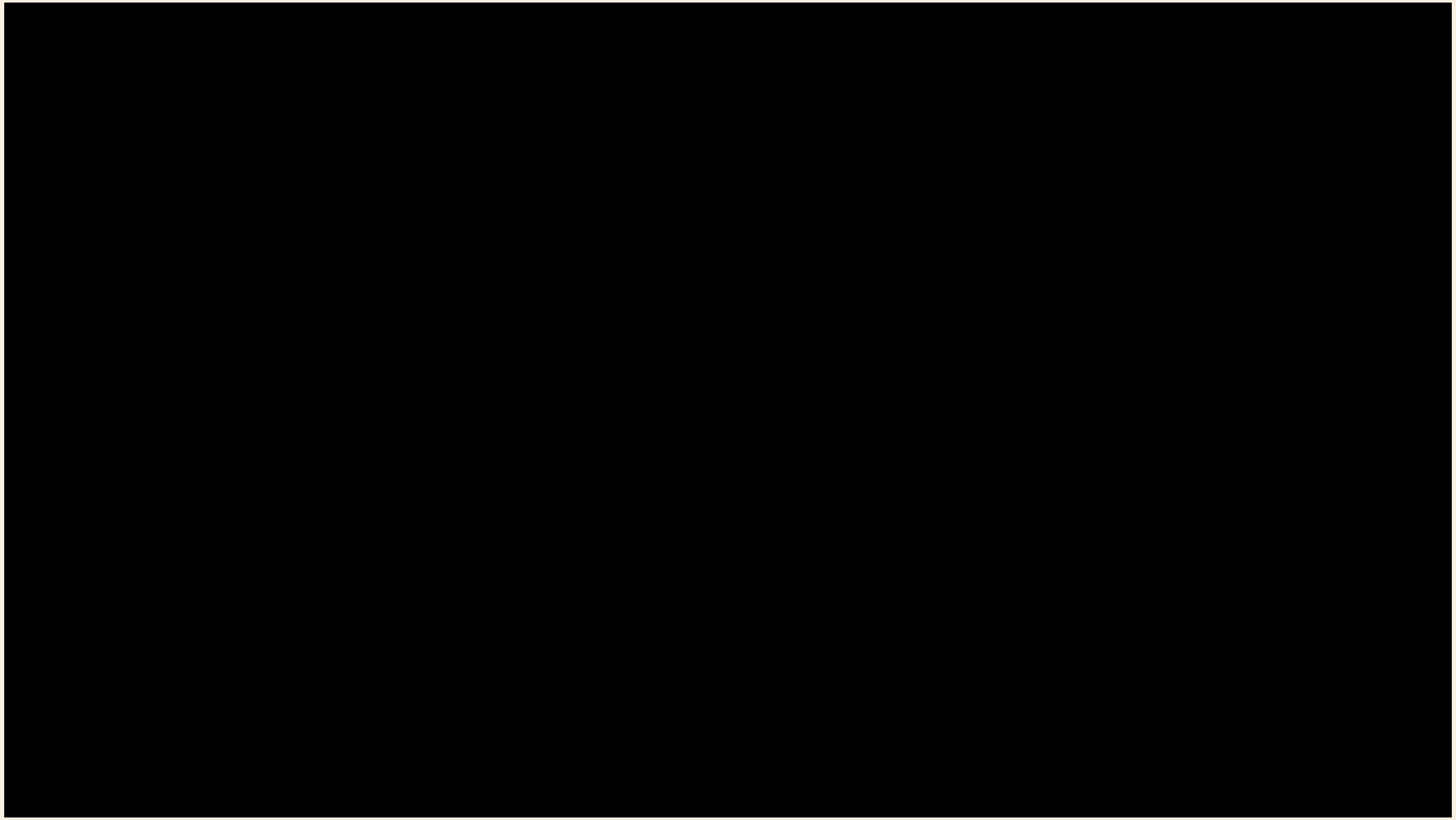
Optimal Control of Differentially Flat Systems is Surprisingly Easy

Logan E. Beaver, Ph.D.

Assistant Professor of Autonomous Systems
Department of Mechanical and Aerospace Engineering
Old Dominion University

