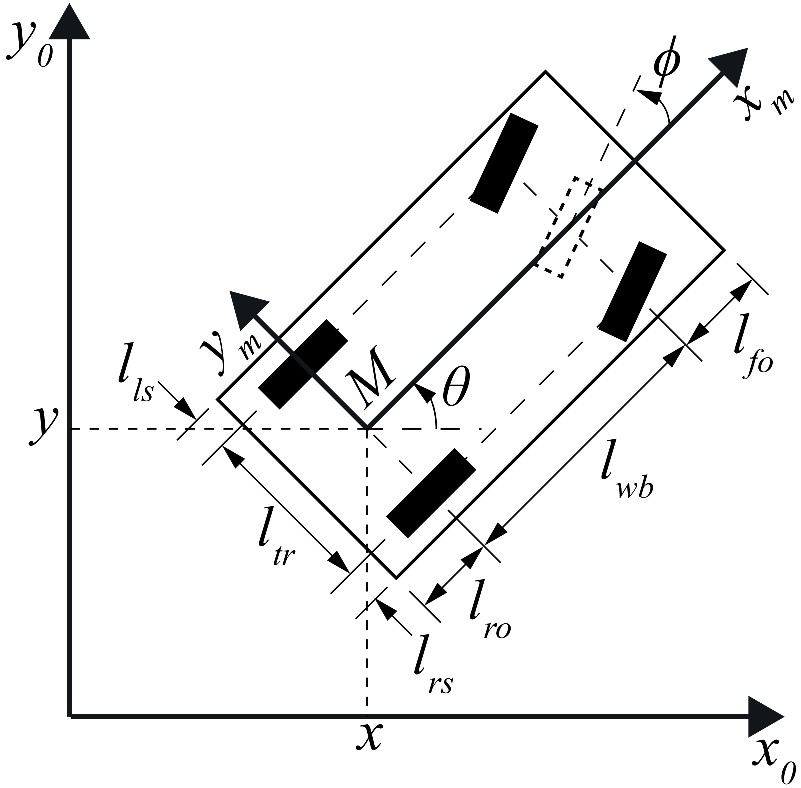


Figure: Kinematic model diagram for a car-like rear-wheel driving robot

$$\begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{\theta} \\ \dot{\phi} \end{bmatrix} = \begin{bmatrix} \cos\theta \\ \sin\theta \\ \tan\phi / l_{wb} \\ 0 \end{bmatrix} v + \begin{bmatrix} 0 \\ 0 \\ 0 \\ 1 \end{bmatrix} \dot{\phi}$$

Where v and $\dot{\phi}$ are the driving and steering velocities.
The relation between ϕ and $\dot{\theta}$ can be expressed as:

$$\phi = \operatorname{atan}\left(\frac{\dot{\theta}\ l_{wb}}{v}\right)$$

Experimental Setup

Velocity, direction of travel, steering and turning signals can be controlled by computer.

Figure: HDK DEL2030DUB

Sensor Based Control With A Weighted Control Scheme

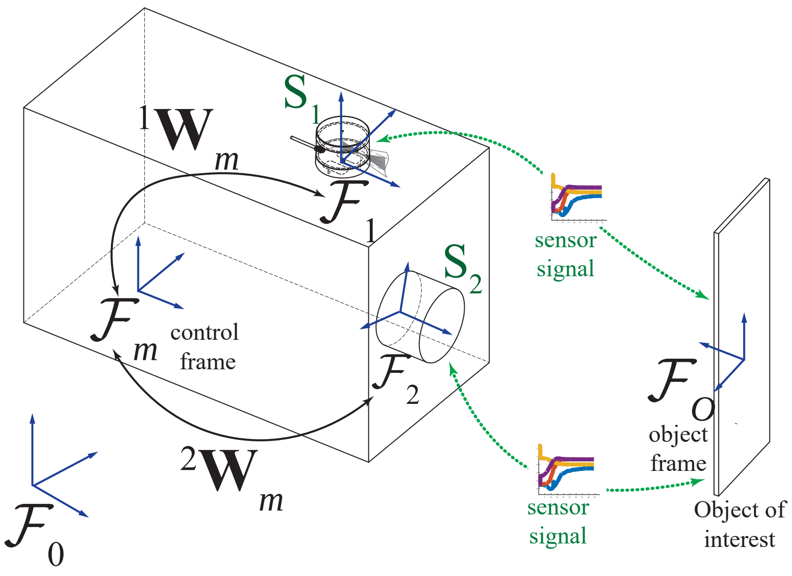


Figure: Multi-sensor model

$$\dot{\mathbf{s}}_i = \mathbf{L}_i \mathbf{v}_i$$

$$\mathbf{v}_i = {}^i\mathbf{W}_m \mathbf{v}_m$$

$$ {}^i\mathbf{W}_m = \left[\begin{array}{c|c} {}^i\mathbf{R}_m & [{}^i\mathbf{t}_m]_\times \, {}^i\mathbf{R}_m \\ \hline \mathbf{0}_{3x3} & {}^i\mathbf{R}_m \end{array} \right]$$

$$\mathbf{L}_s = \mathbf{L}\mathbf{W}_m = \begin{bmatrix} \mathbf{L}_1 & \dots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \dots & \mathbf{L}_k \end{bmatrix} \begin{bmatrix} {}^1\mathbf{W}_m \\ \vdots \\ {}^k\mathbf{W}_m \end{bmatrix}$$

$$\dot{\mathbf{s}} = \mathbf{L}_s \mathbf{v}_m$$

[1]

[1] O. Kermorgant and F. Chaumette, “Dealing with constraints in sensor-based robot control,” *IEEE Transactions on Robotics*, vol. 30, no. 1, pp. 244–257, 2014.

Sensor Based Control With A Weighted Control Scheme

In the current implementation, the sensor measurements are already on F_m, leading to:

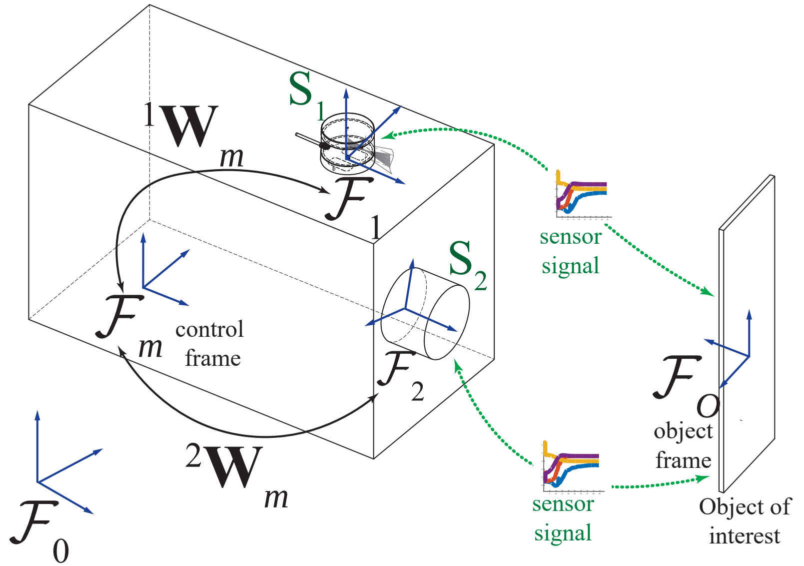


Figure: Multi-sensor model

$$\dot{\mathbf{s}}_i = \mathbf{L}_i \mathbf{v}_i = \mathbf{L}_i \mathbf{v}_m$$

$$\dot{\mathbf{s}} = \mathbf{L}_s \mathbf{v}_m$$

$$\mathbf{L}_s = \begin{bmatrix} \mathbf{L}_1 \\ \vdots \\ \mathbf{L}_k \end{bmatrix}$$

