Autonomous Perpendicular And Parallel Parking Using Multi-Sensor Based Control: Convergence Analysis

David Pérez-Morales, Olivier Kermorgant, Salvador Domínguez-Quijada and Philippe Martinet

Updated: 2017/11/07

CENTRALE NANTES

LS2N LABORATOIRE DES SCIENCES DU NUMÉRIQUE DE NANTES

Overview

Modeling and Notation

Car-Like Robot Rear-Wheel Driving

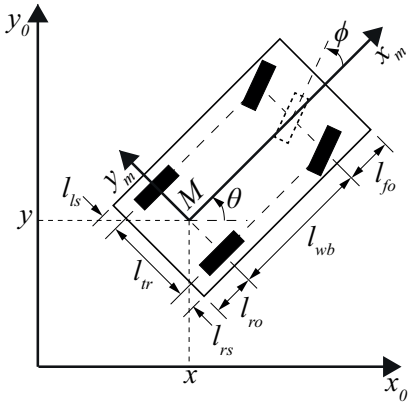


Figure: Kinematic model diagram for a car-like rear-wheel driving robot

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

Where v and $\dot{\phi}$ are the driving and steering velocities.

Experimental Setup

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

Figure: Robotized Renault ZOE

Multi-sensor modeling

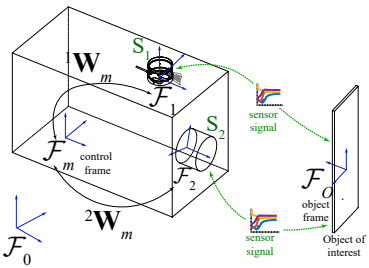


Figure: Multi-sensor model

In a static environment, the sensor feature derivative can be expressed as [1]:

$$\dot{\mathbf{s}}_i = \mathbf{L}_i \mathbf{v}_i = \underset{(d_i \times 6)}{\mathbf{L}_i} \; \underset{(6\times 6)}{{}^i\mathbf{W}_m} \; \underset{(6\times 1)}{\mathbf{v}_m} \tag{2}$$

$$\underset{(d\times 6)}{\mathbf{L}_s} = \mathbf{L}\mathbf{W}_m = \underset{(d\times 6k)}{\begin{bmatrix} \mathbf{L}_1 & \dots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \dots & \mathbf{L}_k \end{bmatrix}} \underset{(6k\times 6)}{\begin{bmatrix} {}^1\mathbf{W}_m \\ \vdots \\ {}^k\mathbf{W}_m \end{bmatrix}} \tag{3}$$

$$\dot{\mathbf{s}} = \mathbf{L}_s \mathbf{v}_m \tag{4}$$

Under a planar world assumption:

$$\dot{\mathbf{s}}_i = \mathbf{L}_{i_r} \mathbf{v}_{i_r} = \underset{(d_i\times 3)}{\mathbf{L}_{i_r}} \; \underset{(3\times 3)}{{}^i\mathbf{W}_{m_r}} \; \underset{(3\times 1)}{\mathbf{v}_{m_r}} \tag{5}$$

where $\mathbf{v}_{m_r} = [v_{x_m}, v_{y_m}, \dot{\theta}]^T$

[1] Kermorgant and Chaumette, "Dealing with constraints in sensor-based robot control"

Multi-sensor modeling

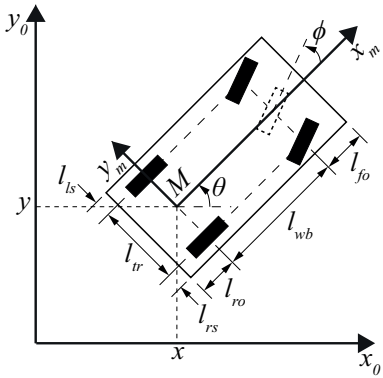


Figure: Kinematic model diagram for a car-like rear-wheel driving robot

Assuming $v_{y_m} = 0$ (no slipping nor skidding)

$$\mathbf{v} = [v_{x_m}, \dot{\theta}]^T \tag{6}$$

$$\dim(\mathbf{L_v}) = (d \times 2)$$

where $v_{x_m} = v$ and $\mathbf{L_v}$ is the corresponding sub-matrix extracted from $\mathbf{L}_{s_r}$.