Goal: Fast Motion Planning



Parc du Haut-Fourneau U4 by TomFPV, Youtube

OUTLINE

01 Flatness + Optimal Control

02 The Problem with Constraints

03 Heuristic Algorithm

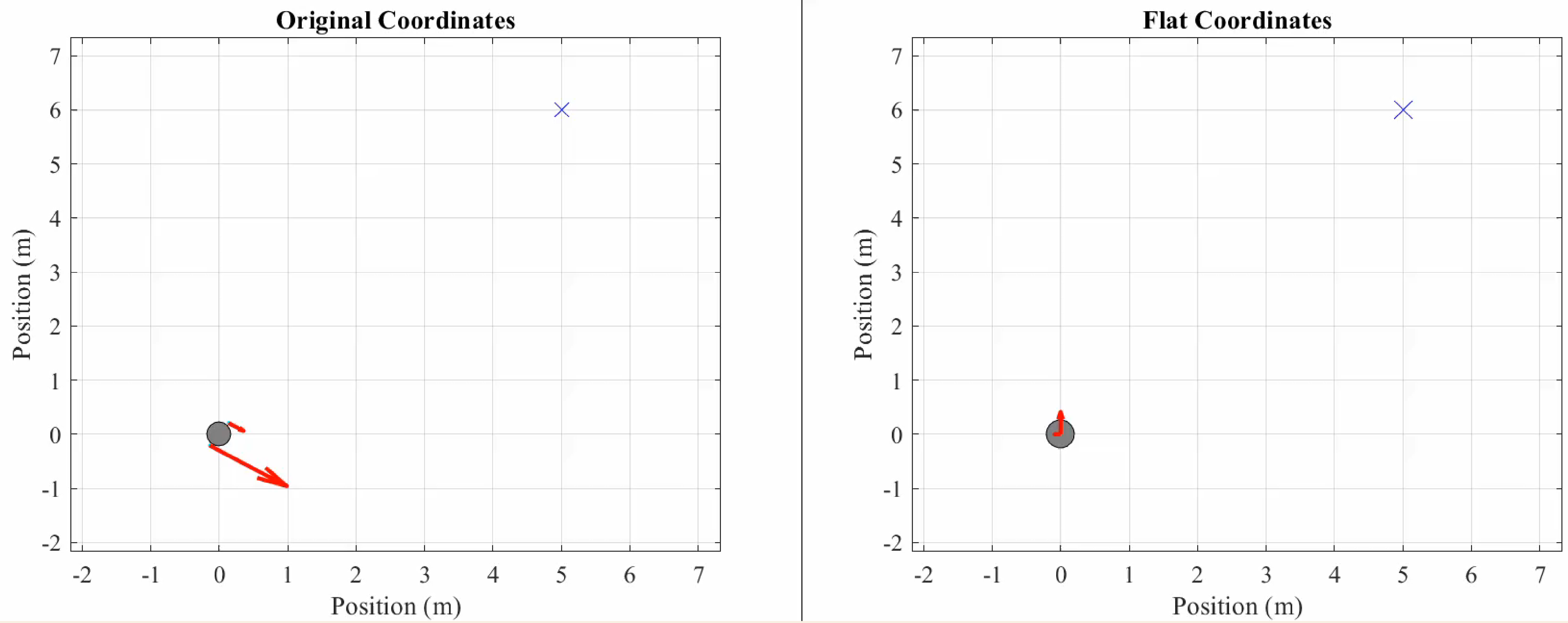
Related Work

- Method of Evolving Junctions
  - Li, Chow, Egerstedt, Lu, and Zho, (2017); Zhai, Hou, Zhang, and Zhou, (2021)
- NOSNOC
  - A.Nurkanović and M. Diehl (2022); A.Nurkanović et al., (2023)
- Collocation Methods
  - Andersson, Gillis, Horn, Rawlings, and Diehl (2019), Ross (2012), Murray (2008), Wächter and Biegler (2006),
- Differential Flatness
  - Fleiss et al., (1995); Petit, Milam, and Murray, (2001); Sira-Ramirez and Agrawal, (2004); Chaplais and Petit, (2007, 2008); Levine, (2011)

Flatness + Optimal Control

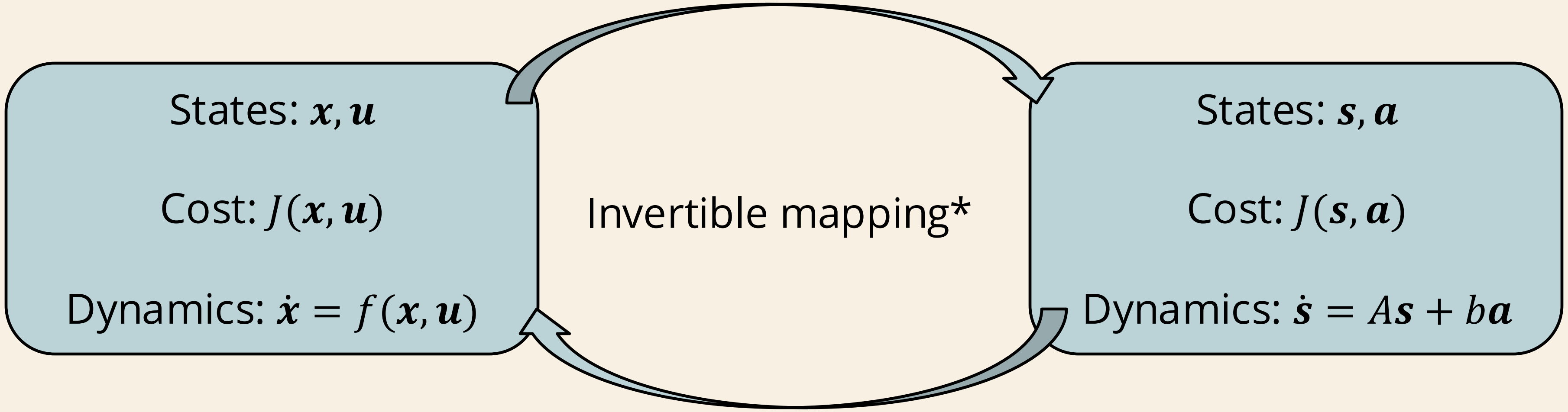
1. **Beaver, L. E**., & Malikopoulos, A. A. (2024). Optimal control of differentially flat systems is surprisingly easy. *Automatica*, *159*, 111404.
2. **Beaver, L. E.** (2023). LQ-OCP: Energy-optimal control for lq problems. *2024 American Control Conference (to appear)*
3. Chalaki, B., **Beaver, L. E**., & Malikopoulos, A. A. (2020). Experimental validation of a real-time optimal controller for coordination of cavs in a multi-lane roundabout. In *2020 IEEE Intelligent Vehicles Symposium (IV)* (pp. 775-780). IEEE.
4. **Beaver, L. E.**, & Malikopoulos, A. A. (2019). A decentralized control framework for energy-optimal goal assignment and trajectory generation. In *2019 IEEE 58th Conference on Decision and Control (CDC)* (pp. 879-884). IEEE.
5. **Beaver, L. E.**, & Malikopoulos, A. A. (2019). A decentralized control framework for energy-optimal goal assignment and trajectory generation. In *2019 IEEE 58th Conference on Decision and Control (CDC)* (pp. 879-884). IEEE.

Differential Flatness



Differential Flatness

- Property of nonlinear systems
- Mapping between coordinates*
- New system has linear dynamics



*no exogenous coordinates, differentially independent

Differential Drive Robot

- Original Coordinates

$$\boldsymbol{p}, \theta, v_L, v_R$$

$$\omega = \frac{v_R - v_L}{L}$$

$$v = \frac{v_R + v_L}{2}$$

- Polar Coordinates

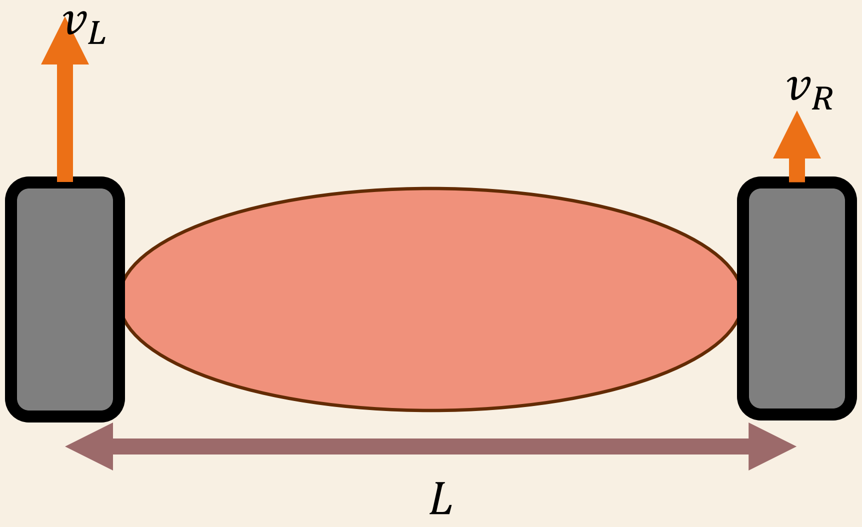
$$\boldsymbol{p}, v, \theta, \omega$$

$$\frac{v_y}{v_x} = \tan(\theta)$$

$$\|\boldsymbol{v}\| = v$$

- Flat Coordinates

$$\boldsymbol{p}, \boldsymbol{v}$$



PMP for Linear Systems

- Follow the *standard approach* of Bryson and Ho

$$H(\boldsymbol{s}, \boldsymbol{a}, \lambda) = J(\boldsymbol{s}, \boldsymbol{a}) + \boldsymbol{\lambda} \cdot \boldsymbol{f}(\boldsymbol{s}, \boldsymbol{a}) + \boldsymbol{\mu} \cdot \boldsymbol{g}(\boldsymbol{s}, \boldsymbol{a})$$

