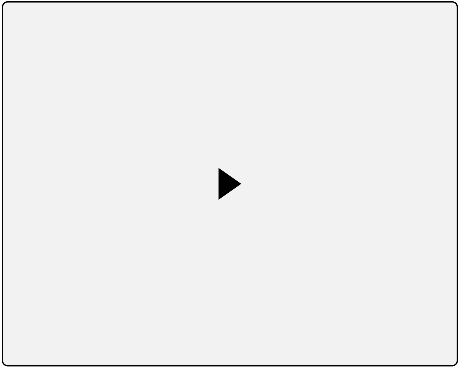
Realistic Optimization-based Driving Using a Constrained Double-Integrator Model

Andreas Belavic

Technical University of Munich

March 31, 2025

Animation



Motivation & Problem Statement

Given a 7-tuple $(\mathcal{X}, \mathcal{U}, x_{\text{initial}}, X_{\text{goal}}, f, J, \{t_i\}_{i=1,\ldots,m})$, the discrete-time optimal trajectory planning problem is defined as:

$$u^* = \underset{u \in \mathcal{U}^{m-1}}{\arg\min} \sum_{i=1}^{m-1} J(x_{i+1}, u_i), \tag{1}$$

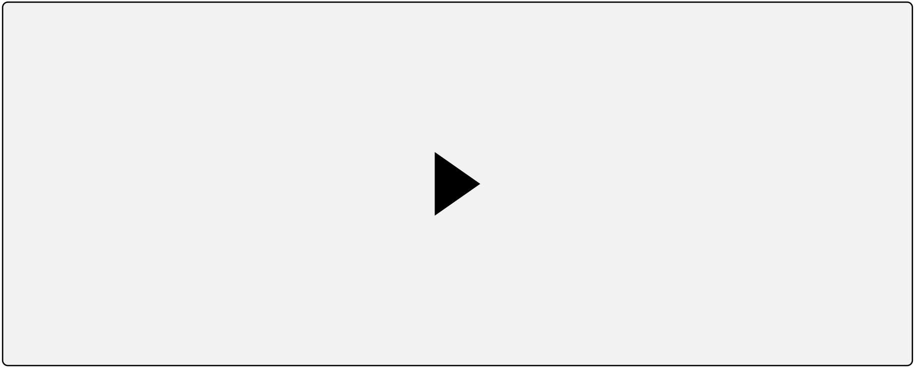
$$\text{s.t.} \quad x_1 = x_{\text{initial}} \tag{2}$$

$$x_m \in X_{\text{goal}} \subseteq \mathcal{X} \tag{3}$$

$$(x_i, u_i) \in \mathcal{C} \subseteq \mathcal{X} \times \mathcal{U} \qquad \text{for all } i \in \{1, \ldots, m-1\} \tag{4}$$

$$x_{i+1} = f(x_i, u_i, \Delta t_i) \qquad \text{for all } i \in \{1, \ldots, m-1\} \tag{5}$$

