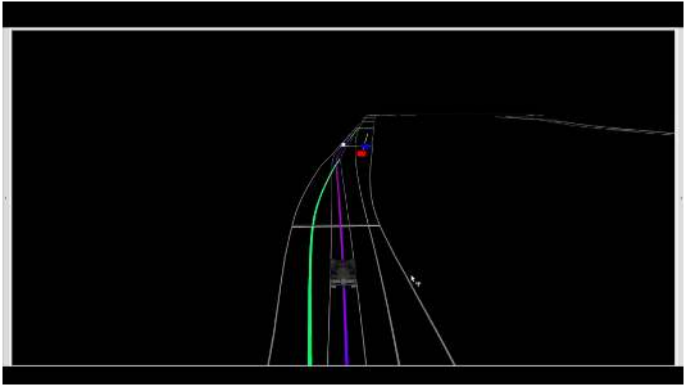
Frenet-Cartesian Model Representations for Automotive Obstacle Avoidance within Nonlinear MPC

Rudolf Reiter, Armin Nurkanović, Jonathan Frey and Moritz Diehl

Systems Control and Optimization Laboratory (syscop)

European Control Conference 2023
June 15, 2023



→play-full-video

- Task: Improve numerical properties of real-time control of autonomous vehicles with obstacles

- Basic approach: use optimization-based control: (Cartesian) NMPC
- Problem: nonconvexities and nonlinearities

- Variation: transform model into curvilinear coordinate frame (Frenet Frame)
- Problem: new coordinate frame makes part of problem more non-smooth

- Our idea: Use redundantly two coordinate frames
- Questions: How to formulate it? Speedup? Other advantages?

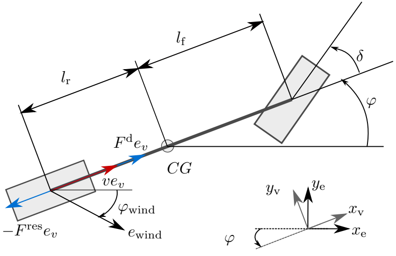
Outline

1. Modeling in two coordinate frames
2. Ways to combine both models
3. Obstacle avoidance
4. NMPC Algorithm
5. Results
6. Conclusion and Discussion

Modeling in two coordinate frames

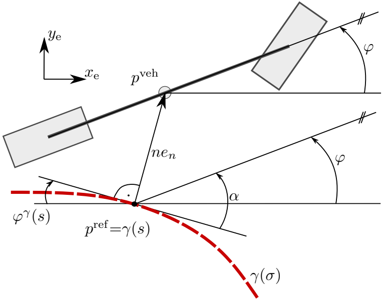
Cartesian Coordinate Frame (CCF)

- Global states $x^{\mathrm{c,C}} = [p_x, p_y, \varphi]^\top$
- Usual physical models
- Some states are CF independent: $x^{\neg\mathrm{c}} = [v, \delta]^\top$

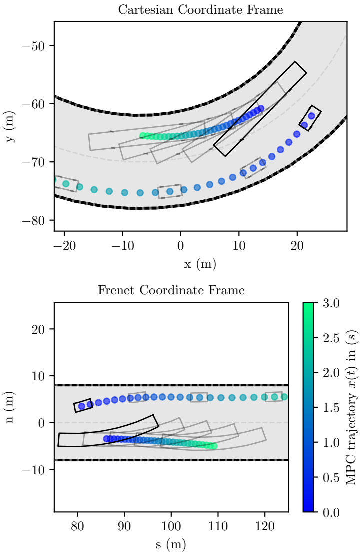


Frenet Coordinate Frame (FCF)

- Projection of states onto road reference $x^{\mathrm{c,F}} = \mathcal{F}_\gamma(x^{\mathrm{c,C}}) = [s, n, \alpha]^\top$
- Results in dynamic system that depends on curvature $\kappa(s)$
- $s^*(p^{\mathrm{veh}}) = \arg\min_\sigma \left\|p^{\mathrm{veh}} - \gamma(\sigma)\right\|_2^2$



Modeling in two coordinate frames



Feature	CCF	FCF
reference definition	✗	✓
boundary constraints	✗	✓
obstacle specification	✓	✗
disturbance specification	✓	✗

Ways to combine both models

Goal:

- Reference definition, boundary constraints $\rightarrow$ FCF
- Obstacle specification, Cartesian disturbance (e.g., wind force) $\rightarrow$ CCF