- Minimize the Hamiltonian

$$\frac{\partial H}{\partial \boldsymbol{a}} = 0$$

$$\dot{\boldsymbol{\lambda}} = \frac{\partial H}{\partial \boldsymbol{s}}$$

- Numerically unstable ODE in general

Linear Systems Trick

$$H(\boldsymbol{s}, \boldsymbol{a}, \lambda) = J(\boldsymbol{s}, \boldsymbol{a}) + \boldsymbol{\lambda} \cdot \boldsymbol{f}(\boldsymbol{s}, \boldsymbol{a}) + \boldsymbol{\mu} \cdot \boldsymbol{g}(\boldsymbol{s}, \boldsymbol{a})$$

- Partial derivatives simplify!

$$\frac{\partial \boldsymbol{f}}{\partial \boldsymbol{s}} = \frac{\partial}{\partial \boldsymbol{s}} A\boldsymbol{s} + b\boldsymbol{a} = A$$

$$\frac{\partial f}{\partial \boldsymbol{a}} = \frac{\partial}{\partial \boldsymbol{a}} A\boldsymbol{s} + b\boldsymbol{a} = b$$

- Resulting ODEs,

$$J_{\boldsymbol{a}} + \boldsymbol{\lambda}^T b + \boldsymbol{\mu}^T \boldsymbol{g}_{\boldsymbol{a}} = 0$$

$$\dot{\boldsymbol{\lambda}}_{\boldsymbol{x}} = -J_{\boldsymbol{s}} - \boldsymbol{\lambda}^T A + \boldsymbol{\mu}^T \boldsymbol{g}_{\boldsymbol{s}}$$

Solution Overview

Nonlinear OCP

$\dot{\boldsymbol{x}} = \boldsymbol{f}(\boldsymbol{x}, \boldsymbol{u})$

↔ Differential Flatness ↔

Integrator System

$\dot{s}_i{}^{(n)} = s_i^{(n+1)}$

$s_i{}^{(k_i)} = a_i$

i chains, k_i links

⇒

Pontryagin Minimum Principle

$H = J + \boldsymbol{\lambda}^T \boldsymbol{f} + \boldsymbol{\mu}^T \boldsymbol{g}$

⇓

Motion primitive generator

$$\sum_{n=0}^{k_i} (-1)^n \frac{d^n}{dt^n} \left(J_{s_i^{(n)}} + \boldsymbol{\mu}^T \boldsymbol{g}_{s_i^{(n)}} \right) = 0$$

⇐

$2^{|\boldsymbol{g}|}$ primitives

↔ Differential Flatness ↔

Motion primitives for original system

⇕ (Nonlinear OCP)

Dynamic Motion Primitives

$$\sum_{n=0}^{k_i} (-1)^n \frac{d^n}{dt^n}\left(J_{s_i^{(n)}} + \boldsymbol{\mu}^T \boldsymbol{g}_{s_i^{(n)}}\right) = 0$$

- Toggle elements of $\boldsymbol{\mu}$ “on” and “off
- Solve ODE + algebraic equations to generate motion primitives
- At most $2^{|\boldsymbol{g}|}$ cases

The Problem with Constraints

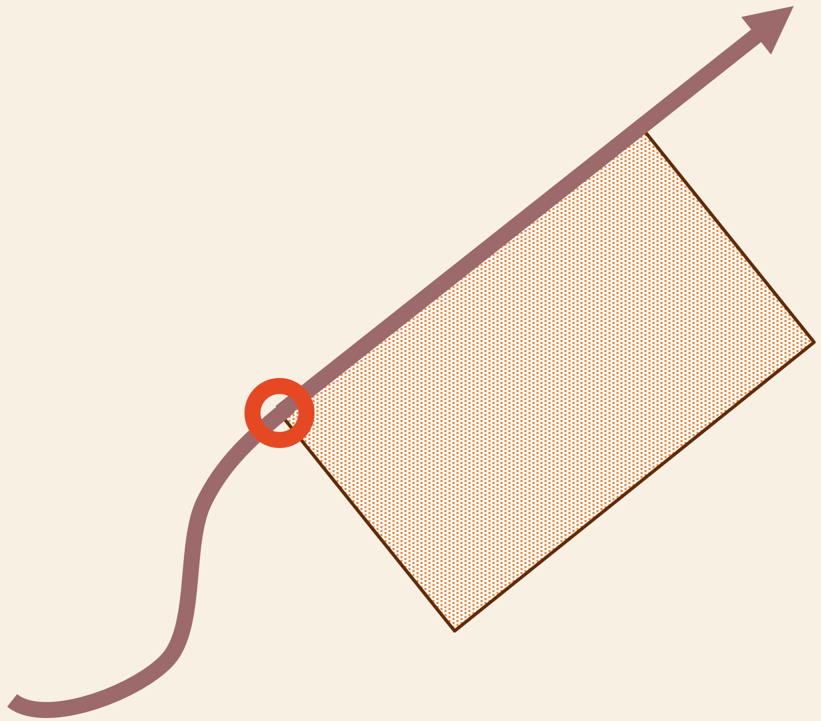
1. **Beaver, L. E.**, Tron, R., & Cassandras, C. G. (2023). A graph-based approach to generate energy-optimal robot trajectories in polygonal environments. *IFAC-PapersOnLine*, *56*(2).
2. Malikopoulos, A. A., **Beaver, L.,** & Chremos, I. V. (2021). Optimal time trajectory and coordination for connected and automated vehicles. *Automatica*, *125*, 109469.

Constraint Junctions

- Known "standard" form at time t_1

$$H(t_1^-) = H(t_1^+) + \boldsymbol{\pi}\frac{\partial \boldsymbol{N}}{\partial t}$$

$$\boldsymbol{\lambda}^T(t_1^-) = \boldsymbol{\lambda}^T(t_1^+) + \boldsymbol{\pi}\frac{\partial \boldsymbol{N}}{\partial \boldsymbol{x}}$$

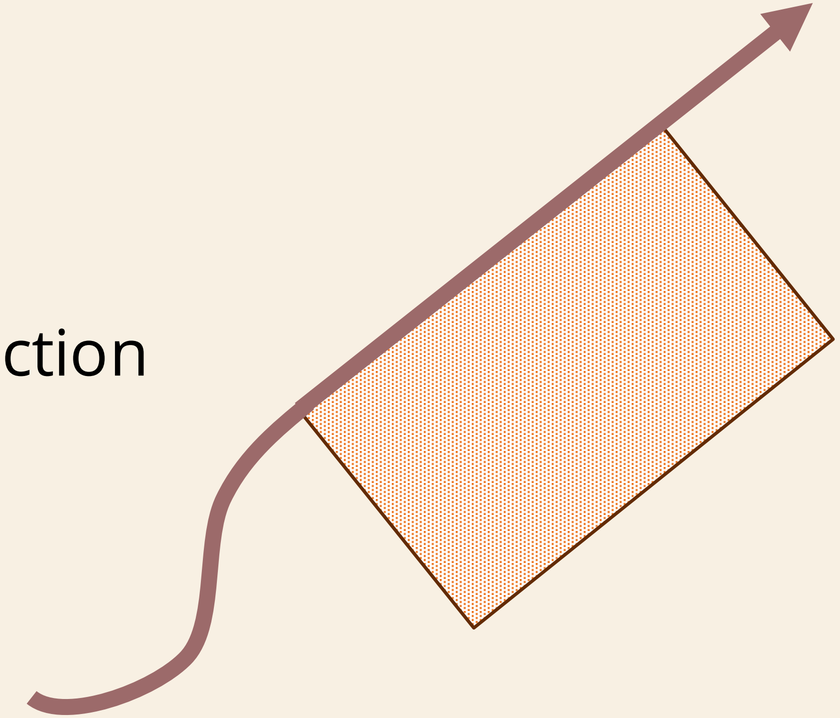


- Easily converts to state conditions

Shooting Method

- Given a constraint sequence,
  - Apply jump conditions for $\boldsymbol{\lambda}, H$
  - For example, $||\boldsymbol{a}||^2$ in cost → continuity in $\boldsymbol{a}$