Weighted error

The weighted multi-sensor error signal is defined as:

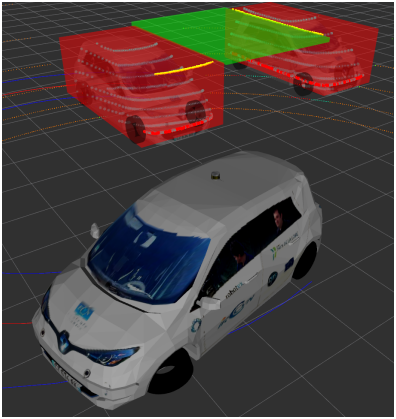
$$\mathbf{e_H} = \mathbf{He} \tag{7}$$

where $\mathbf{e} = \mathbf{s} - \mathbf{s}^*$ is the difference between the current sensor signal $\mathbf{s}$ and its desired value $\mathbf{s}^*$ and $\mathbf{H}$ is a diagonal positive semi-definite weighting matrix that depends on the current value of $\mathbf{s}$. Its associated interaction matrix is $\mathbf{L_H} = \mathbf{HL}_s$.

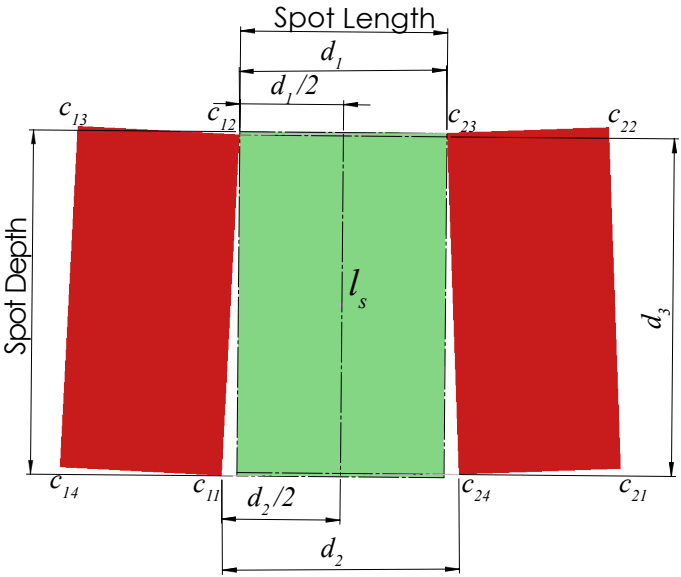
Perception

Perception

Sensor fusion → Filtering and downsampling → Segmentation and clustering → Orientation extraction → Bounding box extraction



Extraction of empty parking place



Parking spots

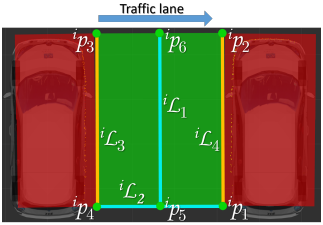


Figure: $\perp$ parking spot model

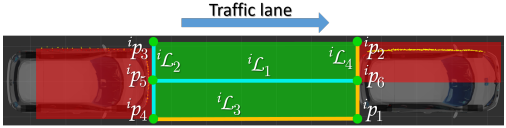


Figure: Parking spot model for reverse $\parallel$ parking maneuvers

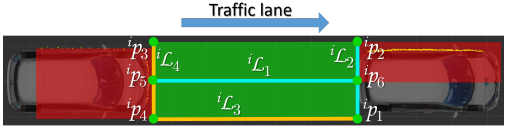


Figure: Parking spot model for forward $\parallel$ parking maneuvers

Table: Pair of points through which each line passes

Line	*Perpendicular*	*Parallel (reverse)*	*Parallel (forward)*
$^i\mathcal{L}_1$	$(^ip_5, ^ip_6)$	$(^ip_5, ^ip_6)$	$(^ip_5, ^ip_6)$
$^i\mathcal{L}_2$	$(^ip_1, ^ip_4)$	$(^ip_3, ^ip_4)$	$(^ip_1, ^ip_2)$
$^i\mathcal{L}_3$	$(^ip_3, ^ip_4)$	$(^ip_1, ^ip_4)$	$(^ip_1, ^ip_4)$
$^i\mathcal{L}_4$	$(^ip_1, ^ip_2)$	$(^ip_1, ^ip_2)$	$(^ip_3, ^ip_4)$