$$\mathbf{v}_m = -\lambda \mathbf{C}\mathbf{e}$$

$$\mathbf{C} = (\mathbf{H}\hat{\mathbf{L}}_\mathbf{s})^+ \mathbf{H}$$

Sensor Based Control - Weighted Error

$$\mathbf{e_H} = \mathbf{He}$$

$$\mathbf{e} = \mathbf{s} - \mathbf{s}^*$$

$$\mathbf{H} = \mathrm{diag}(h_1, \dots, h_d)$$

$$\mathbf{H} = \begin{bmatrix} \mathbf{H}_1 & \dots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \dots & \mathbf{H}_k \end{bmatrix}$$

$$\forall i \in [1, d] : h_i = h_i^t + h_i^c$$

$$h_i^c = \begin{cases} \frac{s_i^{s-} - s_i}{s_i - s_i^-} & \text{if } s_i < s_i^{s-} \\ 0 & \text{otherwise} \end{cases}$$

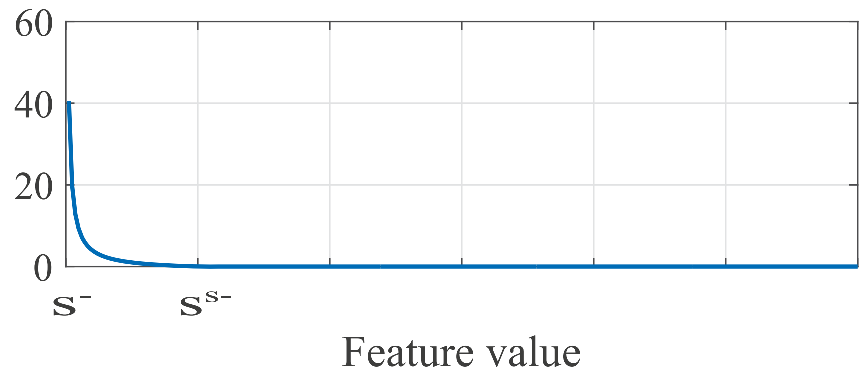
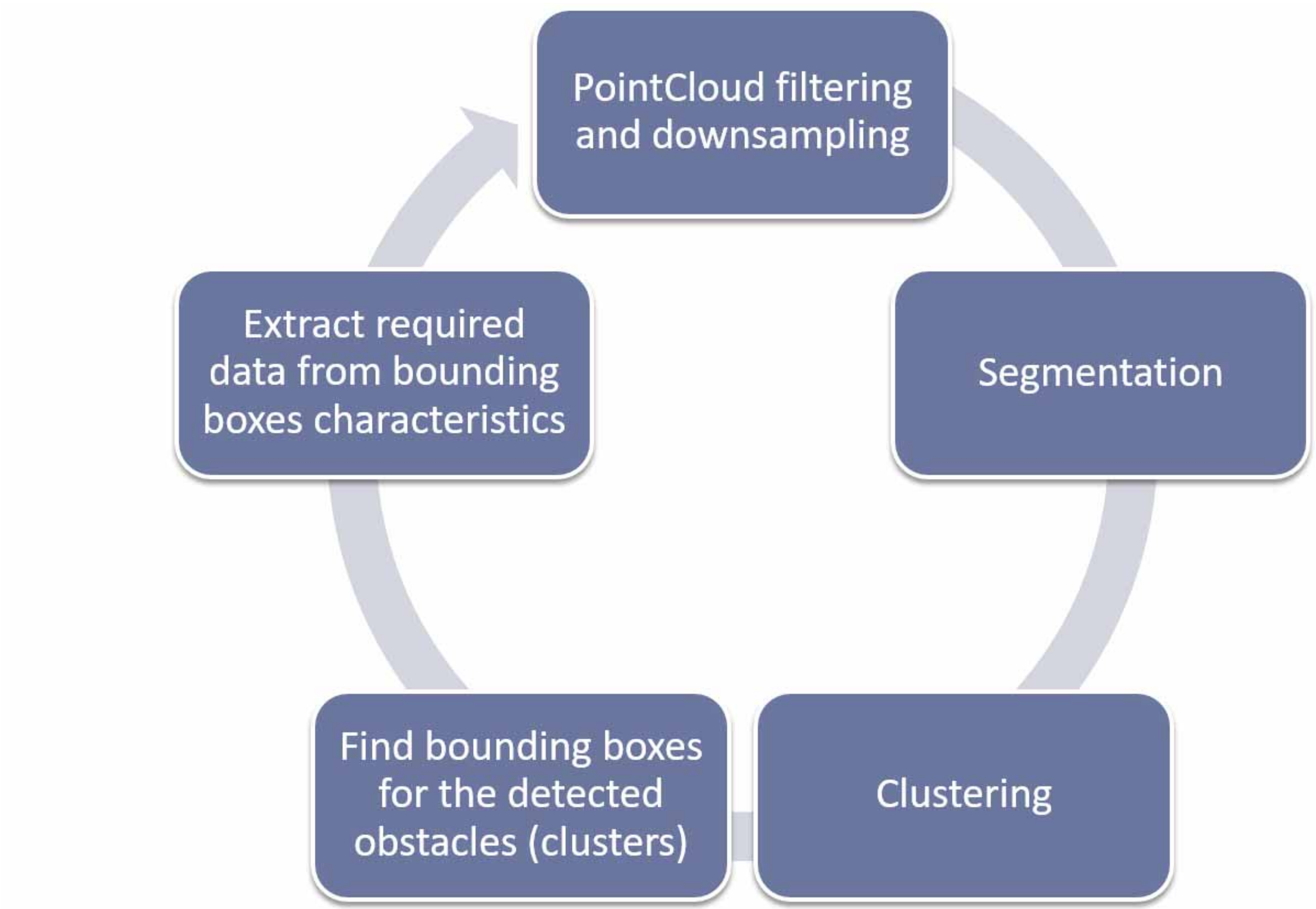


