

# EXPLORATION DE DONNÉES POUR L'OPTIMISATION DE TRAJECTOIRES AÉRIENNES

Cédric Rommel

Directeurs de thèse: Frédéric Bonnans, Pierre Martinon  
Encadrant Safety Line: Baptiste Gregorutti

Soutenance de thèse, 26 octobre 2018



# CONTEXT

# MOTIVATION

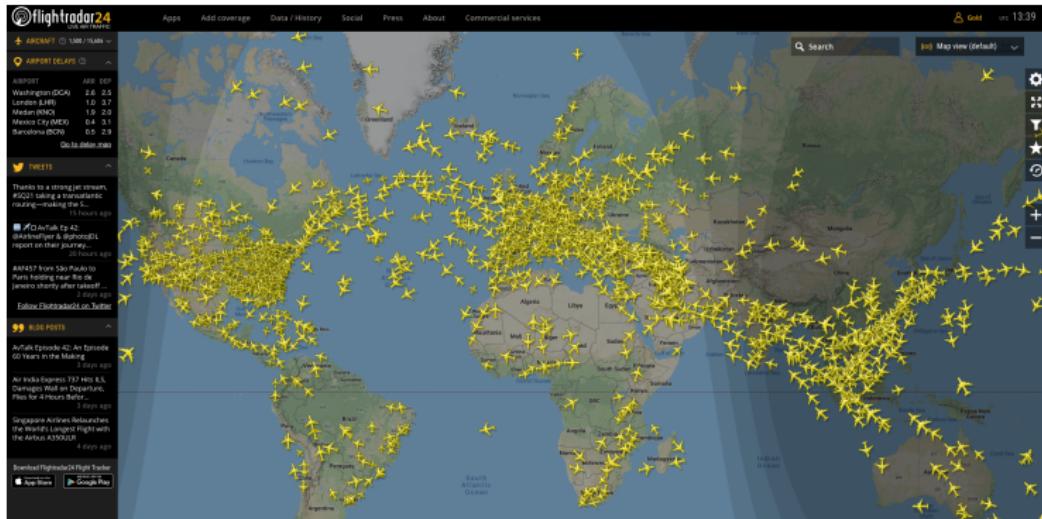


FIGURE: World air traffic - source: [www.flightradar24.com](http://www.flightradar24.com)

# MOTIVATION

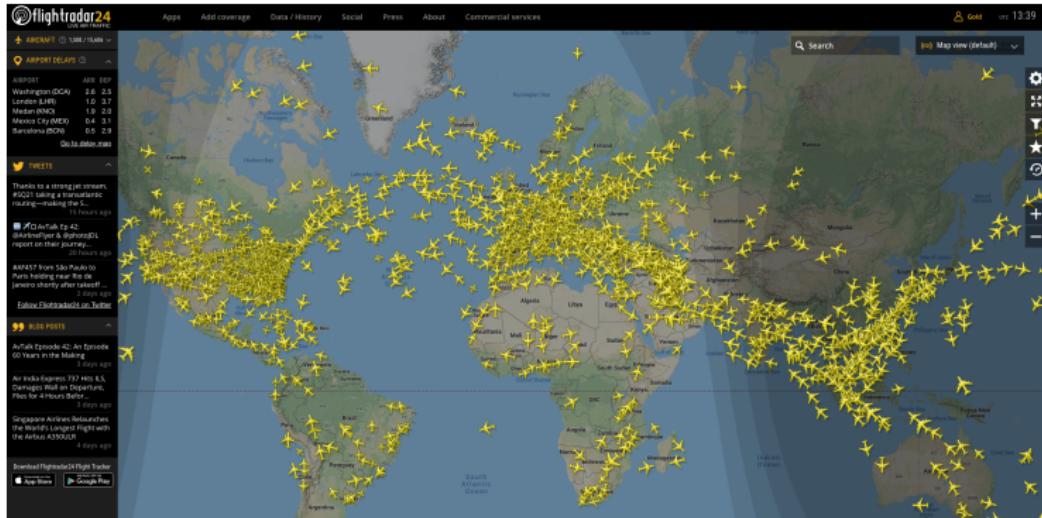


FIGURE: World air traffic - source: [www.flightradar24.com](http://www.flightradar24.com)

- 20 000 airplanes — 80 000 flights per day,

# MOTIVATION

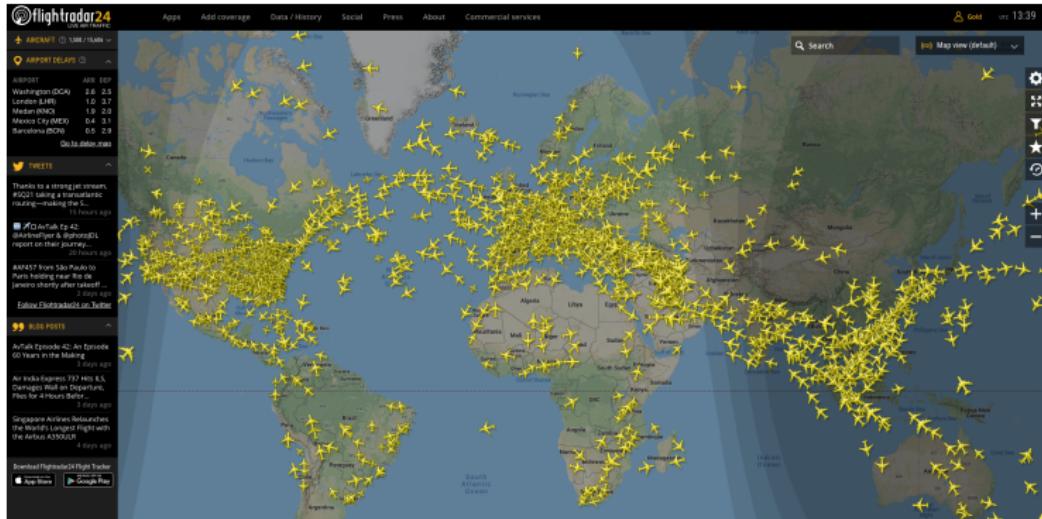
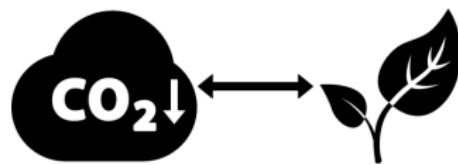


FIGURE: World air traffic - source: [www.flightradar24.com](http://www.flightradar24.com)

- 20 000 airplanes — 80 000 flights per day,
- Should double until 2033,

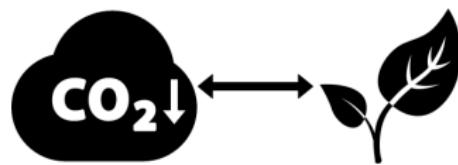
# MOTIVATION

- Most polluting means of transportation,



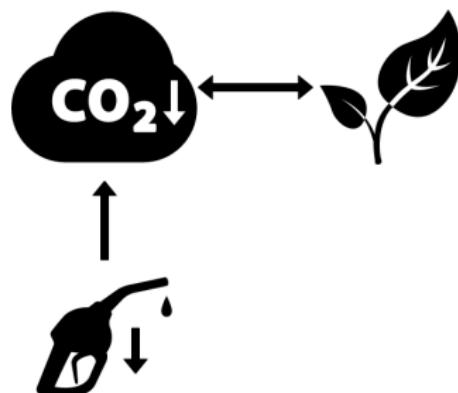
# MOTIVATION

- Most polluting means of transportation,
- Responsible for 3% of  $CO_2$  emissions,



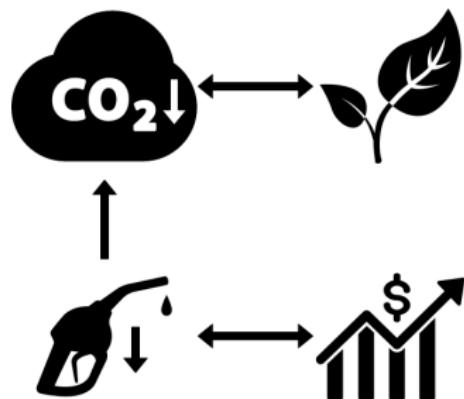
# MOTIVATION

- Most polluting means of transportation,
- Responsible for 3% of  $CO_2$  emissions,



# MOTIVATION

- Most polluting means of transportation,
- Responsible for 3% of  $CO_2$  emissions,
- Fuel  $\simeq$  30% of an airline operational cost,



# MOTIVATION

How to tackle this problem ?

# MOTIVATION

How to tackle this problem ?

- 1 New hardware ?

# MOTIVATION

How to tackle this problem ?

- 1 New hardware ?
- 2 **Better use of existing fleet,**

# MOTIVATION

How to tackle this problem ?

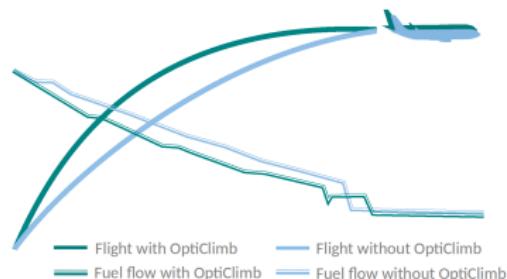
- 1 New hardware ?
- 2 **Better use of existing fleet,**
  - Climb is the most consuming flight phase...

# MOTIVATION

How to tackle this problem ?

- 1 New hardware ?
- 2 Better use of existing fleet,

- Climb is the most consuming flight phase...
- Mostly rectilinear trajectories at full thrust,



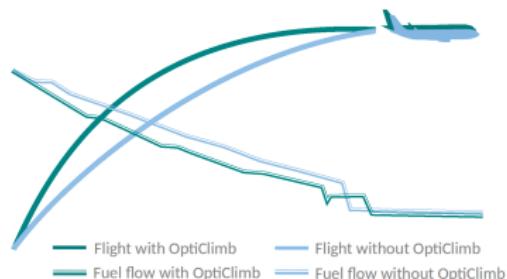
# MOTIVATION

How to tackle this problem ?

1 New hardware ?

2 Better use of existing fleet,

- Climb is the most consuming flight phase...
- Mostly rectilinear trajectories at full thrust,
- Thousands of variables recorded every second,

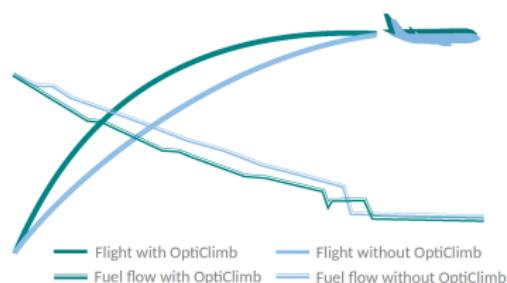


# MOTIVATION

How to tackle this problem ?

- 1 New hardware ?
- 2 Better use of existing fleet,

- Climb is the most consuming flight phase...
- Mostly rectilinear trajectories at full thrust,
- Thousands of variables recorded every second,