$$a(t_1^-) = a(t_1^+)$$



- System of algebraic equations at the junction

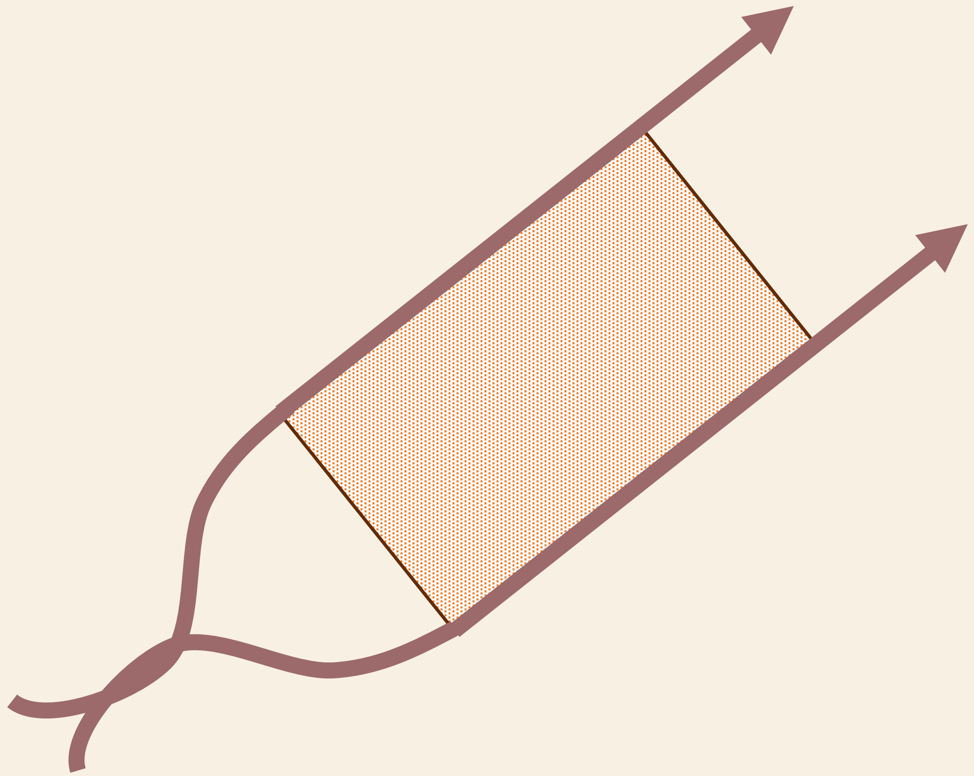
$$x(t_1^-) = x(t_1^+)$$
$$u(t_1^-) = u(t_1^+)$$