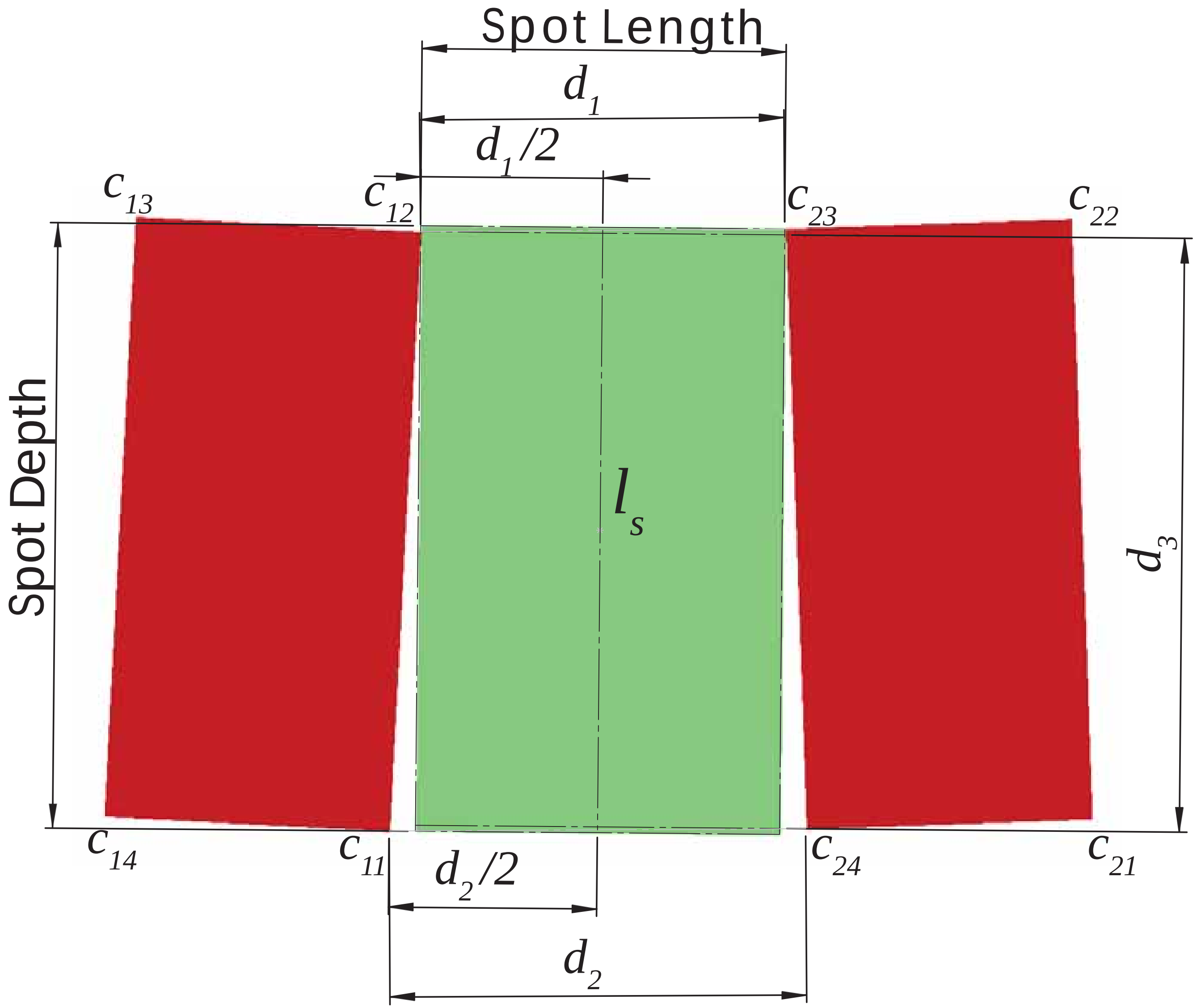
Figure: Weighting h^c for the basic constraint