Interaction Model

Interaction Model

The sensor signals $\mathbf{s}_{i_{\mathcal{L}_j}}$ and reduced interaction matrix $\mathbf{L}_{i_{\mathcal{L}_j}}$ are defined respectively as:

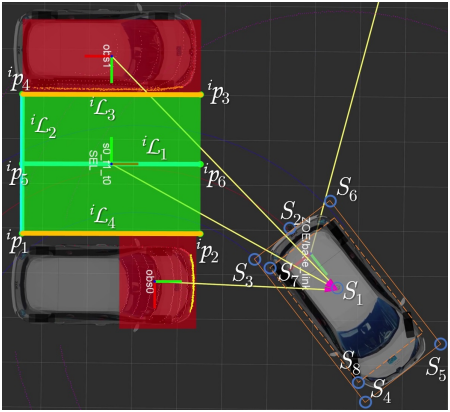


Figure: Sensors' configuration and sensor features

$$\mathbf{s}_{i_{\mathcal{L}_j}} = \left[{}^i\underline{\mathbf{u}}_j(1), {}^i\underline{\mathbf{u}}_j(2), {}^i\mathbf{h}_j(3)\right]^T \tag{8}$$

$$\mathbf{L}_{i_{\mathcal{L}_j}} = \begin{bmatrix} 0 & 0 & {}^i\underline{\mathbf{u}}_j(2) \\ 0 & 0 & -{}^i\underline{\mathbf{u}}_j(1) \\ -{}^i\underline{\mathbf{u}}_j(2) & {}^i\underline{\mathbf{u}}_j(1) & 0 \end{bmatrix} \tag{9}$$

Task sensor features

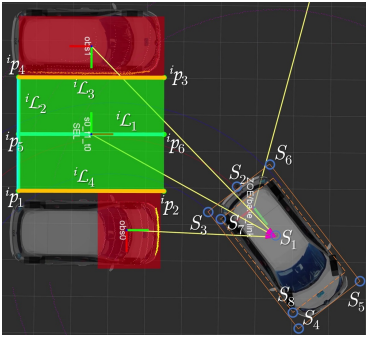


Figure: Sensors' configuration and sensor features

Task sensor features

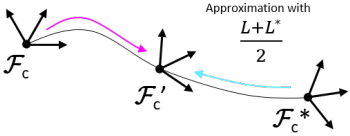
$$\mathbf{s}^t = [\mathbf{s}_{i_{\mathcal{L}_1}}, \mathbf{s}_{i_{\mathcal{L}_2}}]^T \tag{10}$$

$\mathbf{s}^t$ is obtained from S_1 for forward maneuvers and from S_2 for reverse ones.

The corresponding interaction matrix is defined as:

$$\mathbf{L}^t = \frac{\mathbf{L}_{\mathcal{L}} + \mathbf{L}^*_{\mathcal{L}}}{2} \tag{11}$$

where $\mathbf{L}_{\mathcal{L}} = [\mathbf{L}_{i_{\mathcal{L}_1}}, \mathbf{L}_{i_{\mathcal{L}_2}}]^T$ and $\mathbf{L}^*_{\mathcal{L}}$ is equal to the value of $\mathbf{L}_{\mathcal{L}}$ at the desired pose.



Weighting of the task sensor features

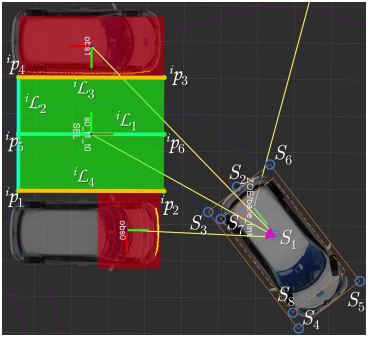


Figure: Sensors' configuration and sensor features

The associated weighting matrix $\mathbf{H}_t$ is defined as:

$$\mathbf{H}^t = \mathrm{diag}(h_1^t, h_2^t, h_3^t, h_4^t, h_5^t, h_6^t) \tag{12}$$

where the values h_3^t and h_6^t are constant while the values of $h_i^t \quad \forall\, i = 1, 2, 4, 5$ are computed using the following smooth weighting function:

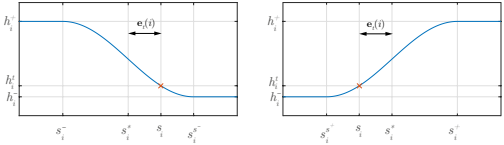


Figure: Weighting function h_i^t

Constraints

Constrained sensor features

$$\mathbf{s}^c = [\mathbf{s}_3, \ldots, \mathbf{s}_8]^T \qquad (13)$$

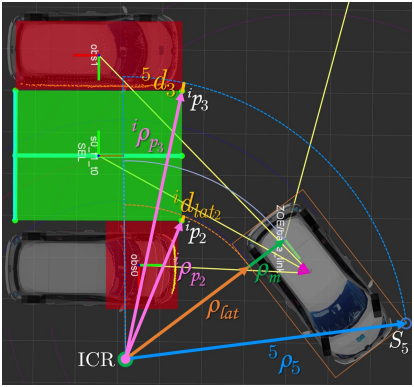


Figure: Radial constraints: all the radii define concentric arcs with center at ICR

The corresponding interaction matrix:

$$\mathbf{L}^c = [\mathbf{L}_3, \ldots, \mathbf{L}_8]^T \qquad (14)$$

Table: Constraints features for ⊥ maneuvers

$\mathbf{s}_i$	*Reverse*	*Forward*
$\mathbf{s}_3$	$[^3\mathbf{h}_2(3), {}^3y_2, {}^3d_{lat_2}]^T$	$^3y_3$
$\mathbf{s}_4$	–	$^4\mathbf{h}_2(3)$
$\mathbf{s}_5$	$^5\mathbf{h}_3(3)$	$[^5\mathbf{h}_2(3), {}^5\mathbf{h}_4(3), {}^5d_2]^T$
$\mathbf{s}_6$	$[^6\mathbf{h}_2(3), {}^6\mathbf{h}_3(3)]^T$	–

Table: Constraints features for || maneuvers

$\mathbf{s}_i$	*Reverse*	*Forward*
$\mathbf{s}_3$	$[^3\mathbf{h}_2(3), {}^3y_2, {}^3d_{lat_3}]^T$	$[^3y_3, {}^3d_{lat_3}]^T$
$\mathbf{s}_4$	$[^4y_2, {}^4d_2]^T$	$^4\mathbf{h}_2(3)$
$\mathbf{s}_5$	–	$^5\mathbf{h}_2(3)$
$\mathbf{s}_6$	$^6\mathbf{h}_2(3)$	–
$\mathbf{s}_7$	$^7\mathbf{h}_3(3)$	$^7\mathbf{h}_3(3)$
$\mathbf{s}_8$	$^8\mathbf{h}_3(3)$	$^7\mathbf{h}_3(3)$

Constraints (reverse perpendicular case)

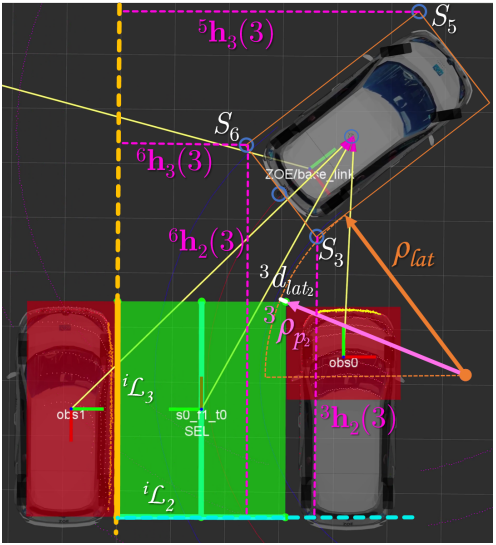


Figure: Constraints required for reverse ⊥ parking maneuvers

Constrained sensor features

$$\mathbf{s}^c = [\mathbf{s}_3, \mathbf{s}_5, \mathbf{s}_6]^T \qquad (15)$$

The corresponding interaction matrix:

$$\mathbf{L}^c = [\mathbf{L}_3, \mathbf{L}_5, \mathbf{L}_6]^T \qquad (16)$$

Control

Control

Control law

$$\mathbf{v} = \text{argmin}||\mathbf{L}_{\mathbf{H}}^{t}\mathbf{v} + \lambda \mathbf{e}_{\mathbf{H}}^{t}||^2 \quad \text{s.t. } \mathbf{A}\mathbf{v} \leq \mathbf{b} \tag{17}$$

with:

$$\mathbf{A} = [\mathbf{L}^c, -\mathbf{L}^c]^T \tag{18}$$

$$\mathbf{b} = [\alpha(\mathbf{s}^{c^+} - \mathbf{s}^c), -\alpha(\mathbf{s}^{c^-} - \mathbf{s}^c)]^T \tag{19}$$

where α is a gain constant, λ is the control gain and $[\mathbf{s}^{c^-}, \mathbf{s}^{c^+}]$ is the desired interval in which we want to keep $\mathbf{s}^c$.

Bounding the control signals

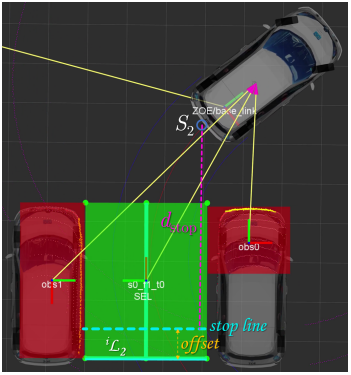


Figure: Distance to stop line

The control signals v and ϕ and their increments are bounded as shown below:

$$|v| \leq v_{max} \tag{20}$$

$$|\phi| \leq \phi_{max} \tag{21}$$

$$(v_{n-1} - \Delta_{dec}) \leq v_n \leq (v_{n-1} + \Delta_{acc}) \tag{22}$$

$$(\phi_{n-1} - \Delta_{\phi}) \leq \phi_n \leq (\phi_{n-1} + \Delta_{\phi}) \tag{23}$$

