Motion Planning and Differential Flatness

Marcus Greiff
marcus.greiff@control.lth.se

15/10-2020

Guest Lecture - Linköping

Introduction

Context

- 4^{th}-year PhD student at Lund Universtiry
- Supervised by Anders Robertsson
- K. Berntorp (former student, now at MERL)
- B. Olofsson (former student, now at LiU/LU)

---

[1] M. Greiff and K. Berntorp, "Projections in Adaptive Mixture Kalman Filtering for GNSS Positioning", ACC, 2020.

[2] M. Greiff, K. Berntorp and A. Robertsson, "Measurement dimension reduction in Gaussian filtering", CCTA, 2020.

[3] M. Greiff, K. Berntorp and A. Robertsson, "Exploiting Linear Substructure In LRKFs (Extended) ", CDC, 2020.

[4] E. Lefeber, M. Greiff and A. Robertsson, "Filtered Output Feedback Tracking Control of a Quadrotor UAV", IFAC, 2020.

[5] M. Greiff, Z. Sun and A. Robertsson, "Attitude Control on SU(2): Stability, Robustness, and Similarities", L-CSS, 2020.

Introduction

Context

- 4^{th}-year PhD student at Lund Universtiry
- Supervised by Anders Robertsson
- K. Berntorp (former student, now at MERL)
- B. Olofsson (former student, now at LiU/LU)

Current work

(i) MERL - Estimation Theory
  - Satellite positioning [1]
  - Nonlinear Filtering [2, 3]

(ii) LU - Nonlinear and Robust Control
  - Focus on aerial vehicles
  - Filtered output feedback [4, 5]

---

[1] M. Greiff and K. Berntorp, "Projections in Adaptive Mixture Kalman Filtering for GNSS Positioning", ACC, 2020.

[2] M. Greiff, K. Berntorp and A. Robertsson, "Measurement dimension reduction in Gaussian filtering", CCTA, 2020.

[3] M. Greiff, K. Berntorp and A. Robertsson, "Exploiting Linear Substructure In LRKFs (Extended) ", CDC, 2020.

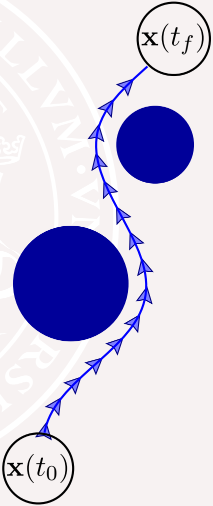
[4] E. Lefeber, M. Greiff and A. Robertsson, "Filtered Output Feedback Tracking Control of a Quadrotor UAV", IFAC, 2020.

[5] M. Greiff, Z. Sun and A. Robertsson, "Attitude Control on SU(2): Stability, Robustness, and Similarities", L-CSS, 2020.

Introduction

Motion planning?

- Necessary in practical experiments
- Taught by Björn at LU in 2017



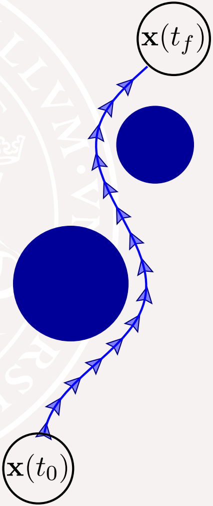
Introduction

Motion planning?

- Necessary in practical experiments
- Taught by Björn at LU in 2017

Disclaimer

- Not an expert in the field
- Share some useful ideas



Introduction

Motion planning?

- Necessary in practical experiments
- Taught by Björn at LU in 2017

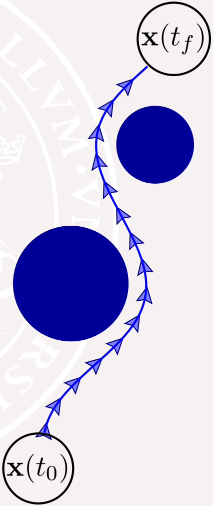
Disclaimer

- Not an expert in the field
- Share some useful ideas

Presentation Purpose

+ Introduce differential flatness (DF)
+ Convex polynomial optimization (CPO)
+ Sequential quadratic programming (SQP)

= Theoretical and practical examples



Introduction

Motion planning?

- Necessary in practical experiments
- Taught by Björn at LU in 2017

Disclaimer

- Not an expert in the field
- Share some useful ideas

Presentation Purpose

+ Introduce differential flatness (DF)
+ Convex polynomial optimization (CPO)
+ Sequential quadratic programming (SQP)

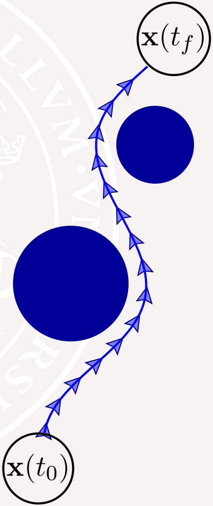
= Theoretical and practical examples

Motivation

DF Applicable to ground, surface, and aerial vehicles

CPO Powerful method enabled by DF

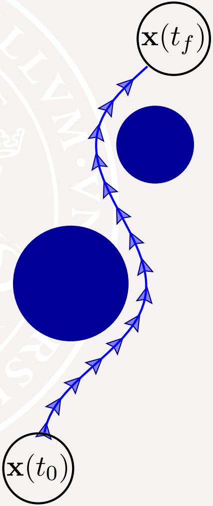
SQP Generally useful, enforce constraints in CPO



Introduction

Motion planning

$$\dot{\mathbf{x}}(t) = \mathbf{f}(\mathbf{x}(t), \mathbf{u}(t))$$
$$\mathbf{x}(t) \in \mathcal{X}, \quad \mathbf{u}(t) \in \mathcal{U}$$
$$t \in [t_0, t_f]$$



[6] T. Glück et al., “Swing-up control of a triple pendulum”, Automatica, 2013.

[6] Link to cool video: `https://www.youtube.com/watch?v=cyN-CRNrb3E`

[7] Link to nice slides by Boyd `https://web.stanford.edu/class/ee364b/lectures/seq_slides.pdf`

[8] C. Richter, “Polynomial trajectory planning for aggressive quadrotor flight in dense indoor environments”, 2013.

Introduction

Motion planning

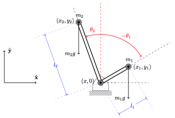
$$\dot{\mathbf{x}}(t) = \mathbf{f}(\mathbf{x}(t), \mathbf{u}(t))$$
$$\mathbf{x}(t) \in \mathcal{X}, \quad \mathbf{u}(t) \in \mathcal{U}$$
$$t \in [t_0, t_f]$$

Approaches

- Collocation-based (BVPs) [6]
- Sampling-based (RRT, PF methods)
- Optimization-based (MPC, SQP [7], CPO [8])

Simulation example A

The BVP-method in [6] applied to a variation of the under-actuated two-pendulum cart problem, and solved with the `bvp5c()` function in Matlab.



Under-actuated two-pendulum cart process.

---

[6] T. Glück et al., "Swing-up control of a triple pendulum", Automatica, 2013.

[6] Link to cool video: `https://www.youtube.com/watch?v=cyN-CRNrb3E`

[7] Link to nice slides by Boyd `https://web.stanford.edu/class/ee364b/lectures/seq_slides.pdf`

[8] C. Richter, "Polynomial trajectory planning for aggressive quadrotor flight in dense indoor environments", 2013.

Introduction

Motion planning

$$\dot{\mathbf{x}}(t) = \mathbf{f}(\mathbf{x}(t), \mathbf{u}(t))$$
$$\mathbf{x}(t) \in \mathcal{X}, \quad \mathbf{u}(t) \in \mathcal{U}$$
$$t \in [t_0, t_f]$$

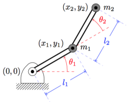
Approaches

- Collocation-based (BVPs) [6]
- Sampling-based (RRT, PF methods)
- Optimization-based (MPC, SQP [7], CPO [8])

Simulation example B

A minimum torque SQP-method in [7] initialized with an interpolation between

$$\theta_1(t_0) = 0 \qquad \theta_1(t_f) = \pi$$
$$\theta_2(t_0) = \frac{-\pi}{2} \qquad \theta_2(t_f) = \frac{\pi}{2}$$



A fully actuated planar two-link robotic arm.

---

[6] T. Glück et al., "Swing-up control of a triple pendulum", Automatica, 2013.

[6] Link to cool video: `https://www.youtube.com/watch?v=cyN-CRNrb3E`

[7] Link to nice slides by Boyd `https://web.stanford.edu/class/ee364b/lectures/seq_slides.pdf`

[8] C. Richter, "Polynomial trajectory planning for aggressive quadrotor flight in dense indoor environments", 2013.

Introduction

Motion planning

$$\dot{\mathbf{x}}(t) = \mathbf{f}(\mathbf{x}(t), \mathbf{u}(t))$$
$$\mathbf{x}(t) \in \mathcal{X}, \quad \mathbf{u}(t) \in \mathcal{U}$$
$$t \in [t_0, t_f]$$

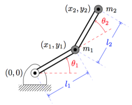
Approaches

- Collocation-based (BVPs) [6]
- Sampling-based (RRT, PF methods)
- Optimization-based (MPC, SQP [7], CPO [8])

Simulation example C

A minimum torque SQP-method in [7] initialized with an interpolation between

$$\theta_1(t_0) = 0 \qquad \theta_1(t_f) = \pi + 2\pi$$
$$\theta_2(t_0) = \frac{-\pi}{2} \qquad \theta_2(t_f) = \frac{\pi}{2} + 2\pi$$



A fully actuated planar two-link robotic arm.

[6] T. Glück et al., “Swing-up control of a triple pendulum”, Automatica, 2013.

[6] Link to cool video: `https://www.youtube.com/watch?v=cyN-CRNrb3E`

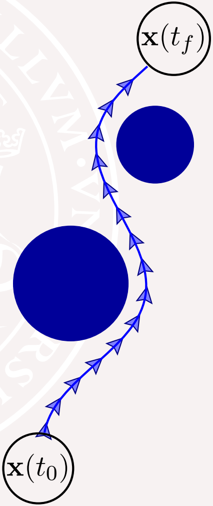
[7] Link to nice slides by Boyd `https://web.stanford.edu/class/ee364b/lectures/seq_slides.pdf`

[8] C. Richter, “Polynomial trajectory planning for aggressive quadrotor flight in dense indoor environments”, 2013.

Introduction

Motion planning

$$\dot{\mathbf{x}}(t) = \mathbf{f}(\mathbf{x}(t), \mathbf{u}(t))$$
$$\mathbf{x}(t) \in \mathcal{X}, \quad \mathbf{u}(t) \in \mathcal{U}$$
$$t \in [t_0, t_f]$$



Approaches

- Collocation-based (BVPs) [6]
- Sampling-based (RRT, PF methods)
- Optimization-based (MPC, SQP [7], CPO [8])

Older and newer solutions combined

- Defining a set of flat outputs
- Path planning by polynomial optimization
- Form QP to speed up/slow down time

---

[6] T. Glück et al., “Swing-up control of a triple pendulum”, Automatica, 2013.

[6] Link to cool video: `https://www.youtube.com/watch?v=cyN-CRNrb3E`

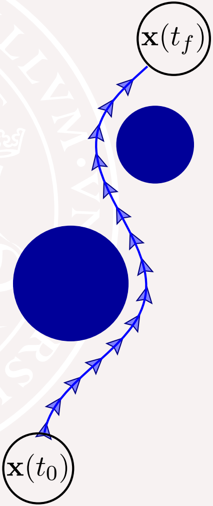
[7] Link to nice slides by Boyd `https://web.stanford.edu/class/ee364b/lectures/seq_slides.pdf`

[8] C. Richter, “Polynomial trajectory planning for aggressive quadrotor flight in dense indoor environments”, 2013.

Introduction

Motion planning

$$\dot{\mathbf{x}}(t) = \mathbf{f}(\mathbf{x}(t), \mathbf{u}(t))$$
$$\mathbf{x}(t) \in \mathcal{X}, \quad \mathbf{u}(t) \in \mathcal{U}$$
$$t \in [t_0, t_f]$$



Approaches

- Collocation-based (BVPs) [6]
- Sampling-based (RRT, PF methods)
- Optimization-based (MPC, SQP [7], CPO [8])

Older and newer solutions combined

- Defining a set of flat outputs
- Path planning by polynomial optimization
- Form QP to speed up/slow down time

---

[6] T. Glück et al., "Swing-up control of a triple pendulum", Automatica, 2013.

