AIRCRAFT TRAJECTORY OPTIMIZATION
UNDER UNKNOWN DYNAMICS

C. Rommel$^{1,2}$, J. F. Bonnans$^{1}$,
B. Gregorutti$^{2}$ and P. Martinon$^{1}$

CMAP Ecole Polytechnique - INRIA$^{1}$
Safety Line$^{2}$

PGMODO Days - November 21st 2018
Optimal control and applications session
Motivation - Optimal Control

\[ \dot{x}(t) = g(u(t), x(t)) + \varepsilon(t) \]
Motivation - Optimal Control

\[ \dot{x}(t) = g(u(t), x(t)) + \varepsilon(t) \]

Optimal Control Problem

\[ \min_{(x, u) \in X \times U} \int_0^{t_f} C(u(t), x(t)) \, dt, \]

s.t. \begin{aligned}
\dot{x}(t) &= g(u(t), x(t)) + \varepsilon(t), \quad \text{for a.e. } t \in [0, t_f], \\
\text{Other constraints...} 
\end{aligned} 

(OCP)

Use of past data to learn how to control a system efficiently

"Model-based reinforcement learning" - [Recht, 2018]
Motivation - Optimal Control

\[ \dot{x}(t) = g(u(t), x(t)) + \varepsilon(t) \]

Optimal Control Problem

\[
\min_{(x,u) \in X \times U} \int_0^{t_f} C(u(t), x(t)) \, dt,
\]

s.t. \[
\begin{aligned}
\dot{x}(t) &= g(u(t), x(t)) + \varepsilon(t), \quad \text{for a.e. } t \in [0, t_f], \\
\text{Other constraints...}
\end{aligned}
\]

(OCP)

Use of past data to learn how to control a system efficiently

"Model-based reinforcement learning" - [Recht, 2018]
Motivation - Optimal Control

\[ \dot{x}(t) = g(u(t), x(t)) + \varepsilon(t) \]

Optimal Control Problem

\[ \begin{align*}
\min_{(x,u) \in X \times U} & \quad \int_0^{t_f} C(u(t), x(t)) \, dt, \\
\text{s.t.} & \quad \dot{x}(t) = \hat{g}(u(t), x(t)), \quad \text{for a.e. } t \in [0, t_f], \\
& \quad \text{Other constraints...}
\end{align*} \]  

(OCP)
Motivation - Optimal Control

\[ \dot{x}(t) = g(u(t), x(t)) + \varepsilon(t) \]

Optimal Control Problem

\[
\min_{(x,u) \in X \times U} \int_0^{t_f} C(u(t), x(t)) \, dt,
\]

s.t. \[
\begin{cases}
\dot{x}(t) = \hat{g}(u(t), x(t)), & \text{for a.e. } t \in [0, t_f], \\
\text{Other constraints...}
\end{cases}
\] (OCP)

Use of past data to learn how to control a system efficiently

"Model-based reinforcement learning" - [Recht, 2018]
Motivation - Optimal Control

Optimal Control Problem

\[
\begin{align*}
\dot{x}(t) &= g(u(t), x(t)) + \varepsilon(t) \\
\min_{(x,u)\in X\times U} &\int_0^{t_f} C(u(t), x(t)) dt, \\
\text{s.t.} &\begin{cases}
\dot{x}(t) = \hat{g}(u(t), x(t)), & \text{for a.e. } t \in [0, t_f], \\
\text{Other constraints...}
\end{cases}
\end{align*}
\]

Use of past data to learn how to control a system efficiently

“Model-based reinforcement learning” - [Recht, 2018]
Flight optimization
Flight optimization

\[ \text{CO}_2 \]
Dynamics are learned from QAR data

Black box
Dynamics are learned from QAR data

Black box

Recorded flights = functional data
Trajectory acceptability

\[
\min_{(x,u)\in X \times U} \int_0^{t_f} C(u(t), x(t)) \, dt,
\]

s.t.
\[
\begin{align*}
\dot{x}(t) &= \hat{g}(u(t), x(t)), \quad \text{a.e. } t \in [0, t_f], \\
\text{Other constraints...}
\end{align*}
\]
**TRAJECTORY ACCEPTABILITY**