- Linear for a "guess" of t_1
  - 1 Equation + 1 unknown

Optimal Trajectories

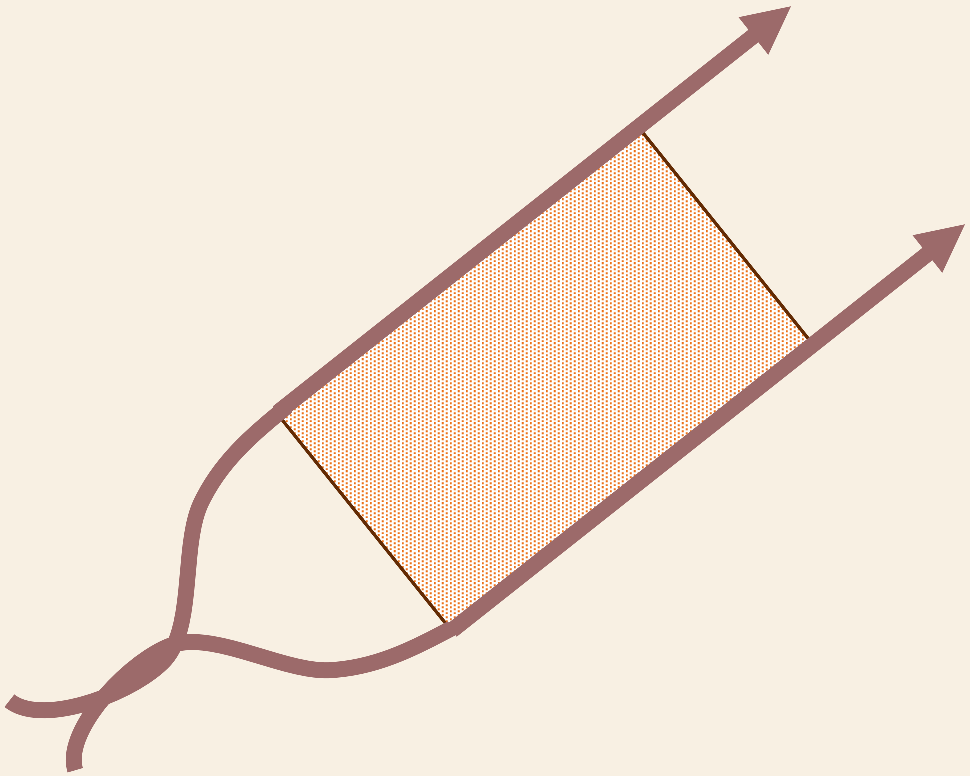
- Given a constraint, easy to solve

- **What is the optimal sequence?**

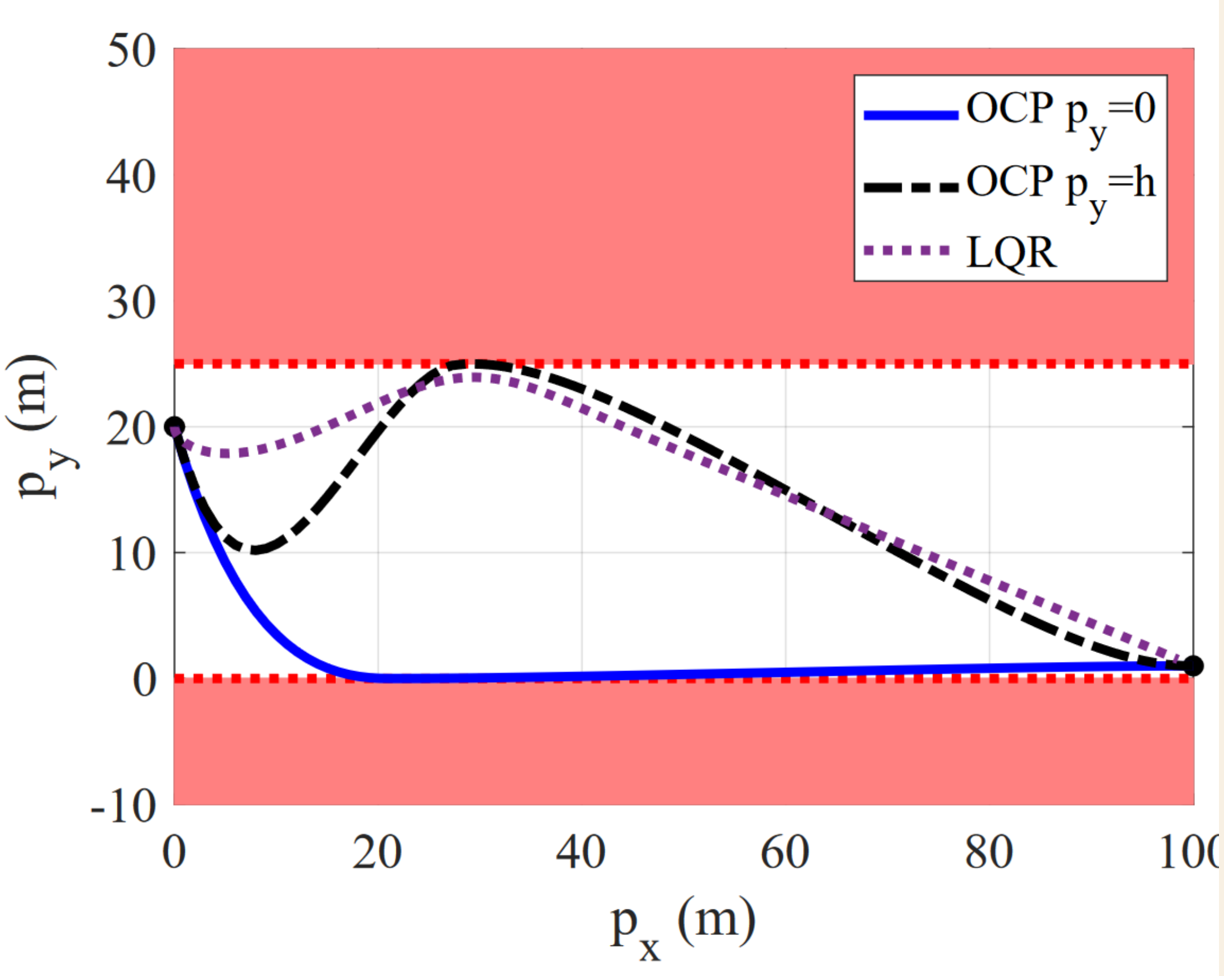


Implications

- Both solutions satisfy PMP
- Both solutions are **convex**
- **Any** constraint breaks convexity!
- NP-Hard integer program?



ACC Example



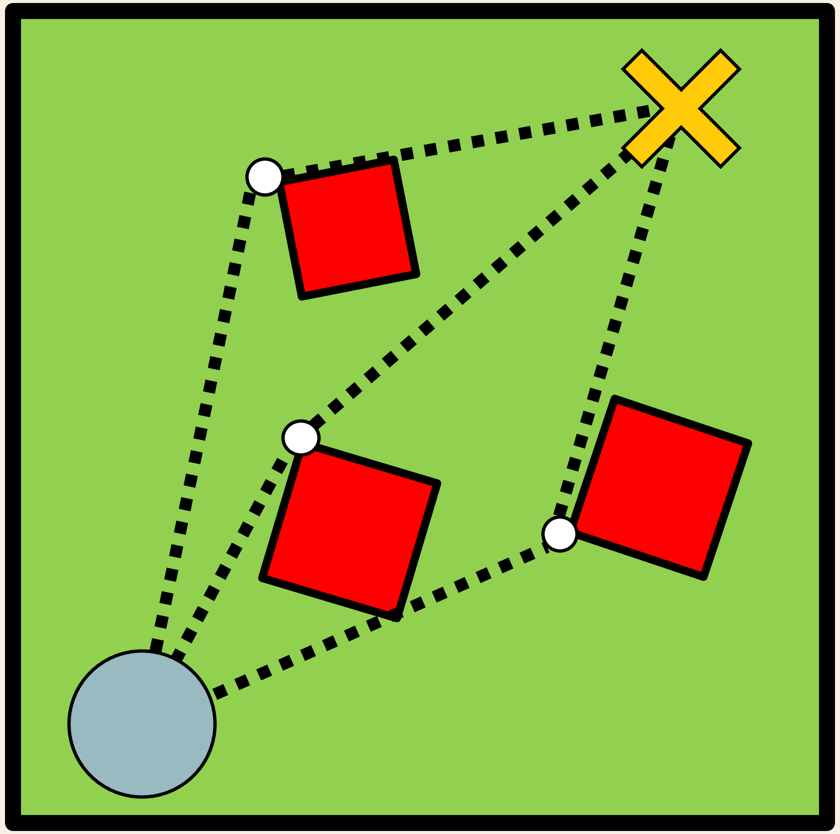
	Energy Cost	Optimality Gap
OCP with $v_y(t_1) = 0$	8,458 units	0%
OCP with $v_y(t_1) = h$	8,561 units	1.2%

Heuristic Algorithm

1. **Beaver, L. E.**, Tron, R., & Cassandras, C. G. (2023). A graph-based approach to generate energy-optimal robot trajectories in polygonal environments. *IFAC-PapersOnLine*, *56*(2).
2. Malikopoulos, A. A., **Beaver, L.,** & Chremos, I. V. (2021). Optimal time trajectory and coordination for connected and automated vehicles. *Automatica*, *125*, 109469.