# OPTICLIMB



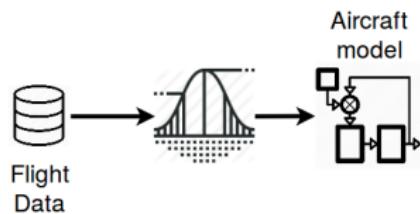
Flight  
Data

**Time**



*Many days before flight...*

# OPTICLIMB

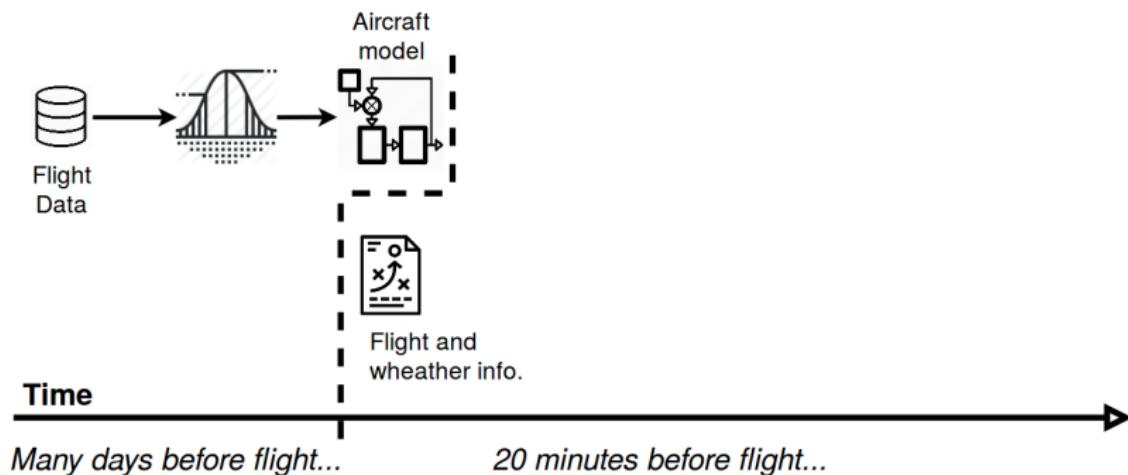


**Time**

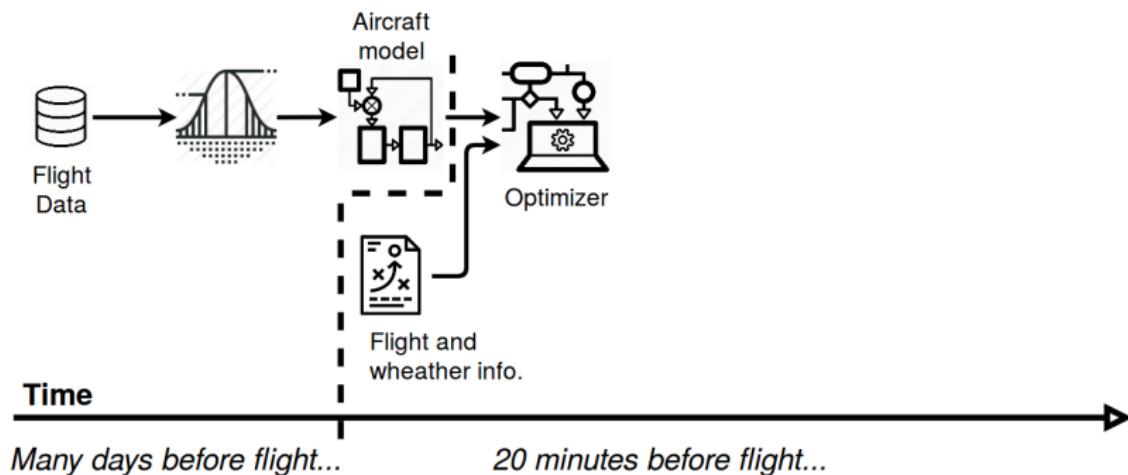


*Many days before flight...*

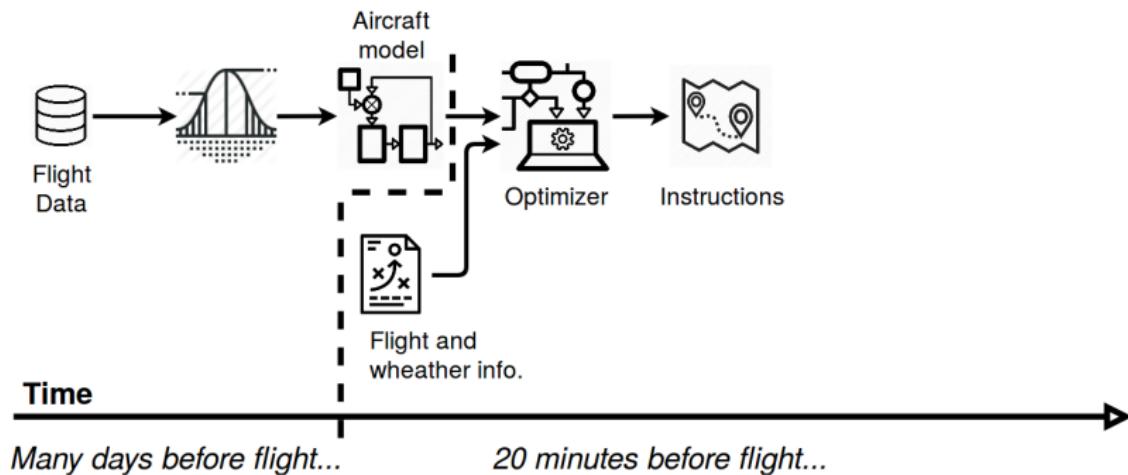
# OPTICLIMB



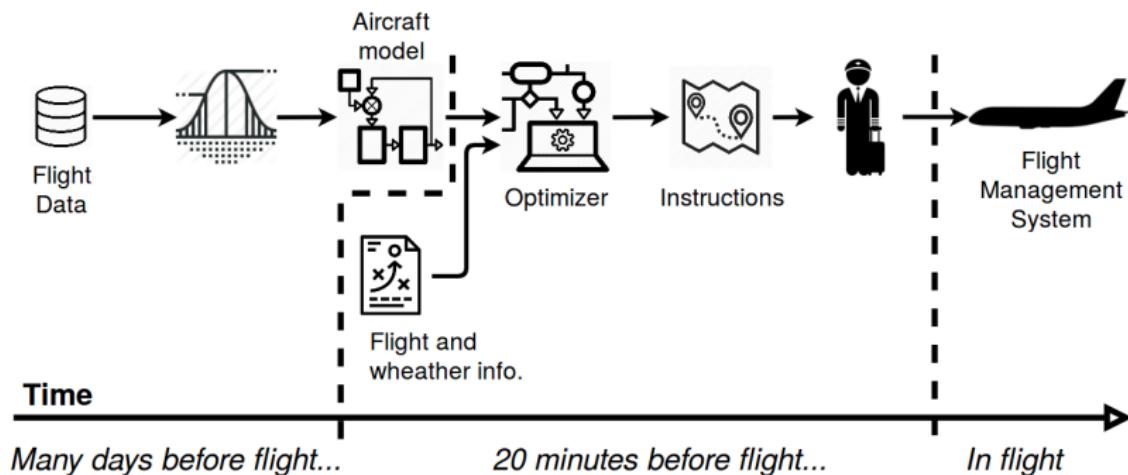
# OPTICLIMB



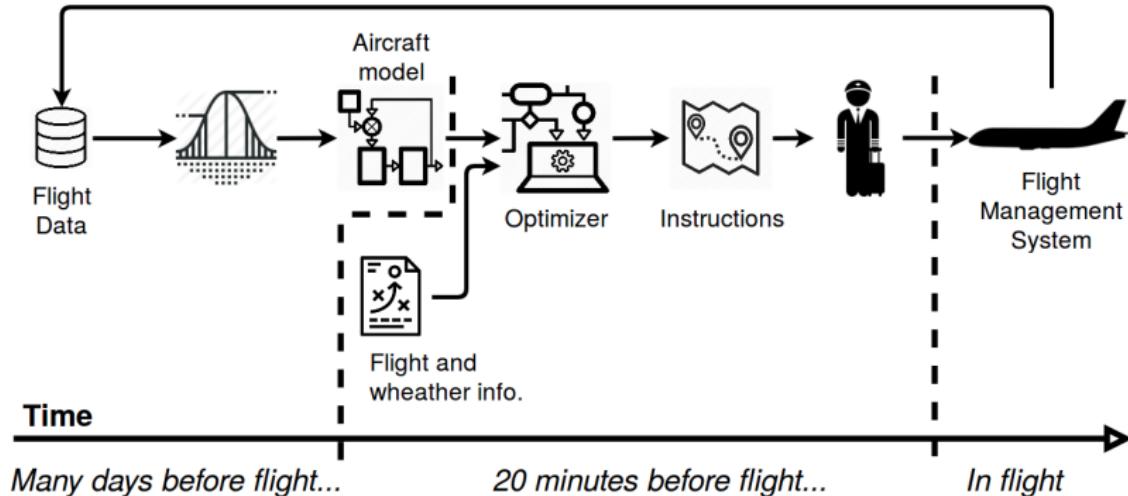
# OPTICLIMB



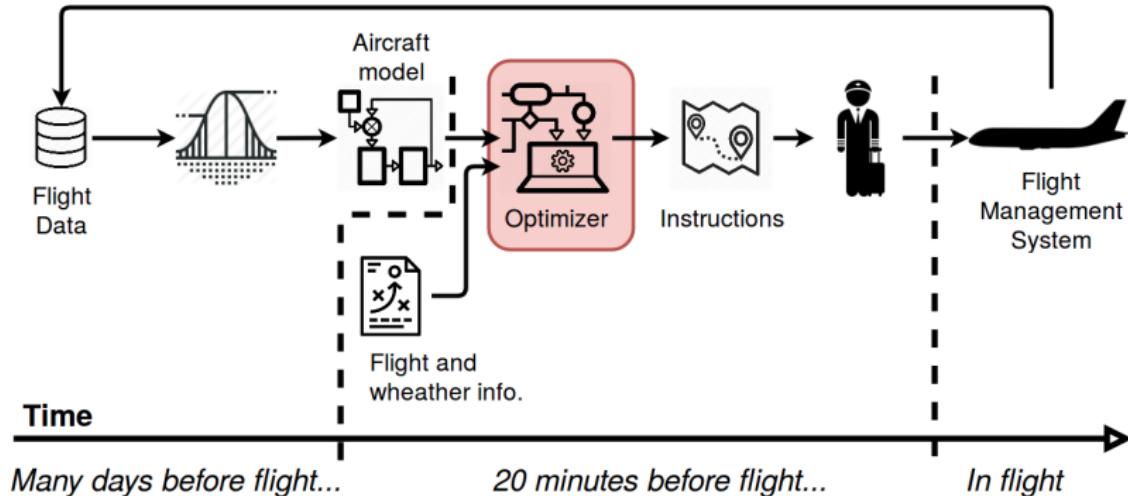
# OPTICLIMB



# OPTICLIMB



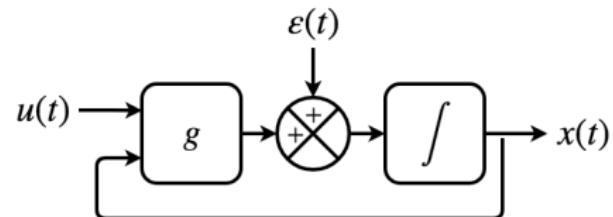
# OPTICLIMB



# TRAJECTORY OPTIMIZATION

Dynamics:

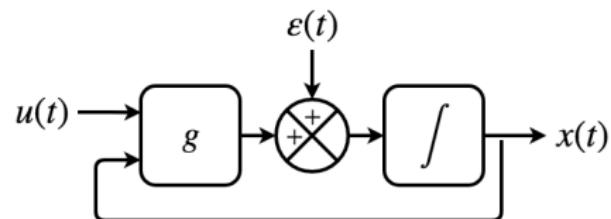
$$\dot{x}(t) = g(\mathbf{u}(t), \mathbf{x}(t)) + \varepsilon(t)$$



# TRAJECTORY OPTIMIZATION

Dynamics:

$$\dot{x}(t) = g(\mathbf{u}(t), \mathbf{x}(t)) + \varepsilon(t)$$

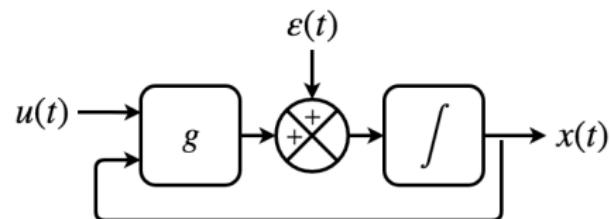


Optimization objective:  $\int_0^{t_f} C(\mathbf{u}(t), \mathbf{x}(t)) dt$

# TRAJECTORY OPTIMIZATION

Dynamics:

$$\dot{x}(t) = g(\mathbf{u}(t), \mathbf{x}(t)) + \varepsilon(t)$$

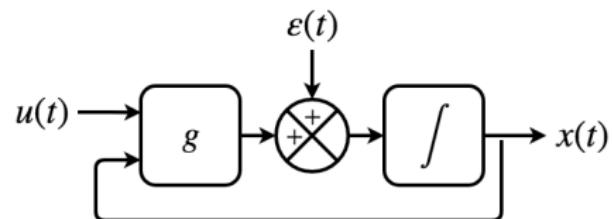


Optimization objective:  $\int_0^{t_f} C(\mathbf{u}(t), \mathbf{x}(t)) dt \Leftarrow \text{min}$ ,

# TRAJECTORY OPTIMIZATION

Dynamics:

$$\dot{x}(t) = g(\mathbf{u}(t), \mathbf{x}(t)) + \varepsilon(t)$$

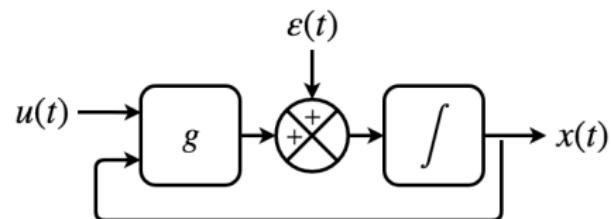


Optimization objective:  $\int_0^{t_f} C(\mathbf{u}(t), \mathbf{x}(t)) dt \Leftarrow \text{↗, } \text{📍, } \text{⌚️}$

# TRAJECTORY OPTIMIZATION

Dynamics:

$$\dot{x}(t) = g(\mathbf{u}(t), \mathbf{x}(t)) + \varepsilon(t)$$



Optimization objective:  $\int_0^{t_f} C(\mathbf{u}(t), \mathbf{x}(t)) dt \Leftarrow \text{fuel}, \text{path}, \text{time}$

Flight constraints:

$$\begin{cases} \Phi(\mathbf{x}(0), \mathbf{x}(t_f)) \in K_\Phi \\ \mathbf{u}(t) \in U_{ad}, \quad \mathbf{x}(t) \in X_{ad}, \\ c(\mathbf{u}(t), \mathbf{x}(t)) \leq 0, \end{cases}$$

Initial and final conditions  
Flight domain  
Operational path constraints

# TRAJECTORY OPTIMIZATION

## OPTIMAL CONTROL PROBLEM

$$\begin{aligned} & \min_{(\mathbf{x}, \mathbf{u}) \in \mathbb{X} \times \mathbb{U}} \int_0^{t_f} C(\mathbf{u}(t), \mathbf{x}(t)) dt, \\ \text{s.t. } & \left\{ \begin{array}{ll} \dot{\mathbf{x}}(t) = g(\mathbf{u}(t), \mathbf{x}(t)) + \varepsilon(t), & \text{a.e. } t \in [0, t_f], \\ \Phi(\mathbf{x}(0), \mathbf{x}(t_f)) \in K_\Phi, & \\ \mathbf{u}(t) \in U_{ad}, \quad \mathbf{x}(t) \in X_{ad}, & \text{a.e. } t \in [0, t_f], \\ c(\mathbf{u}(t), \mathbf{x}(t)) \leq 0, & \text{a.e. } t \in [0, t_f]. \end{array} \right. \end{aligned} \quad (\text{OCP})$$

# TRAJECTORY OPTIMIZATION

## OPTIMAL CONTROL PROBLEM

$$\begin{aligned} & \min_{(\mathbf{x}, \mathbf{u}) \in \mathbb{X} \times \mathbb{U}} \int_0^{t_f} C(\mathbf{u}(t), \mathbf{x}(t)) dt, \\ \text{s.t. } & \left\{ \begin{array}{ll} \dot{\mathbf{x}}(t) = \mathbf{g}(\mathbf{u}(t), \mathbf{x}(t)) + \varepsilon(t), & \text{a.e. } t \in [0, t_f], \\ \Phi(\mathbf{x}(0), \mathbf{x}(t_f)) \in K_\Phi, & \\ \mathbf{u}(t) \in U_{ad}, \quad \mathbf{x}(t) \in X_{ad}, & \text{a.e. } t \in [0, t_f], \\ c(\mathbf{u}(t), \mathbf{x}(t)) \leq 0, & \text{a.e. } t \in [0, t_f]. \end{array} \right. \end{aligned} \quad (\text{OCP})$$

# TRAJECTORY OPTIMIZATION

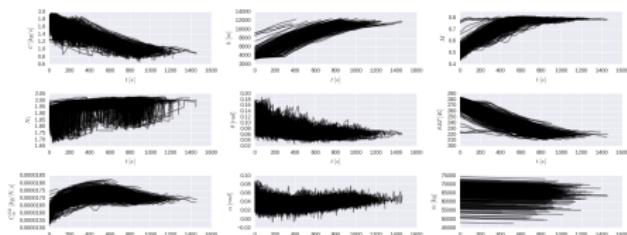
## OPTIMAL CONTROL PROBLEM

$$\begin{aligned} & \min_{(\mathbf{x}, \mathbf{u}) \in \mathbb{X} \times \mathbb{U}} \int_0^{t_f} C(\mathbf{u}(t), \mathbf{x}(t)) dt, \\ \text{s.t. } & \left\{ \begin{array}{ll} \dot{\mathbf{x}}(t) = \mathbf{g}(\mathbf{u}(t), \mathbf{x}(t)) + \varepsilon(t), & \text{a.e. } t \in [0, t_f], \\ \Phi(\mathbf{x}(0), \mathbf{x}(t_f)) \in K_\Phi, & \\ \mathbf{u}(t) \in U_{ad}, \quad \mathbf{x}(t) \in X_{ad}, & \text{a.e. } t \in [0, t_f], \\ c(\mathbf{u}(t), \mathbf{x}(t)) \leq 0, & \text{a.e. } t \in [0, t_f]. \end{array} \right. \end{aligned} \quad (\text{OCP})$$

## SYSTEM IDENTIFICATION



Black box



QAR data

# TRAJECTORY OPTIMIZATION

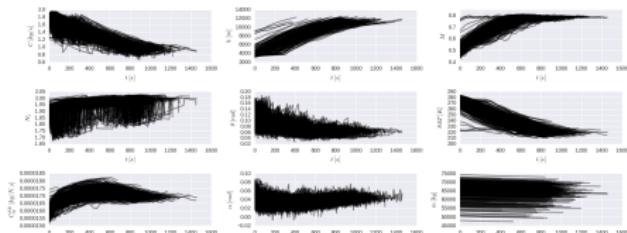
## APPROXIMATE OPTIMAL CONTROL PROBLEM

$$\begin{aligned} & \min_{(\mathbf{x}, \mathbf{u}) \in \mathbb{X} \times \mathbb{U}} \int_0^{t_f} C(\mathbf{u}(t), \mathbf{x}(t)) dt, \\ \text{s.t. } & \left\{ \begin{array}{ll} \dot{\mathbf{x}}(t) = \hat{g}(\mathbf{u}(t), \mathbf{x}(t)), & \text{a.e. } t \in [0, t_f], \\ \Phi(\mathbf{x}(0), \mathbf{x}(t_f)) \in K_\Phi, & \\ \mathbf{u}(t) \in U_{ad}, \quad \mathbf{x}(t) \in X_{ad}, & \text{a.e. } t \in [0, t_f], \\ c(\mathbf{u}(t), \mathbf{x}(t)) \leq 0, & \text{a.e. } t \in [0, t_f]. \end{array} \right. \end{aligned} \quad (\text{A OCP})$$

## SYSTEM IDENTIFICATION

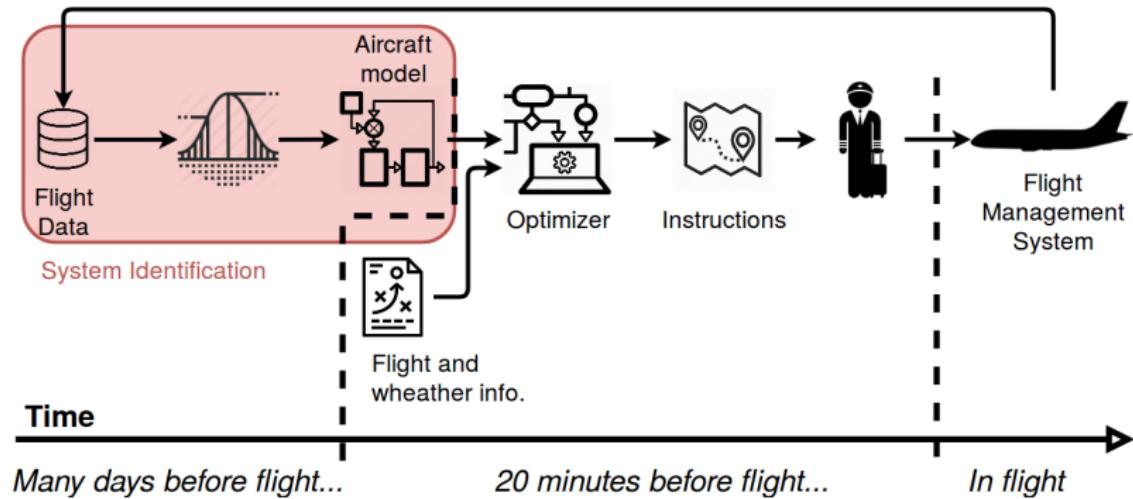


Black box



QAR data

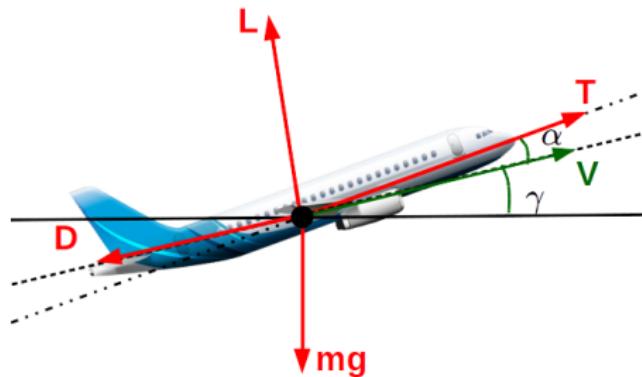
# SYSTEM IDENTIFICATION



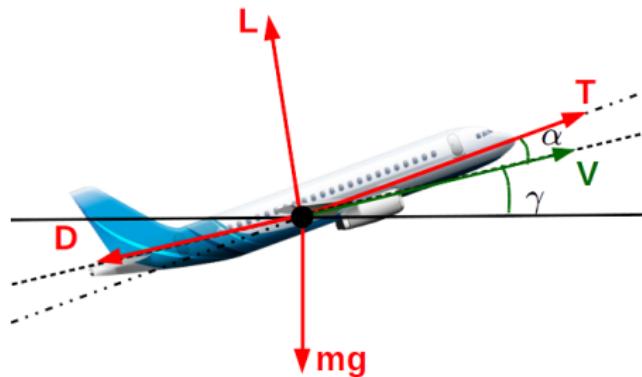
- 1** Context - *Chapter 1*
- 2** System Identification - *Chapter 4*
- 3** Trajectory Acceptability - *Chapters 5 and 6*

# SYSTEM IDENTIFICATION

# FLIGHT DYNAMICS

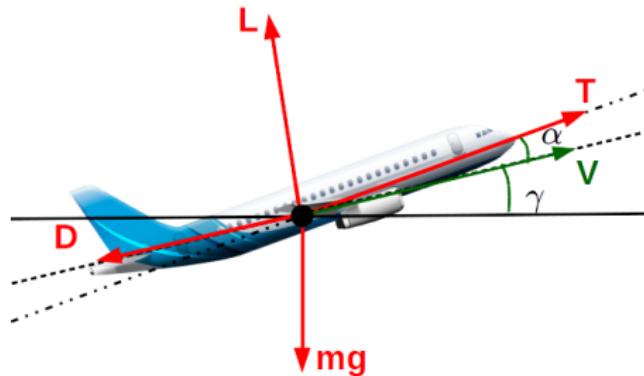


# FLIGHT DYNAMICS



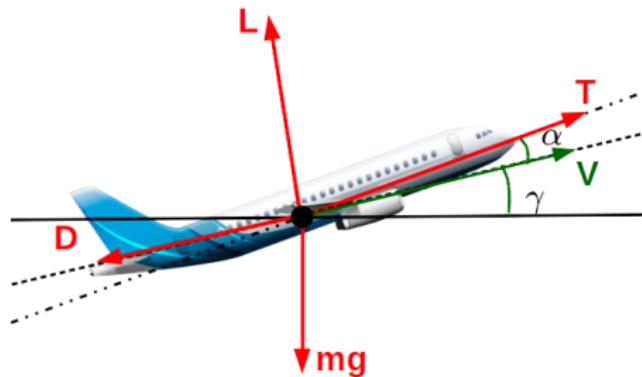
$$\left\{ \begin{array}{l} \dot{h} = V \sin \gamma \\ \dot{V} = \frac{T \cos \alpha - D - mg \sin \gamma}{m} \\ \dot{\gamma} = \frac{T \sin \alpha + L - mg \cos \gamma}{mV} \\ \dot{m} = -\frac{T}{I_{sp}} \end{array} \right.$$

