

Non-linear Control under State Constraints with Validated Trajectories

Joris TILLET, Luc JAULIN, Fabrice LE BARS

ENSTA Bretagne

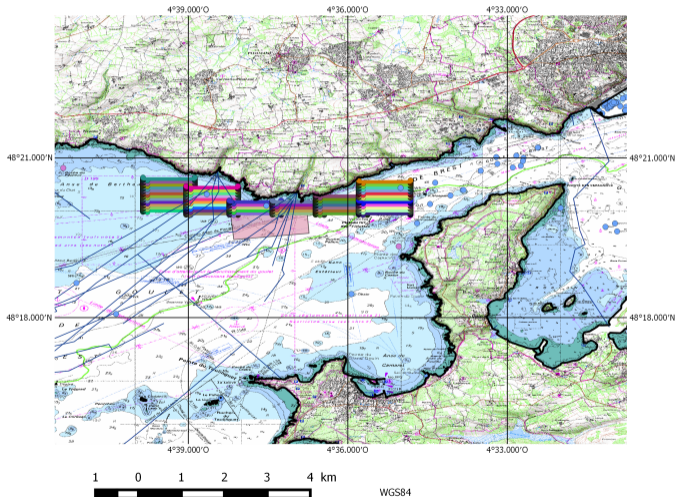
International Online Seminar on Interval Methods in Control Engineering
December 4, 2020



Searching for the *Cordelière*



Searching for the *Cordelière*



Searching for the *Cordelière*



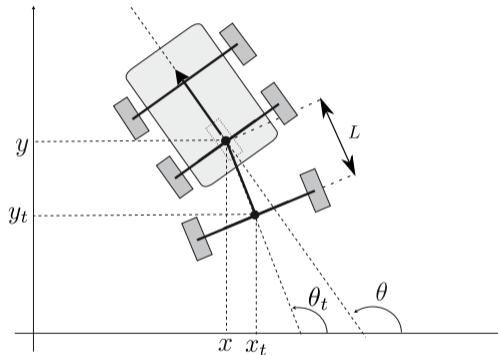
- 1 Command of a car-trailer system
- 2 Constraint validation

- 1 Command of a car-trailer system
- 2 Constraint validation

Chosen model

At any time the robot is described by its state vector \mathbf{x} :

$$\mathbf{x} = \begin{pmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \end{pmatrix} = \begin{pmatrix} x \\ y \\ \theta \\ \theta_t \\ v \end{pmatrix} .$$



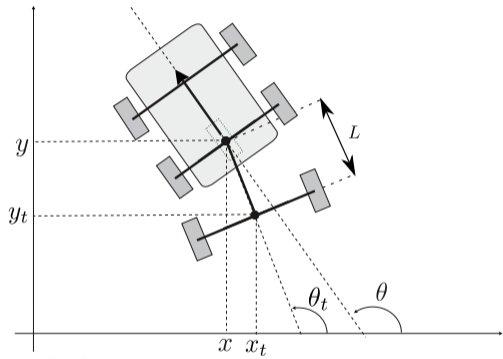
Its state is measured with sensors:

$$\mathbf{y}_{\text{mes}} = \mathbf{h}(\mathbf{x}).$$

Chosen model

Its dynamics is given by the following differential equation where f is the evolution function of the system:

$$\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}, \mathbf{u}) = \begin{cases} x_5 \cos(x_3) \\ x_5 \sin(x_3) \\ u_1 \\ \frac{x_5}{L} \sin(x_3 - x_4) \\ u_2 \end{cases}$$



$\mathbf{u} = \begin{pmatrix} u_1 \\ u_2 \end{pmatrix}$ is the input of the system, used to control the robot.