Shooting Method Decomposition

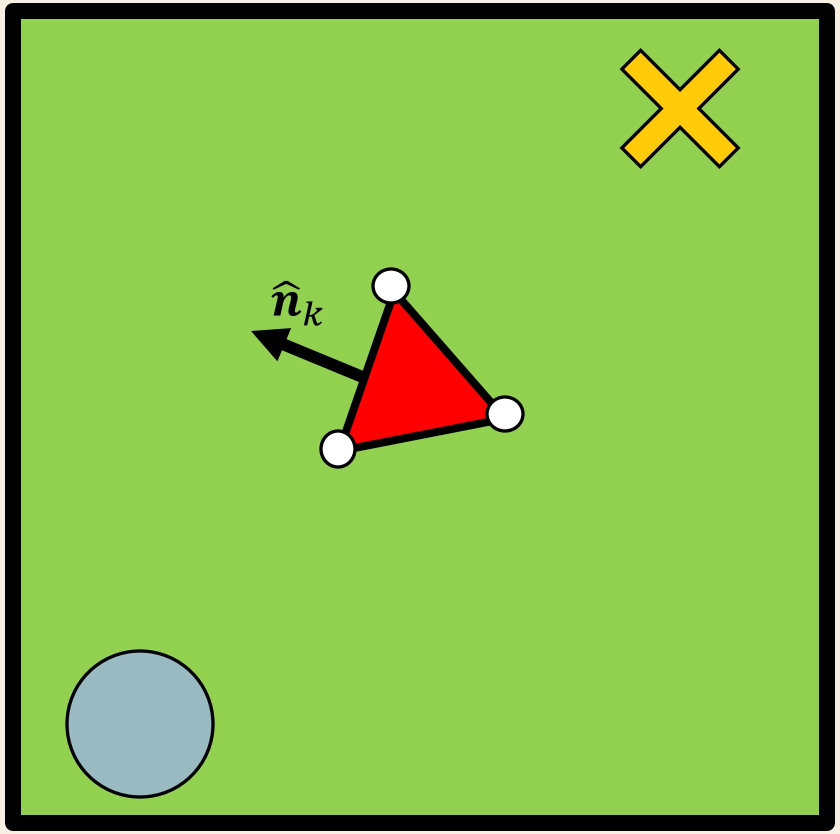
- Motion primitives connected with junctions
- Solve optimal trajectory given the active constraints
- Which constraints to consider?



Resulting Splines + Optimality

$$\sum_{n=0}^{k_i} (-1)^n \frac{d^n}{dt^n}\left(J_{s_i^{(n)}} + \boldsymbol{\mu}^T \boldsymbol{g}_{s_i^{(n)}}\right) = 0$$

- Unconstrained: $\ddot{\boldsymbol{u}}^* = \boldsymbol{0}$
- Constrained: $\boldsymbol{u}^* \cdot \hat{\boldsymbol{n}}_k = \boldsymbol{0}$



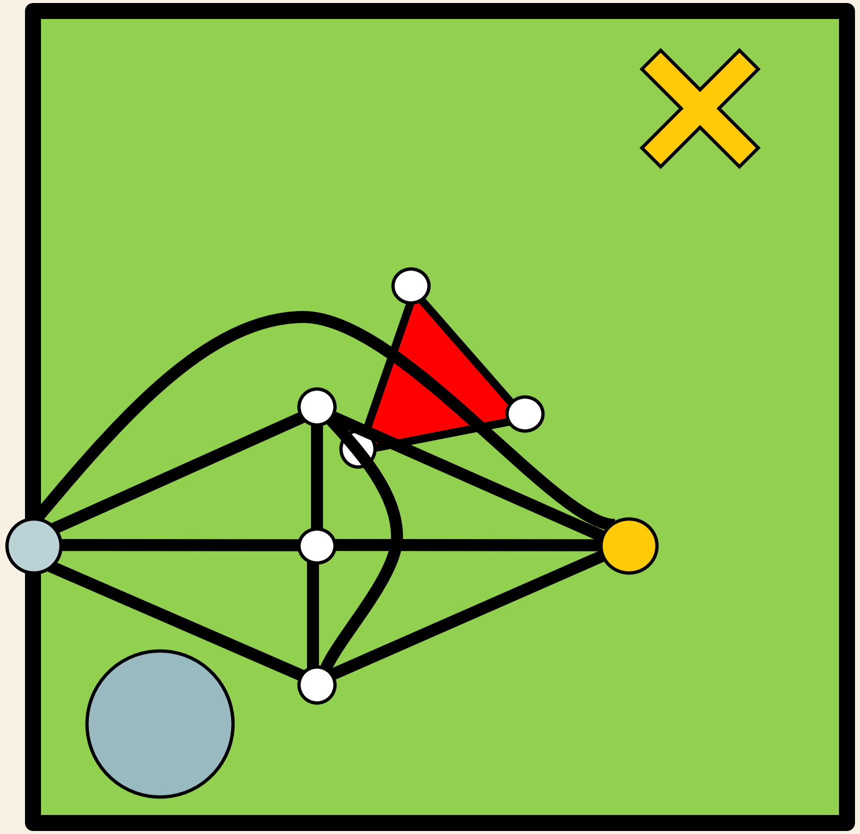
At junctions:

- Continuity in $\boldsymbol{x}^*, \boldsymbol{u}^*, {\boldsymbol{v}^*}^T \dot{\boldsymbol{u}}^*$
- Unknown $\boldsymbol{x}^*, \boldsymbol{t}^*$ at junction