# FLIGHT DYNAMICS



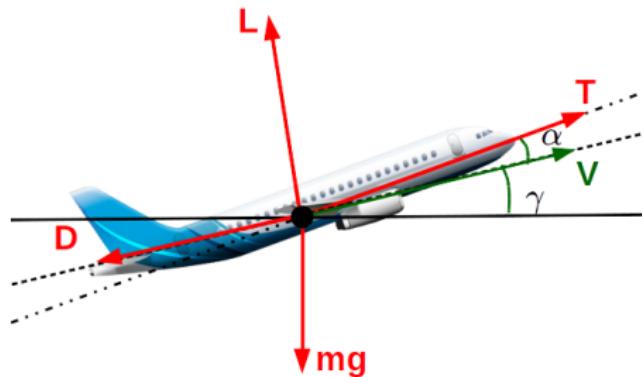
$$\left\{ \begin{array}{l} \dot{h} = V \sin \gamma + \dot{W}_z \\ \dot{V} = \frac{T \cos \alpha - D - mg \sin \gamma - m \dot{W}_{xv}}{m} \\ \dot{\gamma} = \frac{(T \sin \alpha + L) \cos \mu - mg \cos \gamma - m \dot{W}_{zv}}{mV} \\ \dot{m} = -\frac{T}{I_{sp}} \end{array} \right.$$

# FLIGHT DYNAMICS



$$\left\{ \begin{array}{l} \dot{h} = V \sin \gamma \\ \dot{V} = \frac{T \cos \alpha - D - mg \sin \gamma}{m} \\ \dot{\gamma} = \frac{T \sin \alpha + L - mg \cos \gamma}{mV} \\ \dot{m} = -\frac{T}{I_{sp}} \end{array} \right.$$

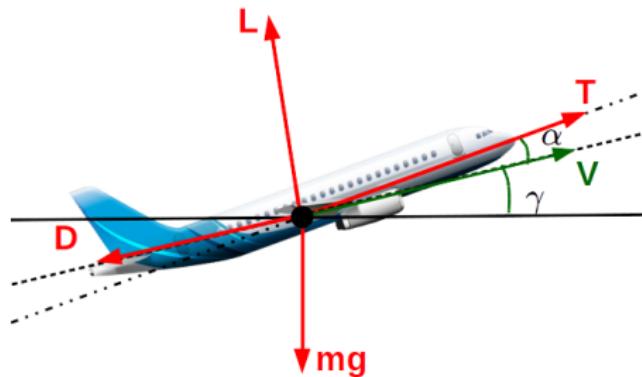
# FLIGHT DYNAMICS



States:  $x = (h, V, \gamma, m)$

$$\left\{ \begin{array}{l} \dot{h} = V \sin \gamma \\ \dot{V} = \frac{T \cos \alpha - D - mg \sin \gamma}{m} \\ \dot{\gamma} = \frac{T \sin \alpha + L - mg \cos \gamma}{mV} \\ \dot{m} = -\frac{T}{I_{sp}} \end{array} \right.$$

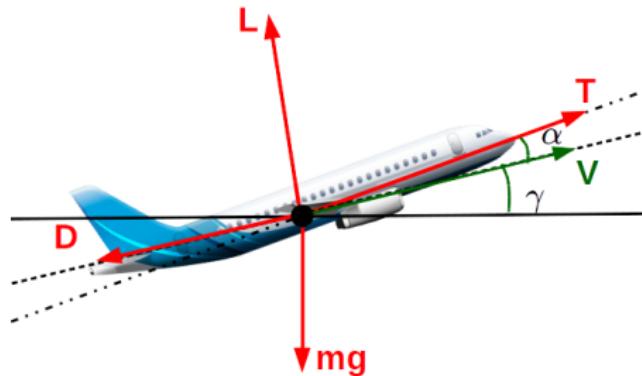
# FLIGHT DYNAMICS



**States:**  $x = (h, V, \gamma, m)$   
**Controls:**  $u = (\alpha, N_1)$

$$\begin{cases} \dot{h} = V \sin \gamma \\ \dot{V} = \frac{T \cos \alpha - D - mg \sin \gamma}{m} \\ \dot{\gamma} = \frac{T \sin \alpha + L - mg \cos \gamma}{mV} \\ \dot{m} = -\frac{T}{I_{sp}} \end{cases}$$

# FLIGHT DYNAMICS



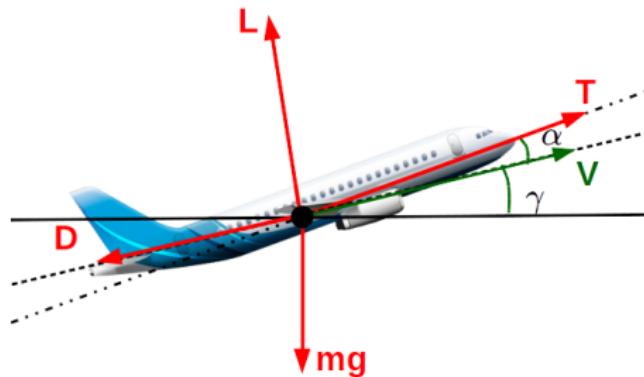
**States:**  $x = (h, V, \gamma, m)$

**Controls:**  $u = (\alpha, N_1)$

**Unknown functions of  $x, u$**

$$\left\{ \begin{array}{l} \dot{h} = V \sin \gamma \\ \dot{V} = \frac{T \cos \alpha - D - mg \sin \gamma}{m} \\ \dot{\gamma} = \frac{T \sin \alpha + L - mg \cos \gamma}{mV} \\ \dot{m} = -\frac{T}{I_{sp}} \end{array} \right.$$

# FLIGHT DYNAMICS



**States:**  $x = (h, V, \gamma, m)$

**Controls:**  $u = (\alpha, N_1)$

**Unknown functions of  $x, u$**

$$\left\{ \begin{array}{l} \dot{h} = V \sin \gamma \\ \dot{V} = \frac{T(u, x) \cos \alpha - D(u, x) - mg \sin \gamma}{m} \\ \dot{\gamma} = \frac{T(u, x) \sin \alpha + L(u, x)}{mV} - mg \cos \gamma \\ \dot{m} = -\frac{T(u, x)}{I_{sp}(u, x)} \end{array} \right.$$

# PHYSICAL MODELS OF NESTED FUNCTIONS

$$\begin{cases} T \text{ function of } (N_1, M, \rho) \\ D \text{ function of } (q, M, \alpha) \\ L \text{ function of } (q, M, \alpha) \\ I_{sp} \text{ function of } (SAT, M, h) \end{cases}$$

# PHYSICAL MODELS OF NESTED FUNCTIONS

$$\left\{ \begin{array}{l} T \text{ function of } (N_1, M, \rho) = \varphi_T(\mathbf{x}, \mathbf{u}) \\ D \text{ function of } (q, M, \alpha) = \varphi_D(\mathbf{x}, \mathbf{u}) \\ L \text{ function of } (q, M, \alpha) = \varphi_L(\mathbf{x}, \mathbf{u}) \\ I_{sp} \text{ function of } (SAT, M, h) = \varphi_{I_{sp}}(\mathbf{x}, \mathbf{u}) \end{array} \right.$$

# PHYSICAL MODELS OF NESTED FUNCTIONS

$$\begin{cases} T(\mathbf{x}, \mathbf{u}, \quad) = N_1 \times P_T(\rho, M) \\ D(\mathbf{x}, \mathbf{u}, \quad) = q \times P_D(\alpha, M) \\ L(\mathbf{x}, \mathbf{u}, \quad) = q \times P_L(\alpha, M) \\ I_{sp}(\mathbf{x}, \mathbf{u}, \quad) = SAT \times P_{Isp}(h, M) \end{cases}$$

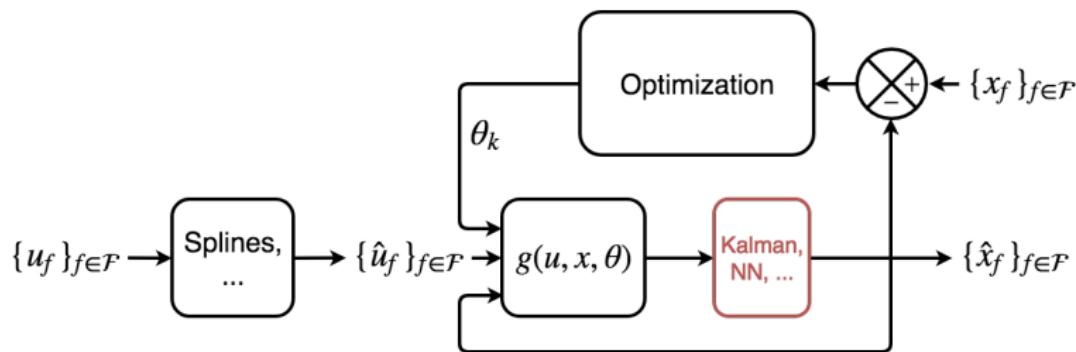
# PHYSICAL MODELS OF NESTED FUNCTIONS

$$\left\{ \begin{array}{l} T(\mathbf{x}, \mathbf{u}, \boldsymbol{\theta}_T) = N_1 \times P_T(\rho, M) = X_T \cdot \boldsymbol{\theta}_T \\ D(\mathbf{x}, \mathbf{u}, \boldsymbol{\theta}_D) = q \times P_D(\alpha, M) = X_D \cdot \boldsymbol{\theta}_D \\ L(\mathbf{x}, \mathbf{u}, \boldsymbol{\theta}_L) = q \times P_L(\alpha, M) = X_L \cdot \boldsymbol{\theta}_L \\ I_{sp}(\mathbf{x}, \mathbf{u}, \boldsymbol{\theta}_{Isp}) = SAT \times P_{Isp}(h, M) = X_{Isp} \cdot \boldsymbol{\theta}_{Isp} \end{array} \right.$$

$$X_T = N_1 \begin{pmatrix} 1 \\ \rho \\ M \\ \rho^2 \\ \rho M \\ M^2 \\ \vdots \end{pmatrix}, X_D = X_L = q \begin{pmatrix} 1 \\ \alpha \\ M \\ \alpha^2 \\ \alpha M \\ M^2 \\ \vdots \end{pmatrix}, X_{Isp} = SAT \begin{pmatrix} 1 \\ h \\ M \\ h^2 \\ hM \\ M^2 \\ \vdots \end{pmatrix}.$$

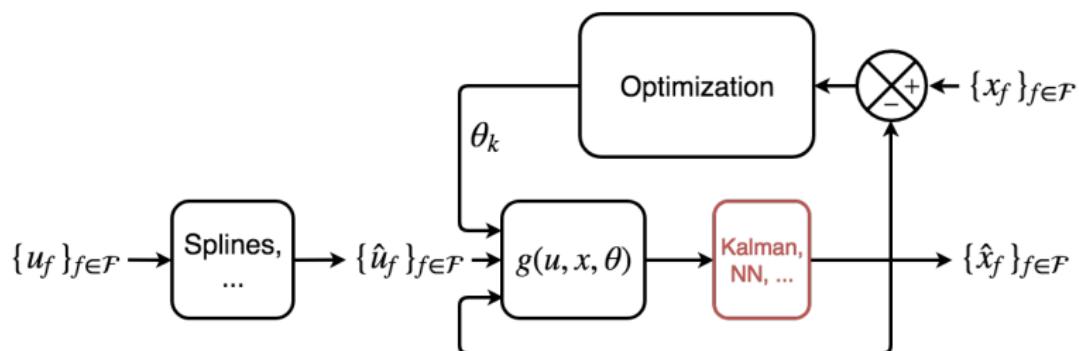
# STATE-OF-THE-ART - [JATEGAONKAR, 2006]

- Output-Error Method
- Filter-Error Method



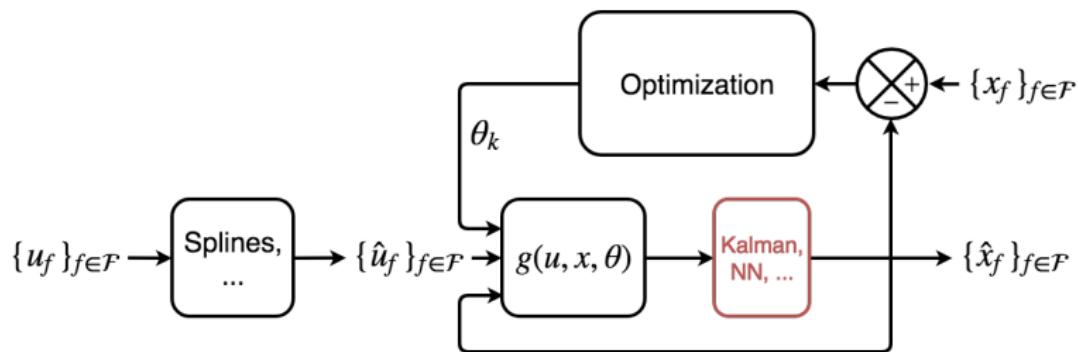
# STATE-OF-THE-ART - [JATEGAONKAR, 2006]

- Output-Error Method
  - Filter-Error Method
- } Less scalable to many trajectories



# STATE-OF-THE-ART - [JATEGAONKAR, 2006]

- Output-Error Method
  - Filter-Error Method
- } Less scalable to many trajectories

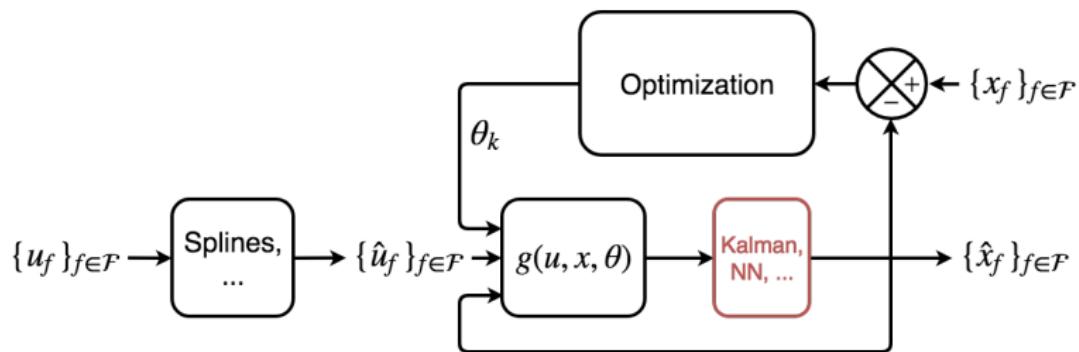


## ■ Equation-Error Method

$$\dot{x}(t) = g(\mathbf{u}(t), \mathbf{x}(t), \boldsymbol{\theta}) + \varepsilon(t), \quad t \in [0, t_f]$$

# STATE-OF-THE-ART - [JATEGAONKAR, 2006]

- Output-Error Method
  - Filter-Error Method
- } Less scalable to many trajectories

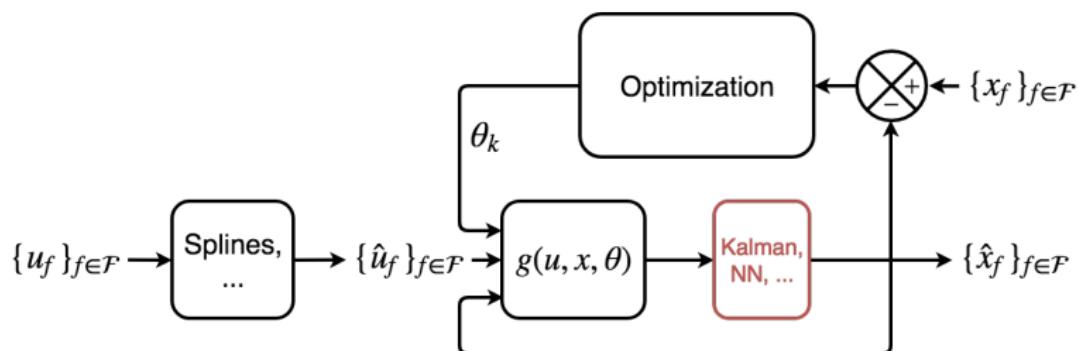


- **Equation-Error Method**

$$\dot{x}_i = g(\mathbf{u}_i, \mathbf{x}_i, \boldsymbol{\theta}) + \varepsilon_i, \quad i = 1, \dots, N$$

# STATE-OF-THE-ART - [JATEGAONKAR, 2006]

- Output-Error Method
  - Filter-Error Method
- } Less scalable to many trajectories

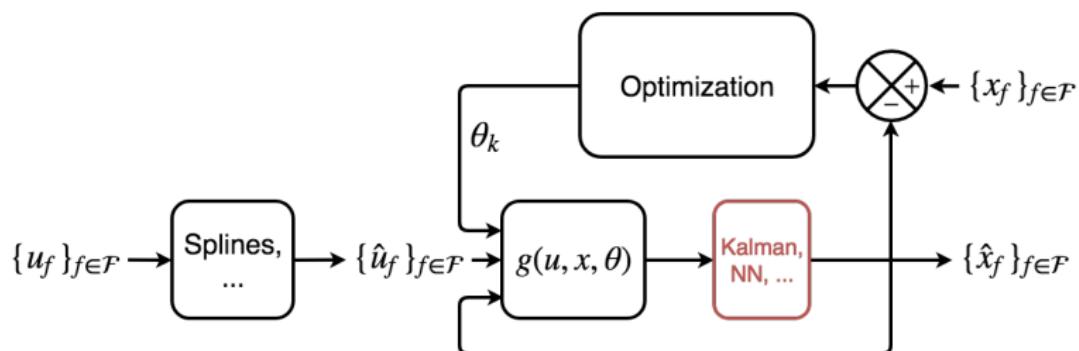


- **Equation-Error Method**

$$\min_{\theta} \sum_{i=1}^N \ell(\dot{x}_i, g(\mathbf{u}_i, \mathbf{x}_i, \theta))$$

# STATE-OF-THE-ART - [JATEGAONKAR, 2006]

- Output-Error Method
  - Filter-Error Method
- } Less scalable to many trajectories