[6] Link to cool video: `https://www.youtube.com/watch?v=cyN-CRNrb3E`

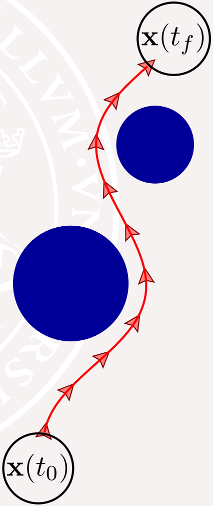
[7] Link to nice slides by Boyd `https://web.stanford.edu/class/ee364b/lectures/seq_slides.pdf`

[8] C. Richter, "Polynomial trajectory planning for aggressive quadrotor flight in dense indoor environments", 2013.

Introduction

Motion planning

$$\dot{\mathbf{x}}(t) = \mathbf{f}(\mathbf{x}(t), \mathbf{u}(t))$$
$$\mathbf{x}(t) \in \mathcal{X}, \quad \mathbf{u}(t) \in \mathcal{U}$$
$$t \in [t_0, t_f]$$



Approaches

- Collocation-based (BVPs) [6]
- Sampling-based (RRT, PF methods)
- Optimization-based (MPC, SQP [7], CPO [8])

Older and newer solutions combined

- Defining a set of flat outputs
- Path planning by polynomial optimization
- Form QP to speed up/slow down time

---

[6] T. Glück et al., "Swing-up control of a triple pendulum", Automatica, 2013.

[6] Link to cool video: `https://www.youtube.com/watch?v=cyN-CRNrb3E`

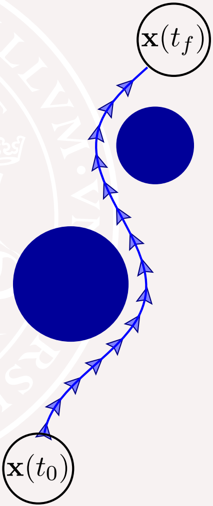
[7] Link to nice slides by Boyd `https://web.stanford.edu/class/ee364b/lectures/seq_slides.pdf`

[8] C. Richter, "Polynomial trajectory planning for aggressive quadrotor flight in dense indoor environments", 2013.

Introduction

Problem formulation

Find (1) a flat output space $\mathcal{F}$, and (2) a feasible flat-output trajectory $\gamma(t) \in \mathcal{F}$, which drives the system from an initial state $\mathbf{x}(t_0)$ to a terminal state $\mathbf{x}(t_f)$, and (3) find an augmented trajectory, $\gamma^*(t)$, minimising t_f without altering the shape of $\gamma(t)$ in $\mathcal{F}$ given constraints in $\mathbf{x}(t) \in \mathcal{X}$ and $\mathbf{u}(t) \in \mathcal{U}$.

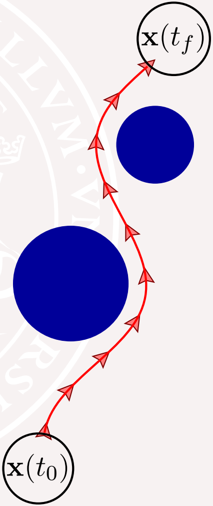


[9] M. Greiff, "A Time-warping Transformation for Differentially Flat Systems", ACC, 2018.

Introduction

Problem formulation

Find (1) a flat output space $\mathcal{F}$, and (2) a feasible flat-output trajectory $\color{blue}{\gamma(t)} \in \mathcal{F}$, which drives the system from an initial state $\mathbf{x}(t_0)$ to a terminal state $\mathbf{x}(t_f)$, and (3) find an augmented trajectory, $\color{red}{\gamma^*(t)}$, minimising t_f without altering the shape of $\color{blue}{\gamma(t)}$ in $\mathcal{F}$ given constraints in $\mathbf{x}(t) \in \mathcal{X}$ and $\mathbf{u}(t) \in \mathcal{U}$.

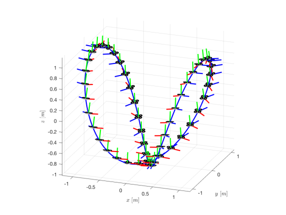


[9] M. Greiff, “A Time-warping Transformation for Differentially Flat Systems”, ACC, 2018.

Introduction

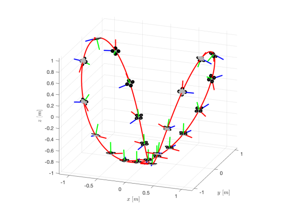
Problem formulation

> *Find (1) a flat output space $\mathcal{F}$, and (2) a feasible flat-output trajectory $\boldsymbol{\gamma}(t) \in \mathcal{F}$, which drives the system from an initial state $\mathbf{x}(t_0)$ to a terminal state $\mathbf{x}(t_f)$, and (3) find an augmented trajectory, $\boldsymbol{\gamma}^*(t)$, minimising t_f without altering the shape of $\boldsymbol{\gamma}(t)$ in $\mathcal{F}$ given constraints in $\mathbf{x}(t) \in \mathcal{X}$ and $\mathbf{u}(t) \in \mathcal{U}$.*



Outline

1. Introduce the concept of differential flatness
2. Plan path in flat output space $\mathcal{F}$
3. QP to warp the rate at which time flows [9]
4. Demonstrate approach in three control examples



[9] M. Greiff, "A Time-warping Transformation for Differentially Flat Systems", ACC, 2018.

1. Differentially Flat Dynamics - Definitions

Swedish	English	Meaning
Platthet	*Flatness*	1. Having a level surface without raised areas or indentations. 2. Lack of emotion or enthusiasm.
Plattityd	*Platitude*	A statement which is considered meaningless and boring.

1. Differentially Flat Dynamics - Definitions

Swedish	**English**	**Meaning**
Platthet	*Flatness*	1. Having a level surface without raised areas or indentations. 2. Lack of emotion or enthusiasm.
Plattityd	*Platitude*	A statement which is considered meaningless and boring.

Remark (Jokingly by Anders and Rolf)

Differentiell platthet är bara en plattityd

Differential flatness is only a platitude

1. Differentially Flat Dynamics - Definitions

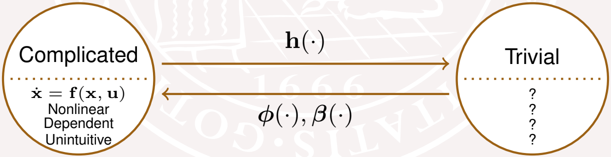
Swedish	English	Meaning
Platthet	*Flatness*	1. Having a level surface without raised areas or indentations. 2. Lack of emotion or enthusiasm.
Plattityd	*Platitude*	A statement which is considered meaningless and boring.

Conceptually not “meaningless”, but concerns “boring/trivial” systems

1. Differentially Flat Dynamics - Definitions

Swedish	English	Meaning
Platthet	*Flatness*	1. Having a level surface without raised areas or indentations. 2. Lack of emotion or enthusiasm.
Plattityd	*Platitude*	A statement which is considered meaningless and boring.

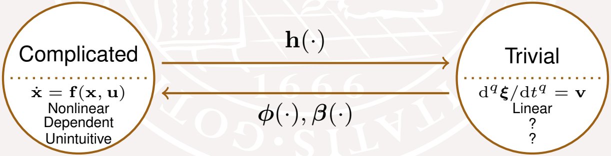
Conceptually not “meaningless”, but concerns “boring/trivial” systems



1. Differentially Flat Dynamics - Definitions

Swedish	**English**	**Meaning**
Platthet	*Flatness*	1. Having a level surface without raised areas or indentations. 2. Lack of emotion or enthusiasm.
Plattityd	*Platitude*	A statement which is considered meaningless and boring.

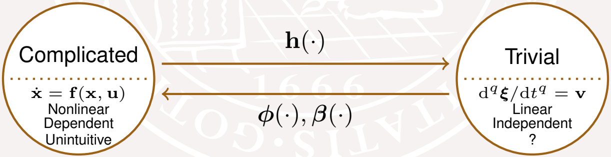
Conceptually not "meaningless", but concerns "boring/trivial" systems



1. Differentially Flat Dynamics - Definitions

Swedish	**English**	**Meaning**
Platthet	*Flatness*	1. Having a level surface without raised areas or indentations. 2. Lack of emotion or enthusiasm.
Plattityd	*Platitude*	A statement which is considered meaningless and boring.

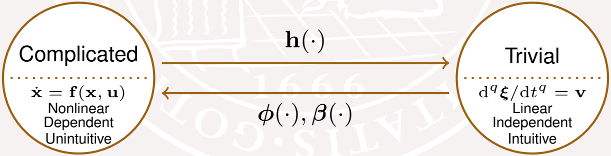
Conceptually not "meaningless", but concerns "boring/trivial" systems



1. Differentially Flat Dynamics - Definitions

Swedish	**English**	**Meaning**
Platthet	*Flatness*	1. Having a level surface without raised areas or indentations. 2. Lack of emotion or enthusiasm.
Plattityd	*Platitude*	A statement which is considered meaningless and boring.

Conceptually not “meaningless”, but concerns “boring/trivial” systems



1. Differentially Flat Dynamics - Definition

Definition (Differential Flatness [10])

A system, $\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}, \mathbf{u})$, with $\mathbf{x} \in \mathbb{R}^n$, $\mathbf{u} \in \mathbb{R}^m$, where $\mathbf{f}$ is a smooth vector field is *differentially flat* if there exists a set of *flat outputs*,

$$\boldsymbol{\gamma} = \mathbf{h}(\mathbf{x}, \mathbf{u}, \dot{\mathbf{u}}, \cdots, \mathbf{u}^{(r)}) \in \mathbb{R}^m,$$

such that

$$\mathbf{x} = \boldsymbol{\phi}(\boldsymbol{\gamma}, \dot{\boldsymbol{\gamma}}, \cdots, \boldsymbol{\gamma}^{(q)}), \qquad \mathbf{u} = \boldsymbol{\beta}(\boldsymbol{\gamma}, \dot{\boldsymbol{\gamma}}, \cdots, \boldsymbol{\gamma}^{(q)}),$$

where $\{\mathbf{h}, \boldsymbol{\phi}, \boldsymbol{\beta}\}$ are smooth functions.

[10] M. Fliess, J. Lévine, P. Martin, and P. Rouchon, “Flatness and defect of non-linear systems,” 1995.

[11] G. Rigatos, “Nonlinear control and filtering using differential flatness approaches,” 2015.

1. Differentially Flat Dynamics - Definition

Definition (Differential Flatness [10])

A system, $\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}, \mathbf{u})$, with $\mathbf{x} \in \mathbb{R}^n$, $\mathbf{u} \in \mathbb{R}^m$, where $\mathbf{f}$ is a smooth vector field is *differentially flat* if there exists a set of *flat outputs*,

$$\boldsymbol{\gamma} = \mathbf{h}(\mathbf{x}, \mathbf{u}, \dot{\mathbf{u}}, \cdots, \mathbf{u}^{(r)}) \in \mathbb{R}^m,$$

such that

$$\mathbf{x} = \boldsymbol{\phi}(\boldsymbol{\gamma}, \dot{\boldsymbol{\gamma}}, \cdots, \boldsymbol{\gamma}^{(q)}), \qquad \mathbf{u} = \boldsymbol{\beta}(\boldsymbol{\gamma}, \dot{\boldsymbol{\gamma}}, \cdots, \boldsymbol{\gamma}^{(q)}),$$

where $\{\mathbf{h}, \boldsymbol{\phi}, \boldsymbol{\beta}\}$ are smooth functions.