Possible formulations of NMPC:

- Use only one CF, approximate and simplify non-smooth constraints
- Model dynamics in *one* CF, use $\mathcal{F}_\gamma$ or ${\mathcal{F}_\gamma}^{-1}$ to obtain *other* states
- Model dynamics redundantly in *both* CFs

Possible formulations of NMPC:

- Use only one CF, approximate and simplify non-smooth constraints
  - Main frame CCF: approximate $\mathcal{F}_\gamma$ with artificial path state (MPCC) *(Not reviewed here)*
  - Main frame FCF: over-approximate obstacles $\rightarrow$ conventional
- Model dynamics in *one* CF, use $\mathcal{F}_\gamma$ or ${\mathcal{F}_\gamma}^{-1}$ to obtain *other* states
- Model dynamics redundantly in *both* CFs

Ways to combine both models

Direct elimination

Possible formulations of NMPC:

- Use only one CF, approximate and simplify non-smooth constraints
- Model dynamics in *one* CF, use $\mathcal{F}_\gamma$ or ${\mathcal{F}_\gamma}^{-1}$ to obtain *other* states
  - Main frame CCF ✗: $\mathcal{F}_\gamma$ is an nonlinear optimization problem by itself
  - Main frame FCF ✓: ${\mathcal{F}_\gamma}^{-1}$ can be obtained efficiently $\rightarrow$ direct elimination
- Model dynamics redundantly in *both* CFs

Ways to combine both models

Lifting

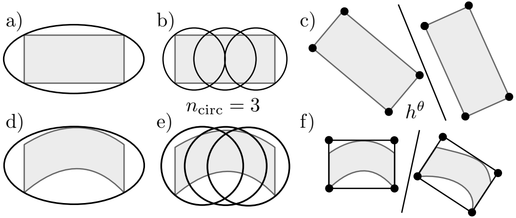
Possible formulations of NMPC:

- Use only one CF, approximate and simplify non-smooth constraints
- Model dynamics in *one* CF, use $\mathcal{F}_\gamma$ or ${\mathcal{F}_\gamma}^{-1}$ to obtain *other* states
- Model dynamics redundantly in *both* CFs
  - Lifting to higher dimension
  - Number of states n_x increases from 5 to 8 $\rightarrow$ lifting

Obstacle avoidance

Comparison of several different obstacle avoidance formulations

1. Ellipse - circle
2. Covering circles
3. Separating hyper-planes



NMPC Problem

Direct elimination

$$
\begin{aligned}
\min_{\substack{x_0^{\mathrm{F}},\ldots,x_N^{\mathrm{F}},\\ u_0,\ldots,u_{N-1}\\ \theta_1,\ldots,\theta_{n_{\mathrm{opp}}}}} \quad & \sum_{k=0}^{N-1} \|u_k\|_R^2 + \left\|x_k^{\mathrm{F}} - x_{\mathrm{ref},k}^{\mathrm{F}}\right\|_Q^2 + \left\|x_N^{\mathrm{F}} - x_{\mathrm{ref},N}^{\mathrm{F}}\right\|_{Q_N}^2 \\
\text{s.t.} \quad & x_0^{\mathrm{F}} = \hat{x}_0^{\mathrm{F}}, \\
& x_{i+1}^{\mathrm{F}} = \Phi^{\mathrm{F}}(x_i^{\mathrm{F}}, u_i, \Delta t), \qquad i = 0,\ldots,N-1, \\
& \underline{u} \leq u_i \leq \overline{u}, \qquad i = 0,\ldots,N-1, \\
& \underline{x}^{\mathrm{F}} \leq x_i^{\mathrm{F}} \leq \overline{x}^{\mathrm{F}}, \qquad i = 0,\ldots,N, \\
& \underline{x}^{\mathrm{c,C}} \leq \mathcal{F}_\gamma^{-1}(x^{\mathrm{c,F}}) \leq \overline{x}^{\mathrm{c,C}}, \qquad i = 0,\ldots,N, \\
& \underline{a}^{\mathrm{lat}} \leq a_{\mathrm{lat}}^{\mathrm{F}}(x_i) \leq \overline{a}^{\mathrm{lat}}, \qquad i = 0,\ldots,N, \\
& v_N \leq \overline{v}_N, \\
& \mathcal{F}_\gamma^{-1}(x^{\mathrm{c,F}}) \in \mathcal{P}(x_i^{\mathrm{c,opp,j}}, \theta_j), \qquad i = 0,\ldots,N-1, \\
& \qquad j = 1,\ldots,n_{\mathrm{opp}}.
\end{aligned}
\tag{1}
$$