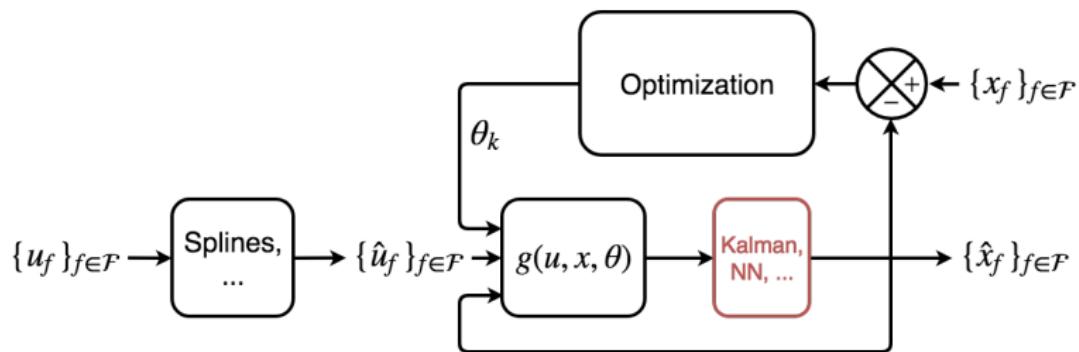


- **Equation-Error Method**      Ex: *(Nonlinear) Least-Squares*

$$\min_{\theta} \sum_{i=1}^N \left\| \dot{x}_i - g(\mathbf{u}_i, \mathbf{x}_i, \theta) \right\|_2^2$$

# STATE-OF-THE-ART - [JATEGAONKAR, 2006]

- Output-Error Method
  - Filter-Error Method
- } Less scalable to many trajectories



- **Equation-Error Method**      Ex: *(Nonlinear) Least-Squares*

$$\min_{\theta} \sum_{i=1}^N \left\| Y(\mathbf{u}_i, \mathbf{x}_i, \dot{\mathbf{x}}_i) - G(\mathbf{u}_i, \mathbf{x}_i, \dot{\mathbf{x}}_i, \theta) \right\|_2^2$$

# LEVERAGING THE DYNAMICS STRUCTURE

$$\left\{ \begin{array}{l} \dot{h} = V \sin \gamma \\ \dot{V} = \frac{T(\mathbf{u}, \mathbf{x}, \boldsymbol{\theta}_T) \cos \alpha - D(\mathbf{u}, \mathbf{x}, \boldsymbol{\theta}_D) - mg \sin \gamma}{m} \\ \dot{\gamma} = \frac{T(\mathbf{u}, \mathbf{x}, \boldsymbol{\theta}_T) \sin \alpha + L(\mathbf{u}, \mathbf{x}, \boldsymbol{\theta}_L) - mg \cos \gamma}{mV} \\ \dot{m} = -\frac{T(\mathbf{u}, \mathbf{x}, \boldsymbol{\theta}_T)}{I_{sp}(\mathbf{u}, \mathbf{x}, \boldsymbol{\theta}_{Isp})} \end{array} \right.$$

# LEVERAGING THE DYNAMICS STRUCTURE

$$\left\{ \begin{array}{l} \dot{h} = V \sin \gamma \\ \dot{V} = \frac{T(\mathbf{u}, \mathbf{x}, \boldsymbol{\theta}_T) \cos \alpha - D(\mathbf{u}, \mathbf{x}, \boldsymbol{\theta}_D) - mg \sin \gamma}{m} \\ \dot{\gamma} = \frac{T(\mathbf{u}, \mathbf{x}, \boldsymbol{\theta}_T) \sin \alpha + L(\mathbf{u}, \mathbf{x}, \boldsymbol{\theta}_L) - mg \cos \gamma}{mV} \\ \dot{m} = -\frac{T(\mathbf{u}, \mathbf{x}, \boldsymbol{\theta}_T)}{I_{sp}(\mathbf{u}, \mathbf{x}, \boldsymbol{\theta}_{Isp})} \end{array} \right.$$

- Nonlinear in states and controls

# LEVERAGING THE DYNAMICS STRUCTURE

$$\left\{ \begin{array}{l} \dot{h} = V \sin \gamma \\ \dot{V} = \frac{T(\mathbf{u}, \mathbf{x}, \boldsymbol{\theta}_T) \cos \alpha - D(\mathbf{u}, \mathbf{x}, \boldsymbol{\theta}_D) - mg \sin \gamma}{m} \\ \dot{\gamma} = \frac{T(\mathbf{u}, \mathbf{x}, \boldsymbol{\theta}_T) \sin \alpha + L(\mathbf{u}, \mathbf{x}, \boldsymbol{\theta}_L) - mg \cos \gamma}{mV} \\ \dot{m} = -\frac{T(\mathbf{u}, \mathbf{x}, \boldsymbol{\theta}_T)}{I_{sp}(\mathbf{u}, \mathbf{x}, \boldsymbol{\theta}_{Isp})} \end{array} \right.$$

- Nonlinear in states and controls
- Nonlinear in parameters

## LEVERAGING THE DYNAMICS STRUCTURE

$$\left\{ \begin{array}{l} \dot{h} = V \sin \gamma \\ m\dot{V} + mg \sin \gamma = T(\mathbf{u}, \mathbf{x}, \boldsymbol{\theta}_T) \cos \alpha - D(\mathbf{u}, \mathbf{x}, \boldsymbol{\theta}_D) \\ mV\dot{\gamma} + mg \cos \gamma = T(\mathbf{u}, \mathbf{x}, \boldsymbol{\theta}_T) \sin \alpha + L(\mathbf{u}, \mathbf{x}, \boldsymbol{\theta}_L) \\ 0 = T(\mathbf{u}, \mathbf{x}, \boldsymbol{\theta}_T) + \dot{m}l_{sp}(\mathbf{u}, \mathbf{x}, \boldsymbol{\theta}_{lsp}) \end{array} \right.$$

- Nonlinear in states and controls
- Nonlinear in parameters

# LEVERAGING THE DYNAMICS STRUCTURE

$$\left\{ \begin{array}{l} \dot{h} = V \sin \gamma \\ m\dot{V} + mg \sin \gamma = (X_T \cdot \theta_T) \cos \alpha - X_D \cdot \theta_D + \varepsilon_1 \\ mV\dot{\gamma} + mg \cos \gamma = (X_T \cdot \theta_T) \sin \alpha + X_L \cdot \theta_L + \varepsilon_2 \\ 0 = X_T \cdot \theta_T + \dot{m}(X_{Isp} \cdot \theta_{Isp}) + \varepsilon_3 \end{array} \right.$$

$$\overbrace{Y(\mathbf{u}, \mathbf{x}, \dot{\mathbf{x}})} \quad \overbrace{G(\mathbf{u}, \mathbf{x}, \dot{\mathbf{x}}, \boldsymbol{\theta})}$$

- Nonlinear in states and controls
- ~~Nonlinear in parameters~~ → Linear in parameters

# LEVERAGING THE DYNAMICS STRUCTURE

$$\left\{ \begin{array}{l} \dot{h} = V \sin \gamma \\ m\dot{V} + mg \sin \gamma = (\mathbf{X}_T \cdot \boldsymbol{\theta}_T) \cos \alpha - X_D \cdot \boldsymbol{\theta}_D + \varepsilon_1 \\ mV\dot{\gamma} + mg \cos \gamma = (\mathbf{X}_T \cdot \boldsymbol{\theta}_T) \sin \alpha + X_L \cdot \boldsymbol{\theta}_L + \varepsilon_2 \\ 0 = X_T \cdot \boldsymbol{\theta}_T + \dot{m}(\mathbf{X}_{Isp} \cdot \boldsymbol{\theta}_{Isp}) + \varepsilon_3 \end{array} \right.$$

$$\underbrace{Y(\mathbf{u}, \mathbf{x}, \dot{\mathbf{x}})}_{Y(\mathbf{u}, \mathbf{x}, \dot{\mathbf{x}})} \quad \underbrace{G(\mathbf{u}, \mathbf{x}, \dot{\mathbf{x}}, \boldsymbol{\theta})}_{G(\mathbf{u}, \mathbf{x}, \dot{\mathbf{x}}, \boldsymbol{\theta})}$$

- Nonlinear in states and controls
- ~~Nonlinear in parameters~~ → Linear in parameters

# LEVERAGING THE DYNAMICS STRUCTURE

$$\left\{ \begin{array}{l} \dot{h} = V \sin \gamma \\ Y_1 = X_{T1} \cdot \theta_T - X_D \cdot \theta_D + \varepsilon_1 \\ Y_2 = X_{T2} \cdot \theta_T + X_L \cdot \theta_L + \varepsilon_2 \\ Y_3 = X_T \cdot \theta_T + X_{Ispm} \cdot \theta_{Isp} + \varepsilon_3 \end{array} \right.$$

- Nonlinear in states and controls
- ~~Nonlinear in parameters~~ → Linear in parameters

# LEVERAGING THE DYNAMICS STRUCTURE

$$\left\{ \begin{array}{l} \dot{h} = V \sin \gamma \\ Y_1 = X_{T1} \cdot \theta_T - X_D \cdot \theta_D + \varepsilon_1 \\ Y_2 = X_{T2} \cdot \theta_T + X_L \cdot \theta_L + \varepsilon_2 \\ Y_3 = X_T \cdot \theta_T + X_{Ispm} \cdot \theta_{Isp} + \varepsilon_3 \end{array} \right.$$

- Nonlinear in states and controls
- ~~Nonlinear in parameters~~ → Linear in parameters
- Structured
- Coupling

# LEVERAGING THE DYNAMICS STRUCTURE

$$\left\{ \begin{array}{l} \dot{h} = V \sin \gamma \\ Y_1 = X_{T1} \cdot \theta_T - X_D \cdot \theta_D + \varepsilon_1 \\ Y_2 = X_{T2} \cdot \theta_T + X_L \cdot \theta_L + \varepsilon_2 \\ Y_3 = X_T \cdot \theta_T + X_{Ispm} \cdot \theta_{Isp} + \varepsilon_3 \end{array} \right.$$

- Nonlinear in states and controls
- ~~Nonlinear in parameters~~ → Linear in parameters
- Structured
- Coupling ↪ **Multi-task Learning**

# MULTI-TASK REGRESSION

Aircraft:

$$\begin{cases} Y_1 = X_{T1} \cdot \theta_T - X_D \cdot \theta_D + \varepsilon_1 \\ Y_2 = X_{T2} \cdot \theta_T + X_L \cdot \theta_L + \varepsilon_2 \\ Y_3 = X_T \cdot \theta_T + X_{lspm} \cdot \theta_{lsp} + \varepsilon_3 \end{cases}$$

General:

$$\begin{cases} Y_1 = X_{c,1} \cdot \theta_c + X_1 \cdot \theta_1 + \varepsilon_1 \\ Y_2 = X_{c,2} \cdot \theta_c + X_2 \cdot \theta_2 + \varepsilon_2 \\ \vdots \\ Y_K = X_{c,K} \cdot \theta_c + X_K \cdot \theta_K + \varepsilon_K \end{cases}$$

**Coupling parameters**, **Task specific parameters**

Many other examples:

- *Giant squid neurons* [FitzHugh, 1961, Nagumo et al., 1962],
- *Susceptible-infectious-recovered models* [Anderson and May, 1992],
- *Mechanical systems*,...

# MULTI-TASK REGRESSION

Aircraft:

$$\begin{cases} Y_1 = X_{T1} \cdot \theta_T - X_D \cdot \theta_D + \varepsilon_1 \\ Y_2 = X_{T2} \cdot \theta_T + X_L \cdot \theta_L + \varepsilon_2 \\ Y_3 = X_T \cdot \theta_T + X_{lspm} \cdot \theta_{lsp} + \varepsilon_3 \end{cases}$$

General:

$$\begin{cases} Y_1 = X_{c,1} \cdot \theta_c + X_1 \cdot \theta_1 + \varepsilon_1 \\ Y_2 = X_{c,2} \cdot \theta_c + X_2 \cdot \theta_2 + \varepsilon_2 \\ \vdots \\ Y_K = X_{c,K} \cdot \theta_c + X_K \cdot \theta_K + \varepsilon_K \end{cases}$$

Coupling parameters , Task specific parameters

Multi-task Linear Least-Squares:

$$\min_{\theta} \sum_{k=1}^K \sum_{i=1}^N (Y_{k,i} - X_{c,k,i} \cdot \theta_c - X_{k,i} \cdot \theta_k)^2$$

# MULTI-TASK REGRESSION

Aircraft:

$$\begin{cases} Y_1 = X_{T1} \cdot \theta_T - X_D \cdot \theta_D + \varepsilon_1 \\ Y_2 = X_{T2} \cdot \theta_T + X_L \cdot \theta_L + \varepsilon_2 \\ Y_3 = X_T \cdot \theta_T + X_{lspm} \cdot \theta_{lsp} + \varepsilon_3 \end{cases}$$

General:

$$\begin{cases} Y_1 = X_{c,1} \cdot \theta_c + X_1 \cdot \theta_1 + \varepsilon_1 \\ Y_2 = X_{c,2} \cdot \theta_c + X_2 \cdot \theta_2 + \varepsilon_2 \\ \vdots \\ Y_K = X_{c,K} \cdot \theta_c + X_K \cdot \theta_K + \varepsilon_K \end{cases}$$

Coupling parameters , Task specific parameters

Multi-task Linear Least-Squares:

$$\min_{\theta} \sum_{i=1}^N \|Y_i - X_i \theta\|_2^2$$

with  $\theta = (\theta_c, \theta_1, \dots, \theta_K) \in \mathbb{R}^p$ ,  $p = d_c + \sum_{k=1}^K d_k$ ,  
 $Y_i \in \mathbb{R}^K$  and  $X_i \in \mathbb{R}^{K \times p}$ .

# MULTI-TASK REGRESSION

Aircraft:

$$\begin{cases} Y_1 = X_{T1} \cdot \theta_T - X_D \cdot \theta_D + \varepsilon_1 \\ Y_2 = X_{T2} \cdot \theta_T + X_L \cdot \theta_L + \varepsilon_2 \\ Y_3 = X_T \cdot \theta_T + X_{lspm} \cdot \theta_{lsp} + \varepsilon_3 \end{cases}$$

General:

$$\begin{cases} Y_1 = X_{c,1} \cdot \theta_c + X_1 \cdot \theta_1 + \varepsilon_1 \\ Y_2 = X_{c,2} \cdot \theta_c + X_2 \cdot \theta_2 + \varepsilon_2 \\ \vdots \quad \vdots \\ Y_K = X_{c,K} \cdot \theta_c + X_K \cdot \theta_K + \varepsilon_K \end{cases}$$

**Coupling parameters**, **Task specific parameters**

Multi-task Linear Least-Squares:

Block-sparse Coupling Structure

$$\min_{\theta} \sum_{i=1}^N \left\| \begin{pmatrix} Y_{1,i} \\ \vdots \\ Y_{K,i} \end{pmatrix} - \begin{pmatrix} X_{c,1,i}^\top & X_{1,i}^\top & 0 & 0 & \dots & 0 \\ X_{c,2,i}^\top & 0 & X_{2,i}^\top & 0 & \dots & 0 \\ \vdots & 0 & 0 & \ddots & 0 & 0 \\ X_{c,K,i}^\top & 0 & 0 & \dots & 0 & X_{K,i}^\top \end{pmatrix} \begin{pmatrix} \theta_c \\ \theta_1 \\ \vdots \\ \theta_K \end{pmatrix} \right\|_2^2$$

# FEATURE SELECTION

Our model:

$$T = N_1(\theta_{T,1} + \theta_{T,2}\rho + \theta_{T,3}M + \theta_{T,4}\rho^2 + \theta_{T,5}\rho M + \theta_{T,6}M^2 + \theta_{T,7}\rho^3 + \theta_{T,8}\rho^2M + \theta_{T,9}\rho M^2 + \theta_{T,10}M^3 + \theta_{T,11}\rho^4 + \theta_{T,12}\rho^3M + \theta_{T,13}\rho^2M^2 + \theta_{T,14}\rho M^3 + \theta_{T,15}M^4).$$

Mattingly's model [Mattingly et al., 1992]:

$$T = N_1(\theta_{T,1}\rho + \theta_{T,2}\rho M^3).$$

# FEATURE SELECTION

Our model:

$$T = N_1(\theta_{T,1} + \theta_{T,2}\rho + \theta_{T,3}M + \theta_{T,4}\rho^2 + \theta_{T,5}\rho M + \theta_{T,6}M^2 + \theta_{T,7}\rho^3 + \theta_{T,8}\rho^2M + \theta_{T,9}\rho M^2 + \theta_{T,10}M^3 + \theta_{T,11}\rho^4 + \theta_{T,12}\rho^3M + \theta_{T,13}\rho^2M^2 + \theta_{T,14}\rho M^3 + \theta_{T,15}M^4).$$

Mattingly's model [Mattingly et al., 1992]:

$$T = N_1(\theta_{T,1}\rho + \theta_{T,2}\rho M^3).$$

⇒ High risk of overfitting

# FEATURE SELECTION

Our (sparse) model:

$$T = N_1(\theta_{T,1} + \theta_{T,2}\rho + \theta_{T,3}M + \theta_{T,4}\rho^2 + \theta_{T,5}\rho M + \theta_{T,6}M^2 + \theta_{T,7}\rho^3 + \theta_{T,8}\rho^2M + \theta_{T,9}\rho M^2 + \theta_{T,10}M^3 + \theta_{T,11}\rho^4 + \theta_{T,12}\rho^3M + \theta_{T,13}\rho^2M^2 + \theta_{T,14}\rho M^3 + \theta_{T,15}M^4).$$

Mattingly's model [Mattingly et al., 1992]:

$$T = N_1(\theta_{T,1}\rho + \theta_{T,2}\rho M^3).$$

⇒ High risk of overfitting

# FEATURE SELECTION

Our (sparse) model:

$$T = N_1(\theta_{T,1} + \theta_{T,2}\rho + \theta_{T,3}M + \theta_{T,4}\rho^2 + \theta_{T,5}\rho M + \theta_{T,6}M^2 + \theta_{T,7}\rho^3 + \theta_{T,8}\rho^2M + \theta_{T,9}\rho M^2 + \theta_{T,10}M^3 + \theta_{T,11}\rho^4 + \theta_{T,12}\rho^3M + \theta_{T,13}\rho^2M^2 + \theta_{T,14}\rho M^3 + \theta_{T,15}M^4).$$

Mattingly's model [Mattingly et al., 1992]:

$$T = N_1(\theta_{T,1}\rho + \theta_{T,2}\rho M^3).$$

Sparse models are:

- Less susceptible to overfitting,
- More compliant with physical models,
- More interpretable,
- Lighter/Faster.