Convex vs Non-Convex



Solver Times

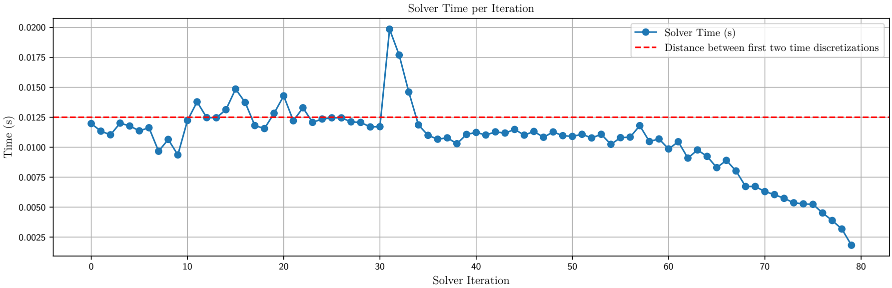


Figure: Convex

Solver Times

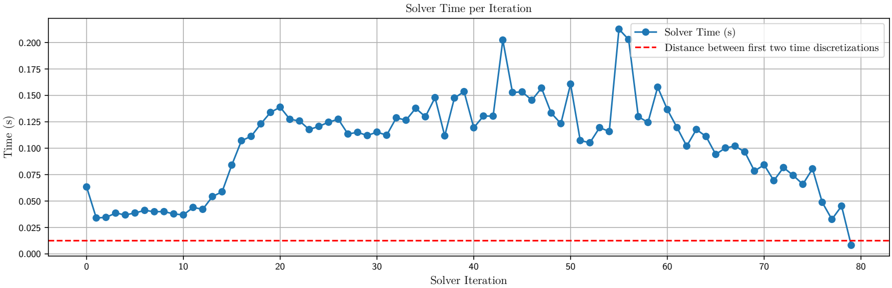
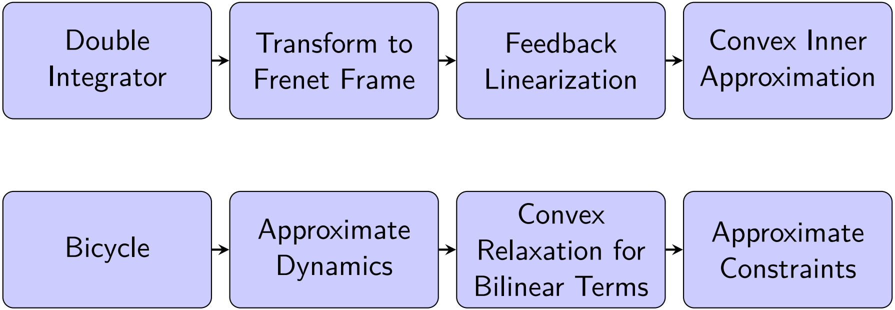


Figure: Non-Convex

Convex Optimization in Motion Planning

- Non-convex formulations often lack robust convergence criteria.
- Our approach transforms the problem into a convex formulation to ensure:
  - Reliable convergence guarantees.
  - Predictable and efficient computation times.
  - Efficient checkability of the convexity.

Overview of the Methodology



Vehicle Modeling: Double Integrator Model (DI)

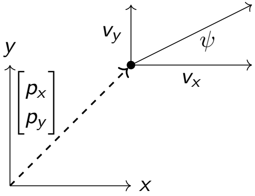


Figure: Double Integrator Model

- Transform to Frenet Frame, introduce Assumption $\xi = 0$.
- Feedback Linearization.
- Convex Inner Approximation.

Coordinate Transformation: Frenet Frame

- Transforms global coordinates into a path-following system.
- Simplifies handling of road curvature and lateral deviations.
- This transformation introduces nonlinear vehicle dynamics.

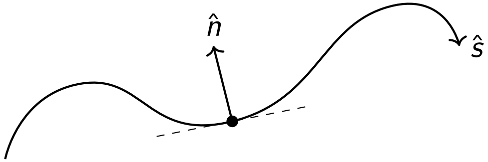


Figure: Frenet Frame Representation

$$\xi = \psi - \theta, \text{ where } \theta \text{ is the angle of the reference path.} \tag{6}$$

$$C(s) = \frac{d\theta}{ds} \tag{7}$$

Resulting Model & Simplifications

$$x_{di} = \begin{bmatrix} s & n & \xi & \dot{s} & \dot{n} & \dot{\psi} \end{bmatrix}^T \tag{8}$$

$$u_{di} = \begin{bmatrix} a_x & a_y & a_\psi \end{bmatrix}^T \tag{9}$$

$$\overbrace{\begin{bmatrix} \dot{s} \\ \dot{n} \\ \dot{\psi} - C(s)\dot{s} \\ \dfrac{a_{x,tn} + 2\dot{n}\, C(s)\dot{s} + n\, C'(s)\dot{s}^2}{1 - n\, C(s)} \\ a_{y,tn} - C(s)\dot{s}^2(1 - n\, C(s)) \\ a_\psi \end{bmatrix}}^{\dot{x}_{di} =} \quad \overset{\xi = 0}{\Longrightarrow} \quad \begin{bmatrix} \dot{s} \\ \dot{n} \\ 0 \\ \dfrac{a_x + 2\dot{n}\, C(s)\dot{s} + n\, C'(s)\dot{s}^2}{1 - n\, C(s)} \\ a_y - C(s)\dot{s}^2(1 - n\, C(s)) \\ \ddot{\theta} \end{bmatrix}$$