Objective: make the trailer follow a dynamics $\Psi(\mathbf{y})$
 \Leftrightarrow make the error \mathbf{e} converges toward 0.

$$\mathbf{e} = \dot{\mathbf{y}} - \Psi(\mathbf{y}).$$

Command law

The feedback linearization method is used to find the command law. We use the following equivalent flattened system:

$$\mathbf{f}_z(\mathbf{z}) + \mathbf{g}_z(\mathbf{z}) \cdot \mathbf{a} = \begin{pmatrix} \dot{z}_1 \\ \dot{z}_2 \\ \dot{z}_3 \\ \dot{z}_4 \\ \dot{z}_5 \\ \dot{z}_6 \end{pmatrix} = \begin{pmatrix} z_3 \cos(z_5) \\ z_3 \sin(z_5) \\ z_4 \\ 0 \\ z_6 \\ 0 \end{pmatrix} + \begin{pmatrix} 0 \\ 0 \\ 0 \\ a_1 \\ 0 \\ a_2 \end{pmatrix}$$

with z_5 the heading of the trailer, (z_3, z_6) the speed vector of the front car expressed in the trailer frame.

a_1 controls the acceleration of the trailer and a_2 its rotation rate.

As output, we have the center of the trailer $\mathbf{h}_z(\mathbf{z}) = \begin{pmatrix} z_1 \\ z_2 \end{pmatrix}$.

We compute the successive derivatives of the error until we get a in an expression.

$$\mathbf{e} = \dot{\mathbf{y}} - \Psi(\mathbf{y}) \quad (1)$$

$$= \begin{pmatrix} \dot{z}_1 \\ \dot{z}_2 \end{pmatrix} - \begin{pmatrix} y_2 \\ -(y_1^2 - 1)y_2 - y_1 \end{pmatrix} \quad (2)$$

$$= \begin{pmatrix} z_3 \cos(z_5) \\ z_3 \sin(z_5) \end{pmatrix} - \begin{pmatrix} y_2 \\ -(y_1^2 - 1)y_2 - y_1 \end{pmatrix} \quad (3)$$

Here Ψ is the Van der Pol vector field.

Using Lie derivatives:

$$\mathbf{e} = \mathcal{L}_{\mathbf{f}_z} \mathbf{h}_z(\mathbf{z}) - \Psi(\mathbf{h}_z(\mathbf{z})) \quad (4)$$

$$\dot{\mathbf{e}} = \mathcal{L}_{\mathbf{f}_z}^2 \mathbf{h}_z(\mathbf{z}) - \mathcal{L}_{\mathbf{f}_z} \Psi(\mathbf{h}_z(\mathbf{z})) \quad (5)$$

$$\ddot{\mathbf{e}} = \mathcal{L}_{\mathbf{f}_z}^3 \mathbf{h}_z(\mathbf{z}) + \mathcal{L}_{\mathbf{g}_z} \mathcal{L}_{\mathbf{f}_z}^2 \mathbf{h}_z(\mathbf{z}) \cdot a - \mathcal{L}_{\mathbf{f}_z}^2 \Psi(\mathbf{h}_z(\mathbf{z})) \quad (6)$$

Using Lie derivatives:

$$\mathbf{e} = \mathcal{L}_{\mathbf{f}_z} \mathbf{h}_z(\mathbf{z}) - \Psi(\mathbf{h}_z(\mathbf{z})) \quad (4)$$

$$\dot{\mathbf{e}} = \mathcal{L}_{\mathbf{f}_z}^2 \mathbf{h}_z(\mathbf{z}) - \mathcal{L}_{\mathbf{f}_z} \Psi(\mathbf{h}_z(\mathbf{z})) \quad (5)$$

$$\ddot{\mathbf{e}} = \mathcal{L}_{\mathbf{f}_z}^3 \mathbf{h}_z(\mathbf{z}) + \mathcal{L}_{\mathbf{g}_z} \mathcal{L}_{\mathbf{f}_z}^2 \mathbf{h}_z(\mathbf{z}) \cdot a - \mathcal{L}_{\mathbf{f}_z}^2 \Psi(\mathbf{h}_z(\mathbf{z})) \quad (6)$$