# BLOCK-SPARSE LASSO

Lasso [Tibshirani, 1994]:  $\{(X_i, Y_i)\}_{i=1}^N \subset \mathbb{R}^{d+1}$  i.i.d sample,

$$\min_{\boldsymbol{\theta}} \sum_{i=1}^N (Y_i - X_i \cdot \boldsymbol{\theta})^2 + \lambda \|\boldsymbol{\theta}\|_1.$$

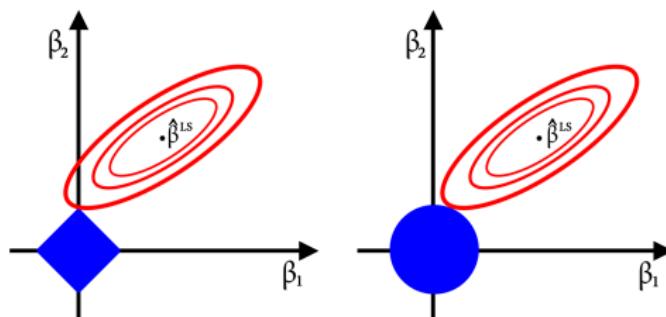


FIGURE:  $^1$ Sparsity induced by  $L^1$  norm in Lasso.

# BLOCK-SPARSE LASSO

$$\min_{\boldsymbol{\theta}} \sum_{k=1}^K \sum_{i=1}^N (Y_{k,i} - X_{c,k,i} \cdot \boldsymbol{\theta}_c - X_{k,i} \cdot \boldsymbol{\theta}_k)^2 + \lambda_c \|\boldsymbol{\theta}_c\|_1 + \sum_{k=1}^K \lambda_k \|\boldsymbol{\theta}_k\|_1$$

# BLOCK-SPARSE LASSO

Block-sparse structure preserved  $\rightsquigarrow$  **Equivalent to Lasso problem**

$$\min_{\boldsymbol{\theta}} \sum_{k=1}^K \sum_{i=1}^N (Y_{k,i} - X_{c,k,i} \cdot \boldsymbol{\theta}_c - X_{k,i} \cdot \boldsymbol{\theta}_k)^2 + \lambda_c \|\boldsymbol{\theta}_c\|_1 + \sum_{k=1}^K \lambda_k \|\boldsymbol{\theta}_k\|_1$$

# BLOCK-SPARSE LASSO

Block-sparse structure preserved  $\rightsquigarrow$  **Equivalent to Lasso problem**

$$\min_{\beta} \sum_{i=1}^N \|Y_i - B_i\beta\|_2^2 + \lambda_c \|\beta\|_1$$

with  $\beta = (\boldsymbol{\theta}_c, \frac{\lambda_1}{\lambda_c} \boldsymbol{\theta}_1, \dots, \frac{\lambda_K}{\lambda_c} \boldsymbol{\theta}_K) \in \mathbb{R}^p$ ,  $p = d_c + \sum_{k=1}^K d_k$ ,  
 $Y_i \in \mathbb{R}^K$  and  $B_i \in \mathbb{R}^{K \times p}$ .

# BLOCK-SPARSE LASSO

Block-sparse structure preserved  $\rightsquigarrow$  **Equivalent to Lasso problem**

$$\min_{\boldsymbol{\theta}} \sum_{i=1}^N \|Y_i - X_i \boldsymbol{\theta}\|_2^2 + \lambda_c \|\boldsymbol{\theta}\|_1$$

with  $\boldsymbol{\theta} = (\boldsymbol{\theta}_c, \boldsymbol{\theta}_1, \dots, \boldsymbol{\theta}_K) \in \mathbb{R}^p$ ,  $p = d_c + \sum_{k=1}^K d_k$ ,  
 $Y_i \in \mathbb{R}^K$  and  $X_i \in \mathbb{R}^{K \times p}$ ,

In practice, we choose  $\lambda_k = \lambda_c$ , for all  $k = 1, \dots, 3$  and

$$X_i = \begin{pmatrix} X_{T1,i}^\top & -X_{D,i}^\top & 0 & 0 \\ X_{T2,i}^\top & 0 & X_{L,i}^\top & 0 \\ X_{T,i}^\top & 0 & 0 & X_{Ispm,i}^\top \end{pmatrix}, \quad Y_i = \begin{pmatrix} Y_{1,i} \\ Y_{2,i} \\ Y_{3,i} \end{pmatrix}$$

# BOOTSTRAP IMPLEMENTATION

**High correlations between features...**

# BOOTSTRAP IMPLEMENTATION

**High correlations between features...**  
⇒ **Inconsistent selections via the lasso !**

# BOOTSTRAP IMPLEMENTATION

High correlations between features...

⇒ Inconsistent selections via the lasso !

---

Bolasso - Bach [2008]

---

training data  $\mathcal{T} = \{(X_i, Y_i)\}_{i=1}^N \subset \mathbb{R}^{K \times (K+1)} \times \mathbb{R}^K$ ,

**Require:** number of bootstrap replicates  $b$ ,

$L^1$  penalty parameter  $\lambda_c$ ,

- 1: **for**  $k = 1$  **to**  $b$  **do**
  - 2:   Generate bootstrap sample  $\mathcal{T}_k$ ,
  - 3:   Compute Block sparse Lasso estimate  $\hat{\theta}^k$  from  $\mathcal{T}_k$ ,
  - 4:   Compute support  $J_k = \{j, \hat{\theta}_j^k \neq 0\}$ ,
  - 5: **end for**
  - 6: Compute intersection  $J = \bigcap_{k=1}^b J_k$ ,
  - 7: Compute  $\hat{\theta}_J$  from selected features using Least-Squares.
-

# BOOTSTRAP IMPLEMENTATION

**High correlations between features...**  
⇒ **Inconsistent selections via the lasso !**

---

Bolasso - Bach [2008]

---

training data  $\mathcal{T} = \{(X_i, Y_i)\}_{i=1}^N \subset \mathbb{R}^{K \times (K+1)} \times \mathbb{R}^K$ ,

**Require:** number of bootstrap replicates  $b$ ,

$L^1$  penalty parameter  $\lambda_c$ ,

- 1: **for**  $k = 1$  **to**  $b$  **do**
  - 2:   Generate bootstrap sample  $\mathcal{T}_k$ ,
  - 3:   Compute Block sparse Lasso estimate  $\hat{\theta}^k$  from  $\mathcal{T}_k$ ,
  - 4:   Compute support  $J_k = \{j, \hat{\theta}_j^k \neq 0\}$ ,
  - 5: **end for**
  - 6: Compute intersection  $J = \bigcap_{k=1}^b J_k$ ,
  - 7: Compute  $\hat{\theta}_J$  from selected features using Least-Squares.
- 

- Consistency even under high correlations proved in Bach [2008],

# BOOTSTRAP IMPLEMENTATION

**High correlations between features...**  
⇒ **Inconsistent selections via the lasso !**

---

Bolasso - Bach [2008]

---

training data  $\mathcal{T} = \{(X_i, Y_i)\}_{i=1}^N \subset \mathbb{R}^{K \times (K+1)} \times \mathbb{R}^K$ ,

**Require:** number of bootstrap replicates  $b$ ,  
 $L^1$  penalty parameter  $\lambda_c$ ,

- 1: **for**  $k = 1$  **to**  $b$  **do**
  - 2:   Generate bootstrap sample  $\mathcal{T}_k$ ,
  - 3:   Compute Block sparse Lasso estimate  $\hat{\theta}^k$  from  $\mathcal{T}_k$ ,
  - 4:   Compute support  $J_k = \{j, \hat{\theta}_j^k \neq 0\}$ ,
  - 5: **end for**
  - 6: Compute intersection  $J = \bigcap_{k=1}^b J_k$ ,
  - 7: Compute  $\hat{\theta}_J$  from selected features using Least-Squares.
- 

- Consistency even under high correlations proved in Bach [2008],
- Efficient implementations exist: LARS [Efron et al., 2004].

# PROBLEM WITH INTRA-GROUP CORRELATIONS

$$\min_{\theta} \sum_{i=1}^N \|Y_i - X_i \theta\|_2^2 + \lambda_c \|\theta\|_1 \Rightarrow \hat{\theta}_T = \hat{\theta}_{Ispl} = 0!$$

# PROBLEM WITH INTRA-GROUP CORRELATIONS

$$\min_{\theta} \sum_{i=1}^N \|Y_i - X_i \theta\|_2^2 + \lambda_c \|\theta\|_1 \Rightarrow \hat{\theta}_T = \hat{\theta}_{Ispl} = 0!$$

$$\begin{cases} Y_1 = X_{T1} \cdot \theta_T - X_D \cdot \theta_D + \varepsilon_1 \\ Y_2 = X_{T2} \cdot \theta_T + X_L \cdot \theta_L + \varepsilon_2 \\ 0 = X_T \cdot \theta_T + X_{Ispl} \cdot \theta_{Ispl} + \varepsilon_3 \end{cases}$$

# PROBLEM WITH INTRA-GROUP CORRELATIONS

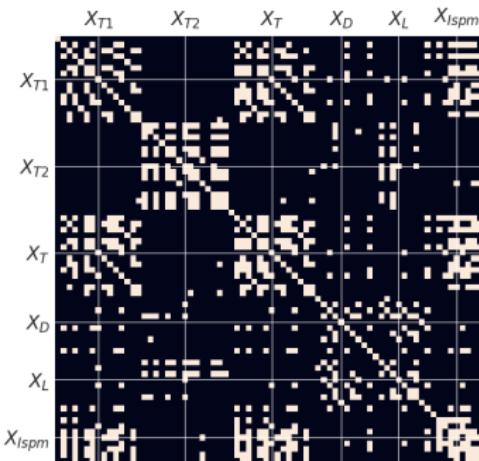
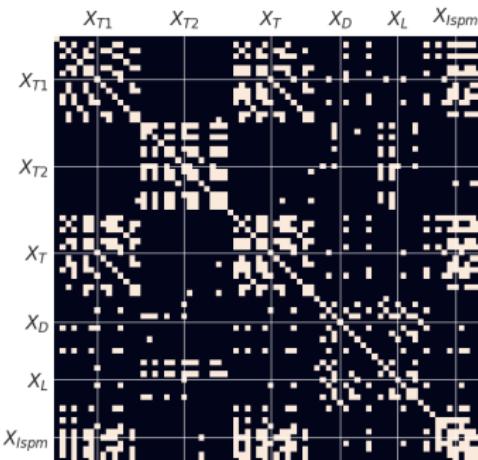


FIGURE: Features correlations  
higher than 0.9 in absolute  
value in white.

# PROBLEM WITH INTRA-GROUP CORRELATIONS



$\Rightarrow \theta \mapsto \sum_{i=1}^N \|Y_i - X_i\theta\|_2^2$  not injective...

**III-posed problem !**

FIGURE: Features correlations higher than 0.9 in absolute value in white.

# PROBLEM WITH INTRA-GROUP CORRELATIONS

$$\left\{ \begin{array}{l} Y_1 = X_{T1} \cdot \theta_T - X_D \cdot \theta_D + \varepsilon_1 \\ Y_2 = X_{T2} \cdot \theta_T + X_L \cdot \theta_L + \varepsilon_2 \\ 0 = X_T \cdot \theta_T + X_{Ispm} \cdot \theta_{Isp} + \varepsilon_3 \end{array} \right.$$

$$\min_{\boldsymbol{\theta}} \sum_{i=1}^N \|Y_i - X_i \boldsymbol{\theta}\|_2^2 + \lambda_c \|\boldsymbol{\theta}\|_1$$

Prior model  $\tilde{l}_{sp}$  from Roux [2005]  $\rightsquigarrow \tilde{l}_{sp,i} = \tilde{l}_{sp}(\mathbf{u}_i, \mathbf{x}_i)$ ,  $i = 1, \dots, N$ .

# PROBLEM WITH INTRA-GROUP CORRELATIONS

$$\left\{ \begin{array}{l} Y_1 = X_{T1} \cdot \theta_T - X_D \cdot \theta_D + \varepsilon_1 \\ Y_2 = X_{T2} \cdot \theta_T + X_L \cdot \theta_L + \varepsilon_2 \\ 0 = X_T \cdot \theta_T + X_{Ispm} \cdot \theta_{Isp} + \varepsilon_3 \end{array} \right.$$

$$\min_{\boldsymbol{\theta}} \sum_{i=1}^N \left( \|Y_i - X_i \boldsymbol{\theta}\|_2^2 + \lambda_t \|\tilde{I}_{sp,i} - X_{Isp,i} \cdot \boldsymbol{\theta}_{Isp}\|_2^2 \right) + \lambda_c \|\boldsymbol{\theta}\|_1$$

Prior model  $\tilde{I}_{sp}$  from Roux [2005]  $\rightsquigarrow \tilde{I}_{sp,i} = \tilde{I}_{sp}(\mathbf{u}_i, \mathbf{x}_i)$ ,  $i = 1, \dots, N$ .

# PROBLEM WITH INTRA-GROUP CORRELATIONS

$$\begin{cases} Y_1 = X_{T1} \cdot \theta_T - X_D \cdot \theta_D + \varepsilon_1 \\ Y_2 = X_{T2} \cdot \theta_T + X_L \cdot \theta_L + \varepsilon_2 \\ 0 = X_T \cdot \theta_T + X_{Ispm} \cdot \theta_{Isp} + \varepsilon_3 \\ \sqrt{\lambda_t} \tilde{I}_{sp} = \sqrt{\lambda_t} X_{Isp} \cdot \theta_{Isp} + \varepsilon_4 \end{cases}$$

$$\min_{\theta} \sum_{i=1}^N \left( \|Y_i - X_i \theta\|_2^2 + \lambda_t \|\tilde{I}_{sp,i} - X_{Isp,i} \cdot \theta_{Isp}\|_2^2 \right) + \lambda_c \|\theta\|_1$$

Prior model  $\tilde{I}_{sp}$  from Roux [2005]  $\rightsquigarrow \tilde{I}_{sp,i} = \tilde{I}_{sp}(\mathbf{u}_i, \mathbf{x}_i)$ ,  $i = 1, \dots, N$ .

# PROBLEM WITH INTRA-GROUP CORRELATIONS

$$\begin{cases} Y_1 = X_{T1} \cdot \theta_T - X_D \cdot \theta_D + \varepsilon_1 \\ Y_2 = X_{T2} \cdot \theta_T + X_L \cdot \theta_L + \varepsilon_2 \\ 0 = X_T \cdot \theta_T + X_{Ispm} \cdot \theta_{Isp} + \varepsilon_3 \\ \sqrt{\lambda_t} \tilde{I}_{sp} = \sqrt{\lambda_t} X_{Isp} \cdot \theta_{Isp} + \varepsilon_4 \end{cases}$$

$$\min_{\theta} \sum_{i=1}^N \| \tilde{Y}_i - \tilde{X}_i \theta \|_2^2 + \lambda_c \|\theta\|_1$$

$$\tilde{Y}_i = \begin{pmatrix} Y_{1,i} \\ Y_{2,i} \\ 0 \\ \sqrt{\lambda_t} \tilde{I}_{sp,i} \end{pmatrix}, \quad \tilde{X}_i = \begin{pmatrix} X_{T1,i}^\top & -X_{D,i}^\top & 0 & 0 \\ X_{T2,i}^\top & 0 & X_{L,i}^\top & 0 \\ X_{T,i}^\top & 0 & 0 & X_{Ispm,i}^\top \\ 0 & 0 & 0 & \sqrt{\lambda_t} X_{Isp,i}^\top \end{pmatrix},$$

Prior model  $\tilde{I}_{sp}$  from Roux [2005]  $\rightsquigarrow \tilde{I}_{sp,i} = \tilde{I}_{sp}(\mathbf{u}_i, \mathbf{x}_i)$ ,  $i = 1, \dots, N$ .

