Control-Theoretic Data Smoothing

Biswa Dip Dey
P. S. Krishnaprasad

Department of Electrical and Computer Engineering
&
Institute for Systems Research
University of Maryland, College Park
Maryland, USA

53rd IEEE Conference on Decision and Control - 2014
Los Angeles, California, USA

December 17, 2014
**Background and Motivation**

- **Objective:** Given a time series of noisy data, reconstruct/generate a path in the underlying space which traverses through the data points.
  - Curve Reconstruction
  - Quantum Information Processing (Quantum State Traversal) Brody et al., PRL 2012
  - Computer Vision (Curve Completion) Ben-Yosef et al., PAMI 2012

- **Issues:** This inverse problem is ill-posed.
  - Non-unique
  - High sensitivity to noise

- **Our Approach:**
  - Regularized Inversion
    - Introduce a generative model to treat the data points as output from an underlying dynamical system.
    - Impose regularization by adding a penalty term to fit error.
**Background and Motivation**

- **Objective**: Given a time series of noisy data, reconstruct/generate a path in the underlying space which traverses through the data points.
  - Curve Reconstruction
  - Quantum Information Processing (Quantum State Traversal) Brody et al., PRL 2012
  - Computer Vision (Curve Completion) Ben-Yosef et al., PAMI 2012

- **Issues**: This inverse problem is ill-posed.
  - Non-unique
  - High sensitivity to noise
**Objective:** Given a time series of noisy data, reconstruct/generate a path in the underlying space which traverses through the data points.

- Curve Reconstruction
- Quantum Information Processing (Quantum State Traversal) Brody et al., PRL 2012
- Computer Vision (Curve Completion) Ben-Yosef et al., PAMI 2012

**Issues:** This inverse problem is ill-posed.

- Non-unique
- High sensitivity to noise

**Our Approach:** Regularized Inversion

- Introduce a generative model to treat the data points as output from an underlying dynamical system.
- Impose regularization by adding a penalty term to fit error.
Background and Motivation

Measured Data $\{r_i\}_{i=0}^N$ → Regularized Inversion → Smoothened Data $r(\cdot)$

Generative Model

\[ \dot{x}(t) = Ax(t) + Bu(t) \]
or,

\[ \dot{q}(t) = f(t, q(t), u(t)) \]
or,

\[ \dot{g}(t) = g(t)\xi_u(t) \]

Cost

Minimize $\left( \sum_{i=0}^{N} d^2(r(t_i), r_i) + \lambda \int_{t_0}^{t_N} L(u(t)) dt \right)$
Outline

1. Data Smoothing in a Euclidean Setting ($\mathbb{R}^n$)
   - Maximum Principle
   - Sketch of Proof
   - Example Problem

2. Data Smoothing in Matrix Lie-Group Setting ($G$)
   - Maximum Principle
   - Sketch of Proof
   - Lie-Poisson Reduction
   - Example Problem

3. Conclusion
Data Smoothing in a Euclidean Setting ($\mathbb{R}^n$)

- Maximum Principle
- Sketch of Proof
- Example Problem

Data Smoothing in Matrix Lie-Group Setting ($G$)

- Maximum Principle
- Sketch of Proof
- Lie-Poisson Reduction
- Example Problem

Conclusion
**Maximum Principle for Data Smoothing on \( \mathbb{R}^n \)**

**Data Smoothing as a Regularized Inversion**

Minimize \( J(q(t_0), u) = \int_{t_0}^{t_N} L(t, q(t), u(t)) \, dt + \sum_{i=0}^{N} F_i(q(t_i)) \) \hspace{1cm} (1)

subject to: \( \dot{q}(t) = f(t, q(t), u(t)), \quad q : [t_0, t_N] \to \mathbb{R}^n, \quad u \in U = \{ u : [t_0, t_N] \to U \} \)

**PMP for data smoothing (Theorem 2.2)**

Let \( u^* \) be an optimal control input for (1), and \( q^* \) denote the corresponding state trajectory. Then, by defining a pre-Hamiltonian as \( H(t, q, p, u) = \langle p, f(t, q, u) \rangle - L(t, q, u) \), we can show that there exists a costate trajectory \( p : [t_0, t_N] \to \mathbb{R}^n \) such that

\[
\dot{q}^*(t) = \frac{\partial H}{\partial p}(t, q^*(t), p(t), u^*(t)), \quad \dot{p}(t) = -\frac{\partial H}{\partial q}(t, q^*(t), p(t), u^*(t)),
\]

and,

\[ H(t, q^*, p, u^*) = \max_{u \in U} H(t, q^*, p, u) \] \hspace{1cm} (2)

at the points of continuity. Moreover, the penalties on intermediate state yield jump discontinuities given by

\[ p(t_i^+) - p(t_i^-) = \frac{\partial F_i(q(t_i))}{\partial q(t_i)}, \quad i = 0, 1, \ldots, N. \] \hspace{1cm} (4)