Useful? Which of (A) and (B) would you rather do planning for?

$$(A) \quad \frac{\mathrm{d}\mathbf{x}}{\mathrm{d}t} = \mathbf{f}(\mathbf{x}, \mathbf{u}),$$
$$(B) \quad \frac{\mathrm{d}^q\boldsymbol{\gamma}}{\mathrm{d}t^q} = \mathbf{v}(t)$$

1. Differentially Flat Dynamics - Definition

Definition (Differential Flatness [10])

A system, $\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}, \mathbf{u})$, with $\mathbf{x} \in \mathbb{R}^n$, $\mathbf{u} \in \mathbb{R}^m$, where $\mathbf{f}$ is a smooth vector field is *differentially flat* if there exists a set of *flat outputs*,

$$\boldsymbol{\gamma} = \mathbf{h}(\mathbf{x}, \mathbf{u}, \dot{\mathbf{u}}, \cdots, \mathbf{u}^{(r)}) \in \mathbb{R}^m,$$

such that

$$\mathbf{x} = \boldsymbol{\phi}(\boldsymbol{\gamma}, \dot{\boldsymbol{\gamma}}, \cdots, \boldsymbol{\gamma}^{(q)}), \qquad \mathbf{u} = \boldsymbol{\beta}(\boldsymbol{\gamma}, \dot{\boldsymbol{\gamma}}, \cdots, \boldsymbol{\gamma}^{(q)}),$$

where $\{\mathbf{h}, \boldsymbol{\phi}, \boldsymbol{\beta}\}$ are smooth functions.

$$(A) \quad \frac{\mathrm{d}\mathbf{x}}{\mathrm{d}t} = \mathbf{f}(\mathbf{x}, \mathbf{u}),$$

$$(B) \quad \frac{\mathrm{d}}{\mathrm{d}t} \begin{bmatrix} \boldsymbol{\gamma} \\ \dot{\boldsymbol{\gamma}} \\ \vdots \\ \boldsymbol{\gamma}^{q-1} \end{bmatrix} = \begin{bmatrix} \mathbf{0}_m & \mathbf{I}_m & \cdots & \mathbf{0}_m \\ \mathbf{0}_m & \mathbf{0}_m & \ddots & \mathbf{0}_m \\ \vdots & \ddots & \ddots & \vdots \\ \mathbf{0}_m & \mathbf{0}_m & \cdots & \mathbf{0}_m \end{bmatrix} \begin{bmatrix} \boldsymbol{\gamma} \\ \dot{\boldsymbol{\gamma}} \\ \vdots \\ \boldsymbol{\gamma}^{q-1} \end{bmatrix} + \begin{bmatrix} \mathbf{0}_m \\ \mathbf{0}_m \\ \vdots \\ \mathbf{I}_m \end{bmatrix} \mathbf{v}$$

1. Differentially Flat Dynamics - Definition

Definition (Differential Flatness [10])

A system, $\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}, \mathbf{u})$, with $\mathbf{x} \in \mathbb{R}^n$, $\mathbf{u} \in \mathbb{R}^m$, where $\mathbf{f}$ is a smooth vector field is *differentially flat* if there exists a set of *flat outputs*,

$$\boldsymbol{\gamma} = \mathbf{h}(\mathbf{x}, \mathbf{u}, \dot{\mathbf{u}}, \cdots, \mathbf{u}^{(r)}) \in \mathbb{R}^m,$$

such that

$$\mathbf{x} = \boldsymbol{\phi}(\boldsymbol{\gamma}, \dot{\boldsymbol{\gamma}}, \cdots, \boldsymbol{\gamma}^{(q)}), \qquad \mathbf{u} = \boldsymbol{\beta}(\boldsymbol{\gamma}, \dot{\boldsymbol{\gamma}}, \cdots, \boldsymbol{\gamma}^{(q)}),$$

where $\{\mathbf{h}, \boldsymbol{\phi}, \boldsymbol{\beta}\}$ are smooth functions.

$$(A) \quad \frac{\mathrm{d}\mathbf{x}}{\mathrm{d}t} = \mathbf{f}(\mathbf{x}, \mathbf{u}),$$

$$(B) \quad \frac{\mathrm{d}}{\mathrm{d}t}\begin{bmatrix} \gamma_i \\ \dot{\gamma}_i \\ \vdots \\ \gamma_i^{q-1} \end{bmatrix} = \begin{bmatrix} 0 & 1 & \cdots & 0 \\ 0 & 0 & \ddots & 0 \\ \vdots & \ddots & \ddots & \vdots \\ 0 & 0 & \cdots & 0 \end{bmatrix} \begin{bmatrix} \gamma_i \\ \dot{\gamma}_i \\ \vdots \\ \gamma_i^{q-1} \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ \vdots \\ 1 \end{bmatrix} v_i \quad \forall i = 1, \ldots, m$$

1. Differentially Flat Dynamics - Definition

Definition (Differential Flatness [10])

A system, $\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}, \mathbf{u})$, with $\mathbf{x} \in \mathbb{R}^n$, $\mathbf{u} \in \mathbb{R}^m$, where $\mathbf{f}$ is a smooth vector field is *differentially flat* if there exists a set of *flat outputs*,

$$\boldsymbol{\gamma} = \mathbf{h}(\mathbf{x}, \mathbf{u}, \dot{\mathbf{u}}, \cdots, \mathbf{u}^{(r)}) \in \mathbb{R}^m,$$

such that

$$\mathbf{x} = \boldsymbol{\phi}(\boldsymbol{\gamma}, \dot{\boldsymbol{\gamma}}, \cdots, \boldsymbol{\gamma}^{(q)}), \qquad \mathbf{u} = \boldsymbol{\beta}(\boldsymbol{\gamma}, \dot{\boldsymbol{\gamma}}, \cdots, \boldsymbol{\gamma}^{(q)}),$$

where $\{\mathbf{h}, \boldsymbol{\phi}, \boldsymbol{\beta}\}$ are smooth functions.

Useful? Yes, simplifies planning problem significantly.

1. Differentially Flat Dynamics - Definition

Definition (Differential Flatness [10])

A system, $\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}, \mathbf{u})$, with $\mathbf{x} \in \mathbb{R}^n$, $\mathbf{u} \in \mathbb{R}^m$, where $\mathbf{f}$ is a smooth vector field is *differentially flat* if there exists a set of *flat outputs*,

$$\boldsymbol{\gamma} = \mathbf{h}(\mathbf{x}, \mathbf{u}, \dot{\mathbf{u}}, \cdots, \mathbf{u}^{(r)}) \in \mathbb{R}^m,$$

such that

$$\mathbf{x} = \boldsymbol{\phi}(\boldsymbol{\gamma}, \dot{\boldsymbol{\gamma}}, \cdots, \boldsymbol{\gamma}^{(q)}), \qquad \mathbf{u} = \boldsymbol{\beta}(\boldsymbol{\gamma}, \dot{\boldsymbol{\gamma}}, \cdots, \boldsymbol{\gamma}^{(q)}),$$

where $\{\mathbf{h}, \boldsymbol{\phi}, \boldsymbol{\beta}\}$ are smooth functions.

Useful? Yes, simplifies planning problem significantly.

Constructive? Yes, but it can be challenging to find $\{\mathbf{h}, \boldsymbol{\phi}, \boldsymbol{\beta}\}$...!

1. Differentially Flat Dynamics - Definition

Definition (Differential Flatness [10])

A system, $\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}, \mathbf{u})$, with $\mathbf{x} \in \mathbb{R}^n$, $\mathbf{u} \in \mathbb{R}^m$, where $\mathbf{f}$ is a smooth vector field is *differentially flat* if there exists a set of *flat outputs*,

$$\boldsymbol{\gamma} = \mathbf{h}(\mathbf{x}, \mathbf{u}, \dot{\mathbf{u}}, \cdots, \mathbf{u}^{(r)}) \in \mathbb{R}^m,$$

such that

$$\mathbf{x} = \boldsymbol{\phi}(\boldsymbol{\gamma}, \dot{\boldsymbol{\gamma}}, \cdots, \boldsymbol{\gamma}^{(q)}), \qquad \mathbf{u} = \boldsymbol{\beta}(\boldsymbol{\gamma}, \dot{\boldsymbol{\gamma}}, \cdots, \boldsymbol{\gamma}^{(q)}),$$

where $\{\mathbf{h}, \boldsymbol{\phi}, \boldsymbol{\beta}\}$ are smooth functions.

Useful? Yes, simplifies planning problem significantly.

Constructive? Yes, but it can be challenging to find $\{\mathbf{h}, \boldsymbol{\phi}, \boldsymbol{\beta}\}$...!

Example: Feedback-linearization, Chapter 13 in [12]

[10] M. Fliess, J. Lévine, P. Martin, and P. Rouchon, “Flatness and defect of non-linear systems,” 1995.

[11] G. Rigatos, “Nonlinear control and filtering using differential flatness approaches,” 2015.

[12] H. Khalil et al., “Nonlinear Systems”, available online as a pdf.

1. Differentially Flat Dynamics - Toy Example

Consider a system $\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}) + \mathbf{g}(\mathbf{x})u$, configured on $\mathcal{C} = \mathbb{R}^3$, where

$$\mathbf{f}(\mathbf{x}) = \begin{bmatrix} -x_1 + x_2 \\ x_1 - x_2 - x_1 x_3 \\ x_1 + x_1 x_2 - 2x_3 \end{bmatrix}, \qquad \mathbf{g}(\mathbf{x}) = \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix}$$

[12] H. Khalil et al., “Nonlinear Systems”, available online as a pdf.

1. Differentially Flat Dynamics - Toy Example

Consider a system $\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}) + \mathbf{g}(\mathbf{x})u$, configured on $\mathcal{C} = \mathbb{R}^3$, where

$$\mathbf{f}(\mathbf{x}) = \begin{bmatrix} -x_1 + x_2 \\ x_1 - x_2 - x_1 x_3 \\ x_1 + x_1 x_2 - 2x_3 \end{bmatrix}, \qquad \mathbf{g}(\mathbf{x}) = \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix}$$

- Show how to derive flat outputs
- Enough details to do it yourselves
- Temporarily a bit more mathematical
- Don't worry if it is a bit tricky to follow

[12] H. Khalil et al., "Nonlinear Systems", available online as a pdf.

1. Differentially Flat Dynamics - Toy Example

Consider a system $\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}) + \mathbf{g}(\mathbf{x})u$, configured on $\mathcal{C} = \mathbb{R}^3$, where

$$\mathbf{f}(\mathbf{x}) = \begin{bmatrix} -x_1 + x_2 \\ x_1 - x_2 - x_1 x_3 \\ x_1 + x_1 x_2 - 2x_3 \end{bmatrix}, \qquad \mathbf{g}(\mathbf{x}) = \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix}$$

Definition (Lie Derivative [12])

The so-called Lie-derivative of $\mathbf{h}$ with respect to $\mathbf{f}$,

$$(\mathcal{L}_{\mathbf{f}}\mathbf{h})(\mathbf{x}) = \frac{\partial \mathbf{h}}{\partial \mathbf{x}}\mathbf{f}(\mathbf{x})$$

denotes a change in $\mathbf{h}$ along the trajectories of the system $\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x})$

[12] H. Khalil et al., “Nonlinear Systems”, available online as a pdf.

1. Differentially Flat Dynamics - Toy Example

Consider a system $\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}) + \mathbf{g}(\mathbf{x})u$, configured on $\mathcal{C} = \mathbb{R}^3$, where

$$\mathbf{f}(\mathbf{x}) = \begin{bmatrix} -x_1 + x_2 \\ x_1 - x_2 - x_1 x_3 \\ x_1 + x_1 x_2 - 2x_3 \end{bmatrix}, \qquad \mathbf{g}(\mathbf{x}) = \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix}$$

1. Find an output $\gamma = h(\mathbf{x})$ yielding full relative degree, i.e. such that

$$\begin{aligned} L_{\mathbf{g}} h(\mathbf{x}) &= 0 \\ L_{\mathbf{g}} L_{\mathbf{f}} h(\mathbf{x}) &= 0 \\ L_{\mathbf{g}} L_{\mathbf{f}}^2 h(\mathbf{x}) &\neq 0 \end{aligned}$$

[12] H. Khalil et al., “Nonlinear Systems”, available online as a pdf.