Feedback Linearization Technique

- Linearize the vehicle's nonlinear dynamics.

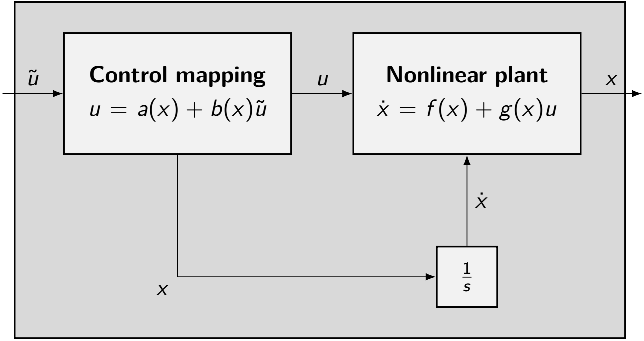


Figure: Feedback control structure for a nonlinear system.

Technique detailed in Section 3.1.4 of the thesis.

Resulting Model

$$\tilde{x}_{di} = \begin{bmatrix} s, & n, & \dot{s}, & \dot{n} \end{bmatrix}^T \tag{10}$$

$$\tilde{u}_{di} := \begin{bmatrix} u_t & u_n \end{bmatrix}^T \tag{11}$$

$$\frac{d\tilde{x}_{di}}{dt} = \begin{bmatrix} \dot{s} \\ \dot{n} \\ u_t \\ u_n \end{bmatrix} \tag{12}$$

Constraint Handling in the Framework

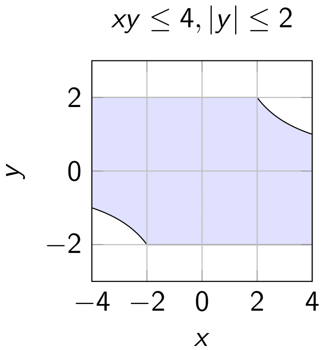
- Feedback linearization leads to nonlinear physical constraints: acceleration, velocity, and road boundaries.
- Using convex inner approximations for non-convex constraints.

$$\hat{C} = \left\{ \begin{bmatrix} \tilde{x}_{di} \\ \tilde{u}_{di} \end{bmatrix} \middle| \; N \begin{bmatrix} \tilde{x}_{di} \\ \tilde{u}_{di} \end{bmatrix} \leq b \right\} \subseteq \mathcal{C}, \tag{13}$$

- We apply quantifier elimination, using two approaches:
  - Interval Fitting and
  - Cylindrical Algebraic Decomposition (CAD).

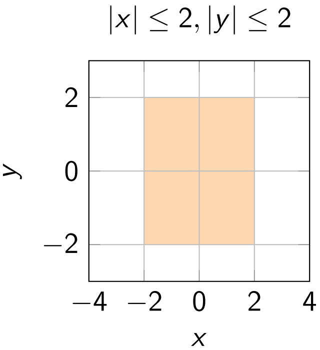
$$\tilde{C} = \left\{ \begin{bmatrix} \dot{s} \\ u_t \\ u_n \end{bmatrix} \middle| \; \begin{bmatrix} \tilde{x}_{di} \\ \tilde{u}_{di} \end{bmatrix} \in \mathcal{C}, \quad \forall \begin{bmatrix} s \\ n \\ \dot{n} \end{bmatrix} \in \begin{bmatrix} \underline{s}, \overline{s} \\ \underline{n}, \overline{n} \\ \underline{\dot{n}}, \overline{\dot{n}} \end{bmatrix} \right\}. \tag{14}$$

Interval Fitting Illustration



$\Longrightarrow$

Interval Fitting



Cylindrical Algebraic Decomposition

- It is applied to polynomials and divides the space into cylindrical cells.
- Example for $x^2 + bx + 1$.