Using Lie derivatives:

$$\mathbf{e} = \mathcal{L}_{\mathbf{f}_z} \mathbf{h}_z(\mathbf{z}) - \Psi(\mathbf{h}_z(\mathbf{z})) \quad (4)$$

$$\dot{\mathbf{e}} = \mathcal{L}_{\mathbf{f}_z}^2 \mathbf{h}_z(\mathbf{z}) - \mathcal{L}_{\mathbf{f}_z} \Psi(\mathbf{h}_z(\mathbf{z})) \quad (5)$$

$$\ddot{\mathbf{e}} = \mathcal{L}_{\mathbf{f}_z}^3 \mathbf{h}_z(\mathbf{z}) + \mathcal{L}_{\mathbf{g}_z} \mathcal{L}_{\mathbf{f}_z}^2 \mathbf{h}_z(\mathbf{z}) \cdot a - \mathcal{L}_{\mathbf{f}_z}^2 \Psi(\mathbf{h}_z(\mathbf{z})) \quad (6)$$

We want e to converge exponentially toward 0:

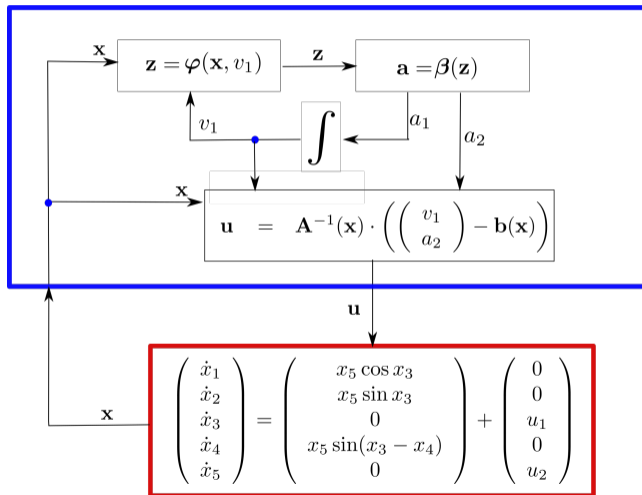
$$\ddot{e} + 2\dot{e} + e = 0$$

The full expression of the solution is computed with symbolic calculus.
Then we obtain the expression for $\mathbf{a} = \beta(\mathbf{z})$, and deduce the one for $\mathbf{u}(\mathbf{x})$.

Finally:

$$\mathbf{f}(\mathbf{x}, \mathbf{u}) = \mathbf{f}(\mathbf{x}, \mathbf{u}(\mathbf{x})) = \mathbf{f}^*(\mathbf{x})$$

Command law



$$A(x) = \begin{pmatrix} \mathcal{L}_{g_1} z_3 & \mathcal{L}_{g_2} z_3 \\ \mathcal{L}_{g_1} \mathcal{L}_f z_5 & \mathcal{L}_{g_2} \mathcal{L}_f z_5 \end{pmatrix}$$

$$b(x) = \begin{pmatrix} \mathcal{L}_f z_3 \\ \mathcal{L}_f^2 z_5 \end{pmatrix}$$

- 1 Command of a car-trailer system
- 2 Constraint validation**

Constraint expression

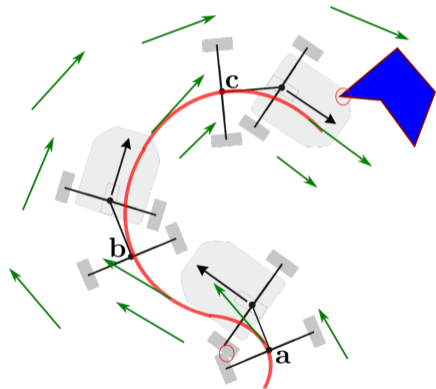
The constraint is expressed as a condition on a function of the state vector:

$$\begin{cases} \dot{\mathbf{x}} &= \mathbf{f}^*(\mathbf{x}) \\ \mathbf{H}(\mathbf{x}) &\geq 0 \end{cases} .$$