$x^{\mathrm{F}} \in \mathbb{R}^5 \ldots$ Frenet states, $x^{\mathrm{c,C}} \in \mathbb{R}^3 \ldots$ Cartesian position states, $\mathcal{P} \ldots$ obstacle-free set
$\theta \ldots$ hyperplane variables, $\mathcal{F}_\gamma^{-1} \ldots$ inverse Frenet transformation, $\Phi^{\mathrm{F}}(\cdot) \ldots$ integrator

NMPC Problem

Lifted

$$
\begin{aligned}
\min_{\substack{x_0^{\mathrm{d}},\dots,x_N^{\mathrm{d}},\\ u_0,\dots,u_{N-1}\\ \theta_1,\dots,\theta_{n_{\mathrm{opp}}}}} \quad & \sum_{k=0}^{N-1} \|u_k\|_R^2 + \left\|x_k^{\mathrm{F}} - x_{\mathrm{ref},k}^{\mathrm{F}}\right\|_Q^2 + \left\|x_N^{\mathrm{F}} - x_{\mathrm{ref},N}^{\mathrm{F}}\right\|_{Q_N}^2 \\
\text{s.t.} \quad & x_0^{\mathrm{d}} = \hat{x}_0^{\mathrm{d}}, \\
& x_{i+1}^{\mathrm{d}} = \Phi^{\mathrm{d}}(x_i^{\mathrm{d}}, u_i, \Delta t), i = 0, \dots, N-1, \\
& \underline{u} \leq u_i \leq \overline{u}, \qquad i = 0, \dots, N-1, \\
& \underline{x}^{\mathrm{d}} \leq x_i^{\mathrm{d}} \leq \overline{x}^{\mathrm{d}}, \qquad i = 0, \dots, N, \\
& \underline{a}^{\mathrm{lat}} \leq a_{\mathrm{lat}}(x_i^{\mathrm{d}}) \leq \overline{a}^{\mathrm{lat}}, i = 0, \dots, N, \\
& v_N \leq \overline{v}_N, \\
& x_i^{\mathrm{c,C}} \in \mathcal{P}(x_i^{\mathrm{c,opp,j}}, \theta_j), \; i = 0, \dots, N-1, \\
& \qquad j = 1, \dots, n_{\mathrm{opp}}.
\end{aligned} \tag{2}
$$

$x^{\mathrm{F}} \in \mathbb{R}^5 \dots$ Frenet states, $x^{\mathrm{d}} \in \mathbb{R}^8 \dots$ lifted states, $\mathcal{P} \dots$ obstacle-free set
$\theta \dots$ hyperplane variables, $\Phi^{\mathrm{d}}(\cdot) \dots$ model integration function

Results

Setup:

- Simulation on randomized scenarios with three obstacles to overtake
- Two scenarios:
  - Truck-sized obstacles
  - Car-sized obstacles
- Obstacle formulations:
  - Ellipsoids
  - Covering circles (1,3,5,7)
  - Separating hyper-planes
- Coordinate formulations:
  - Conventional (over-approximation)
  - Direct elimination
  - Lifted ODE

Evaluation:

- Computation time
- Maximum progress

Results

car-sized

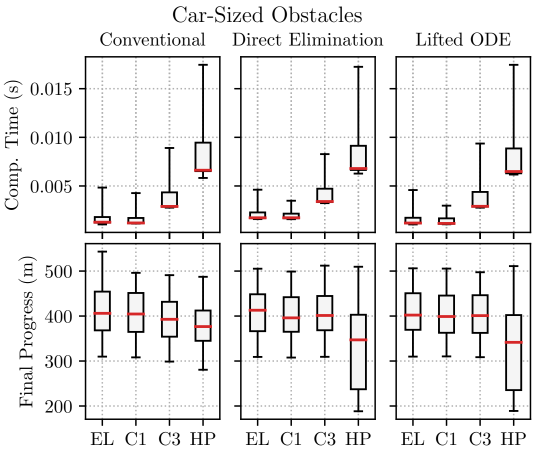


Figure: Box-plot comparison of the NMPC solution timings for each real-time iteration and the final progress after 20 seconds for different obstacle formulations for car-sized vehicles.