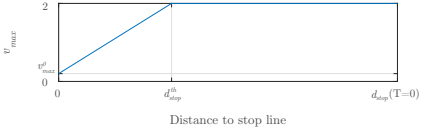


Figure: Deceleration profile

Results

Convergence Analysis - Exhaustive Simulations

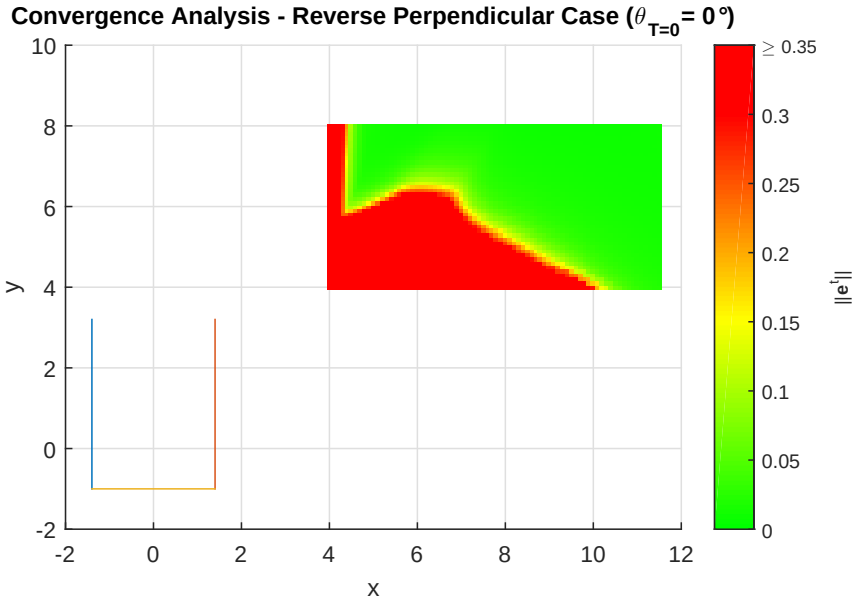


Figure: Reverse ⊥ case, spot length = 4m and width = 2.7m

Convergence Analysis - Exhaustive Simulations

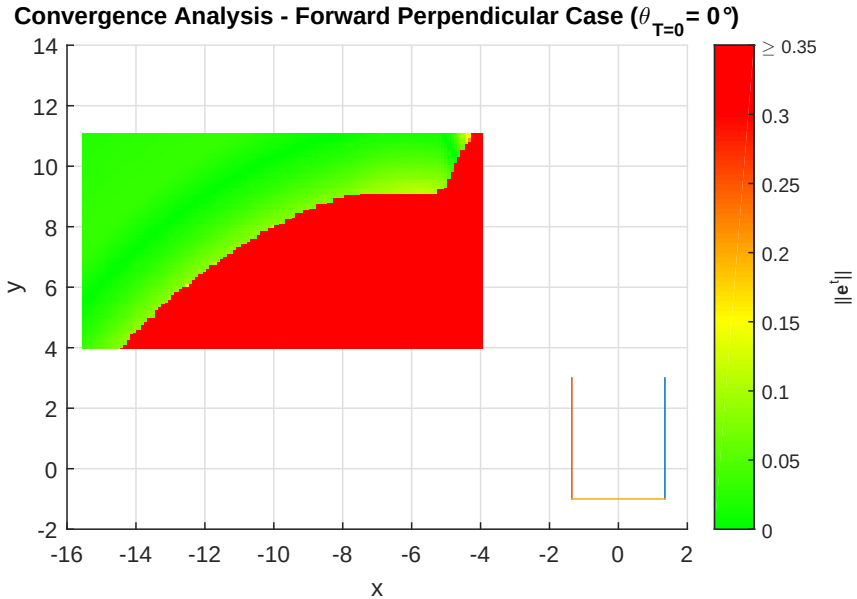


Figure: Forward ⊥ case, spot length = 4m and width = 2.7m

Convergence Analysis - Exhaustive Simulations

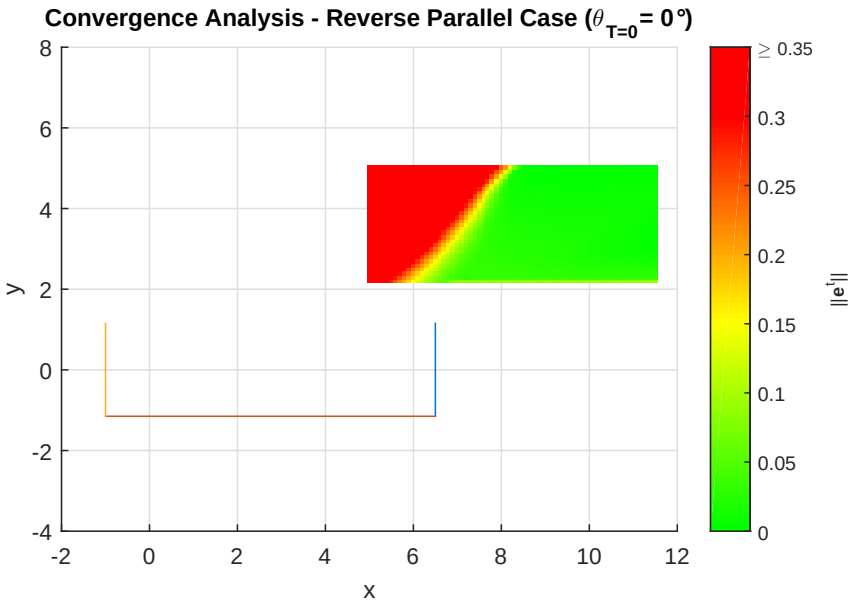


Figure: Reverse || case, spot length = 7.5m and width = 2.3m

Convergence Analysis - Exhaustive Simulations

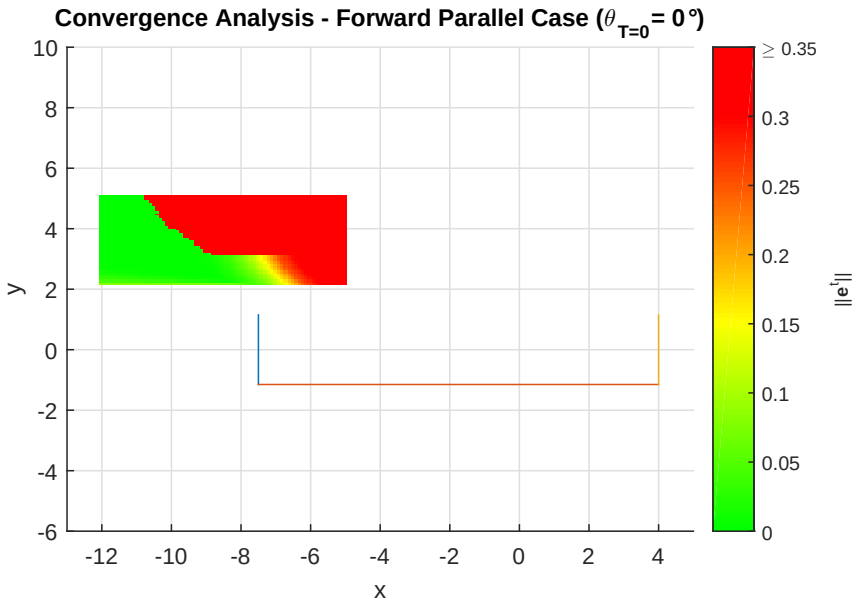


Figure: Forward || case, spot length = 11.5m and width=2.3m

Real Experimentation

Figure: Reverse $\perp$ parking maneuver (https://youtu.be/Lm5-pFiV5pA)

Play/Pause

Convergence Analysis - Real Experimentation

The initial position of the vehicle (denoted by a black ×) lies inside of the region of attraction (ROA).