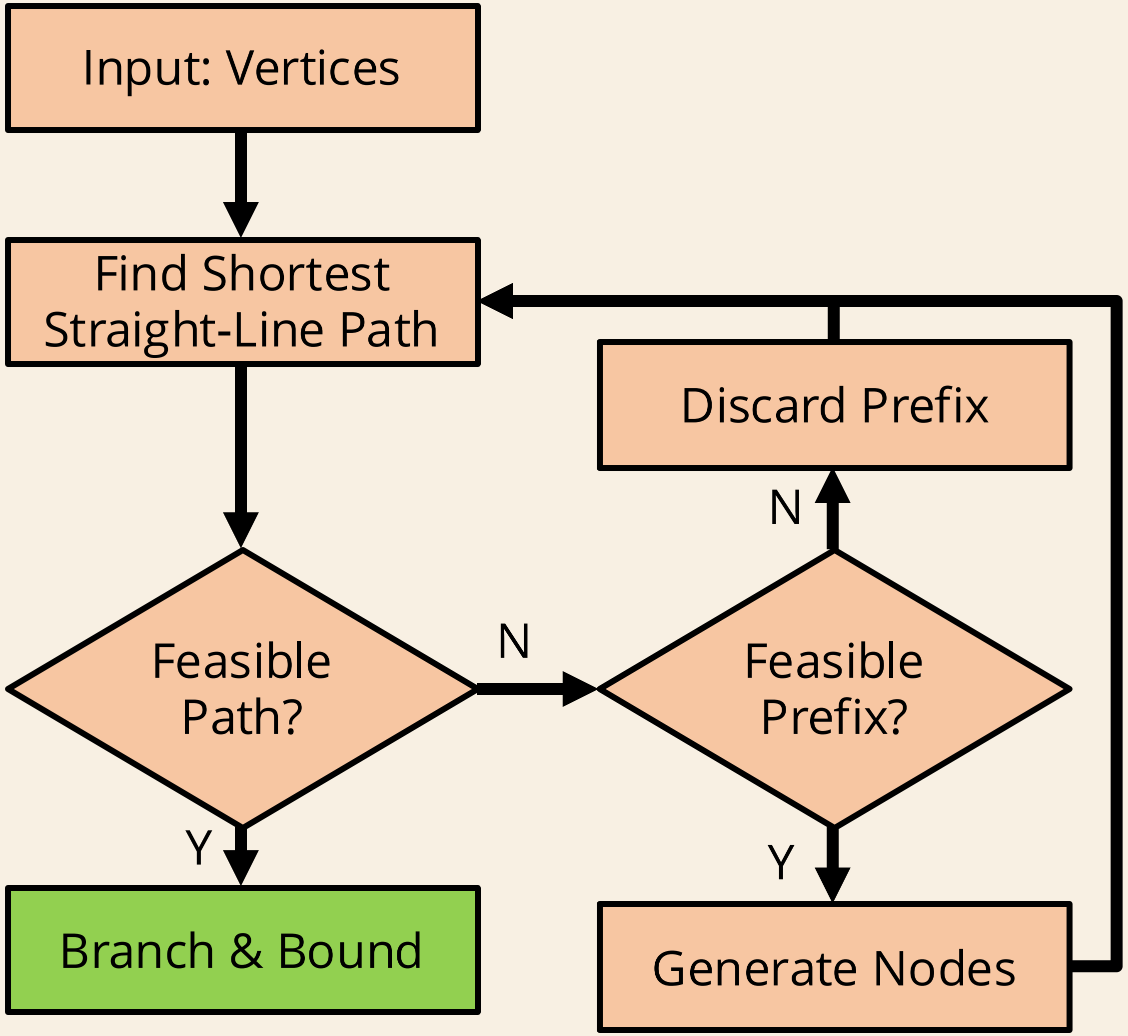
The Integer Problem

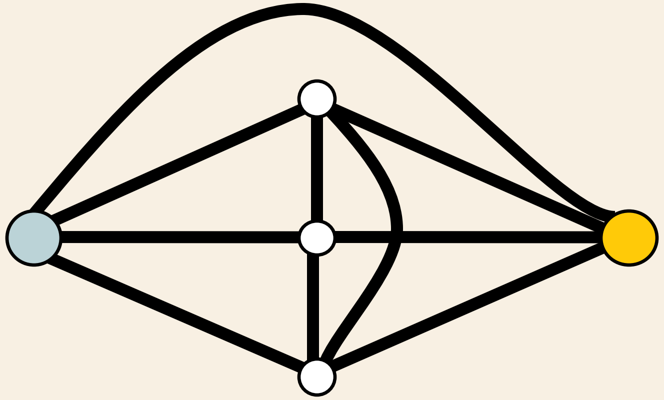
- OCP is easy, given constraints

- Which constraints to select?
  - Fully-connected graph, unknown cost
  - No admissible heuristic