1. Differentially Flat Dynamics - Toy Example

Consider a system $\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}) + \mathbf{g}(\mathbf{x})u$, configured on $\mathcal{C} = \mathbb{R}^3$, where

$$\mathbf{f}(\mathbf{x}) = \begin{bmatrix} -x_1 + x_2 \\ x_1 - x_2 - x_1 x_3 \\ x_1 + x_1 x_2 - 2x_3 \end{bmatrix}, \qquad \mathbf{g}(\mathbf{x}) = \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix}$$

1. Find an output $\gamma = h(\mathbf{x})$ yielding full relative degree, i.e. such that

$$L_{\mathbf{g}} h(\mathbf{x}) = 0$$
$$L_{\mathbf{g}} L_{\mathbf{f}} h(\mathbf{x}) = 0$$
$$L_{\mathbf{g}} L_{\mathbf{f}}^2 h(\mathbf{x}) \neq 0$$

such as

$$h(\mathbf{x}) = a\left(\frac{1}{2}x_1^2 - x_3\right) + b, \qquad a \in \mathbb{R}\backslash\{0\}, b \in \mathbb{R}.$$

[12] H. Khalil et al., “Nonlinear Systems”, available online as a pdf.

1. Differentially Flat Dynamics - Toy Example

Consider a system $\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}) + \mathbf{g}(\mathbf{x})u$, configured on $\mathcal{C} = \mathbb{R}^3$, where

$$\mathbf{f}(\mathbf{x}) = \begin{bmatrix} -x_1 + x_2 \\ x_1 - x_2 - x_1 x_3 \\ x_1 + x_1 x_2 - 2x_3 \end{bmatrix}, \qquad \mathbf{g}(\mathbf{x}) = \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix}$$

1. Find an output $\gamma = h(\mathbf{x})$ yielding full relative degree, i.e. such that
2. With $a = 1, b = 0$, we find a feedback linearization

$$\gamma = L_{\mathbf{f}}^0 h(x) = x_1^2/2 - x_3$$
$$\dot{\gamma} = L_{\mathbf{f}}^1 h(x) = -x_1^2 - x_1 + 2x_3$$
$$\ddot{\gamma} = L_{\mathbf{f}}^2 h(x) = 2x_1^2 + 3x_1 - x_2 - 4x_3$$

which (surprisingly) turns out to be a surjective map $\mathbf{x} = \phi(\gamma, \dot{\gamma}, \ddot{\gamma})$.

[12] H. Khalil et al., “Nonlinear Systems”, available online as a pdf.

1. Differentially Flat Dynamics - Toy Example

Consider a system $\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}) + \mathbf{g}(\mathbf{x})u$, configured on $\mathcal{C} = \mathbb{R}^3$, where

$$\mathbf{f}(\mathbf{x}) = \begin{bmatrix} -x_1 + x_2 \\ x_1 - x_2 - x_1 x_3 \\ x_1 + x_1 x_2 - 2x_3 \end{bmatrix}, \qquad \mathbf{g}(\mathbf{x}) = \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix}$$

1. Find an output $\gamma = h(\mathbf{x})$ yielding full relative degree, i.e. such that
2. With $a = 1, b = 0$, we find a surjective map $\mathbf{x} = \boldsymbol{\phi}(\gamma, \dot{\gamma}, \ddot{\gamma})$.

[12] H. Khalil et al., “Nonlinear Systems”, available online as a pdf.

1. Differentially Flat Dynamics - Toy Example

Consider a system $\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}) + \mathbf{g}(\mathbf{x})u$, configured on $\mathcal{C} = \mathbb{R}^3$, where

$$\mathbf{f}(\mathbf{x}) = \begin{bmatrix} -x_1 + x_2 \\ x_1 - x_2 - x_1x_3 \\ x_1 + x_1x_2 - 2x_3 \end{bmatrix}, \qquad \mathbf{g}(\mathbf{x}) = \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix}$$

1. Find an output $\gamma = h(\mathbf{x})$ yielding full relative degree, i.e. such that
2. With $a = 1, b = 0$, we find a surjective map $\mathbf{x} = \phi(\gamma, \dot{\gamma}, \ddot{\gamma})$.
3. With $\gamma(t) = h(\mathbf{x})$, the endogenous feedback law

$$u = \frac{1}{L_{\mathbf{g}}L_{\mathbf{f}}^2 h(\mathbf{x})}[-L_{\mathbf{f}}^3 h(\mathbf{x}) + v] = 4x_2 - 8x_1 + 8x_3 + x_1x_3 - 4x_1^2 - v$$

results in a system

$$\frac{d^3\gamma(t)}{dt^3} = v(t),$$

As $\mathbf{x}$ is known from ϕ, and v is known from $\dddot{\gamma}$, we also know $u = \beta(\gamma, \dot{\gamma}, \ddot{\gamma}, \dddot{\gamma})$.

[12] H. Khalil et al., “Nonlinear Systems”, available online as a pdf.

1. Differentially Flat Dynamics - Toy Example

Consider a system $\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}) + \mathbf{g}(\mathbf{x})u$, configured on $\mathcal{C} = \mathbb{R}^3$, where

$$\mathbf{f}(\mathbf{x}) = \begin{bmatrix} -x_1 + x_2 \\ x_1 - x_2 - x_1 x_3 \\ x_1 + x_1 x_2 - 2x_3 \end{bmatrix}, \qquad \mathbf{g}(\mathbf{x}) = \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix}$$

1. Find an output $\gamma = h(\mathbf{x})$ yielding full relative degree, i.e. such that
2. With $a = 1, b = 0$, we find a surjective map $\mathbf{x} = \phi(\gamma, \dot{\gamma}, \ddot{\gamma})$.
3. With $\gamma(t) = h(\mathbf{x})$, the endogenous feedback law results in a system

$$\frac{d^3\gamma(t)}{dt^3} = v(t),$$

As $\mathbf{x}$ is known from ϕ, and v is known from $\dddot{\gamma}$, we also know $u = \beta(\gamma, \dot{\gamma}, \ddot{\gamma}, \dddot{\gamma})$.

[12] H. Khalil et al., “Nonlinear Systems”, available online as a pdf.

1. Differentially Flat Dynamics - Toy Example

Consider a system $\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}) + \mathbf{g}(\mathbf{x})u$, configured on $\mathcal{C} = \mathbb{R}^3$, where

$$\mathbf{f}(\mathbf{x}) = \begin{bmatrix} -x_1 + x_2 \\ x_1 - x_2 - x_1x_3 \\ x_1 + x_1x_2 - 2x_3 \end{bmatrix}, \qquad \mathbf{g}(\mathbf{x}) = \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix}$$

1. Find an output $\gamma = h(\mathbf{x})$ yielding full relative degree, i.e. such that
2. With $a = 1, b = 0$, we find a surjective map $\mathbf{x} = \boldsymbol{\phi}(\gamma, \dot{\gamma}, \ddot{\gamma})$.
3. With $\gamma(t) = h(\mathbf{x})$, the endogenous feedback law results in a system

$$\frac{d^3\gamma(t)}{dt^3} = v(t),$$

   As $\mathbf{x}$ is known from ϕ, and v is known from $\dddot{\gamma}$, we also know $u = \beta(\gamma, \dot{\gamma}, \ddot{\gamma}, \dddot{\gamma})$.
4. Sufficiently smooth trajectories, here $\gamma \in C^2(\mathbb{R})$, can be followed!

[12] H. Khalil et al., “Nonlinear Systems”, available online as a pdf.

1. Differentially Flat Dynamics - Toy Example

Consider a system $\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}) + \mathbf{g}(\mathbf{x})u$, configured on $\mathcal{C} = \mathbb{R}^3$, where

$$\mathbf{f}(\mathbf{x}) = \begin{bmatrix} -x_1 + x_2 \\ x_1 - x_2 - x_1x_3 \\ x_1 + x_1x_2 - 2x_3 \end{bmatrix}, \qquad \mathbf{g}(\mathbf{x}) = \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix}$$

1. Find an output $\gamma = h(\mathbf{x})$ yielding full relative degree, i.e. such that
2. With $a = 1, b = 0$, we find a surjective map $\mathbf{x} = \boldsymbol{\phi}(\gamma, \dot{\gamma}, \ddot{\gamma})$.
3. With $\gamma(t) = h(\mathbf{x})$, the endogenous feedback law results in a system

$$\frac{d^3\gamma(t)}{dt^3} = v(t),$$

   As $\mathbf{x}$ is known from $\boldsymbol{\phi}$, and v is known from $\dddot{\gamma}$, we also know $u = \beta(\gamma, \dot{\gamma}, \ddot{\gamma}, \dddot{\gamma})$.
4. Sufficiently smooth trajectories, here $\gamma \in C^2(\mathbb{R})$, can be followed!

[12] H. Khalil et al., “Nonlinear Systems”, available online as a pdf.

1. Differentially Flat Dynamics - Toy Example

Consider a system $\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}) + \mathbf{g}(\mathbf{x})u$, configured on $\mathcal{C} = \mathbb{R}^3$, where

$$\mathbf{f}(\mathbf{x}) = \begin{bmatrix} -x_1 + x_2 \\ x_1 - x_2 - x_1 x_3 \\ x_1 + x_1 x_2 - 2x_3 \end{bmatrix}, \qquad \mathbf{g}(\mathbf{x}) = \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix}$$

1. Find an output $\gamma = h(\mathbf{x})$ yielding full relative degree, i.e. such that
2. With $a = 1, b = 0$, we find a surjective map $\mathbf{x} = \phi(\gamma, \dot{\gamma}, \ddot{\gamma})$.
3. With $\gamma(t) = h(\mathbf{x})$, the endogenous feedback law results in a system

$$\frac{d^3\gamma(t)}{dt^3} = v(t),$$

   As $\mathbf{x}$ is known from ϕ, and v is known from $\dddot{\gamma}$, we also know $u = \beta(\gamma, \dot{\gamma}, \ddot{\gamma}, \dddot{\gamma})$.
4. Sufficiently smooth trajectories, here $\gamma \in C^2(\mathbb{R})$, can be followed!

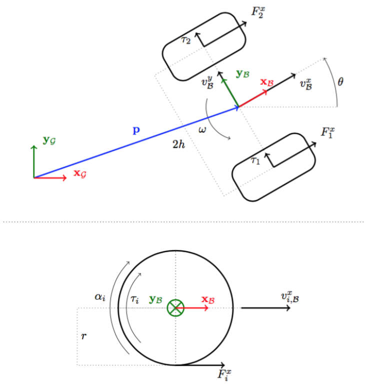
Disclaimer: Not always possible. What about other systems?

[12] H. Khalil et al., “Nonlinear Systems”, available online as a pdf.

1. Differentially Flat Dynamics - UGV (unconstrained)

Unmanned ground vehicle (UGV) configured on $\mathcal{C} = SE(2)$,

$$\mathbf{x} = \begin{bmatrix} \theta \\ p^x_{\mathcal{G}} \\ p^y_{\mathcal{G}} \\ \omega \\ v^x_{\mathcal{B}} \\ v^y_{\mathcal{B}} \end{bmatrix}$$		Attitude Translation in $\mathbf{x}_{\mathcal{G}}$ Translation in $\mathbf{y}_{\mathcal{G}}$ Attitude rate Velocity in $\mathbf{x}_{\mathcal{B}}$ Velocity in $\mathbf{y}_{\mathcal{B}}$
$$\mathbf{u} = \begin{bmatrix} \tau_1 \\ \tau_2 \end{bmatrix}$$		Torque along $\mathbf{x}_{\mathcal{B}}$ at wheel 1 Torque along $\mathbf{y}_{\mathcal{B}}$ at wheel 2



with $\mathcal{X} \subseteq \mathbb{R}^6$, $\mathcal{U} \subseteq \mathbb{R}^2$, and dynamics

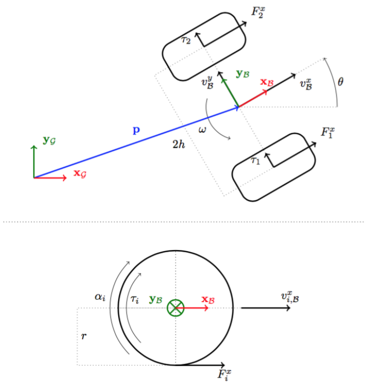
$$\begin{aligned}
\dot{\theta}(t) &= \omega_{\mathcal{B}}(t) \\
\dot{p}^x_{\mathcal{G}}(t) &= v^x_{\mathcal{B}}(t)\cos(\theta(t)) - v^y_{\mathcal{B}}(t)\sin(\theta(t)) \\
\dot{p}^y_{\mathcal{G}}(t) &= v^x_{\mathcal{B}}(t)\sin(\theta(t)) + v^y_{\mathcal{B}}(t)\cos(\theta(t)) \\
\dot{\omega}(t) &= (h/(Jr))(\tau_1(t) - \tau_2(t)) \\
\dot{v}^x_{\mathcal{B}}(t) &= \omega(t)v^y_{\mathcal{B}}(t) + (r/m)(\tau_1(t) + \tau_2(t)) \\
\dot{v}^y_{\mathcal{B}}(t) &= -\omega(t)v^x_{\mathcal{B}}(t)
\end{aligned}$$