Perception

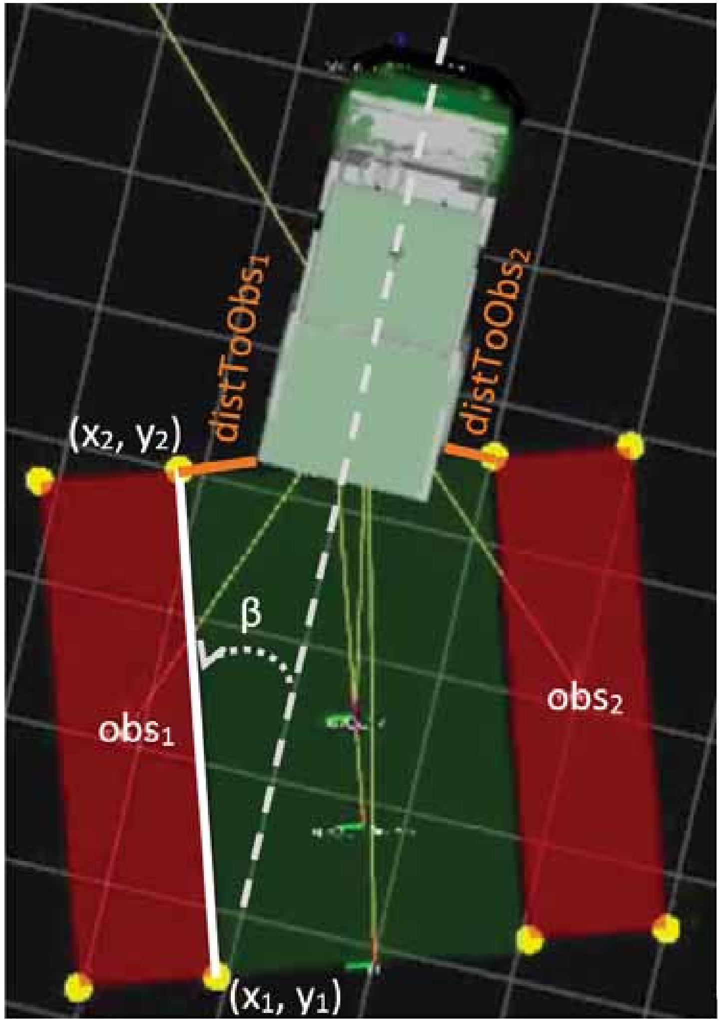
PointCloud Processing



Extraction of empty parking place



Extraction Of Current Features For Sensor Based Control



Control sensor feature

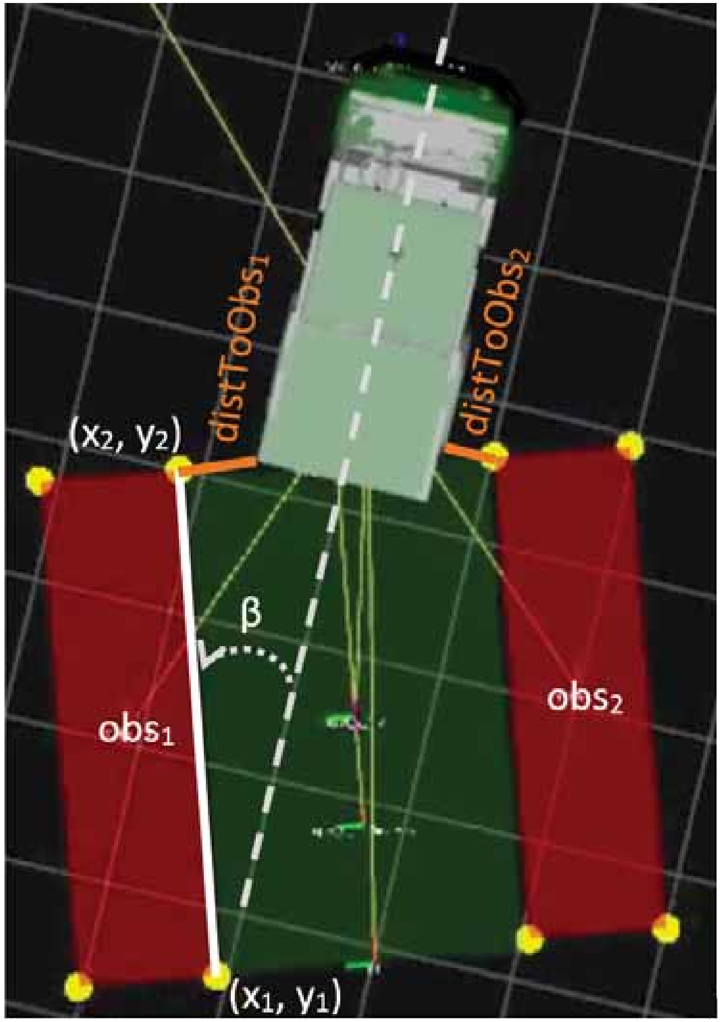
The control sensor feature $\mathbf{s}_1$ is extracted from the closest side (of the two largest sides) of obs_1 to the vehicle. The point (x_1, y_1) is chosen to be always the one further to the back of the vehicle.

$$\mathbf{s}_1 = [x_1, y_1, \beta]^T$$

$$\mathbf{L}_{s_1} = \begin{bmatrix} -1 & y_1 \\ 0 & -x_1 \\ 0 & 1 \end{bmatrix}$$

$$\mathbf{L}_i = \frac{L_{s_i} + L^*_{s_i}}{2}$$

Extraction Of Current Features For Sensor Based Control



Constraints sensor features

The constraints sensor features $\mathbf{s}_2$ and $\mathbf{s}_3$ correspond to, respectively, the closest distance between either the left or right side of the vehicle and obs_1 or obs_2.

$$\forall j \in [1,2] : s_{1+j} = distToObs_j = \sqrt{x_{obs_j}^2 + y_{obs_j}^2}$$

$$\mathbf{L}_{s_{1+j}} = \begin{bmatrix} -\frac{x_{obs_j}}{distToObs_j} & 0 \end{bmatrix}$$

$$\mathbf{L}_i = \frac{L_{s_i} + L_{s_i}^*}{2}$$

RESULTS

Results - Offline Data

Figure: Extraction of empty parking place using offline data (speed: 2x)

Simulation Results - Path Planning Approach

Figure: Results of the implementation of a state-of-the-art path planning approach[2] (speed: 1.75x)

[2] P. Petrov, F. Nashashibi, and M. Marouf, "Path Planning and Steering control for an Automatic Perpendicular Parking Assist System," *7th Workshop on Planning, Perception and Navigation for Intelligent Vehicles, PPNIV'15*, pp. 143–148, 2015.

Simulation Results - Path Planning Approach

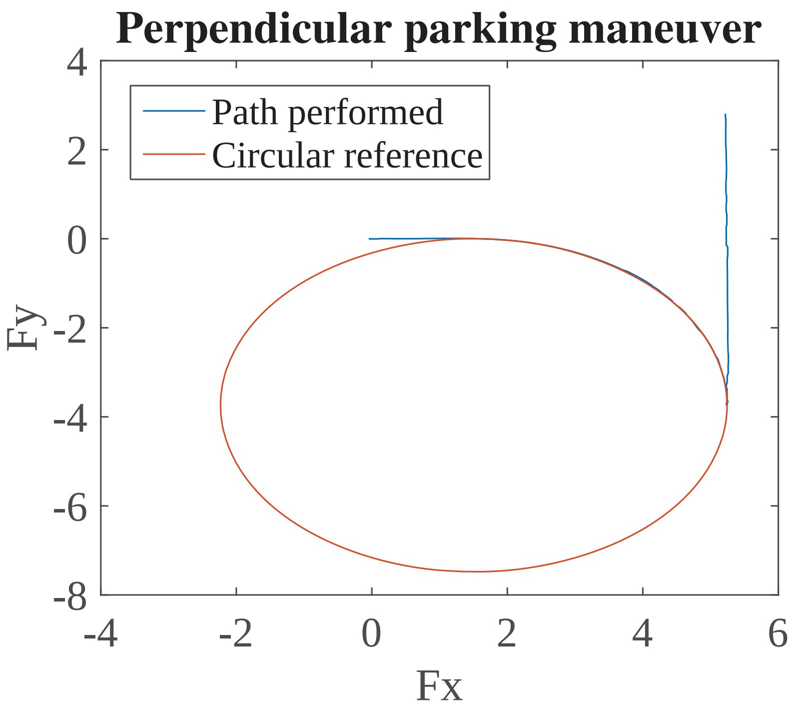


Figure: Path performed by the vehicle and the reference circle that connects positions 1 and 2

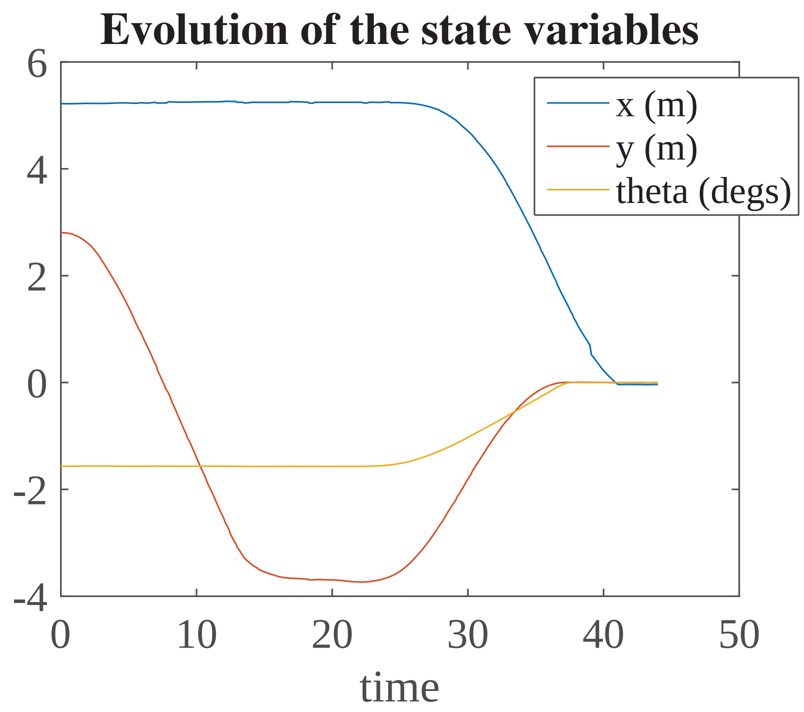


Figure: Evolution of the state variables over time

Final error values: $x = -37.7$ mm, $y = -2.6$ mm, $\theta = 0.000\,08°$.