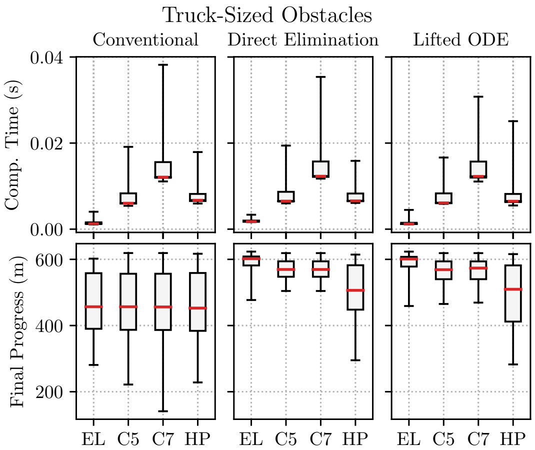
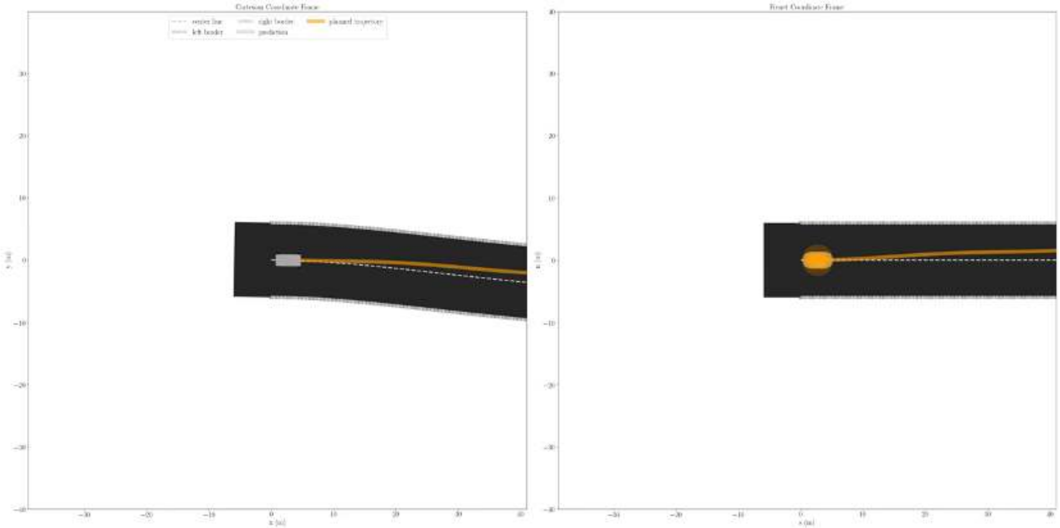


Figure: Box-plot comparison of the NMPC solution timings for each real-time iteration and the final progress after 20 seconds for different obstacle formulations for truck-sized vehicles.

Computation times (ms) for truck-sized obstacles	Conventional	Direct Elimination		Lifted ODE	
EL	1.5 ± 0.4	1.9 ± 0.2	28.9%	$\mathbf{1.4 \pm 0.3}$	$\mathbf{-6.6\%}$
C5	7.2 ± 1.9	7.6 ± 1.7	5.5%	7.2 ± 1.8	-0.0%
C7	14.0 ± 3.2	14.0 ± 2.8	-0.1%	13.9 ± 2.9	-0.4%
HP	7.5 ± 1.5	7.5 ± 1.5	-0.1%	7.4 ± 1.7	-1.6%
car-sized obstacles					
EL	1.5 ± 0.5	2.0 ± 0.4	29.6%	$\mathbf{1.4 \pm 0.4}$	$\mathbf{-5.7\%}$
C1	1.4 ± 0.4	1.9 ± 0.4	34.0%	1.4 ± 0.4	-3.5%
C3	3.6 ± 1.1	4.0 ± 1.0	12.4%	3.6 ± 1.1	0.6%
HP	8.0 ± 2.3	7.9 ± 1.9	-0.6%	7.7 ± 2.0	-4.0%