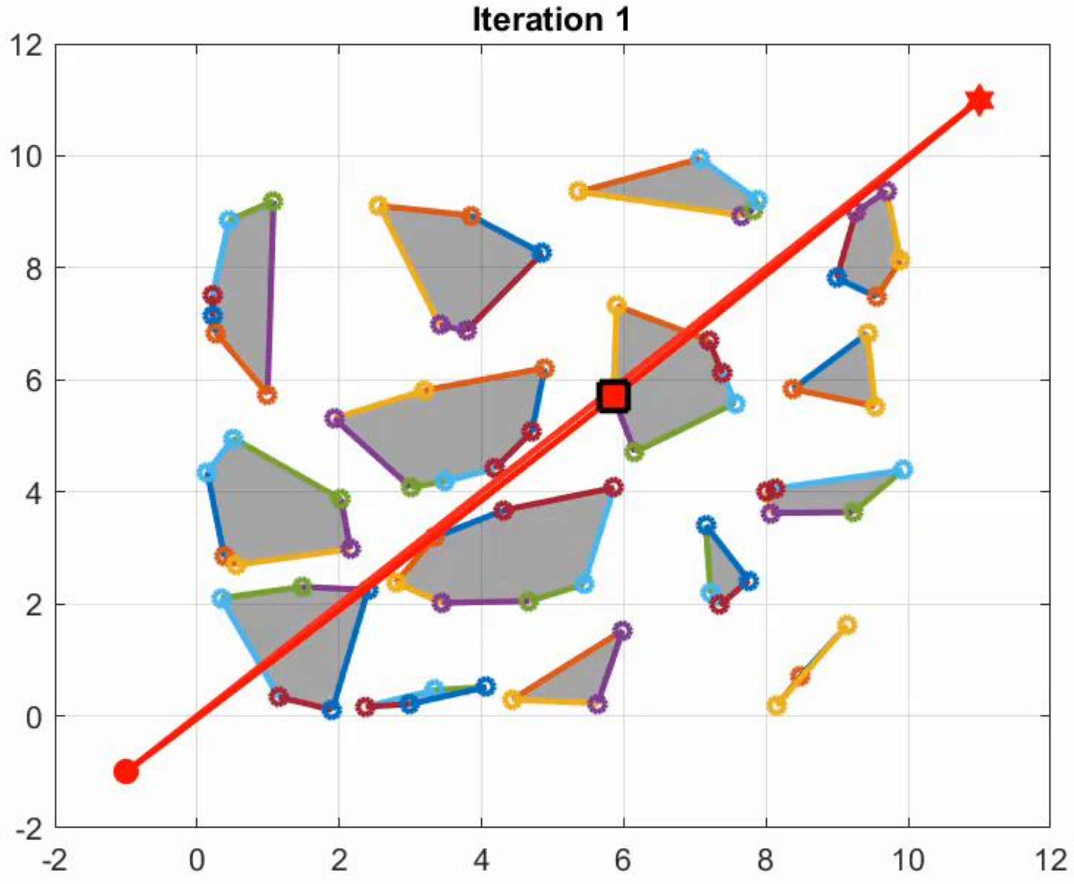


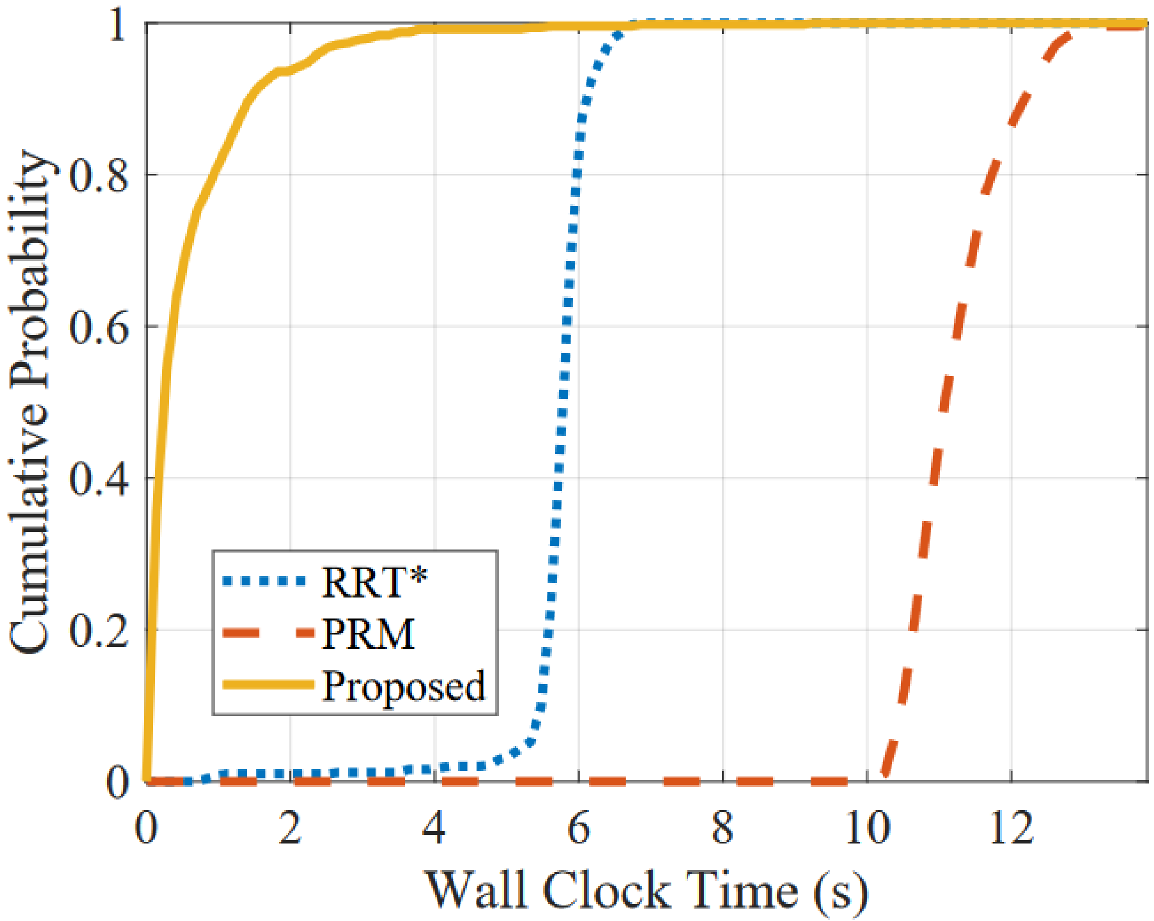
Search-Based Solution





Constrained Motion Planning



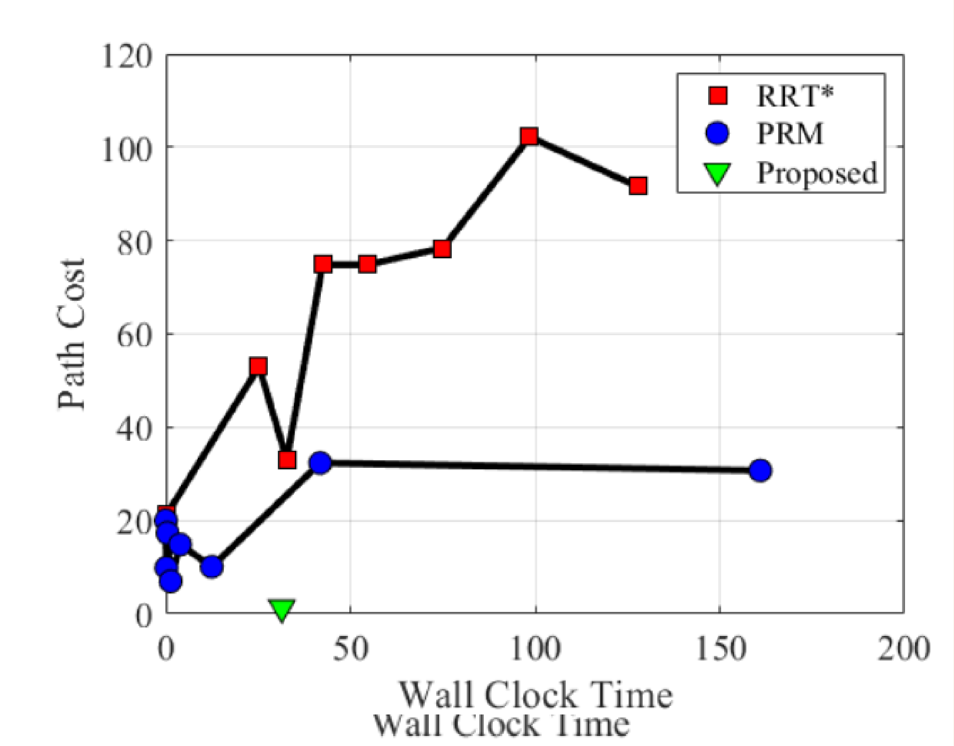


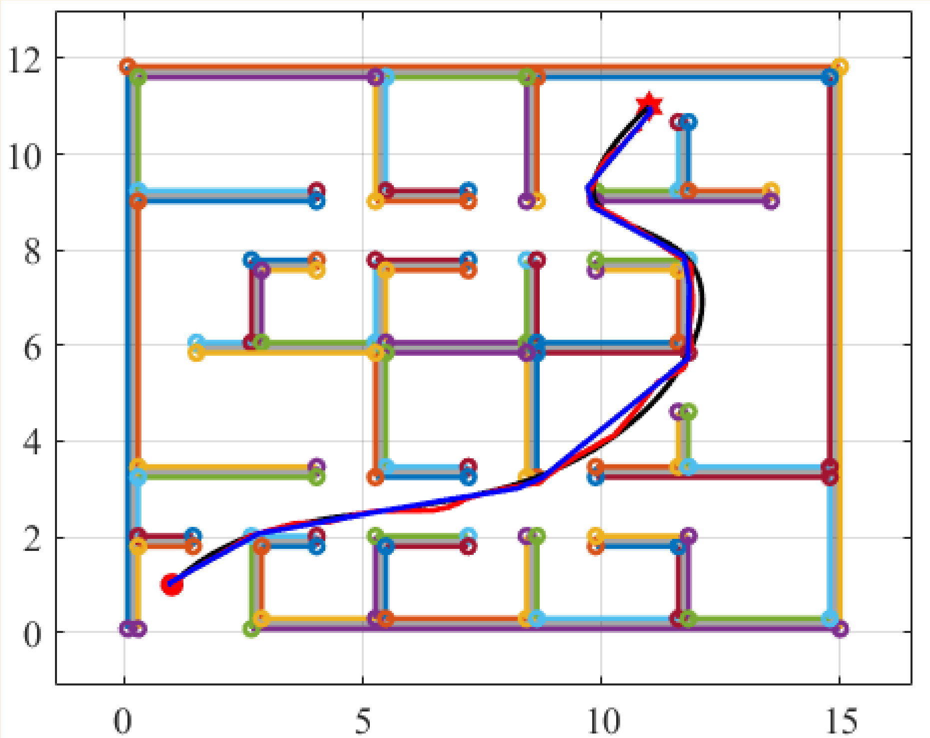
72 constraints (5184 combinations)

1. **Beaver**, Tron, and Cassandras, *A Graph-Based Approach to Generate Energy-Optimal Robot Trajectories in Polygonal Environments*, **2023 IFAC World Congress (to appear)**, 2023

Maze Solving

- Proposed vs RRT and PRM





High Degree of Freedom Systems

- Scales with number of vertices
- “Independent” of dimension
- Environmental preprocessing?

Image: Wikimedia commons

Main Takeaways

- Flatness makes optimal control “easy”
- Out-performs state-of-the-art planning
- Constraints (even linear) cause problems
- Hybrid optimal + sampling techniques?