Simulation Results - Path Planning Approach

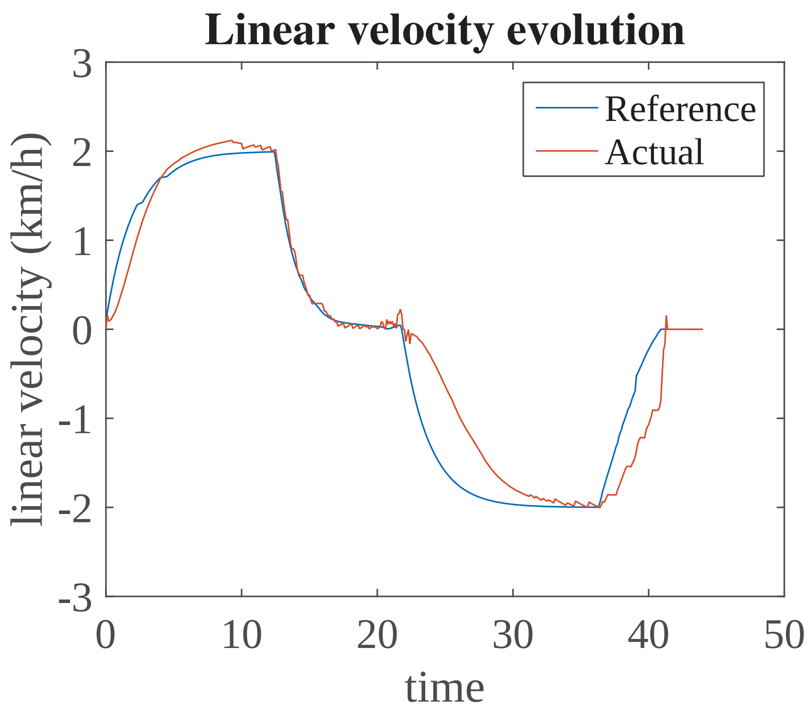


Figure: Evolution of the vehicle's linear velocity - path planning

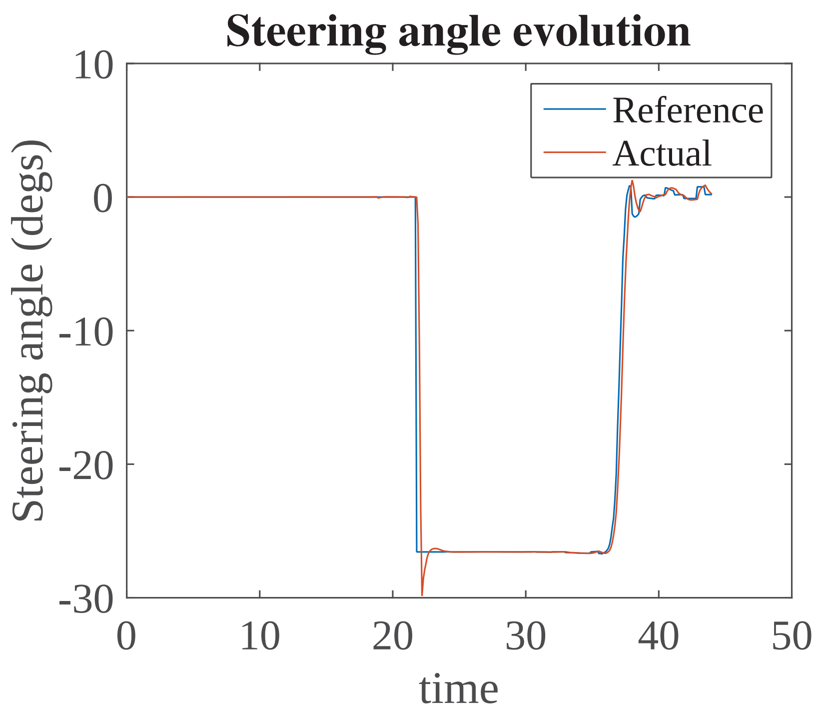


Figure: Evolution of the vehicle's steering angle

Simulation Results - Sensor Based Approach

Figure: Results using the proposed sensor based approach - reverse maneuver (speed: 1.75x)

Simulation Results - Sensor Based Approach

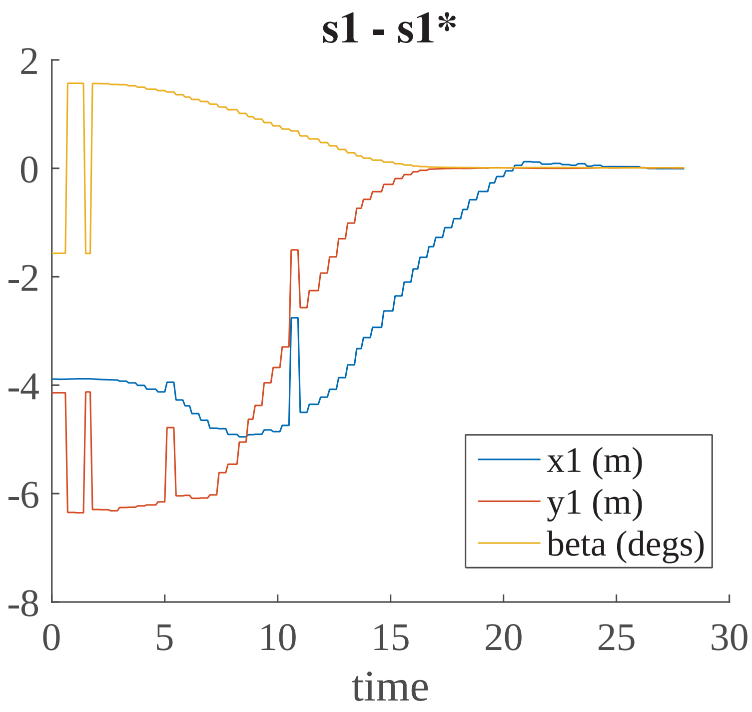


Figure: Evolution of the error signal - reverse maneuver

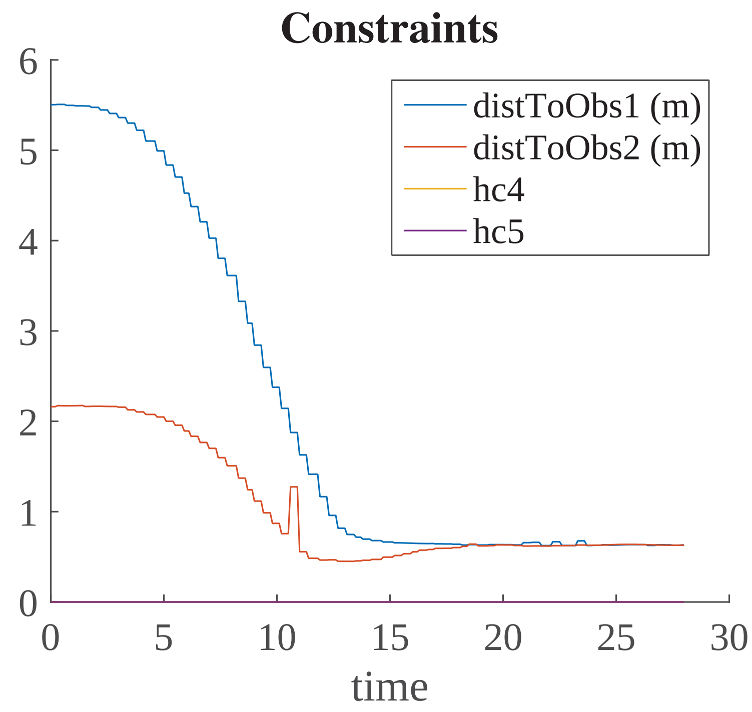


Figure: Evolution of the constraints - reverse maneuver

Final error values: $x_1 = -6.5$ mm, $y_1 = 7.2$ mm, $\beta = 0.0102°$.