1. Differentially Flat Dynamics - UGV (unconstrained)

Unmanned ground vehicle (UGV) configured on $\mathcal{C} = SE(2)$,

$$\mathbf{x} = \begin{bmatrix} \theta \\ p^x_{\mathcal{G}} \\ p^y_{\mathcal{G}} \\ \omega \\ v^x_{\mathcal{B}} \\ v^y_{\mathcal{B}} \end{bmatrix} \quad \begin{matrix} \text{Attitude} \\ \text{Translation in } \mathbf{x}_{\mathcal{G}} \\ \text{Translation in } \mathbf{y}_{\mathcal{G}} \\ \text{Attitude rate} \\ \text{Velocity in } \mathbf{x}_{\mathcal{B}} \\ \text{Velocity in } \mathbf{y}_{\mathcal{B}} \end{matrix}$$

$$\mathbf{u} = \begin{bmatrix} \tau_1 \\ \tau_2 \end{bmatrix} \quad \begin{matrix} \text{Torque along } \mathbf{x}_{\mathcal{B}} \text{ at wheel 1} \\ \text{Torque along } \mathbf{y}_{\mathcal{B}} \text{ at wheel 2} \end{matrix}$$



with $\mathcal{X} \subseteq \mathbb{R}^6$, $\mathcal{U} \subseteq \mathbb{R}^2$, and flat outputs

$$\boldsymbol{\gamma}(t) = \mathbf{h}(\mathbf{x}(t)) = \begin{bmatrix} p^x_{\mathcal{G}}(t) & p^y_{\mathcal{G}}(t) \end{bmatrix}^T \in C^3(\mathbb{R}^2)$$

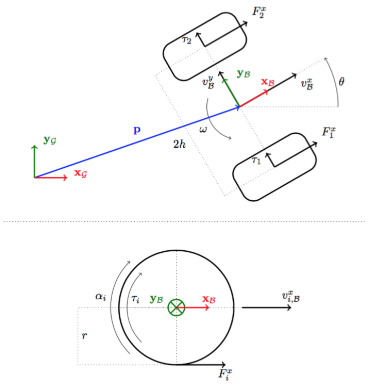
in the flat output space $\mathcal{F}$.

1. Differentially Flat Dynamics - UGV (constrained)

Constrained UGV with no lateral slip configured on $\mathcal{C} = SE(2)$,

$$\mathbf{x} = \begin{bmatrix} p_{\mathcal{G}}^x \\ p_{\mathcal{G}}^y \\ \theta \\ \dot{\alpha}_1 \\ \dot{\alpha}_2 \end{bmatrix} \quad \begin{matrix} \text{Attitude} \\ \text{Translation in } \mathbf{x}_{\mathcal{G}} \\ \text{Translation in } \mathbf{y}_{\mathcal{G}} \\ \text{Angular rate of wheel 1} \\ \text{Angular rate of wheel 2} \end{matrix}$$

$$\mathbf{u} = \begin{bmatrix} \tau_1 \\ \tau_2 \end{bmatrix} \quad \begin{matrix} \text{Torque along } \mathbf{x}_{\mathcal{B}} \text{ at wheel 1} \\ \text{Torque along } \mathbf{y}_{\mathcal{B}} \text{ at wheel 2} \end{matrix}$$



with $\mathcal{X} \subseteq \mathbb{R}^5, \mathcal{U} \subseteq \mathbb{R}^2$, and dynamics

$$\dot{p}_{\mathcal{G}}^x(t) = (\dot{\alpha}_1(t) + \dot{\alpha}_2(t))(r/2)\cos(\theta)$$

$$\dot{p}_{\mathcal{G}}^y(t) = (\dot{\alpha}_1(t) + \dot{\alpha}_2(t))(r/2)\sin(\theta)$$

$$\dot{\theta}(t) = (\dot{\alpha}_2 - \dot{\alpha}_1(t))/(2h)$$

$$\ddot{\alpha}_1(t) = J_1^{-1}\tau_1(t)$$

$$\ddot{\alpha}_2(t) = J_2^{-1}\tau_2(t)$$

1. Differentially Flat Dynamics - UGV (constrained)

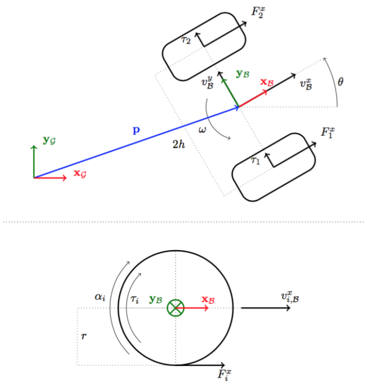
Constrained UGV with no lateral slip configured on $\mathcal{C} = SE(2)$,

$$\mathbf{x} = \begin{bmatrix} p_{\mathcal{G}}^{x} \\ p_{\mathcal{G}}^{y} \\ \theta \\ \dot{\alpha}_1 \\ \dot{\alpha}_2 \end{bmatrix}$$

Attitude
Translation in $\mathbf{x}_{\mathcal{G}}$
Translation in $\mathbf{y}_{\mathcal{G}}$
Angular rate of wheel 1
Angular rate of wheel 2

$$\mathbf{u} = \begin{bmatrix} \tau_1 \\ \tau_2 \end{bmatrix}$$

Torque along $\mathbf{x}_{\mathcal{B}}$ at wheel 1
Torque along $\mathbf{y}_{\mathcal{B}}$ at wheel 2



with $\mathcal{X} \subseteq \mathbb{R}^5, \mathcal{U} \subseteq \mathbb{R}^2$, and flat outputs

$$\boldsymbol{\gamma}(t) = \mathbf{h}(\mathbf{x}(t)) = \begin{bmatrix} p_{\mathcal{G}}^{x}(t) & p_{\mathcal{G}}^{y}(t) \end{bmatrix}^T \in C^2(\mathbb{R}^2)$$

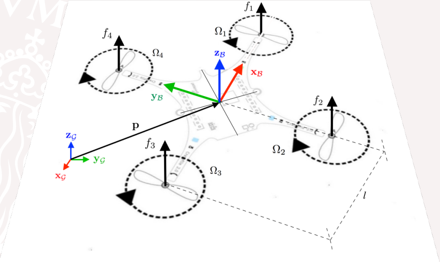
in the flat output space $\mathcal{F}$.

1. Differentially Flat Dynamics - UAV

Unmanned Aerial Vehicle (UAV) configured on $\mathcal{C} = SE(3)$,

$$\mathbf{x} = \begin{bmatrix} \mathbf{p}_\mathcal{G} \\ \mathbf{R} \\ \mathbf{v}_\mathcal{G} \\ \boldsymbol{\omega} \end{bmatrix} \quad \begin{matrix} \text{Translation} \\ \text{Attitude} \\ \text{Translational Velocity} \\ \text{Angular rate} \end{matrix}$$

$$\mathbf{u} = \begin{bmatrix} f_z \\ \tau_x \\ \tau_y \\ \tau_z \end{bmatrix} \quad \begin{matrix} \text{Positive force along } \mathbf{z}_\mathcal{B} \\ \text{Torque along } \mathbf{x}_\mathcal{B} \\ \text{Torque along } \mathbf{y}_\mathcal{B} \\ \text{Torque along } \mathbf{z}_\mathcal{B} \end{matrix}$$



with $\mathcal{X} \subseteq \mathbb{R}^9 \times \mathbb{S}^2, \mathcal{U} \subseteq \mathbb{R}^4$, but we may instead use rotor speeds $\boldsymbol{\Omega} = [\Omega_1, \cdots, \Omega_4]$ as inputs

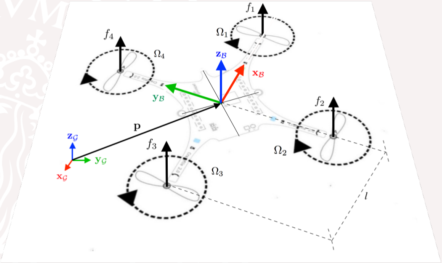
$$\begin{bmatrix} f_z(t) \\ \tau_x(t) \\ \tau_y(t) \\ \tau_z(t) \end{bmatrix} = \begin{bmatrix} k\sum_{i=1}^{4} \Omega_i^2(t) \\ kl(-\Omega_2^2(t) + \Omega_4^2(t)) \\ kl(-\Omega_1^2(t) + \Omega_3^2(t)) \\ \sum_{i=1}^{4} b\Omega_i^2(t) + I_M\dot{\Omega}_i(t) \end{bmatrix}$$

1. Differentially Flat Dynamics - UAV

Unmanned Aerial Vehicle (UAV) configured on $\mathcal{C} = SE(3)$,

$$\mathbf{x} = \begin{bmatrix} \mathbf{p}_\mathcal{G} \\ \mathbf{R} \\ \mathbf{v}_\mathcal{G} \\ \boldsymbol{\omega} \end{bmatrix} \quad \begin{matrix} \text{Translation} \\ \text{Attitude} \\ \text{Translational Velocity} \\ \text{Angular rate} \end{matrix}$$

$$\mathbf{u} = \begin{bmatrix} f_z \\ \tau_x \\ \tau_y \\ \tau_z \end{bmatrix} \quad \begin{matrix} \text{Positive force along } \mathbf{z}_\mathcal{B} \\ \text{Torque along } \mathbf{x}_\mathcal{B} \\ \text{Torque along } \mathbf{y}_\mathcal{B} \\ \text{Torque along } \mathbf{z}_\mathcal{B} \end{matrix}$$



with $\mathcal{X} \subseteq \mathbb{R}^9 \times \mathbb{S}^2, \mathcal{U} \subseteq \mathbb{R}^4$, and flat outputs in $\mathbf{u}$ or $\boldsymbol{\Omega}$,

$$\boldsymbol{\gamma}(t) = \mathbf{h}(\mathbf{x}(t)) = \begin{bmatrix} \mathbf{p}_\mathcal{G}^T(t) & \psi(t) \end{bmatrix}^T \in C^5(\mathbb{R}^4)$$

defines the flat outputs [13, 14].

[13] D. W. Mellinger, "Trajectory generation and control for quadrotors", PhD Thesis, 2012, available online as a pdf.

[14] M. Greiff, "Modelling and control of the crazyflie quadrotor", M.Sc. Thesis, 2017, available online as a pdf.

1. Differentially flat systems - Summary

Main takeaways

1. A very large number of systems are “boring” [11, 15]
2. Ways of finding flat outputs exists (feedback linearization)
3. Almost always found as functions of the system configurations
4. Independent planning in flat output dimensions
5. Plan for smoothness in γ instead of explicitly enforcing $\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}, \mathbf{u})$ in time.

Toy example	$\gamma \in C^2(\mathbb{R}^1)$
UGV (unconstrained)	$\boldsymbol{\gamma} \in C^3(\mathbb{R}^2)$
UGV (constrained)	$\boldsymbol{\gamma} \in C^2(\mathbb{R}^2)$
UAV	$\boldsymbol{\gamma} \in C^5(\mathbb{R}^4)$

[11] G. Rigatos, “Nonlinear control and filtering using differential flatness approaches,” 2015.

[15] R. Murray et al., “Differential flatness of mechanical control systems,” 1995.

1. Differentially flat systems - Summary

Main takeaways

1. A very large number of systems are “boring” [11, 15]
2. Ways of finding flat outputs exists (feedback linearization)
3. Almost always found as functions of the system configurations
4. Independent planning in flat output dimensions
5. Plan for smoothness in γ instead of explicitly enforcing $\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}, \mathbf{u})$ in time.

Toy example	$\gamma \in C^2(\mathbb{R}^1)$
UGV (unconstrained)	$\boldsymbol{\gamma} \in C^3(\mathbb{R}^2)$
UGV (constrained)	$\boldsymbol{\gamma} \in C^2(\mathbb{R}^2)$
UAV	$\boldsymbol{\gamma} \in C^5(\mathbb{R}^4)$

Suitable parameterizations of the flat trajectories?