\[
\begin{align*}
\min_{(x,u)\in \mathbb{X} \times \mathbb{U}} & \int_{0}^{t_f} C(u(t), x(t))\,dt, \\
\text{s.t.} & \quad \dot{x}(t) = \hat{g}(u(t), x(t)), \quad \text{a.e. } t \in [0, t_f], \\
& \quad \text{Other constraints...} \\
\Rightarrow & \quad \hat{z} = (\hat{x}, \hat{u}) \text{ solution of (AOCP).}
\end{align*}
\]
TRAJECTORY ACCEPTABILITY

\[
\min_{(\mathbf{x}, \mathbf{u}) \in \mathbb{X} \times \mathbb{U}} \int_0^{t_f} C(\mathbf{u}(t), \mathbf{x}(t)) \, dt,
\]

\[
\text{s.t. } \begin{cases} 
\dot{\mathbf{x}}(t) = \hat{\mathbf{g}}(\mathbf{u}(t), \mathbf{x}(t)), & \text{a.e. } t \in [0, t_f], \\
\text{Other constraints...} 
\end{cases}
\]

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

\[\square\] Is \(\hat{\mathbf{z}}\) inside the validity region of the dynamics model \(\hat{\mathbf{g}}\) ?
**Trajectory Acceptability**

\[
\min_{(x,u) \in \mathbb{R} \times \mathbb{U}} \int_0^{t_f} C(u(t), x(t)) dt,
\]

\[
\text{s.t. } \begin{cases} 
\dot{x}(t) = \hat{g}(u(t), x(t)), & \text{a.e. } t \in [0, t_f], \\
\text{Other constraints...}
\end{cases}
\]

\[
\Rightarrow \hat{z} = (\hat{x}, \hat{u}) \text{ solution of (AOCP).}
\]

- Is \( \hat{z} \) inside the validity region of the dynamics model \( \hat{g} \)?
- Does it look like a real trajectory?
Trajectory acceptability

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

\[
\Rightarrow \hat{z} = (\hat{x}, \hat{u}) \text{ solution of (AOCP).}
\]

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

Pilots acceptance  Air Traffic Control

1 NATS UK air traffic control
Trajectory acceptability

\[
\min_{(x,u)\in \mathbb{X} \times \mathbb{U}} \int_{0}^{t_f} C(u(t), x(t)) dt, \quad \text{(AOCP)}
\]

s.t. \[
\begin{align*}
\dot{x}(t) &= \hat{g}(u(t), x(t)), & \text{a.e. } t \in [0, t_f], \\
\text{Other constraints...}
\end{align*}
\]

\[\Rightarrow \hat{z} = (\hat{x}, \hat{u}) \text{ solution of (AOCP)}.\]

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

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 $C(\mathbb{T}, E)$, 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 \( Z_t \),
- \( y \) new trajectory,
- \( f_t(y(t)) \) marginal likelihood of \( y \) at \( t \), i.e. likelihood of observing \( Z_t = y(t) \).
How do we aggregate the marginal likelihoods?

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

**Mean marginal likelihood**

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

where $\psi : L^1(E, \mathbb{R}_+) \times \mathbb{R} \to [0; 1]$ is a continuous scaling map,
How do we aggregate the marginal likelihoods?

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

**Mean marginal likelihood**

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

where $\psi : L^1(E, \mathbb{R}_+) \times \mathbb{R} \to [0; 1]$ 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

$$\psi[f_t, y(t)] := \frac{f_t(y(t))}{\max_{z \in E} f_t(z)},$$
How do we aggregate the marginal likelihoods?

Possible scalings are the normalized density

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

or the confidence level

\[
\psi[f_t, y(t)] := \mathbb{P}(f_t(Z_t) \leq f_t(y(t))).
\]
How do we deal with sampled curves?