IRCCyN

ECN Centrale Nantes

Simulation Results - Sensor Based Approach

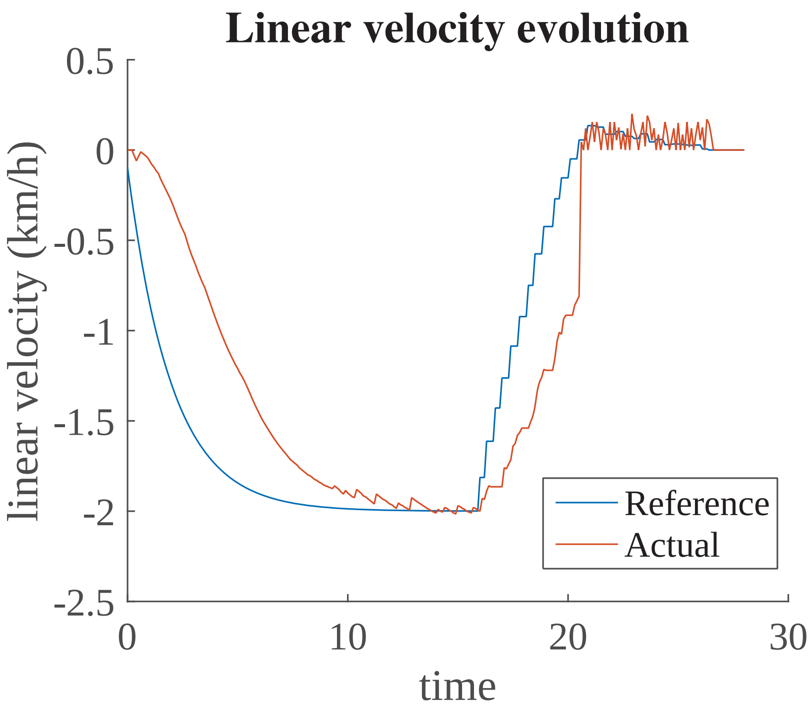


Figure: Evolution of the linear velocity - reverse maneuver

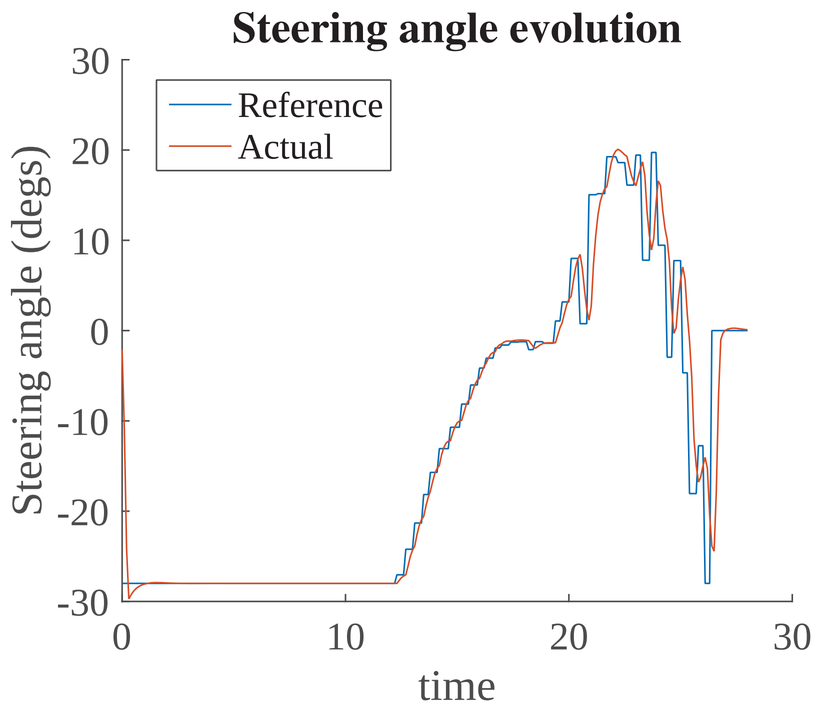


Figure: Evolution of the steering angle - reverse maneuver

Simulation Results - Sensor Based Approach

Figure: Results using the proposed sensor based approach - forward maneuver (speed: 1.75x)