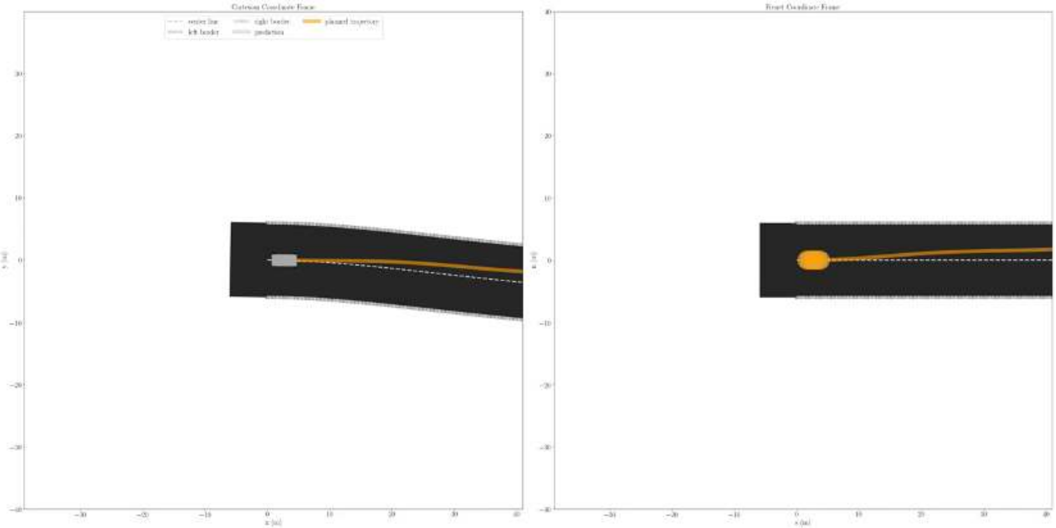
Table: Mean and standard deviation of computation times for different scenarios, obstacle formulations and lifting formulations. Additionally, the difference in percent to the conventional formulation is given.

Under-approximation (using Cartesian dimensions in FCF)

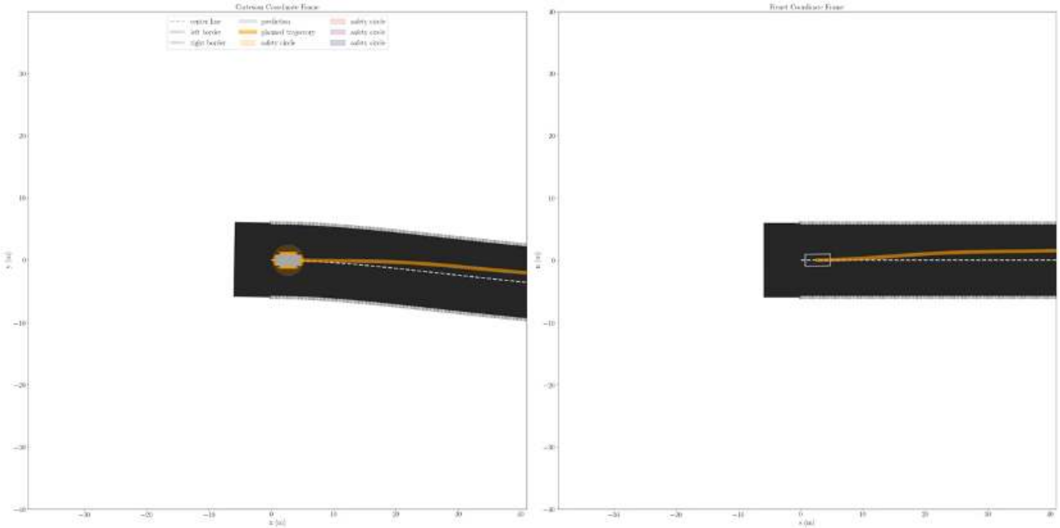


→play-full-video

Safe over-approximation



→play-full-video



→play-full-video

Conclusion and Discussion

- Our idea: Use redundantly two coordinate frames
- Questions:
  Different formulaitons?✓
  Speedup? ✓(-6%)
  Advantages of both CF? ✓$(+25\%$ progress)

Conclusion:

- For large obstacle, the accurate representation is important. If over-approximated, performance loss.
- Using ellipsoidal obstacle constraints in the lifted expression is fastest and accurate
- Formulation general enough for higher fidelity models

Not considered and future work:

- How do results relate to Model Predictive Contouring Control (Cartesian-Based with *simple target model* along reference curve)

Thank you for your attention!

universität freiburg

ELO-X