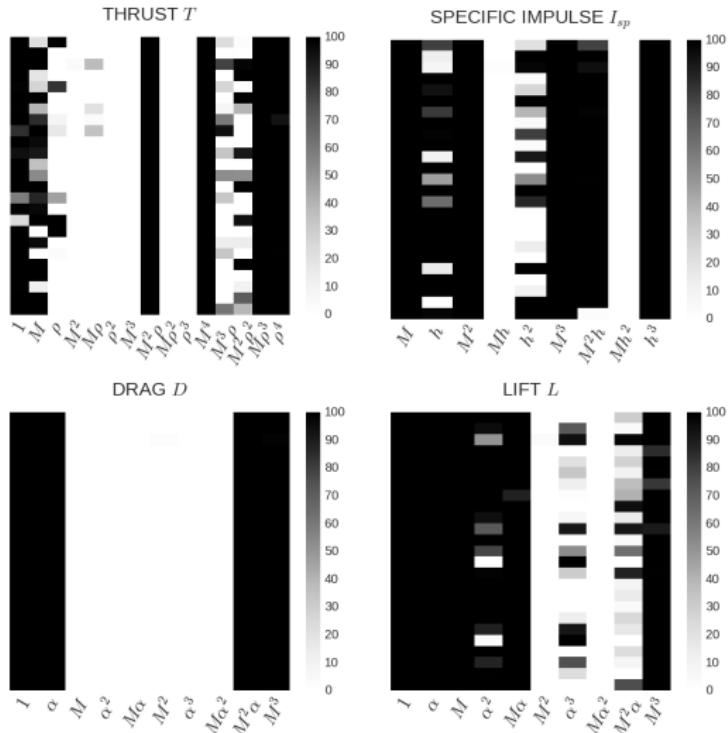
# FEATURE SELECTION RESULTS

- 25 different B737-800,
- 10 471 flights = 8 261 619 observations,

# FEATURE SELECTION RESULTS

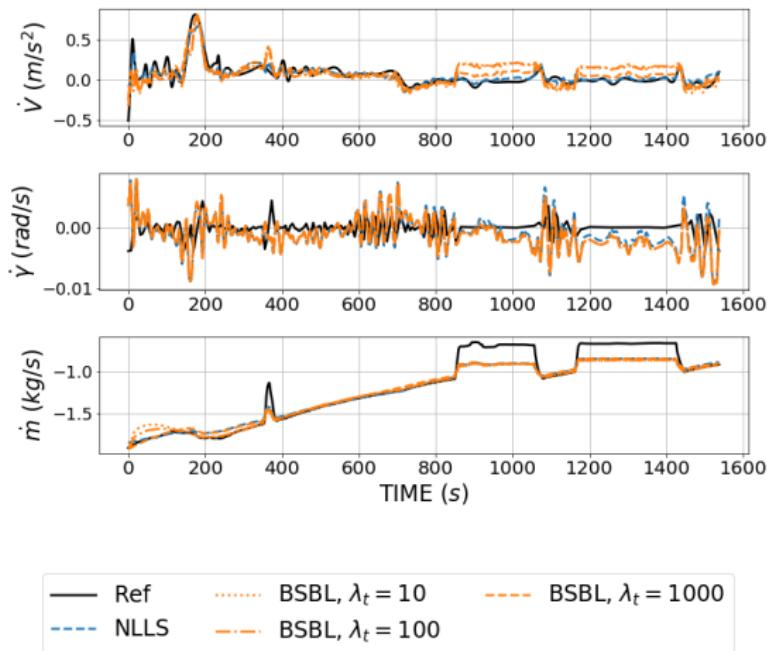
- 25 different B737-800,
- 10 471 flights = 8 261 619 observations,
- Block sparse Bolasso used for  $T$ ,  $D$ ,  $L$  and  $I_{sp}$ ,
- We expect similar model structures,

# FEATURE SELECTION RESULTS

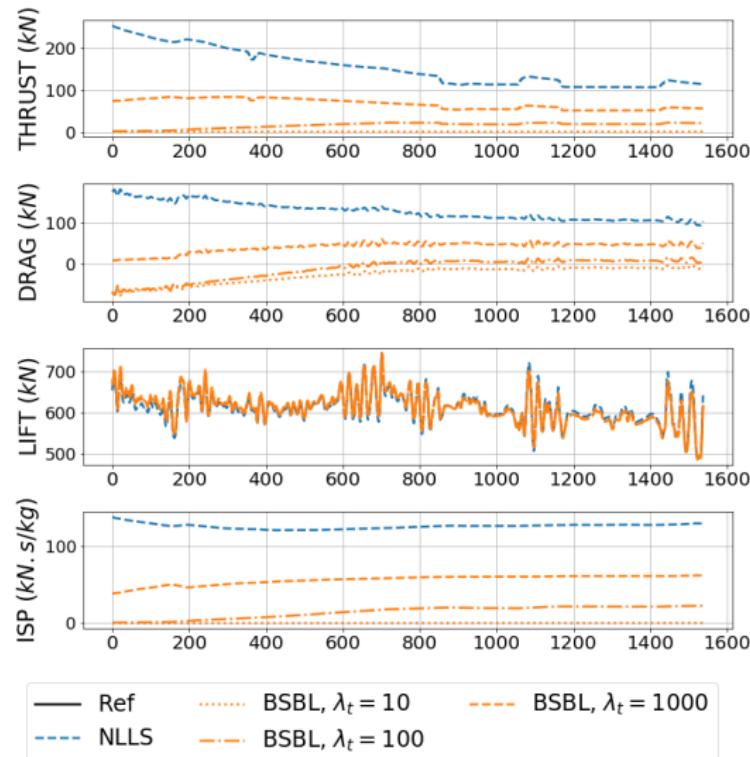


Feature selection results for the thrust, drag, lift and specific impulse models.

# ACCURACY OF DYNAMICS PREDICTIONS



# REALISM OF HIDDEN ELEMENTS



# FLIGHT RESIMULATION

# FLIGHT RESIMULATION

- Last assessment criterion = static;

# FLIGHT RESIMULATION

- Last assessment criterion = static;
- Does not incorporate the fact that the observations are time dependent;

# FLIGHT RESIMULATION

- Last assessment criterion = static;
- Does not incorporate the fact that the observations are time dependent;
- Does not take into account the goal of optimally controlling the aircraft system.

# FLIGHT RESIMULATION

- Last assessment criterion = static;
- Does not incorporate the fact that the observations are time dependent;
- Does not take into account the goal of optimally controlling the aircraft system.

Another possible dynamic criterion:

$$\begin{aligned} \min_{(\mathbf{x}, \mathbf{u})} & \int_{t_0}^{t_n} (\|\mathbf{u}(t) - \mathbf{u}_{test}(t)\|_{\mathbf{u}}^2 + \|\mathbf{x}(t) - \mathbf{x}_{test}(t)\|_{\mathbf{x}}^2) dt \\ \text{s.t. } & \dot{\mathbf{x}}(t) = g(\mathbf{x}(t), \mathbf{u}(t), \hat{\theta}), \end{aligned}$$

where  $\|\cdot\|_{\mathbf{u}}$ ,  $\|\cdot\|_{\mathbf{x}}$  denote scaling norms.

# FLIGHT RESIMULATION

- Last assessment criterion = static;
- Does not incorporate the fact that the observations are time dependent;
- Does not take into account the goal of optimally controlling the aircraft system.

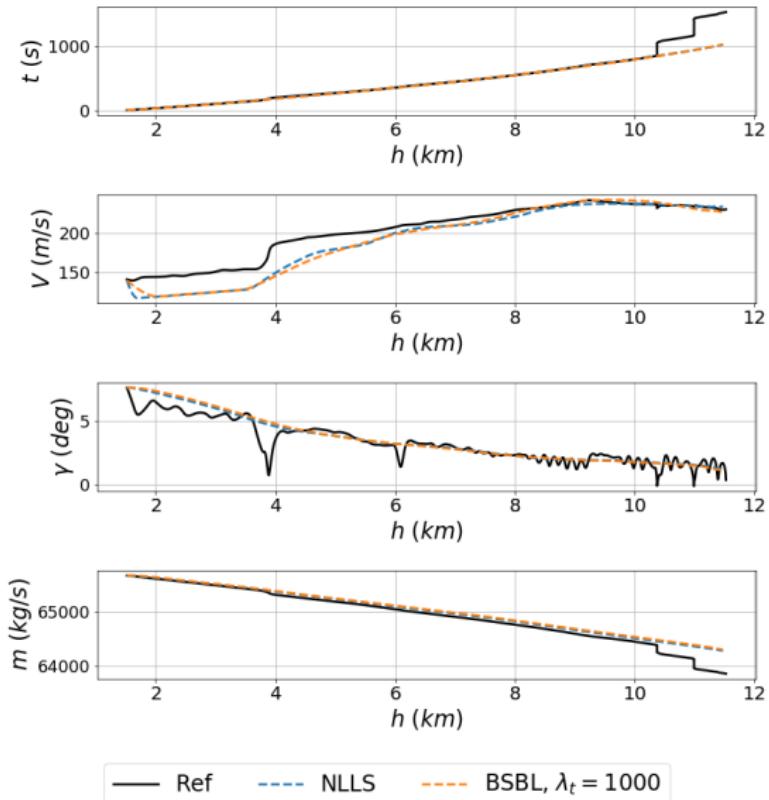
Another possible dynamic criterion:

$$\begin{aligned} \min_{(\mathbf{x}, \mathbf{u})} & \int_{t_0}^{t_n} (\|\mathbf{u}(t) - \mathbf{u}_{test}(t)\|_{\mathbf{u}}^2 + \|\mathbf{x}(t) - \mathbf{x}_{test}(t)\|_{\mathbf{x}}^2) dt \\ \text{s.t. } & \dot{\mathbf{x}}(t) = g(\mathbf{x}(t), \mathbf{u}(t), \hat{\theta}), \end{aligned}$$

where  $\|\cdot\|_{\mathbf{u}}$ ,  $\|\cdot\|_{\mathbf{x}}$  denote scaling norms.

For practical applications:  $t \leftrightarrow h$

# FLIGHT RESIMULATION



# FLIGHT RESIMULATION

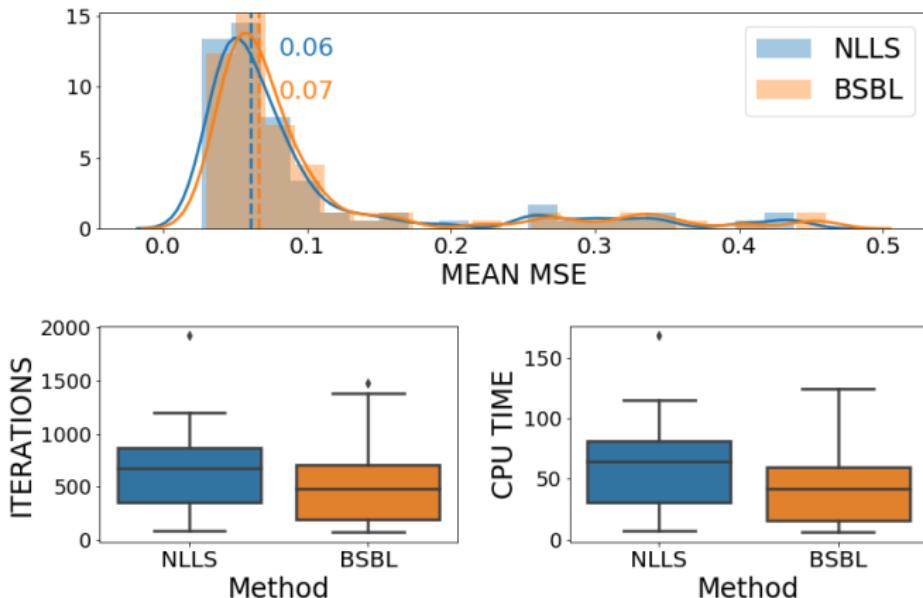


FIGURE: Distribution of the off-sample simulation error and boxplot of the optimization number of iterations and CPU time.

# SYSTEM IDENTIFICATION CONCLUSION

- 1 Proposed Equation-Error Method approaches which extend to the System Identification framework well-known supervised learning techniques (Lasso, Ridge, bootstrap,...),

# SYSTEM IDENTIFICATION CONCLUSION

- 1 Proposed Equation-Error Method approaches which extend to the System Identification framework well-known supervised learning techniques (Lasso, Ridge, bootstrap,...),
- 2 Applicable to large amounts of data,

# SYSTEM IDENTIFICATION CONCLUSION

- 1 Proposed Equation-Error Method approaches which extend to the System Identification framework well-known supervised learning techniques (Lasso, Ridge, bootstrap,...),
- 2 Applicable to large amounts of data,
- 3 Block-sparse estimators are proved to lead to consistent structured feature selection,

# SYSTEM IDENTIFICATION CONCLUSION

- 1 Proposed Equation-Error Method approaches which extend to the System Identification framework well-known supervised learning techniques (Lasso, Ridge, bootstrap,...),
- 2 Applicable to large amounts of data,
- 3 Block-sparse estimators are proved to lead to consistent structured feature selection,
- 4 Can be efficiently trained using LARS algorithm as they are equivalent to successive Lasso problems,

# SYSTEM IDENTIFICATION CONCLUSION

- 1 Proposed Equation-Error Method approaches which extend to the System Identification framework well-known supervised learning techniques (Lasso, Ridge, bootstrap,...),
- 2 Applicable to large amounts of data,
- 3 Block-sparse estimators are proved to lead to consistent structured feature selection,
- 4 Can be efficiently trained using LARS algorithm as they are equivalent to successive Lasso problems,
- 5 Compared to regular Nonlinear Least-Squares:
  - Similar performances in accuracy and training time,

# SYSTEM IDENTIFICATION CONCLUSION

- 1 Proposed Equation-Error Method approaches which extend to the System Identification framework well-known supervised learning techniques (Lasso, Ridge, bootstrap,...),
- 2 Applicable to large amounts of data,
- 3 Block-sparse estimators are proved to lead to consistent structured feature selection,
- 4 Can be efficiently trained using LARS algorithm as they are equivalent to successive Lasso problems,
- 5 Compared to regular Nonlinear Least-Squares:
  - Similar performances in accuracy and training time,
  - No initialization required,

# SYSTEM IDENTIFICATION CONCLUSION

- 1 Proposed Equation-Error Method approaches which extend to the System Identification framework well-known supervised learning techniques (Lasso, Ridge, bootstrap,...),
- 2 Applicable to large amounts of data,
- 3 Block-sparse estimators are proved to lead to consistent structured feature selection,
- 4 Can be efficiently trained using LARS algorithm as they are equivalent to successive Lasso problems,
- 5 Compared to regular Nonlinear Least-Squares:
  - Similar performances in accuracy and training time,
  - No initialization required,
  - Light, interpretable and compact data-dependent models (more than 50% compression),

# SYSTEM IDENTIFICATION CONCLUSION

- 1 Proposed Equation-Error Method approaches which extend to the System Identification framework well-known supervised learning techniques (Lasso, Ridge, bootstrap,...),
- 2 Applicable to large amounts of data,
- 3 Block-sparse estimators are proved to lead to consistent structured feature selection,
- 4 Can be efficiently trained using LARS algorithm as they are equivalent to successive Lasso problems,
- 5 Compared to regular Nonlinear Least-Squares:
  - Similar performances in accuracy and training time,
  - No initialization required,
  - Light, interpretable and compact data-dependent models (more than 50% compression),
  - Faster convergence when applied to control problems.

## TRAJECTORY ACCEPTABILITY

# TRAJECTORY ACCEPTABILITY

$$\begin{aligned} & \min_{(\mathbf{x}, \mathbf{u}) \in \mathbb{X} \times \mathbb{U}} \int_0^{t_f} C(\mathbf{u}(t), \mathbf{x}(t)) dt, \\ \text{s.t. } & \left\{ \begin{array}{l} \dot{\mathbf{x}}(t) = \hat{g}(\mathbf{u}(t), \mathbf{x}(t)), \quad \text{a.e. } t \in [0, t_f], \\ \text{Other constraints...} \end{array} \right. \end{aligned} \tag{AOCP}$$

# TRAJECTORY ACCEPTABILITY

$$\begin{aligned} & \min_{(\mathbf{x}, \mathbf{u}) \in \mathbb{X} \times \mathbb{U}} \int_0^{t_f} C(\mathbf{u}(t), \mathbf{x}(t)) dt, \\ \text{s.t. } & \left\{ \begin{array}{l} \dot{\mathbf{x}}(t) = \hat{g}(\mathbf{u}(t), \mathbf{x}(t)), \quad \text{a.e. } t \in [0, t_f], \\ \text{Other constraints...} \end{array} \right. \end{aligned} \tag{AOCP}$$

$\Rightarrow \hat{\mathbf{z}} = (\hat{\mathbf{x}}, \hat{\mathbf{u}})$  solution of (AOCP).

# TRAJECTORY ACCEPTABILITY

$$\begin{aligned} & \min_{(\mathbf{x}, \mathbf{u}) \in \mathbb{X} \times \mathbb{U}} \int_0^{t_f} C(\mathbf{u}(t), \mathbf{x}(t)) dt, \\ \text{s.t. } & \left\{ \begin{array}{l} \dot{\mathbf{x}}(t) = \hat{g}(\mathbf{u}(t), \mathbf{x}(t)), \quad \text{a.e. } t \in [0, t_f], \\ \text{Other constraints...} \end{array} \right. \end{aligned} \tag{AOCP}$$

$\Rightarrow \hat{\mathbf{z}} = (\hat{\mathbf{x}}, \hat{\mathbf{u}})$  solution of (AOCP).

- Is  $\hat{\mathbf{z}}$  inside the validity region of the dynamics model  $\hat{g}$  ?

# TRAJECTORY ACCEPTABILITY

$$\begin{aligned} & \min_{(\mathbf{x}, \mathbf{u}) \in \mathbb{X} \times \mathbb{U}} \int_0^{t_f} C(\mathbf{u}(t), \mathbf{x}(t)) dt, \\ \text{s.t. } & \left\{ \begin{array}{l} \dot{\mathbf{x}}(t) = \hat{g}(\mathbf{u}(t), \mathbf{x}(t)), \quad \text{a.e. } t \in [0, t_f], \\ \text{Other constraints...} \end{array} \right. \end{aligned} \tag{AOCP}$$

$\Rightarrow \hat{\mathbf{z}} = (\hat{\mathbf{x}}, \hat{\mathbf{u}})$  solution of (AOCP).

- Is  $\hat{\mathbf{z}}$  inside the validity region of the dynamics model  $\hat{g}$  ?
- Does it look like a real trajectory ?

# TRAJECTORY ACCEPTABILITY

$$\begin{aligned} & \min_{(\mathbf{x}, \mathbf{u}) \in \mathbb{X} \times \mathbb{U}} \int_0^{t_f} C(\mathbf{u}(t), \mathbf{x}(t)) dt, \\ \text{s.t. } & \left\{ \begin{array}{l} \dot{\mathbf{x}}(t) = \hat{g}(\mathbf{u}(t), \mathbf{x}(t)), \quad \text{a.e. } t \in [0, t_f], \\ \text{Other constraints...} \end{array} \right. \end{aligned} \tag{AOCP}$$

$\Rightarrow \hat{\mathbf{z}} = (\hat{\mathbf{x}}, \hat{\mathbf{u}})$  solution of (AOCP).

- Is  $\hat{\mathbf{z}}$  inside the validity region of the dynamics model  $\hat{g}$  ?
- Does it look like a real trajectory ?



Pilots acceptance



Air Traffic Control<sup>2</sup>

# TRAJECTORY ACCEPTABILITY

$$\begin{aligned} & \min_{(\mathbf{x}, \mathbf{u}) \in \mathbb{X} \times \mathbb{U}} \int_0^{t_f} C(\mathbf{u}(t), \mathbf{x}(t)) dt, \\ \text{s.t. } & \left\{ \begin{array}{l} \dot{\mathbf{x}}(t) = \hat{g}(\mathbf{u}(t), \mathbf{x}(t)), \quad \text{a.e. } t \in [0, t_f], \\ \text{Other constraints...} \end{array} \right. \end{aligned} \tag{AOCP}$$

$\Rightarrow \hat{\mathbf{z}} = (\hat{\mathbf{x}}, \hat{\mathbf{u}})$  solution of (AOCP).

- Is  $\hat{\mathbf{z}}$  inside the validity region of the dynamics model  $\hat{g}$  ?
- Does it look like a real trajectory ?



Pilots acceptance



Air Traffic Control<sup>2</sup>

**How can we quantify the closeness from the optimized trajectory to the set of real flights?**

# OPTIMIZED TRAJECTORY LIKELIHOOD

**Assumption:** We suppose that the real flights are observations of the same functional random variable  $Z = (Z_t)$  valued in  $\mathcal{C}(\mathbb{T}, E)$ , with  $E$  compact subset of  $\mathbb{R}^d$  and  $\mathbb{T} = [0, t_f]$ .