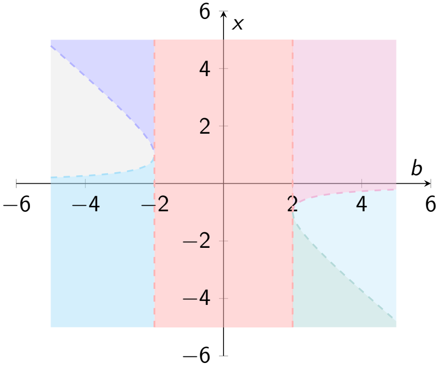


Figure: Illustrating the cells with shaded regions.

Vehicle Modeling: Kinematic Bicycle Model (KST)

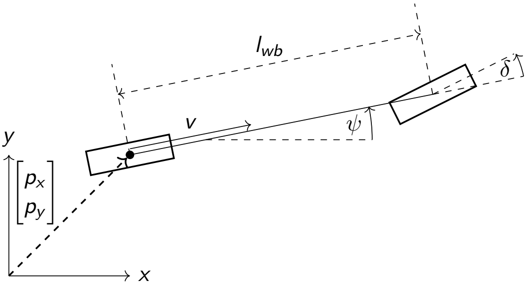


Figure: Bicycle model representation of a vehicle.

- Transform to Frenet Frame.
- Approximate Dynamics.
- Convex Relaxation for Bilinear Terms.
- Approximate Constraints.

KST Dynamics

$$x_{kst} = \begin{bmatrix} s & n & \xi & v & \delta \end{bmatrix}^T \tag{15}$$

$$u_{kst} = \begin{bmatrix} a & v_\delta \end{bmatrix}^T \tag{16}$$

$$\dot{x}_{kst} = \begin{bmatrix} \frac{v \cos \xi}{1 - nC(s)} \\ v \sin \xi \\ \frac{1}{l_{wb}} v \tan \delta - C(s)\dot{s} \\ a \\ v_\delta \end{bmatrix} \tag{17}$$

Small Angle Approximation

- For small steering angles, trigonometric functions can be approximated.
- This simplifies the nonlinear equations in the bicycle model.

$$\begin{bmatrix} \frac{v\cos\xi}{1-nC(s)} \\ v\sin\xi \\ \frac{1}{l_{wb}}v\tan\delta - C(s)\dot{s} \\ a \\ v_\delta \end{bmatrix} \tag{18}$$

McCormick Relaxation for Bilinear Terms

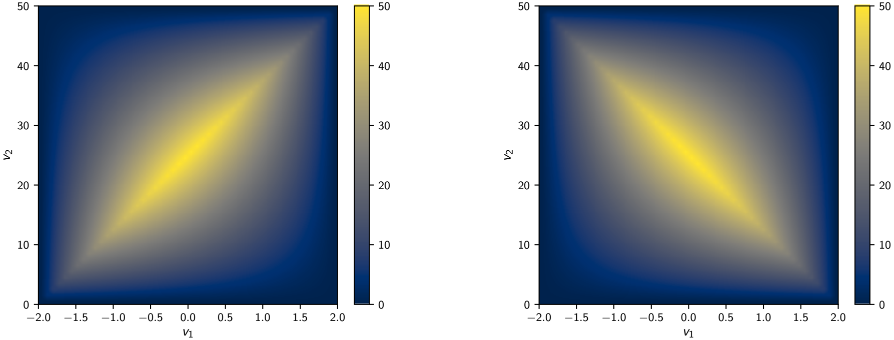
- McCormick relaxation provides a convex envelope for bilinear terms such as $w = v_1 v_2$.
- Given $v_1 \in [\underline{v_1}, \overline{v_1}]$ and $v_2 \in [\underline{v_2}, \overline{v_2}]$, the McCormick envelopes are:

$$
\begin{aligned}
w &\geq \underline{v_1} v_2 + \underline{v_2} v_1 - \underline{v_1}\,\underline{v_2}, \\
w &\geq \overline{v_1} v_2 + \overline{v_2} v_1 - \overline{v_1}\,\overline{v_2}, \\
w &\leq \overline{v_1} v_2 + \underline{v_2} v_1 - \overline{v_1}\,\underline{v_2}, \\
w &\leq \underline{v_1} v_2 + \overline{v_2} v_1 - \underline{v_1}\,\overline{v_2}.
\end{aligned}
$$

- This relaxation transforms the non-convex bilinear constraint into linear inequalities.

Example for Different Bounds

- $-2 \leq v_1 \leq 2$ and $0 \leq v_2 \leq 50$

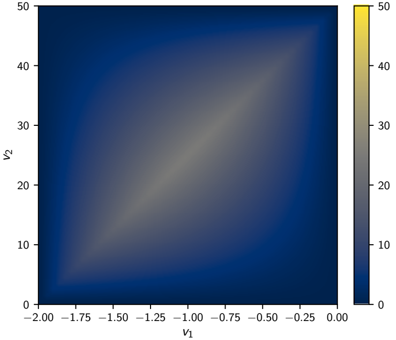


(a) Difference to the upper bound

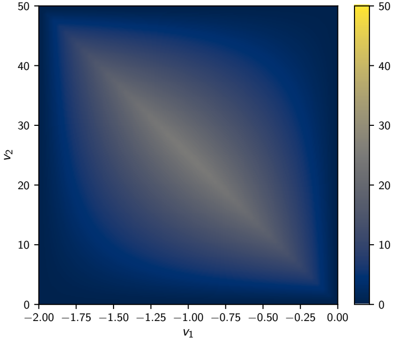
(b) Difference to the lower bound

Example for Different Bounds

- $-2 \leq v_1 \leq 0$ and $0 \leq v_2 \leq 50$



(c) Tighter upper bound on w



(d) Tighter lower bound on w

Implementation Details and Evaluation Metrics

- We define road segments, planner configurations, and soft constraints.
- Simulation scenarios mimic realistic driving conditions.
- Multiple scenarios: straight roads, curved segments.
- Metrics: computational time, road completion, and objective values.

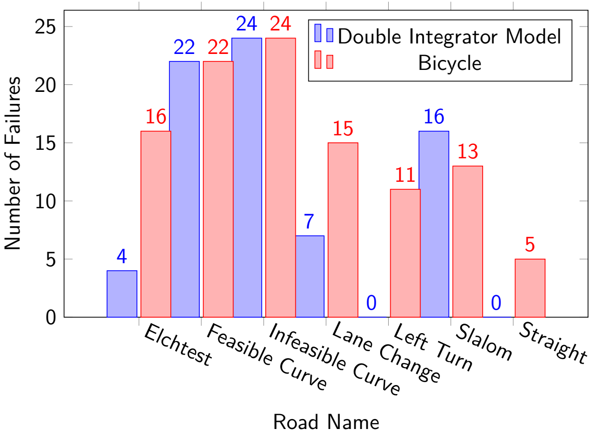
Performance: Computational Efficiency

- Solver times validate the framework's efficiency.
- $t_{\text{conf}}^{(1)} = (3\text{s}, 0.1\text{s}, 10\text{ms}, 40\text{ms})$
- $t_{\text{conf}}^{(2)} = (5\text{s}, 0.1\text{s}, 20\text{ms}, 20\text{ms})$

Model	**Config**	**Avg Time**	**Time Deviation**
DI	$t_{\text{conf}}^{(1)}$	3.9ms	1.0ms
DI	$t_{\text{conf}}^{(2)}$	3.8ms	1.3ms
Bicycle	$t_{\text{conf}}^{(1)}$	9.5ms	2.1ms
Bicycle	$t_{\text{conf}}^{(2)}$	9.4ms	2.9ms

Comparison: Double Integrator vs. Bicycle Model

- Double integrator: Elchtest, Lane Change, and Left Turn.
- Bicycle model: Slalom.
- Straight road: Solver reliability check.