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

$$
\mathcal{T}^D := \{(t^r_j, z^r_j)\}_{1 \leq j \leq n} \subset \mathbb{T} \times E, \quad z^r_j := z^r(t^r_j), \\
\mathcal{Y} := \{(\tilde{t}_j, y_j)\}_{j=1}^{\tilde{n}} \subset \mathbb{T} \times E, \quad y_j := y(\tilde{t}_j).
$$
**How do we deal with sampled curves?**

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

$$
\mathcal{T}^D := \{(t^r_j, z^r_j)\}_{1 \leq j \leq n} \subset \mathbb{T} \times E, \quad z^r_j := z^r(t^r_j),
$$

$$
\mathcal{Y} := \{ (\tilde{t}_j, y_j) \}_{\tilde{n}} \subset \mathbb{T} \times E, \quad y_j := y(\tilde{t}_j).
$$

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

$$
\text{EMML}_m(\mathcal{T}^D, \mathcal{Y}) := \frac{1}{t_f} \sum_{j=1}^{\tilde{n}} \psi[\hat{f}^m_{\tilde{t}_j}, y_j] \Delta \tilde{t}_j.
$$
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 \(\{t_r^j : j = 1, \ldots, n; r = 1, \ldots, m\}\) 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 \(Z_t = (Z_T | T = t) = (Y | X)\).

We can apply SOA conditional density estimation techniques, such as LS-CDE [Sugiyama et al., 2010], and we can use a fine partitioning of the time domain.
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 \( \{ t_r^j : j = 1, \ldots, n; r = 1, \ldots, m \} \) 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 \( \{ t_j^r : j = 1, \ldots, n; r = 1, \ldots, m \} \) 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 \( \{t^r_j : j = 1, \ldots, n; r = 1, \ldots, m\} \) 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 \( \{ t_j^r : j = 1, \ldots, n; r = 1, \ldots, m \} \) 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.
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:

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

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

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

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

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

**Assumption 1 - Positive time density**
\( \nu \in L^\infty(E, \mathbb{R}_+) \) density function of \( T \), s.t.

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

**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 \( \{B_{m,\ell}^q\}_{\ell=1}^q \) of \([0; t_f]\), with binsize \( b_m \), is s.t.

\[
\lim_{m \to \infty} b_m = 0, \quad \lim_{m \to \infty} mb_m = \infty.
\]
**Consistency**

**Assumption 1 - Positive time density**

\( \nu \in L^\infty(\mathbb{E}, \mathbb{R}_+) \) 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 \((t, z) \in \mathbb{T} \times \mathbb{E} \mapsto f_t(z)\) 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 \( \{B^m_{\ell}\}_{\ell=1}^{q^m} \) of \([0; t_f]\), with binsize \( b_m \), is s.t.

\[
\lim_{m \to \infty} b_m = 0, \quad \lim_{m \to \infty} mb_m = \infty.
\]
**Assumption 4 - i.i.d. consistency**

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

The estimator obtained by applying \( \Theta \) to \( S^N_{\rho} \), denoted by

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

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 \( \rho \in \mathcal{G} \),

\[
N \geq N_{\varepsilon, \alpha_1} \Rightarrow \mathbb{P}\left(\left|\hat{\rho}^N(z) - \rho(z)\right| < \varepsilon\right) > 1 - \alpha_1.
\]
Theorem 1
Under assumptions 1 to 4, for any \( z \in E \) and \( t \in \mathbb{T} \), \( \hat{f}_{\ell m}(t)(z) \) consistently approximates the marginal density \( f_t(z) \) as the number of curves \( m \) grows:

\[
\forall \varepsilon > 0, \quad \lim_{m \to \infty} \mathbb{P} \left( |\hat{f}_{\ell m}(z) - f_t(z)| < \varepsilon \right) = 1.
\]
**Theorem 1**

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

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

**Note that:**

- $m \to \infty \neq N \to \infty$, number of samples = random, training data not i.i.d.
**Consistency**

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

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

**Note that:**

- $m \to \infty \neq N \to \infty$, 
- Number of samples = random,
**Consistency**

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

$$\forall \varepsilon > 0, \lim_{m \to \infty} \mathbb{P} \left( |\hat{f}_m^m(z) - f_t(z)| < \varepsilon \right) = 1.$$

Note that:

- $m \to \infty \neq \mathbb{N} \to \infty$,
- Number of samples = random,
- Training data not i.i.d.
Marginal density estimation results
MARGINAL DENSITY ESTIMATION RESULTS
**How good is it compared to other methods?**

<table>
<thead>
<tr>
<th></th>
<th>Estimated Likelihood</th>
<th>Time</th>
</tr>
</thead>
<tbody>
<tr>
<td>MML</td>
<td>0.63 ± 0.07</td>
<td>0.43 ± 0.08</td>
</tr>
<tr>
<td>FPCA</td>
<td>0.16 ± 0.12</td>
<td>6.4e-03 ± 3.8e-03</td>
</tr>
<tr>
<td>LS-CDE</td>
<td>0.77 ± 0.05</td>
<td>0.68 ± 0.04</td>
</tr>
</tbody>
</table>

Training set of $m = 424$ flights yields approximately $334,531$ point observations.