**How likely is it to draw the optimized trajectory from the law of  $Z$  ?**

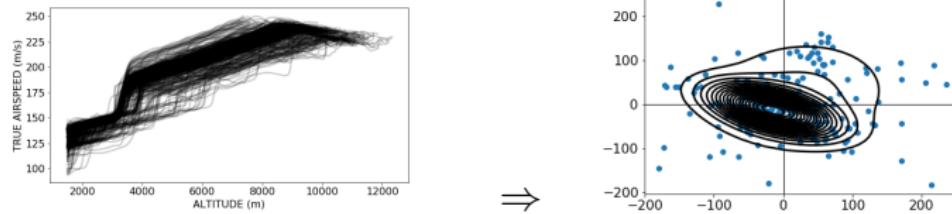
# HOW TO APPLY THIS TO FUNCTIONAL DATA?

**Problem:** Computation of probability densities in infinite dimensional space.

# HOW TO APPLY THIS TO FUNCTIONAL DATA?

**Problem:** Computation of probability densities in infinite dimensional space.

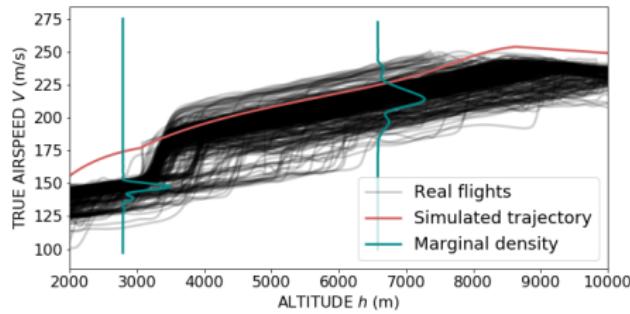
- Standard approach in Functional Data Analysis: use Functional Principal Component Analysis to decompose the data in a small number of coefficients



# HOW TO APPLY THIS TO FUNCTIONAL DATA?

**Problem:** Computation of probability densities in infinite dimensional space.

- Standard approach in Functional Data Analysis: use Functional Principal Component Analysis to decompose the data in a small number of coefficients
- Or: we can use the marginal densities



# HOW DO WE AGGREGATE THE MARGINAL LIKELIHOODS?

- $f_t$  marginal density of  $Z$ , i.e. probability density function of  $Z_t$ ,
- $\mathbf{y}$  new trajectory,
- $f_t(\mathbf{y}(t))$  marginal likelihood of  $\mathbf{y}$  at  $t$ , i.e. likelihood of observing  $Z_t = \mathbf{y}(t)$ .

# HOW DO WE AGGREGATE THE MARGINAL LIKELIHOODS?

- $f_t$  marginal density of  $Z$ , i.e. probability density function of  $Z_t$ ,
- $\mathbf{y}$  new trajectory,
- $f_t(\mathbf{y}(t))$  marginal likelihood of  $\mathbf{y}$  at  $t$ , i.e. likelihood of observing  $Z_t = \mathbf{y}(t)$ .

## MEAN MARGINAL LIKELIHOOD

$$\text{MML}(Z, \mathbf{y}) = \frac{1}{t_f} \int_0^{t_f} \psi[f_t, \mathbf{y}(t)] dt,$$

where  $\psi : L^1(E, \mathbb{R}_+) \times \mathbb{R} \rightarrow [0; 1]$  is a continuous scaling map,

# HOW DO WE AGGREGATE THE MARGINAL LIKELIHOODS?

- $f_t$  marginal density of  $Z$ , i.e. probability density function of  $Z_t$ ,
- $\mathbf{y}$  new trajectory,
- $f_t(\mathbf{y}(t))$  marginal likelihood of  $\mathbf{y}$  at  $t$ , i.e. likelihood of observing  $Z_t = \mathbf{y}(t)$ .

## MEAN MARGINAL LIKELIHOOD

$$\text{MML}(Z, \mathbf{y}) = \frac{1}{t_f} \int_0^{t_f} \psi[f_t, \mathbf{y}(t)] dt,$$

where  $\psi : L^1(E, \mathbb{R}_+) \times \mathbb{R} \rightarrow [0; 1]$  is a continuous scaling map, because marginal densities may have really different shapes.

# HOW DO WE AGGREGATE THE MARGINAL LIKELIHOODS?

Possible scalings are the normalized density

$$\psi[f_t, \mathbf{y}(t)] := \frac{f_t(\mathbf{y}(t))}{\max_{z \in E} f_t(z)},$$

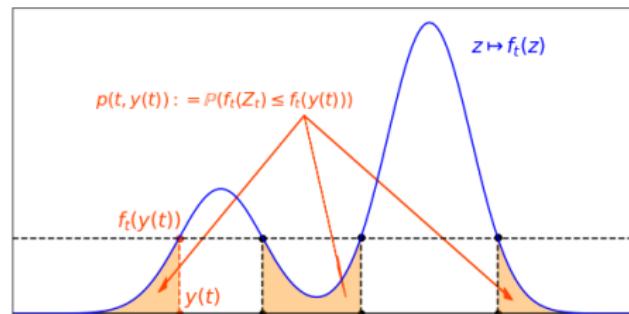
# HOW DO WE AGGREGATE THE MARGINAL LIKELIHOODS?

Possible scalings are the normalized density

$$\psi[f_t, \mathbf{y}(t)] := \frac{f_t(\mathbf{y}(t))}{\max_{z \in E} f_t(z)},$$

or the confidence level

$$\psi[f_t, \mathbf{y}(t)] := \mathbb{P}(f_t(Z_t) \leq f_t(\mathbf{y}(t))).$$



# HOW DO WE DEAL WITH SAMPLED CURVES?

In practice, the  $m$  trajectories are sampled at variable discrete times:

$$\begin{aligned}\mathcal{T}^D &:= \{(t_j^r, z_j^r)\}_{\substack{1 \leq j \leq n \\ 1 \leq r \leq m}} \subset \mathbb{T} \times E, & z_j^r &:= \mathbf{z}^r(t_j^r), \\ \mathcal{Y} &:= \{(\tilde{t}_j, y_j)\}_{j=1}^{\tilde{n}} \subset \mathbb{T} \times E, & y_j &:= \mathbf{y}(\tilde{t}_j).\end{aligned}$$

# HOW DO WE DEAL WITH SAMPLED CURVES?

In practice, the  $m$  trajectories are sampled at variable discrete times:

$$\begin{aligned}\mathcal{T}^D &:= \{(t_j^r, z_j^r)\}_{\substack{1 \leq j \leq n \\ 1 \leq r \leq m}} \subset \mathbb{T} \times E, & z_j^r &:= \mathbf{z}^r(t_j^r), \\ \mathcal{Y} &:= \{(\tilde{t}_j, y_j)\}_{j=1}^{\tilde{n}} \subset \mathbb{T} \times E, & y_j &:= \mathbf{y}(\tilde{t}_j).\end{aligned}$$

Hence, we approximate the MML using a Riemann sum which aggregates consistent estimators  $\hat{f}_{\tilde{t}_j}^m$  of the marginal densities  $f_{\tilde{t}_j}$ :

$$\text{EMML}_m(\mathcal{T}^D, \mathcal{Y}) := \frac{1}{t_f} \sum_{j=1}^{\tilde{n}} \psi[\hat{f}_{\tilde{t}_j}^m, y_j] \Delta \tilde{t}_j.$$

# HOW CAN WE ESTIMATE MARGINAL DENSITIES?

# HOW CAN WE ESTIMATE MARGINAL DENSITIES?

- In practice, the altitude plays the role of time, so we can't assume the same sampling for each trajectory;

# HOW CAN WE ESTIMATE MARGINAL DENSITIES?

- In practice, the altitude plays the role of time, so we can't assume the same sampling for each trajectory;
- Assume sampling times  $\{t_j^r : j = 1, \dots, n; r = 1, \dots, m\}$  to be i.i.d. observations of a r.v.  $T$ , indep.  $Z$ ;

# HOW CAN WE ESTIMATE MARGINAL DENSITIES?

- In practice, the altitude plays the role of time, so we can't assume the same sampling for each trajectory;
- Assume sampling times  $\{t_j^r : j = 1, \dots, n; r = 1, \dots, m\}$  to be i.i.d. observations of a r.v.  $T$ , indep.  $Z$ ;
- Our problem can be seen as a conditional probability density learning problem with  $(X, Y) = (T, Z_T)$ , where  $f_t$  is the density of  $Z_t = (Z_T | T = t) = (Y|X)$ .

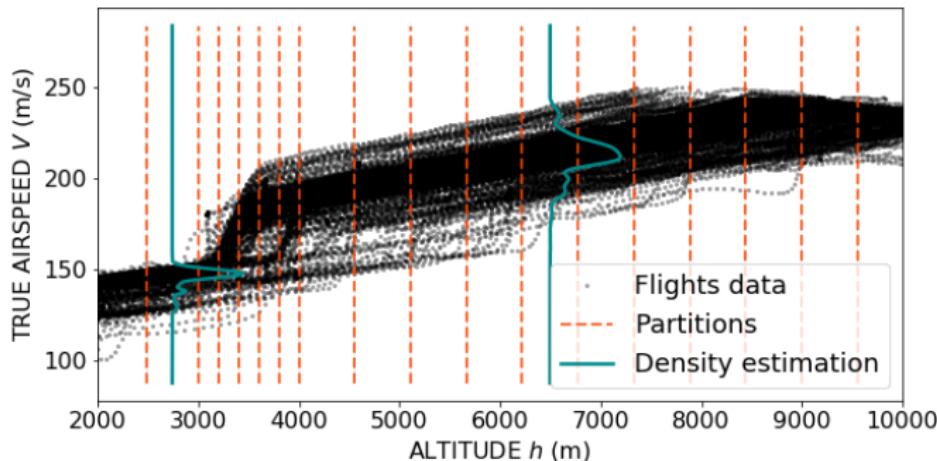
# HOW CAN WE ESTIMATE MARGINAL DENSITIES?

- In practice, the altitude plays the role of time, so we can't assume the same sampling for each trajectory;
  - Assume sampling times  $\{t_j^r : j = 1, \dots, n; r = 1, \dots, m\}$  to be i.i.d. observations of a r.v.  $T$ , indep.  $Z$ ;
  - Our problem can be seen as a conditional probability density learning problem with  $(X, Y) = (T, Z_T)$ , where  $f_t$  is the density of  $Z_t = (Z_T | T = t) = (Y|X)$ .
- 1 We can apply SOA conditional density estimation techniques, such as LS-CDE [Sugiyama et al., 2010],

# HOW CAN WE ESTIMATE MARGINAL DENSITIES?

- In practice, the altitude plays the role of time, so we can't assume the same sampling for each trajectory;
  - Assume sampling times  $\{t_j^r : j = 1, \dots, n; r = 1, \dots, m\}$  to be i.i.d. observations of a r.v.  $T$ , indep.  $Z$ ;
  - Our problem can be seen as a conditional probability density learning problem with  $(X, Y) = (T, Z_T)$ , where  $f_t$  is the density of  $Z_t = (Z_T | T = t) = (Y|X)$ .
- 
- 1 We can apply SOA conditional density estimation techniques, such as LS-CDE [Sugiyama et al., 2010],
  - 2 **We can use a fine partitioning of the time domain.**

# PARTITION BASED MARGINAL DENSITY ESTIMATION



Idea: to average in time the marginal densities over small bins by applying classical multivariate density estimation techniques to each subset.

# CONSISTENCY

We denote by:

- $\Theta : \mathcal{S} \rightarrow L^1(E, \mathbb{R}_+)$  multivariate density estimation statistic,
- $\mathcal{S} = \{(z_k)_{k=1}^N \in E^N : N \in \mathbb{N}^*\}$  set of finite sequences,

# CONSISTENCY

We denote by:

- $\Theta : \mathcal{S} \rightarrow L^1(E, \mathbb{R}_+)$  multivariate density estimation statistic,
- $\mathcal{S} = \{(z_k)_{k=1}^N \in E^N : N \in \mathbb{N}^*\}$  set of finite sequences,
- $m$  the number of random curves;
- $\mathcal{T}_t^m$  subset of data points whose sampling times fall in the bin containing  $t$ ;

# CONSISTENCY

We denote by:

- $\Theta : \mathcal{S} \rightarrow L^1(E, \mathbb{R}_+)$  multivariate density estimation statistic,
- $\mathcal{S} = \{(z_k)_{k=1}^N \in E^N : N \in \mathbb{N}^*\}$  set of finite sequences,
- $m$  the number of random curves;
- $\mathcal{T}_t^m$  subset of data points whose sampling times fall in the bin containing  $t$ ;
- $\hat{f}_t^m := \Theta[\mathcal{T}_t^m]$  estimator trained using  $\mathcal{T}_t^m$ .

# CONSISTENCY

ASSUMPTION 1 - POSITIVE TIME DENSITY

$\nu \in L^\infty(E, \mathbb{R}_+)$  density function of  $T$ , s.t.

$$\nu_+ := \text{ess} \sup_{t \in \mathbb{T}} \nu(t) < \infty, \quad \nu_- := \text{ess} \inf_{t \in \mathbb{T}} \nu(t) > 0.$$

# CONSISTENCY

ASSUMPTION 1 - POSITIVE TIME DENSITY

$\nu \in L^\infty(E, \mathbb{R}_+)$  density function of  $T$ , s.t.

$$\nu_+ := \text{ess sup}_{t \in \mathbb{T}} \nu(t) < \infty, \quad \nu_- := \text{ess inf}_{t \in \mathbb{T}} \nu(t) > 0.$$

ASSUMPTION 2 - LIPSCHITZ IN TIME

Function  $(t, z) \in \mathbb{T} \times E \mapsto f_t(z)$  is continuous and

$$|f_{t_1}(z) - f_{t_2}(z)| \leq L|t_1 - t_2|, \quad L > 0.$$

# CONSISTENCY

ASSUMPTION 1 - POSITIVE TIME DENSITY

$\nu \in L^\infty(E, \mathbb{R}_+)$  density function of  $T$ , s.t.

$$\nu_+ := \text{ess sup}_{t \in \mathbb{T}} \nu(t) < \infty, \quad \nu_- := \text{ess inf}_{t \in \mathbb{T}} \nu(t) > 0.$$

ASSUMPTION 2 - LIPSCHITZ IN TIME

Function  $(t, z) \in \mathbb{T} \times E \mapsto f_t(z)$  is continuous and

$$|f_{t_1}(z) - f_{t_2}(z)| \leq L|t_1 - t_2|, \quad L > 0.$$

ASSUMPTION 3 - SHRINKING BINS

The homogeneous partition  $\{B_\ell^m\}_{\ell=1}^{q_m}$  of  $[0; t_f]$ , with binsize  $b_m$ , is s.t.

$$\lim_{m \rightarrow \infty} b_m = 0, \quad \lim_{m \rightarrow \infty} mb_m = \infty.$$

# CONSISTENCY

## ASSUMPTION 4 - I.I.D. CONSISTENCY

- $\mathcal{G}$  arbitrary family of probability density functions on  $E$ ,  $\rho \in \mathcal{G}$ ,
- $S_\rho^N$  i.i.d sample of size  $N$  drawn from  $\rho$  valued in  $\mathcal{S}$ .

The estimator obtained by applying  $\Theta$  to  $S_\rho^N$ , denoted by

$$\hat{\rho}^N := \Theta[S_\rho^N] \in L^1(E, \mathbb{R}_+),$$

is a (pointwise) consistent density estimator, uniformly in  $\rho$ :

For all  $z \in E, \varepsilon > 0, \alpha_1 > 0$ , there is  $N_{\varepsilon, \alpha_1} > 0$  such that, for any  $\rho \in \mathcal{G}$ ,

$$N \geq N_{\varepsilon, \alpha_1} \Rightarrow \mathbb{P} \left( \left| \hat{\rho}^N(z) - \rho(z) \right| < \varepsilon \right) > 1 - \alpha_1.$$

# CONSISTENCY

## THEOREM 1

Under assumptions 1 to 4, for any  $z \in E$  and  $t \in \mathbb{T}$ ,  $\hat{f}_{\ell^m(t)}^m(z)$  consistently approximates the marginal density  $f_t(z)$  as the number of curves  $m$  grows:

$$\forall \varepsilon > 0, \quad \lim_{m \rightarrow \infty} \mathbb{P} (|\hat{f}_t^m(z) - f_t(z)| < \varepsilon) = 1.$$

# CONSISTENCY

## THEOREM 1

Under assumptions 1 to 4, for any  $z \in E$  and  $t \in \mathbb{T}$ ,  $\hat{f}_{\ell^m(t)}^m(z)$  consistently approximates the marginal density  $f_t(z)$  as the number of curves  $m$  grows:

$$\forall \varepsilon > 0, \quad \lim_{m \rightarrow \infty} \mathbb{P}(|\hat{f}_t^m(z) - f_t(z)| < \varepsilon) = 1.$$

**Note that:**

- $m \rightarrow \infty \neq N \rightarrow \infty$ ,

# CONSISTENCY

## THEOREM 1

Under assumptions 1 to 4, for any  $z \in E$  and  $t \in \mathbb{T}$ ,  $\hat{f}_{\ell^m(t)}^m(z)$  consistently approximates the marginal density  $f_t(z)$  as the number of curves  $m$  grows:

$$\forall \varepsilon > 0, \quad \lim_{m \rightarrow \infty} \mathbb{P}(|\hat{f}_t^m(z) - f_t(z)| < \varepsilon) = 1.$$

**Note that:**

- $m \rightarrow \infty \neq N \rightarrow \infty$ ,
- Number of samples = random,