- Sinusoids
- Bezier curves
- LP-filtered signals
- Polynomials

1. Differentially flat systems - Summary

Main takeaways

1. A very large number of systems are “boring” [11, 15]
2. Ways of finding flat outputs exists (feedback linearization)
3. Almost always found as functions of the system configurations
4. Independent planning in flat output dimensions
5. Plan for smoothness in γ instead of explicitly enforcing $\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}, \mathbf{u})$ in time.

Toy example	$\gamma \in C^2(\mathbb{R}^1)$
UGV (unconstrained)	$\boldsymbol{\gamma} \in C^3(\mathbb{R}^2)$
UGV (constrained)	$\boldsymbol{\gamma} \in C^2(\mathbb{R}^2)$
UAV	$\boldsymbol{\gamma} \in C^5(\mathbb{R}^4)$

Suitable parameterizations of the flat trajectories?

- Sinusoids
- Bezier curves
- LP-filtered signals
- Polynomials

2. Path Planning in Flat Output Space

Path planning with polynomials (CPO) [8]

1. Consider n polynomial splines $P_1(t), ..., P_n(t)$ with $\deg(P_k) = N$, as

$$P_k(t) = \sum_{i=0}^{N} p_{k,i} t^i = \mathbf{p}_{(k)}^T \mathbf{t}(t), \qquad t \in [0, T_k], \qquad \mathbf{p}_{(k)} = [p_{k,0}, ..., p_{k,N}]^T$$

[8] C. Richter, “Polynomial trajectory planning for aggressive quadrotor flight in dense indoor environments”, 2013.

2. Path Planning in Flat Output Space

Path planning with polynomials (CPO) [8]

1. Consider n polynomial splines $P_1(t), ..., P_n(t)$ with $\deg(P_k) = N$
2. Integral cost associated with sum of spline derivatives

$$J(T_k) = \sum_{i=0}^{N} \int_0^{T_k} c_i \left\| \frac{dP_k^{(i)}(t)}{dt} \right\|_2^2 dt = \mathbf{p}_{(k)}^T \mathbf{Q}_{(k)} \mathbf{p}_{(k)}$$

[8] C. Richter, “Polynomial trajectory planning for aggressive quadrotor flight in dense indoor environments”, 2013.

2. Path Planning in Flat Output Space

Path planning with polynomials (CPO) [8]

1. Consider n polynomial splines $P_1(t), \ldots, P_n(t)$ with $\deg(P_k) = N$
2. Integral cost associated with sum of spline derivatives

$$J(T_k) = \sum_{i=0}^{N} \int_0^{T_k} c_i \left\| \frac{dP_k^{(i)}(t)}{dt} \right\|_2^2 dt = \mathbf{p}_{(k)}^T \mathbf{Q}_{(k)} \mathbf{p}_{(k)}$$

Remark (Regarding the cost)

Spline	Name	Objective
$P_k(t)$	Position	Small/Large (often $c_0 = 0$)
$\frac{\mathrm{d}^1}{\mathrm{d}t^1} P_k(t)$	Velocity	Small/Large (often $c_1 = 0$)
$\frac{\mathrm{d}^2}{\mathrm{d}t^2} P_k(t)$	Acceleration	Small (often $c_2 > 0$)
$\frac{\mathrm{d}^3}{\mathrm{d}t^3} P_k(t)$	Jerk	Small (often $c_3 > 0$)
$\frac{\mathrm{d}^4}{\mathrm{d}t^4} P_k(t)$	Snap	Small (often $c_4 \gg 0$)

Minimum snap: $c_4 > 0$, $c_i = 0 \forall i \neq 4$.

Path planning with polynomials (CPO) [8]

1. Consider n polynomial splines $P_1(t), ..., P_n(t)$ with $\deg(P_k) = N$
2. Integral cost associated with sum of spline derivatives

$$J(T_k) = \sum_{i=0}^{N} \int_0^{T_k} c_i \left\| \frac{dP_k^{(i)}(t)}{dt} \right\|_2^2 dt = \mathbf{p}_{(k)}^T \mathbf{Q}_{(k)} \mathbf{p}_{(k)}$$

Remark (Regarding the smoothness)

If we need a function $C^M(\mathbb{R})$, add constraint

$$\frac{\mathrm{d}^m}{\mathrm{d}t^m} P_k(T_k) = \frac{\mathrm{d}^m}{\mathrm{d}t^m} P_{k+1}(0) \quad \forall m = 0, ..., M, \quad k = 1, ..., n-1 \tag{4}$$

which is linear in $\mathbf{p}_{(k)}$ and $\mathbf{p}_{(k+1)}$ given T_k.

2. Path Planning in Flat Output Space

Path planning with polynomials (CPO) [8]

1. Consider n polynomial splines $P_1(t), ..., P_n(t)$ with $\deg(P_k) = N$
2. Integral cost associated with sum of spline derivatives

$$J(T_k) = \sum_{i=0}^{N} \int_0^{T_k} c_i \left\| \frac{dP_k^{(i)}(t)}{dt} \right\|_2^2 dt = \mathbf{p}_{(k)}^T \mathbf{Q}_{(k)} \mathbf{p}_{(k)}$$

3. Sum cost over all splines with $\mathbf{p} = [\mathbf{p}_{(1)}^T, ..., \mathbf{p}_{(n)}^T]^T$

$$\text{Minimize} \quad \sum_{k=1}^{n} J(T_k) \quad \Rightarrow \quad \text{Minimize} \quad \mathbf{p}^T \mathbf{Q} \mathbf{p}$$

$$\text{Subject to} \quad P_k(t) \in C^M(\mathbb{R}) \;\; \forall k = 1, .., n \quad \Rightarrow \quad \text{Subject to} \quad \mathbf{A}\mathbf{p} - \mathbf{b} = \mathbf{0}.$$

4. Do this independently for each flat dimension

[8] C. Richter, “Polynomial trajectory planning for aggressive quadrotor flight in dense indoor environments”, 2013.

2. Path Planning in Flat Output Space

Path planning with polynomials (CPO) [8]

- Consider n polynomials $P_1(t), ..., P_n(t)$ with $\deg(P_k) = N$
- Integral cost associated with sum of spline derivatives
- Sum cost over all splines
- What happens between the endpoints?

Reconsider the toy example

$$\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}) + \mathbf{g}(\mathbf{x})u, \quad \mathbf{f}(\mathbf{x}) = \begin{bmatrix} -x_1 + x_2 \\ x_1 - x_2 - x_1 x_3 \\ x_1 + x_1 x_2 - 2x_3 \end{bmatrix}, \quad \mathbf{g}(\mathbf{x}) = \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix}$$

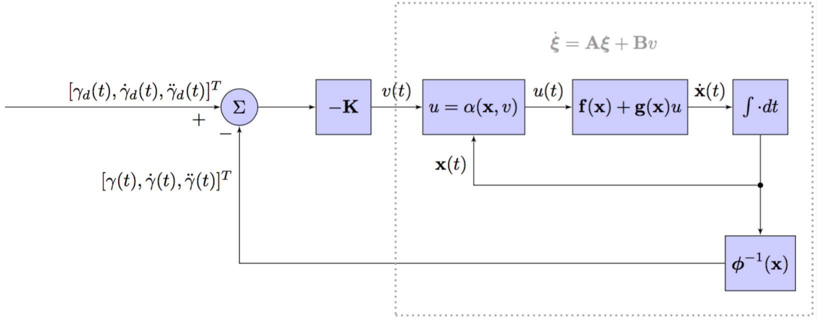
[8] C. Richter, “Polynomial trajectory planning for aggressive quadrotor flight in dense indoor environments”, 2013.

2. Path Planning in Flat Output Space

Path planning with polynomials (CPO) [8]

- Consider n polynomials $P_1(t), ..., P_n(t)$ with $\deg(P_k) = N$
- Integral cost associated with sum of spline derivatives
- Sum cost over all splines
- What happens between the endpoints?

Reconsider the toy example



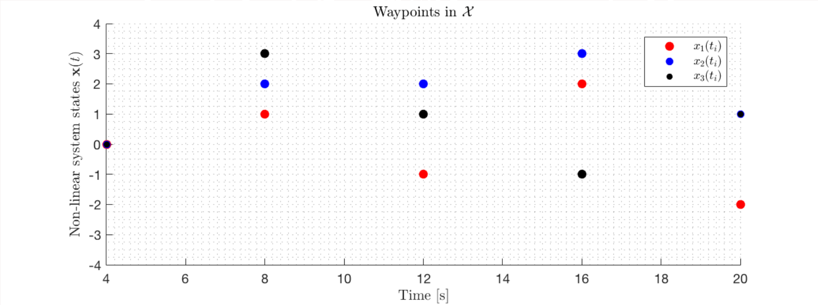
[8] C. Richter, "Polynomial trajectory planning for aggressive quadrotor flight in dense indoor environments", 2013.

2. Path Planning in Flat Output Space

Path planning with polynomials (CPO) [8]

- Consider n polynomials $P_1(t), ..., P_n(t)$ with $\deg(P_k) = N$
- Integral cost associated with sum of spline derivatives
- Sum cost over all splines
- What happens between the endpoints?

Reconsider the toy example



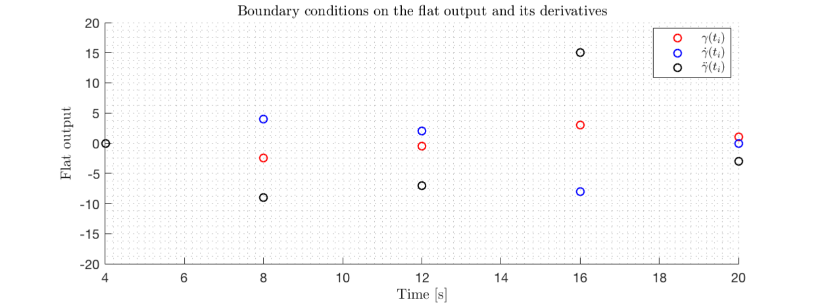
[8] C. Richter, "Polynomial trajectory planning for aggressive quadrotor flight in dense indoor environments", 2013.

2. Path Planning in Flat Output Space

Path planning with polynomials (CPO) [8]

- Consider n polynomials $P_1(t), ..., P_n(t)$ with $\deg(P_k) = N$
- Integral cost associated with sum of spline derivatives
- Sum cost over all splines
- What happens between the endpoints?

Reconsider the toy example

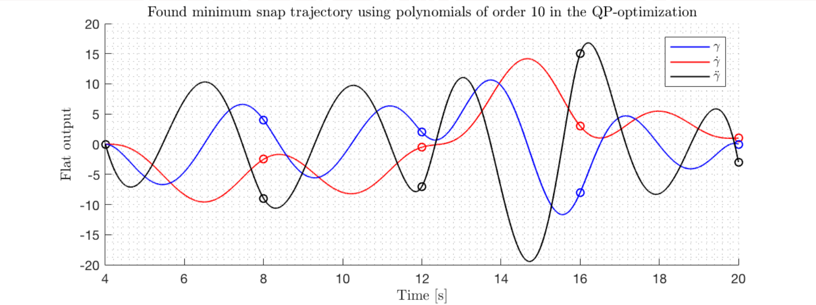


[8] C. Richter, "Polynomial trajectory planning for aggressive quadrotor flight in dense indoor environments", 2013.

Path planning with polynomials (CPO) [8]

- Consider n polynomials $P_1(t), ..., P_n(t)$ with $\deg(P_k) = N$
- Integral cost associated with sum of spline derivatives
- Sum cost over all splines
- What happens between the endpoints?

Reconsider the toy example



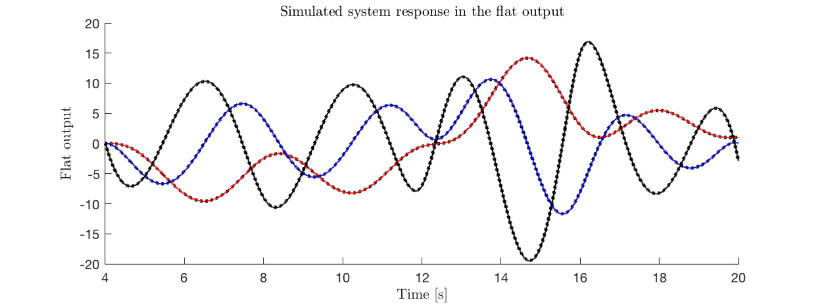
[8] C. Richter, “Polynomial trajectory planning for aggressive quadrotor flight in dense indoor environments”, 2013.