For instance (case **a** in the Figure):

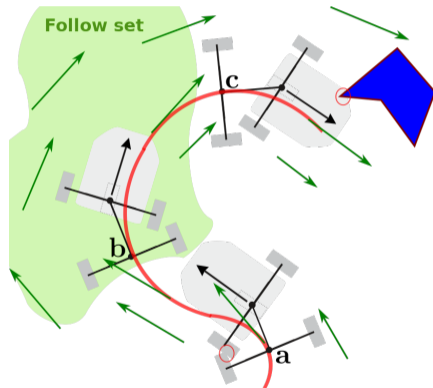
$$\cos(\theta - \theta_t) - \cos(\delta_{lim}) \geq 0$$

where δ_{lim} is the maximum angle of turn of the trailer.



Goal

The goal is to validate the controller with respect to the constraints.
It means we want to compute a "follow set", which represents every place the robot can go safely.



State vector computing

For each output position y , there is only one state x which is possible:

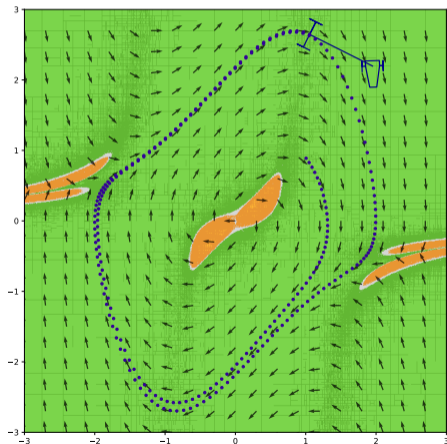
$$\begin{cases} \theta_t = & \arctan2(\dot{y}_2, \dot{y}_1) \\ x, y = & \phi(y_1, y_2, \theta_t) \\ \theta = & \arctan2(\dot{y}, \dot{x}) \end{cases},$$

where ϕ is the function that computes the robot position from the trailer one.
We can compute the expression of the constraint.

To compute the follow set in a guaranteed way, we use tools from interval analysis:
For each 2D-box $[y]$, the interval vector $[x]$ which contains all possible states in this box is computed.
Then the box is painted in green if the constraints are respected, and in orange otherwise.

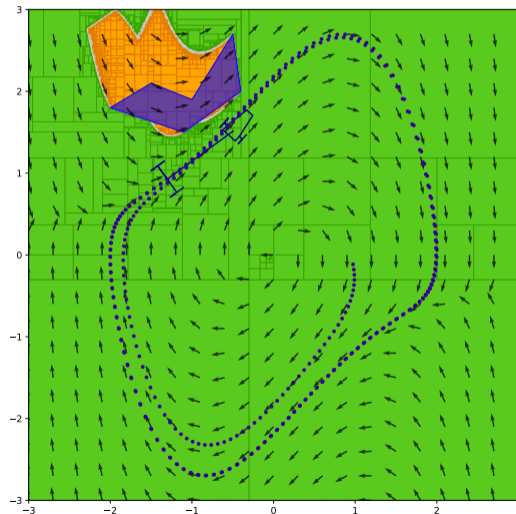
Result

Now we place this result as a background of the simulation on the vector field, and so we can ensure consistency.



Result

An obstacle is represented by the purple polygon.



- Control of a non-linear system with feedback linearization ;
- Automatization of complex calculations of Lie derivative ;
- Expressions of the constraints ;
- Computation of the follow set with interval analysis.

- Work with n trailers and control the position of the last one ;
- Reachability analysis could be performed to find viable domains ;
- Find the *Cordelière*.

Non-linear Control under State Constraints with Validated Trajectories

Joris TILLET, Luc JAULIN, Fabrice LE BARS

ENSTA Bretagne

International Online Seminar on Interval Methods in Control Engineering
December 4, 2020