Simulation Results - Sensor Based Approach

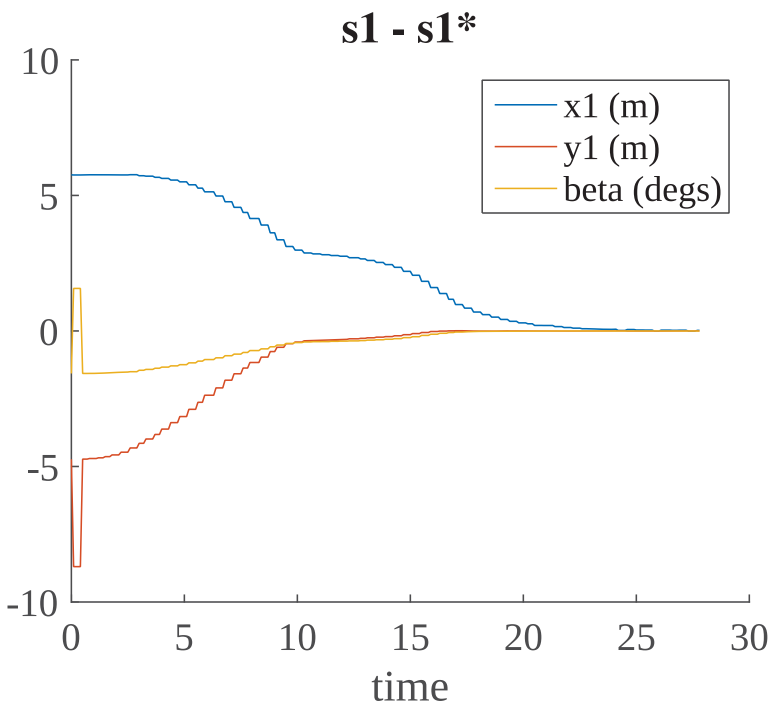


Figure: Evolution of the error signal - forward maneuver

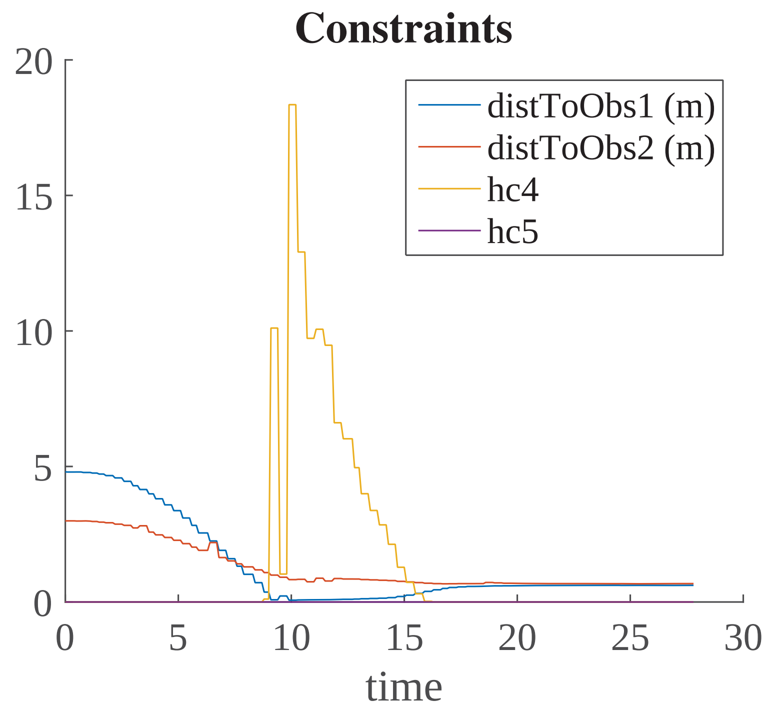


Figure: Evolution of the constraints - forward maneuver

Final error values: $x_1 = 20.9$ mm, $y_1 = 0.7$ mm, $\beta = -0.0019°$.

Simulation Results - Sensor Based Approach

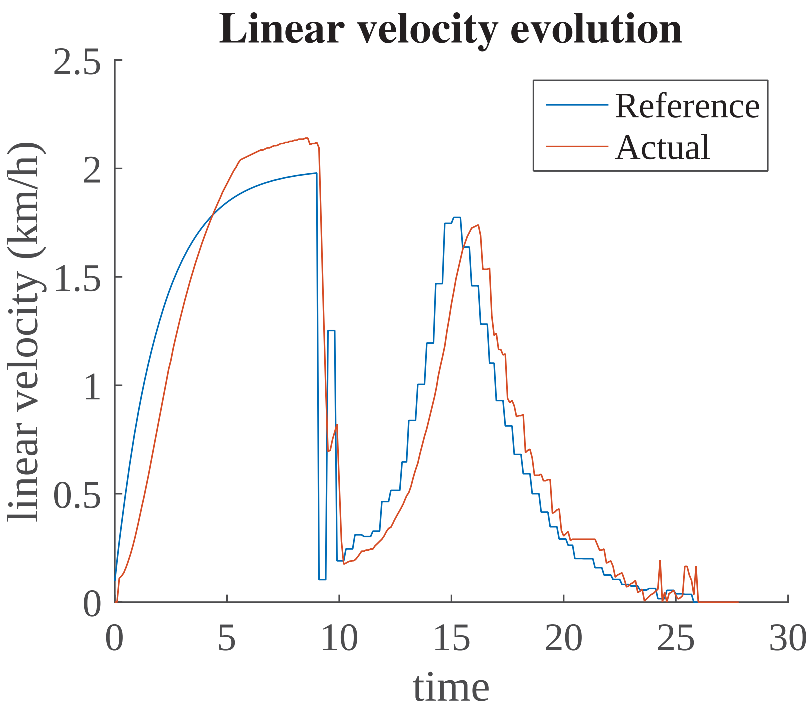


Figure: Evolution of the linear velocity - forward maneuver

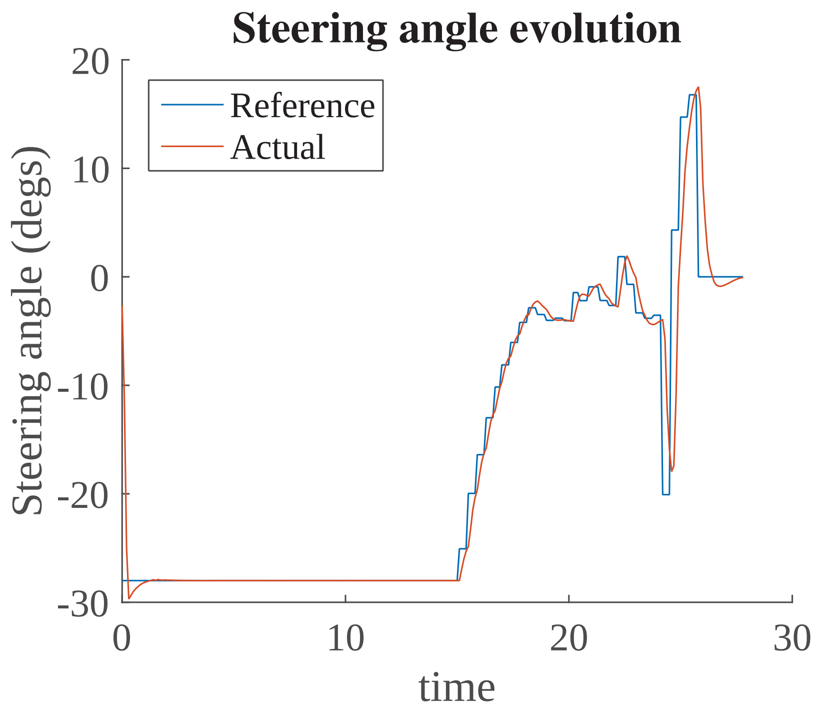


Figure: Evolution of the steering angle - forward maneuver

Real Experimentation Results - Sensor Based Approach

Figure: Preliminary results with the real vehicle using the proposed sensor based approach