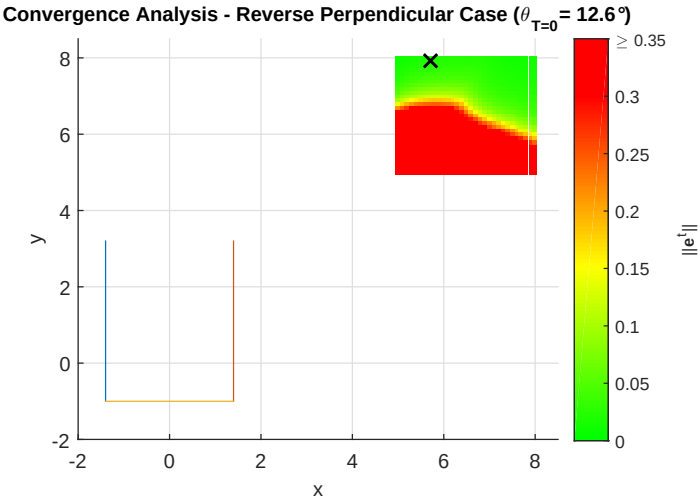
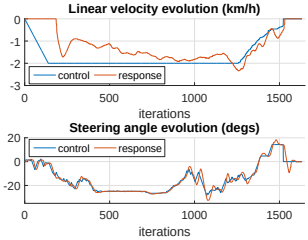
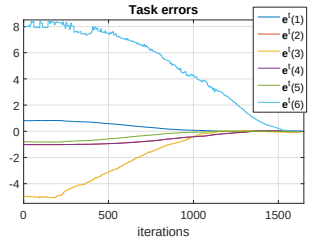


Figure: Reverse ⊥ case, spot length = 4.2m and width = 2.8m

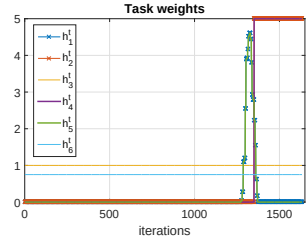
Real Experimentation - Parking Maneuver Signals



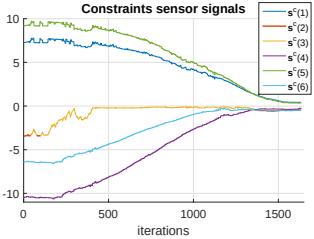
(a) Control signals



(b) Task error signal



(c) Task features' weights



(d) Constraints sensor signals

Figure: Reverse ⊥ parking maneuver signals

Conclusions and Future Work

Conclusions and Future Work

Conclusions:

- The presented technique has been proven to be very versatile and robust.
- The regions of attraction (ROAs) are quite extensive and their boundaries seem natural.

Future work:

- Validate the approach for other parking scenarios by real experimentation.
- To be able to park with multiple maneuvers.