2. Path Planning in Flat Output Space

Path planning with polynomials (CPO) [8]

- Consider n polynomials $P_1(t), ..., P_n(t)$ with $\deg(P_k) = N$
- Integral cost associated with sum of spline derivatives
- Sum cost over all splines
- What happens between the endpoints?

Reconsider the toy example



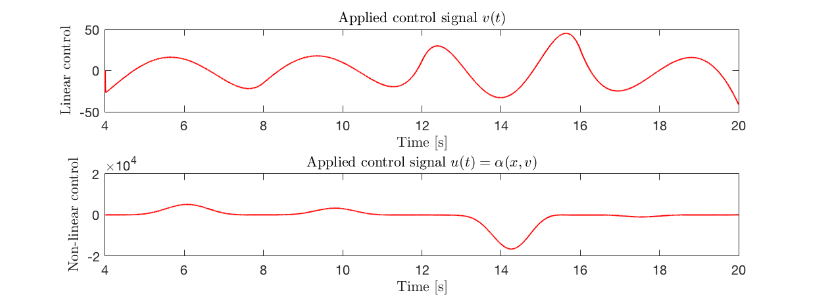
[8] C. Richter, “Polynomial trajectory planning for aggressive quadrotor flight in dense indoor environments”, 2013.

2. Path Planning in Flat Output Space

Path planning with polynomials (CPO) [8]

- Consider n polynomials $P_1(t), \ldots, P_n(t)$ with $\deg(P_k) = N$
- Integral cost associated with sum of spline derivatives
- Sum cost over all splines
- What happens between the endpoints?

Reconsider the toy example



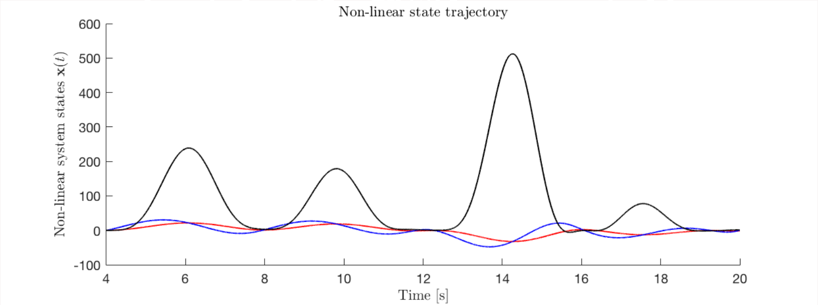
[8] C. Richter, “Polynomial trajectory planning for aggressive quadrotor flight in dense indoor environments”, 2013.

2. Path Planning in Flat Output Space

Path planning with polynomials (CPO) [8]

- Consider n polynomials $P_1(t), ..., P_n(t)$ with $\deg(P_k) = N$
- Integral cost associated with sum of spline derivatives
- Sum cost over all splines
- What happens between the endpoints?

Reconsider the toy example

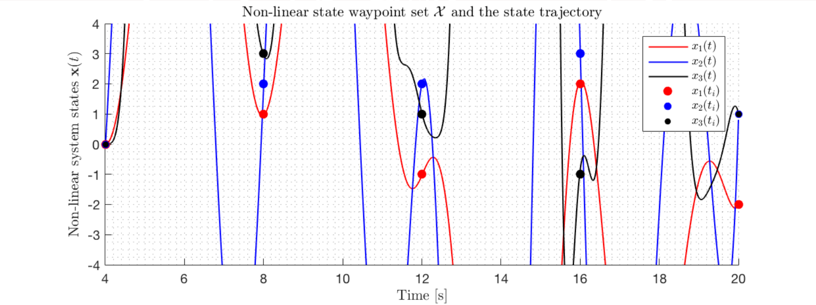


[8] C. Richter, "Polynomial trajectory planning for aggressive quadrotor flight in dense indoor environments", 2013.

Path planning with polynomials (CPO) [8]

- Consider n polynomials $P_1(t), ..., P_n(t)$ with $\deg(P_k) = N$
- Integral cost associated with sum of spline derivatives
- Sum cost over all splines
- What happens between the endpoints?

Reconsider the toy example



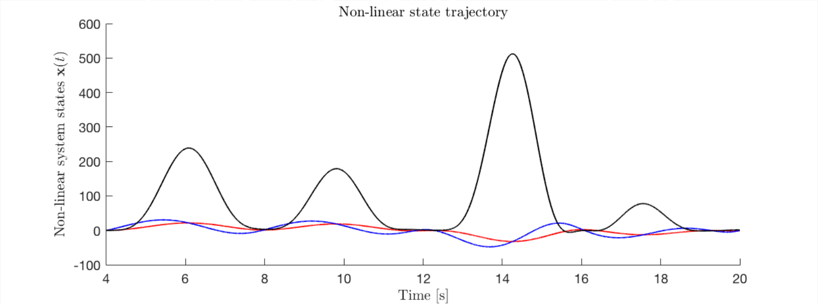
[8] C. Richter, "Polynomial trajectory planning for aggressive quadrotor flight in dense indoor environments", 2013.

2. Path Planning in Flat Output Space

Path planning with polynomials (CPO) [8]

- Consider n polynomials $P_1(t), ..., P_n(t)$ with $\deg(P_k) = N$
- Integral cost associated with sum of spline derivatives
- Sum cost over all splines
- What happens between the endpoints?

Reconsider the toy example



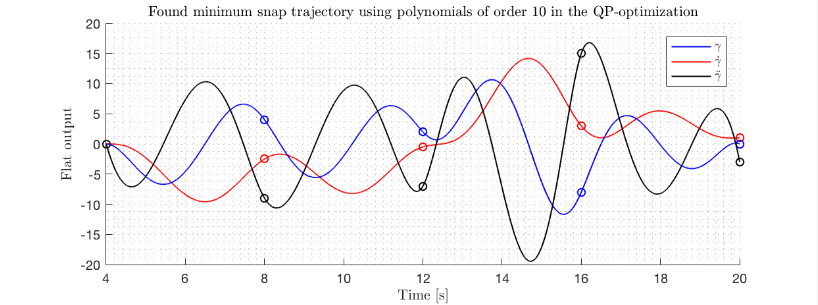
[8] C. Richter, “Polynomial trajectory planning for aggressive quadrotor flight in dense indoor environments”, 2013.

2. Path Planning in Flat Output Space

Path planning with polynomials (CPO) [8]

- Consider n polynomials $P_1(t), ..., P_n(t)$ with $\deg(P_k) = N$
- Integral cost associated with sum of spline derivatives
- Sum cost over all splines
- What happens between the endpoints?

Reconsider the toy example



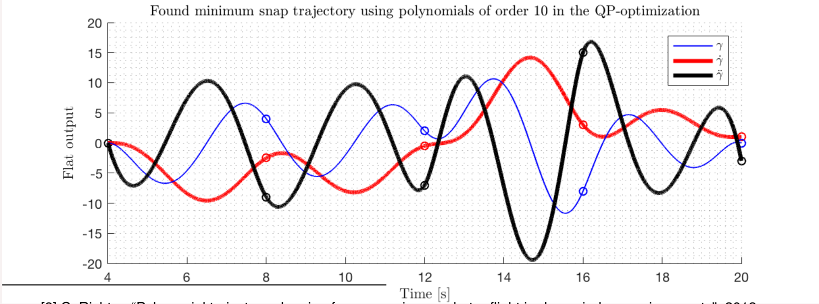
[8] C. Richter, “Polynomial trajectory planning for aggressive quadrotor flight in dense indoor environments”, 2013.

2. Path Planning in Flat Output Space

Path planning with polynomials (CPO) [8]

- Consider n polynomials $P_1(t), ..., P_n(t)$ with $\deg(P_k) = N$
- Integral cost associated with sum of spline derivatives
- Sum cost over all splines
- What happens between the endpoints?

Reconsider the toy example

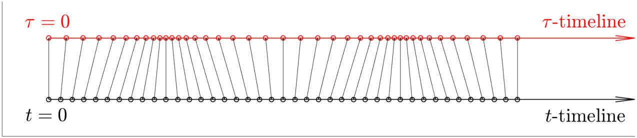


[8] C. Richter, “Polynomial trajectory planning for aggressive quadrotor flight in dense indoor environments”, 2013.

3. Time-warping transformation

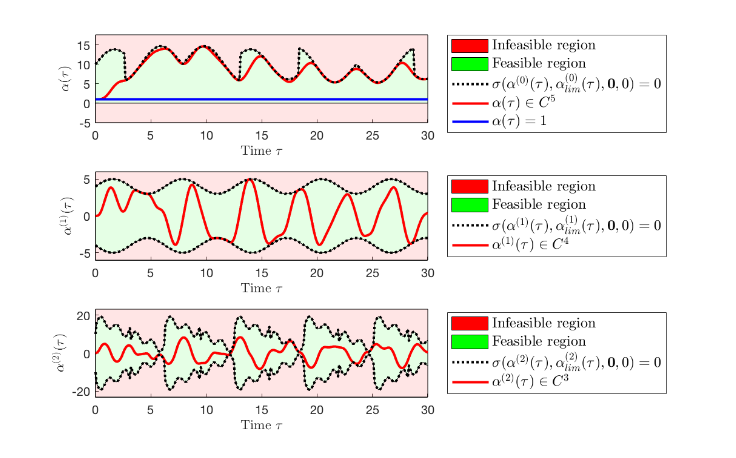
Remark (Change the rate of time)

Let a trajectory $\gamma(\tau)$ be generated in terms of the time unit τ, relating to a second time unit t on which the system evolves, such that $\alpha(\tau)d\tau = dt$ for some $\alpha(\tau) > 0$.



3. Time-warping transformation

Optimization program to maximize $\alpha(\tau)$, enforcing smoothness [16]



[16] M. Greiff, “A Time-Warping Transformation for Time-Optimal Movement”, 2018.

3. Formulating a QP

Given the maps $\{\mathbf{h}, \phi, \boldsymbol{\beta}\}$ for a differentially flat system, and

- A feasible flat trajectory $\boldsymbol{\gamma}(\tau) \in C^M$
- A smoothness constraint on $\alpha(\tau) \in C^{M-1}$

minimize the time $t_f - t_0$ taken to traverse $\boldsymbol{\gamma}(t)$ by maximising $\alpha(\tau)$

3. Formulating a QP

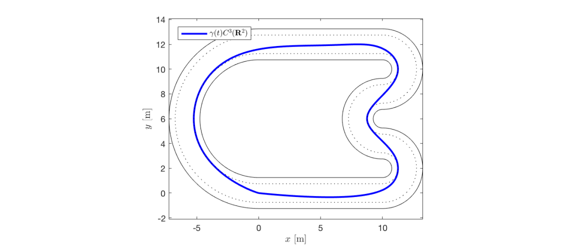
Given the maps $\{\mathbf{h}, \phi, \boldsymbol{\beta}\}$ for a differentially flat system, and

- A feasible flat trajectory $\boldsymbol{\gamma}(\tau) \in C^M$
- A smoothness constraint on $\alpha(\tau) \in C^{M-1}$

minimize the time $t_f - t_0$ taken to traverse $\boldsymbol{\gamma}(t)$ by maximising $\alpha(\tau)$

Example: Consider a UGV, path in $\mathcal{F}$ generated by CPO, velocity constraints

$$-\begin{bmatrix} 5 + 2\sin(\tau) \\ 5 + 2\sin(\tau) \end{bmatrix} \leq \boldsymbol{\gamma}_t^{(1)}(\tau) \leq \begin{bmatrix} 5 + 2\sin(\tau) \\ 5 + 2\sin(\tau) \end{bmatrix},$$



Simulation example D - UGV with and without time-warping

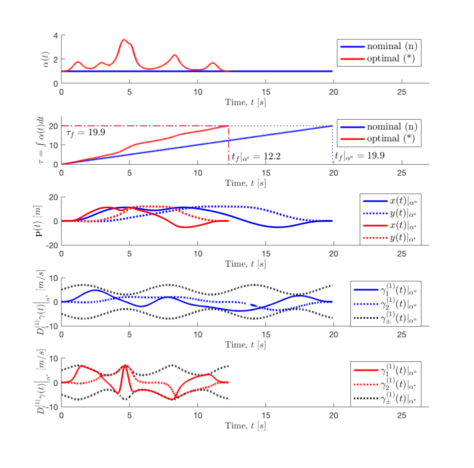


Figure 1: A nominal $\alpha^n(\tau) = 1 \ \ \forall\tau$ and optimal $\alpha(\tau) \in C^3(\mathbb{R})$ subject to sinusoidal constraints.