Real Experimentation Results - Sensor Based Approach

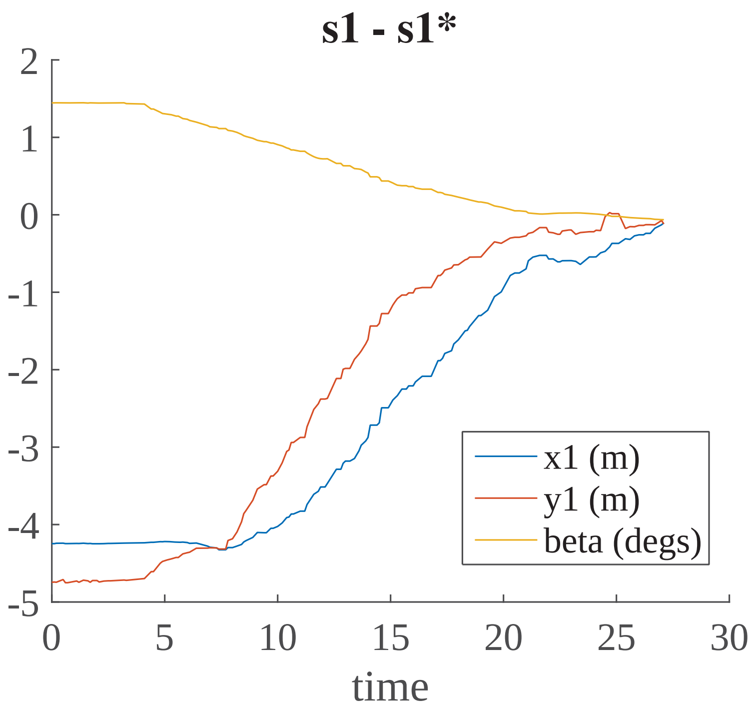


Figure: Evolution of the error signal

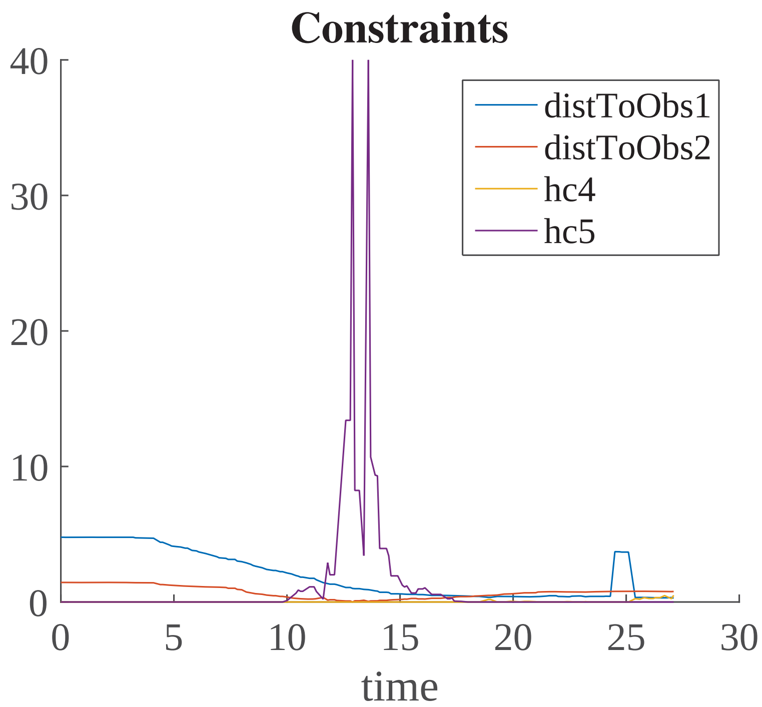


Figure: Evolution of the constraints

Final error values: $x_1 = -10.4$ cm, $y_1 = 11.9$ cm, $\beta = -0.0637°$.

Real Experimentation Results - Sensor Based Approach

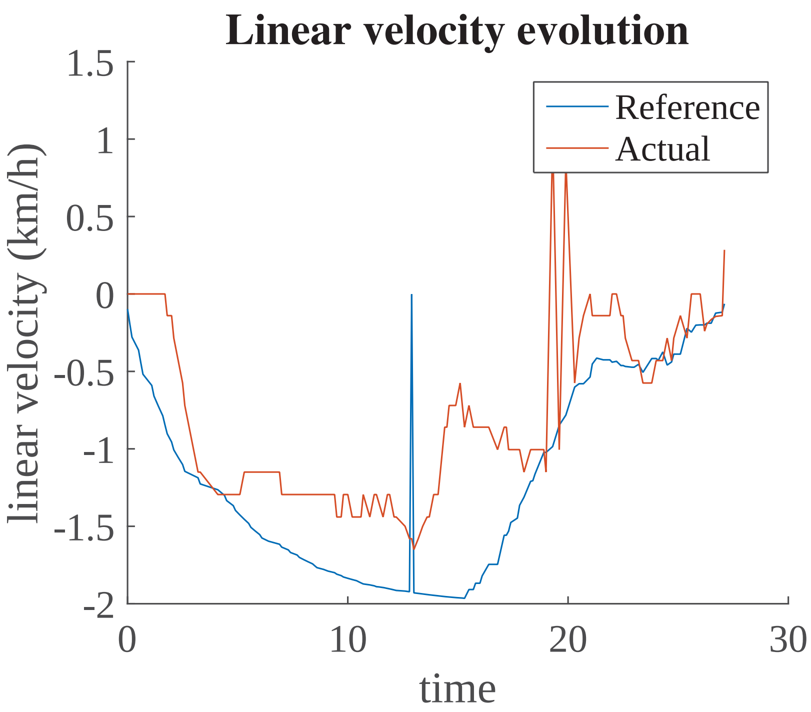


Figure: Evolution of the linear velocity

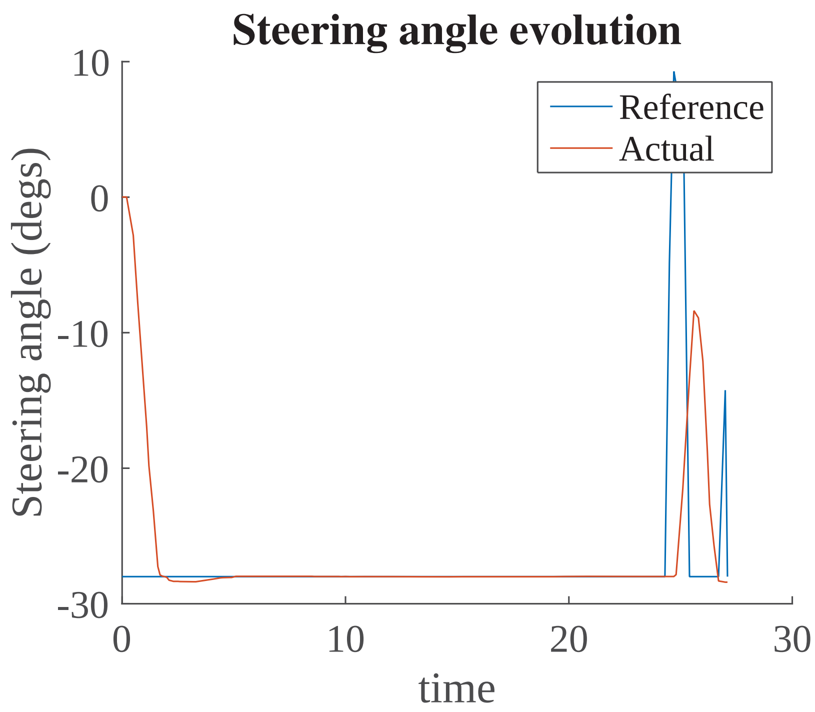


Figure: Evolution of the steering angle

CONCLUSIONS

Conclusions

- Under simulation, the proposed sensor based approach is able to achieve slightly smaller errors in position than the path planning approach.
- The proposed approach is very versatile, requiring only some minor modifications to change between reverse and forward maneuvers.
- The proposed approach has been proved to be a valid one considering the results obtained from simulation and real experimentation.
- More work has to be done to improve the performance when using the real vehicle.

Acknowledgments

This work was supported by the Mexican National Council for Science and Technology (CONACYT), Baja California State Council for Science and Technological Innovation (COCIT BC) and the Mexican Ministry of Public Education (SEP).