Test set of 150 flights.
How good is it compared to other methods?

- Training set of $m = 424$ flights $\approx 334\,531$ point observations,
How good is it compared to other methods?

- Training set of \( m = 424 \) flights \( \sim 334 \, 531 \) point observations,
- Test set of 150 flights

<table>
<thead>
<tr>
<th></th>
<th>Var. Estimated Likelihoods</th>
<th>Tr. Time</th>
</tr>
</thead>
<tbody>
<tr>
<td>MML</td>
<td>0.63 ( \pm 0.07 )</td>
<td>5s</td>
</tr>
<tr>
<td>FPCA</td>
<td>0.16 ( \pm 0.12 )</td>
<td>6.4e-03</td>
</tr>
<tr>
<td>LS-CDE</td>
<td>0.77 ( \pm 0.05 )</td>
<td>14h</td>
</tr>
</tbody>
</table>

- Optimized flights

- Real (50) with operational constraints
- Opt1 (50) without operational constraints
- Opt2 (50) without operational constraints
How good is it compared to other methods?

- Training set of \( m = 424 \) flights \( \sim 334,531 \) point observations,
- Test set of 150 flights

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

<table>
<thead>
<tr>
<th>Var.</th>
<th>Estimated Likelihoods</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Real ( \pm )</td>
</tr>
<tr>
<td>MML</td>
<td>0.63 ( \pm ) 0.07</td>
</tr>
<tr>
<td>FPCA</td>
<td>0.16 ( \pm ) 0.12</td>
</tr>
<tr>
<td>LS-CDE</td>
<td>0.77 ( \pm ) 0.05</td>
</tr>
</tbody>
</table>

- Optimized flights
  - Real (50) with operational constraints
  - Opt1 (50) without operational constraints
  - Opt2 (50) without operational constraints
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:

<table>
<thead>
<tr>
<th>Var.</th>
<th>Estimated Likelihoods</th>
<th>Tr. Time</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Real</td>
<td>Opt1</td>
</tr>
<tr>
<td>MML</td>
<td>0.63 $\pm$ 0.07</td>
<td>0.43 $\pm$ 0.08</td>
</tr>
<tr>
<td>FPCA</td>
<td>0.16 $\pm$ 0.12</td>
<td>6.4E-03 $\pm$ 3.8E-03</td>
</tr>
<tr>
<td>LS-CDE</td>
<td>0.77 $\pm$ 0.05</td>
<td>0.68 $\pm$ 0.04</td>
</tr>
</tbody>
</table>
MML penalty

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

\[
\min_{(x,u) \in X \times U} \int_0^{t_f} C(u(t), x(t)) dt \\
\text{s.t. } \begin{cases} 
\dot{x}(t) = \hat{g}(u(t), x(t)), & \text{a.e. } t \in [0, t_f], \\
\text{Other constraints...} \end{cases}
\]
MML penalty

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

\[
\min_{(x,u) \in X \times U} \int_0^{t_f} C(u(t), x(t)) \, dt - \lambda \text{MML}(Z, x),
\]

s.t. \[\begin{align*}
\dot{x}(t) &= \hat{g}(u(t), x(t)), & \text{a.e. } t \in [0, t_f], \\
\text{Other constraints...}
\end{align*}\] (MML-AOCP)
MML penalty

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

$$\min_{(x,u) \in X \times U} \int_0^{t_f} C(u(t), x(t)) dt - \lambda \text{MML}(Z, x),$$

s.t. \( \dot{x}(t) = \hat{g}(u(t), x(t)), \) a.e. \( t \in [0, t_f], \)

Other constraints...

\(\lambda\) 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,
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