3. Simulation Example - UGV

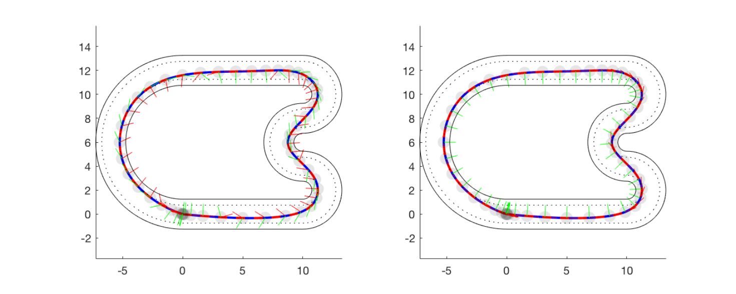


Figure 1: The nominal trajectory $\boldsymbol{\gamma}^n(\tau)$ (blue) and simulated motion along the optimised flat output trajectory $\boldsymbol{\gamma}^*(t)$ with the $SE(2)$ configured UGV (left) and the non-holonomically constrained UGV (right).

Example summary

- Optimal warping found as the problem is convex
- Same flat output trajectory, very different state-trajectories
- Original dynamical system $\dim(\mathbf{x}) = \{5, 6\}$, $\dim(\mathbf{u}) = 2$
- System in the warping MPC formulation $\dim(\mathbf{x}_\alpha) = 3$, $\dim(u_\alpha) = 1$

3. Formulating a SQP

Example: UAV dynamics, starting and finishing in $\boldsymbol{\gamma}(0) = \mathbf{0}$, performing a looping manoeuvre defined by the path

$$\boldsymbol{\gamma}(\tau) = -[\sin(\pi\tau/4), \sin(\pi\tau/2), \cos(\pi\tau/2), -\pi\tau/4]^T.$$

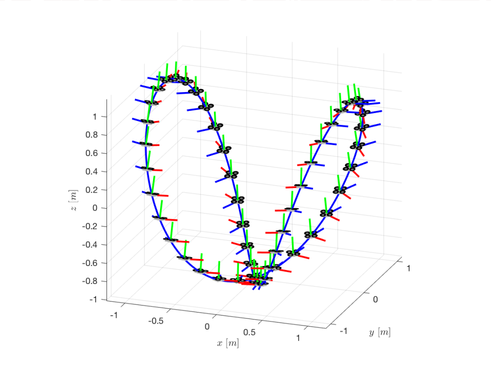


Figure 2: A looping manoeuvre with $\gamma(\tau) \in C^\infty(\mathbb{R}^4)$.

3. Formulating a SQP

Example: UAV dynamics, starting and finishing in $\boldsymbol{\gamma}(0) = \mathbf{0}$, performing a looping manoeuvre defined by the path

$$\boldsymbol{\gamma}(\tau) = -[\sin(\pi\tau/4), \sin(\pi\tau/2), \cos(\pi\tau/2), -\pi\tau/4]^T.$$

Two main saturating constraints

- Velocities:

$$-\begin{bmatrix}5\\5\\5\end{bmatrix} \leq \begin{bmatrix}v^x(t)\\v^y(t)\\v^z(t)\end{bmatrix} \leq \begin{bmatrix}5\\5\\5\end{bmatrix} \qquad [m/s] \qquad \text{(Linear in } \alpha\text{)}$$

- Rotor-speeds:

$$\begin{bmatrix}500\\500\\500\\500\end{bmatrix} \leq \begin{bmatrix}\Omega_1(t)\\\Omega_2(t)\\\Omega_3(t)\\\Omega_4(t)\end{bmatrix} \leq \begin{bmatrix}2400\\2400\\2400\\2400\end{bmatrix} [rad/s] \qquad \text{(Highly nonlinear in } \alpha\text{)}$$

Simulation example E - Looping UAV with and without time-warping

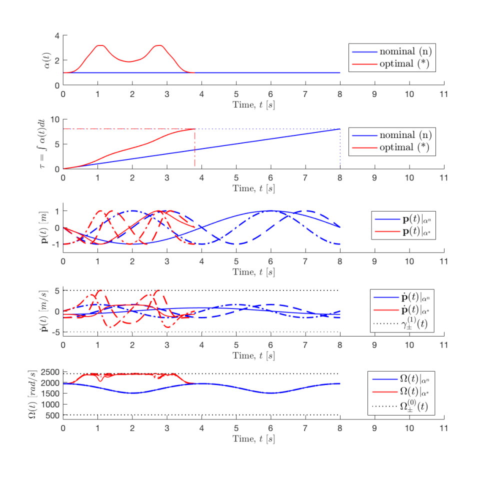


Figure 2: The nominal- (blue) and computed locally time-optimal trajectories (red) for the $SE(3)$-configured UAV during the looping manoeuvre with actuator constraints.

3. Simulation Example - UAV

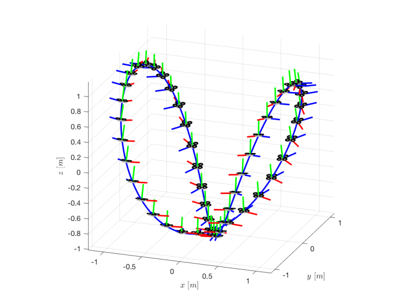


Figure 2: Nominal and locally time-optimal solutions for a looping manoeuvre given actuator constraints.

Example summary

- Locally optimal warping found (problem is now non-convex)
- The posed constraints are close to saturated at almost all times
- Original system $\dim(\mathbf{x}) = \{12, 13\}$, $\dim(\mathbf{u}) = 4$
- System in the SQP formulation $\dim(\mathbf{x}_\alpha) = 5$, $\dim(u_\alpha) = 1$

3. Simulation Example - UAV

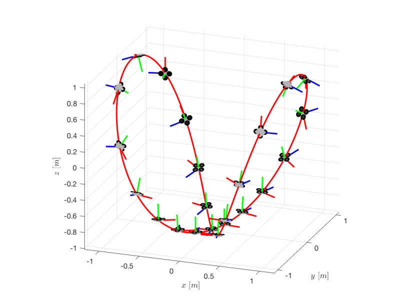


Figure 2: Nominal and locally time-optimal solutions for a looping manoeuvre given actuator constraints.

Example summary

- Locally optimal warping found (problem is now non-convex)
- The posed constraints are close to saturated at almost all times
- Original system $\dim(\mathbf{x}) = \{12, 13\}$, $\dim(\mathbf{u}) = 4$
- System in the SQP formulation $\dim(\mathbf{x}_\alpha) = 5$, $\dim(u_\alpha) = 1$

Summary

Two-step approach

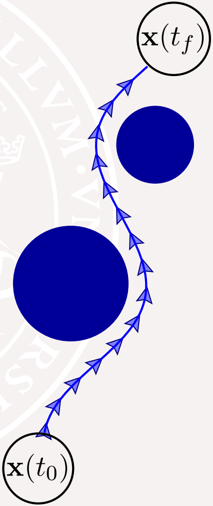
- Path planning of the flat outputs
- Augment higher order derivatives through $\alpha(\tau)$

Contents

- Differential flatness
- Polynomial path planning
- Examples with QP and SQP time-warping

Examples

- Toy example
- UGV with nonholonomic constraints
- UGV with no constraints
- UAV with actuator constraints



Summary

Two-step approach

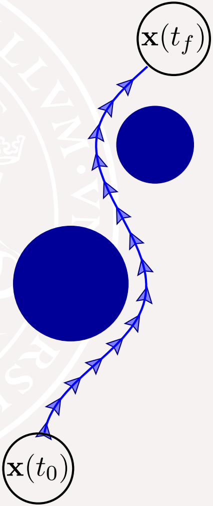
- Path planning of the flat outputs
- Augment higher order derivatives through $\alpha(\tau)$

Contents

- Differential flatness
- Polynomial path planning
- Examples with QP and SQP time-warping

Examples

- Toy example
- UGV with nonholonomic constraints
- UGV with no constraints
- UAV with actuator constraints



Summary

Two-step approach

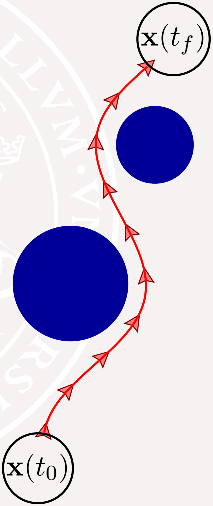
- Path planning of the flat outputs
- Augment higher order derivatives through $\alpha(\tau)$

Contents

- Differential flatness
- Polynomial path planning
- Examples with QP and SQP time-warping

Examples

- Toy example
- UGV with nonholonomic constraints
- UGV with no constraints
- UAV with actuator constraints



References I

[1] M. Greiff and K. Berntorp, "Optimal measurement projections with adaptive mixture kalman filtering for gnss positioning," in *2020 American Control Conference (ACC)*, 2020, pp. 4435–4441.

[2] M. Greiff, A. Robertsson, and K. Berntorp, "Mse-optimal measurement dimension reduction in gaussian filtering," in *2020 IEEE Conference on Control Technology and Applications (CCTA)*, 2020, pp. 126–133.

[3] M. Greiff, K. Berntorp, and A. Robertsson, "Exploiting linear substructure in lrkfs (extended)," 2020.

[4] E. Lefeber, M. Greiff, and A. Robertsson, *Filtered Output Feedback Tracking Control of a Quadrotor UAV*, ser. DC Reports. Eindhoven University of Technology, Dynamics and Control Group, Department of Mechanical Engineering, Eindhoven, The Netherlands, May 2020, no. DC 2020.053.

References II

[5] Z. S. M. Greiff and A. Robertsson, "Attitude control on SU(2): Stability, robustness, and similarities," 2020, to appear in L-CSS.

[6] T. Glück, A. Eder, and A. Kugi, "Swing-up control of a triple pendulum on a cart with experimental validation," *Automatica*, vol. 49, no. 3, pp. 801–808, 2013.

[7] S. Boyd, "Ee364b: Lecture slides - sequential convex programming," 2012, https://web.stanford.edu/class/ee364b/lectures/seq_slides.pdf.

[8] C. Richter, A. Bry, and N. Roy, "Polynomial trajectory planning for aggressive quadrotor flight in dense indoor environments," in *Proceedings of the International Symposium on Robotics Research (ISRR)*, 2013.

References III

[9] M. Greiff, “A time-warping transformation for time-optimal movement in differentially flat systems,” in *2018 Annual American Control Conference (ACC)*. IEEE, 2018, pp. 6723–6730.

[10] M. Fliess, J. Lévine, P. Martin, and P. Rouchon, “Flatness and defect of non-linear systems: introductory theory and examples,” *International journal of control*, vol. 61, no. 6, pp. 1327–1361, 1995.

[11] G. G. Rigatos, *Nonlinear control and filtering using differential flatness approaches: applications to electromechanical systems*. Springer, 2015, vol. 25.

[12] H. K. Khalil and J. W. Grizzle, *Nonlinear systems*. Prentice hall Upper Saddle River, NJ, 2002, vol. 3.

[13] D. W. Mellinger, “Trajectory generation and control for quadrotors,” 2012.

References IV

[14] M. Greiff, "Modelling and control of the crazyflie quadrotor for aggressive and autonomous flight by optical flow driven state estimation," 2017.

[15] R. M. Murray, M. Rathinam, and W. Sluis, "Differential flatness of mechanical control systems: A catalog of prototype systems," in *ASME international mechanical engineering congress and exposition*. Citeseer, 1995.

[16] M. Greiff, "A time-warping transformation for time-optimal movement in differentially flat systems," in *American Control Conference (ACC), 2018*. IEEE, 2018.