# CONSISTENCY

## THEOREM 1

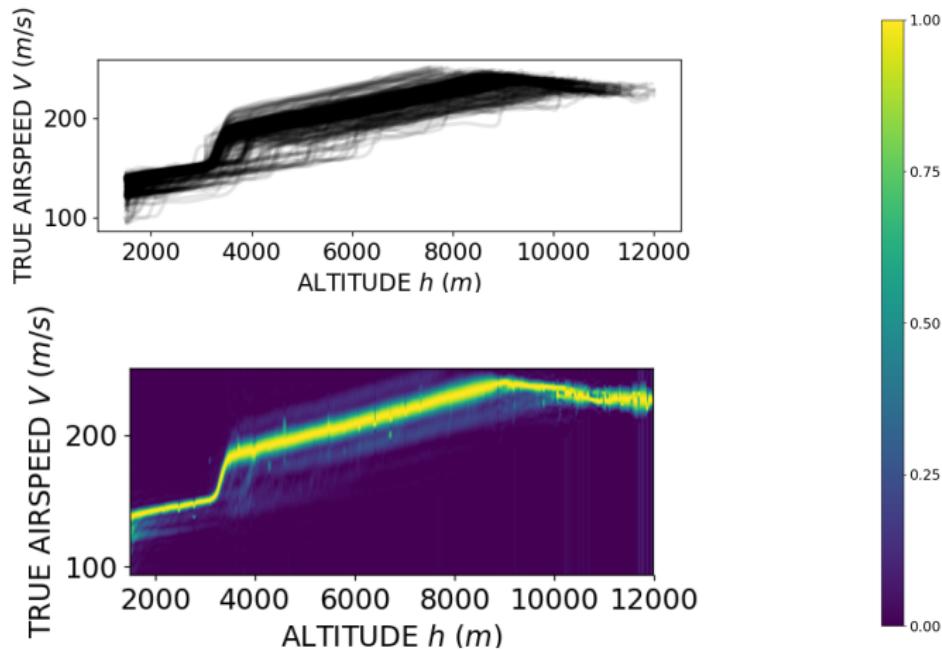
Under assumptions 1 to 4, for any  $z \in E$  and  $t \in \mathbb{T}$ ,  $\hat{f}_{\ell^m(t)}^m(z)$  consistently approximates the marginal density  $f_t(z)$  as the number of curves  $m$  grows:

$$\forall \varepsilon > 0, \quad \lim_{m \rightarrow \infty} \mathbb{P}(|\hat{f}_t^m(z) - f_t(z)| < \varepsilon) = 1.$$

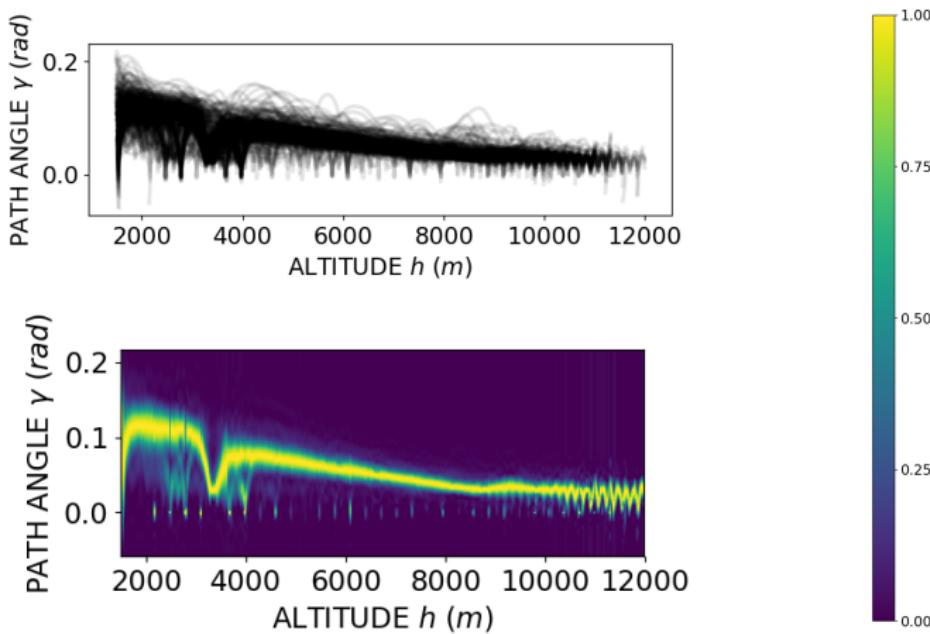
**Note that:**

- $m \rightarrow \infty \neq N \rightarrow \infty$ ,
- Number of samples = random,
- Training data not i.i.d.

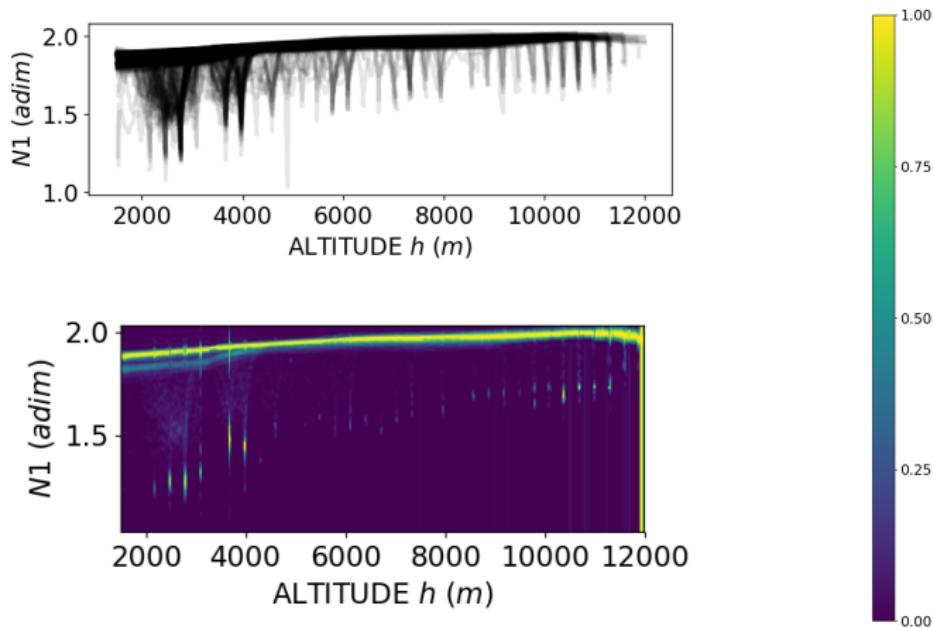
# MARGINAL DENSITY ESTIMATION RESULTS



# MARGINAL DENSITY ESTIMATION RESULTS



# MARGINAL DENSITY ESTIMATION RESULTS



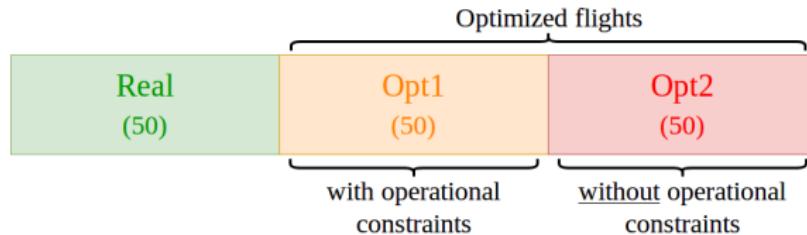
# HOW GOOD IS IT COMPARED TO OTHER METHODS?

# HOW GOOD IS IT COMPARED TO OTHER METHODS?

- Training set of  $m = 424$  flights  $\simeq 334\ 531$  point observations,

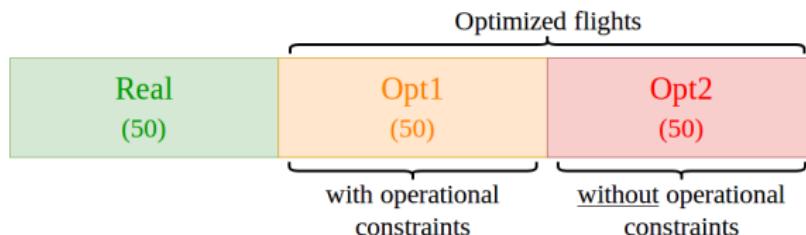
# HOW GOOD IS IT COMPARED TO OTHER METHODS?

- Training set of  $m = 424$  flights  $\simeq 334\ 531$  point observations,
- Test set of 150 flights



# HOW GOOD IS IT COMPARED TO OTHER METHODS?

- Training set of  $m = 424$  flights  $\simeq 334\,531$  point observations,
- Test set of 150 flights

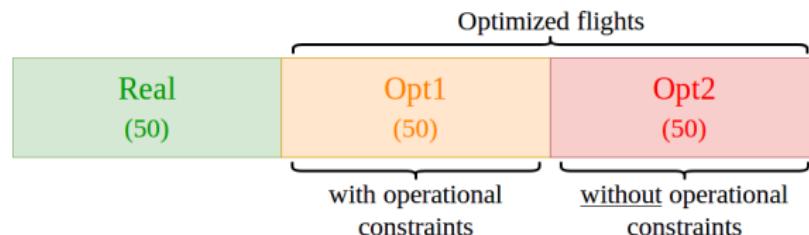


- Discrimination power comparison with (gmm-)FPCA and (integrated) LS-CDE:

VAR.	ESTIMATED LIKELIHOODS		
	REAL	OPT1	OPT2
MML	<b><math>0.63 \pm 0.07</math></b>	<b><math>0.43 \pm 0.08</math></b>	<b><math>0.13 \pm 0.02</math></b>
FPCA	$0.16 \pm 0.12$	$6.4\text{E-}03 \pm 3.8\text{E-}03$	$3.6\text{E-}03 \pm 5.4\text{E-}03$
LS-CDE	$0.77 \pm 0.05$	$0.68 \pm 0.04$	$0.49 \pm 0.06$

# HOW GOOD IS IT COMPARED TO OTHER METHODS?

- Training set of  $m = 424$  flights  $\simeq 334\,531$  point observations,
- Test set of 150 flights



- Discrimination power comparison with (gmm-)FPCA and (integrated) LS-CDE:

VAR.	ESTIMATED LIKELIHOODS			TR. TIME
	REAL	OPT1	OPT2	
MML	<b><math>0.63 \pm 0.07</math></b>	<b><math>0.43 \pm 0.08</math></b>	<b><math>0.13 \pm 0.02</math></b>	5s
FPCA	$0.16 \pm 0.12$	$6.4\text{E-}03 \pm 3.8\text{E-}03$	$3.6\text{E-}03 \pm 5.4\text{E-}03$	20s
LS-CDE	$0.77 \pm 0.05$	$0.68 \pm 0.04$	$0.49 \pm 0.06$	14H

# MML PENALTY

The MML can be used not only to assess the optimization solutions, but also to penalize the optimization itself:

$$\begin{aligned} & \min_{(\mathbf{x}, \mathbf{u}) \in \mathbb{X} \times \mathbb{U}} \int_0^{t_f} C(\mathbf{u}(t), \mathbf{x}(t)) dt \\ \text{s.t. } & \left\{ \begin{array}{l} \dot{\mathbf{x}}(t) = \hat{g}(\mathbf{u}(t), \mathbf{x}(t)), \quad \text{a.e. } t \in [0, t_f], \\ \text{Other constraints...} \end{array} \right. \end{aligned} \quad (\text{AOCP})$$

# MML PENALTY

The MML can be used not only to assess the optimization solutions, but also to penalize the optimization itself:

$$\begin{aligned} & \min_{(\mathbf{x}, \mathbf{u}) \in \mathbb{X} \times \mathbb{U}} \int_0^{t_f} C(\mathbf{u}(t), \mathbf{x}(t)) dt - \lambda \text{MML}(\mathcal{Z}, \mathbf{x}), \\ \text{s.t. } & \left\{ \begin{array}{l} \dot{\mathbf{x}}(t) = \hat{g}(\mathbf{u}(t), \mathbf{x}(t)), \quad \text{a.e. } t \in [0, t_f], \\ \text{Other constraints...} \end{array} \right. \end{aligned} \tag{MML-AOCP}$$

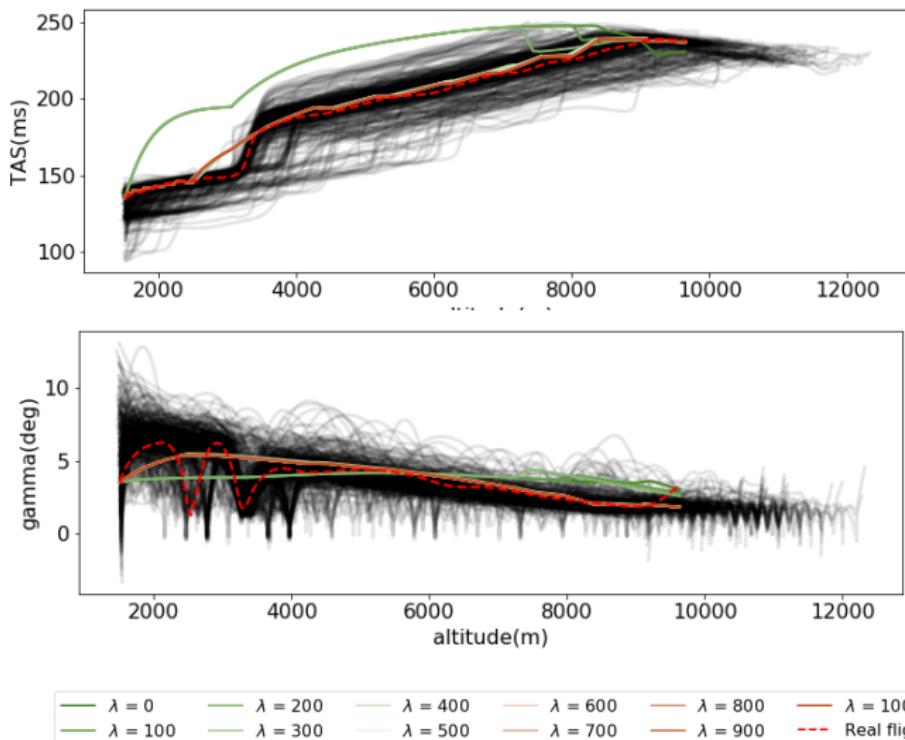
# MML PENALTY

The MML can be used not only to assess the optimization solutions, but also to penalize the optimization itself:

$$\begin{aligned} & \min_{(\mathbf{x}, \mathbf{u}) \in \mathbb{X} \times \mathbb{U}} \int_0^{t_f} C(\mathbf{u}(t), \mathbf{x}(t)) dt - \lambda \text{MML}(\mathcal{Z}, \mathbf{x}), \\ \text{s.t. } & \left\{ \begin{array}{l} \dot{\mathbf{x}}(t) = \hat{g}(\mathbf{u}(t), \mathbf{x}(t)), \quad \text{a.e. } t \in [0, t_f], \\ \text{Other constraints...} \end{array} \right. \end{aligned} \quad (\text{MML-AOCP})$$

- $\lambda$  sets trade-off between a fuel minimization and a likelihood maximization,

# PENALTY EFFECT



# TRAJECTORY ACCEPTABILITY CONCLUSION

- 1 General probabilistic criterion using marginal densities to quantify the closeness between a curve and a set of random trajectories,

# TRAJECTORY ACCEPTABILITY CONCLUSION

- 1 General probabilistic criterion using marginal densities to quantify the closeness between a curve and a set of random trajectories,
- 2 Class of consistent plug-in estimators, based on “histogram” of multivariate density estimators,

# TRAJECTORY ACCEPTABILITY CONCLUSION

- 1 General probabilistic criterion using marginal densities to quantify the closeness between a curve and a set of random trajectories,
- 2 Class of consistent plug-in estimators, based on “histogram” of multivariate density estimators,
- 3 Applicable to the case of aircraft climb trajectories,

# TRAJECTORY ACCEPTABILITY CONCLUSION

- 1 General probabilistic criterion using marginal densities to quantify the closeness between a curve and a set of random trajectories,
- 2 Class of consistent plug-in estimators, based on “histogram” of multivariate density estimators,
- 3 Applicable to the case of aircraft climb trajectories,
  - Competitive with other well-established SOA approaches,

# TRAJECTORY ACCEPTABILITY CONCLUSION

- 1 General probabilistic criterion using marginal densities to quantify the closeness between a curve and a set of random trajectories,
- 2 Class of consistent plug-in estimators, based on “histogram” of multivariate density estimators,
- 3 Applicable to the case of aircraft climb trajectories,
  - Competitive with other well-established SOA approaches,
- 4 Particular Adaptive Kernel and Gaussian mixture implementation,

# TRAJECTORY ACCEPTABILITY CONCLUSION

- 1 General probabilistic criterion using marginal densities to quantify the closeness between a curve and a set of random trajectories,
- 2 Class of consistent plug-in estimators, based on “histogram” of multivariate density estimators,
- 3 Applicable to the case of aircraft climb trajectories,
  - Competitive with other well-established SOA approaches,
- 4 Particular Adaptive Kernel and Gaussian mixture implementation,
  - Showed that it can be used in optimal control problems to obtain solutions close to optimal, and still realistic.

**THANK YOU FOR YOUR ATTENTION**

# REFERENCES

- Anderson, R. M. and May, R. M. (1992). Infectious Diseases of Humans: Dynamics and Control. Oxford university press.
- Bach, F. (2008). Bolasso: model consistent Lasso estimation through the bootstrap. In ICML, pages 33–40.
- Dempster, A. P., Laird, N. M., and Rubin, D. B. (1977). Maximum likelihood from incomplete data via the EM algorithm. JRSS-B, pages 1–38.
- Efron, B., Hastie, T., Johnstone, I., and Tibshirani, R. (2004). Least angle regression. The Annals of Statistics, 32:407–499.
- FitzHugh, R. (1961). Impulses and physiological states in theoretical models of nerve membrane. Biophysical journal, 1(6):445–466.
- Friedman, J., Hastie, T., and Tibshirani, R. (2010). A note on the Group Lasso and a Sparse group Lasso. arXiv:1001.0736.
- Jategaonkar, R. V. (2006). Flight Vehicle System Identification: A Time Domain Methodology. AIAA.
- Mattingly, J. D., Heiser, W. H., and Daley, D. H. (1992). Aircraft Engine Design. University Press.
- Nagumo, J., Arimoto, S., and Yoshizawa, S. (1962). An active pulse transmission line simulating nerve axon. Proceedings of the IRE, 50(10):2061–2070.
- Obozinski, G., Taskar, B., and Jordan, M. I. (2006). Multi-task feature selection. In ICML-06 Workshop on Structural Knowledge Transfer for Machine Learning.
- Roux, E. (2005). Pour une approche analytique de la dynamique du vol. PhD thesis, Supaero.
- Sugiyama, M., Takeuchi, I., Suzuki, T., Kanamori, T., Hachiya, H., and Okanohara, D. (2010). Conditional density estimation via least-squares density ratio estimation. In AISTAT, pages 781–788.
- Tibshirani, R. (1994). Regression shrinkage and selection via the Lasso. JRSS-B, 58:267–288.
- Tikhonov, A. N. (1943). On the stability of inverse problems. In Doklady Akademii Nauk SSSR, volume 39, pages 195–198.
- Yuan, M. and Lin, Y. (2005). Model selection and estimation in regression with grouped variables. JRSS-B, 68:49–67.