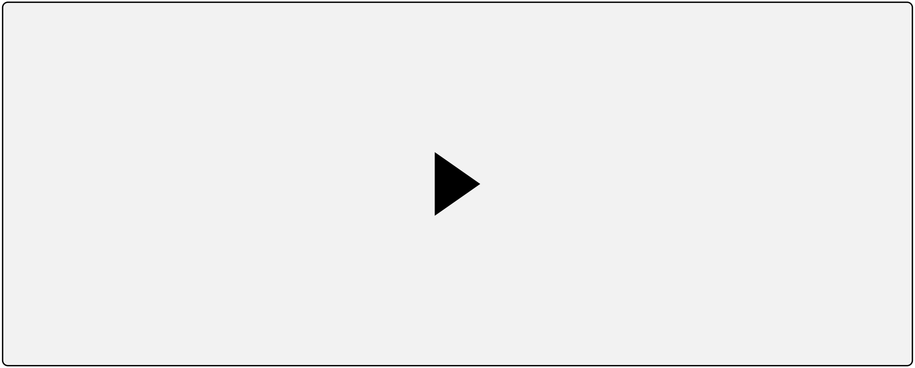


Numerical Experiment: Lane Change

- Scenario: Handling a sharp curve.
- Result: Both models perform well on moderate and slower velocities.
- Difference: DI accounts for lateral acceleration from the initial point onward.

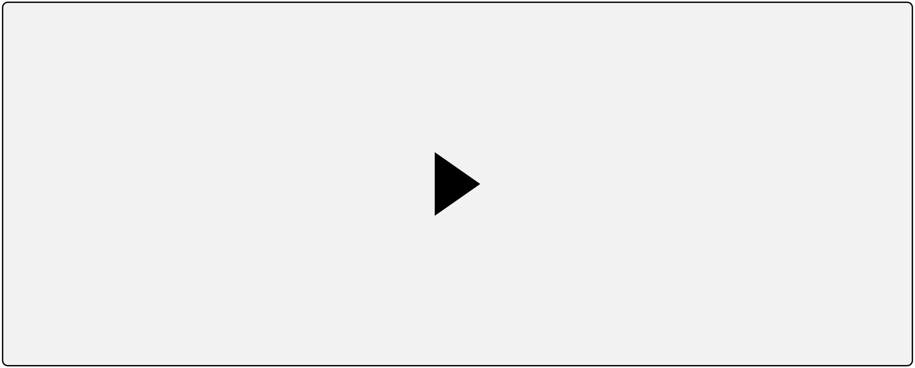
Numerical Experiment: Slalom

- Scenario: Handling a sharp curve.
- Result: KST performs better.
- Inscribed polytope too conservative.



Numerical Experiment: Left Curve

- Scenario: Minimize control derivatives.
- Result: Approximation errors noticeable for KST.



Limitations and Challenges

- Conservative approximations can limit feasibility.
- KST performs worse in curves, due to approximations.
- KST fails Elchtest, due to restrictive friction approximation.

$$\sqrt{a^2 + \left(v\dot{\psi}\right)^2} = \sqrt{a^2 + \left(\frac{v^2}{l_{wb}} \tan(\delta)\right)^2} \leq a_{max} \tag{19}$$

Summary

- Proposed a convex optimization framework for motion planning.
- Demonstrated realistic, safe trajectories with a constrained double-integrator and kinematic bicycle model.
- Validated through extensive simulations and performance evaluations.

Future Work

- Continue work on the KST model.
- Open work on the inscribed polytope, i.e. restrict $n \in [0, \overline{n}]$ in right curves.
- Implement the problem without the need of constraints compilation during runtime.

Animation