Also, boundary values of the costate variables satisfy

\[ p(t_0^-) = p(t_N^+) = 0. \] \hspace{1cm} (5)
Highlights of The Proof

- We introduce a **new state variable**: \( \tilde{q} : [t_0, t_N] \to \mathbb{R} \).

  \[
y(t) \triangleq \begin{pmatrix} \tilde{q}(t) \\ q(t) \end{pmatrix} \in \mathbb{R}^{n+1} \quad \implies \quad \dot{y}(t) = \begin{pmatrix} L(t, q(t), u(t)) \\ f(t, q(t), u(t)) \end{pmatrix}, \quad y(t_i^+) - y(t_i^-) = \begin{pmatrix} F_i(q(t_i)) \\ 0 \end{pmatrix}
\]

  \( \triangleq g(t, y(t), u(t)) \)

- This transforms the problem into the **Mayer form**, as \( J(q(t_0), u) = \tilde{q}(t_N^+) = J(y(t_0), u) \).

- **Perturbed Control** (Needle Variation):

  \[
u_w, I(t) \triangleq \begin{cases} u^*(t) & \text{if } t \notin I \\ w & \text{if } t \in I \end{cases}, \quad w \in U, \quad I = (b - \epsilon a, b) \subset (t_0, t_N), a > 0
\]

- Construct the **perturbed trajectory**, and compute the perturbation in the terminal state \( y(t_N^+) \).

- Construct the **terminal cone** at \( y^*(t_N^+) \), through concatenation of needle variations.
**Example Problem: Trajectory Reconstruction**

**Trajectory Reconstruction**

Minimize \( J(q(t_0), u) = \sum_{i=0}^{N} \| r(t_i) - r_i \|^2 + \lambda \int_{t_0}^{t_N} u^T(t)u(t)dt \)

subject to \( \dot{q}(t) = Aq(t) + Bu(t), \quad r(t) = Cq(t) \)

\( q(t_0) \in \mathbb{R}^9, \quad u \in \mathcal{U} \)

\[ A = \begin{bmatrix} 0 & I_3 & 0 \\ 0 & 0 & I_3 \\ 0 & 0 & 0 \end{bmatrix} \]

\[ B = \begin{bmatrix} 0 \\ 0 \\ I_3 \end{bmatrix} \]

\[ C = \begin{bmatrix} I_3 & 0 & 0 \end{bmatrix} \]

---

**Example Problem: Trajectory Reconstruction**

**Trajectory Reconstruction**

Minimize

\[
J(q(t_0), u) = \sum_{i=0}^{N} \|r(t_i) - r_i\|^2 + \lambda \int_{t_0}^{t_N} u^T(t)u(t)dt
\]

subject to

\[
\dot{q}(t) = Aq(t) + Bu(t), \quad r(t) = Cq(t)
\]

\[
q(t_0) \in \mathbb{R}^9, \quad u \in \mathcal{U}
\]

\[
L(t, q, u) = \lambda u^T u
\]

\[
F_i(q(t_i)) = q(t_i)C^T C q(t_i) - 2q(t_i)C^T r_i + r_i^T r_i
\]

Optimal Control Input

\[
u^*(t) = \frac{1}{2\lambda} B^T p(t)
\]

State-costate Dynamics

\[
\frac{d}{dt} \begin{bmatrix} q^*(t) \\ p(t) \end{bmatrix} = \begin{bmatrix} A & \frac{1}{2\lambda} BB^T \\ 0 & -A^T \end{bmatrix} \begin{bmatrix} q^*(t) \\ p(t) \end{bmatrix}
\]

Boundary Values and Jump Conditions

\[
p(t^-_0) = p(t^+_0) = 0
\]

\[
p(t^+_i) - p(t^-_N) = 2C^T [Cq(t_i) - r_i]
\]

---

Data Smoothing in a Euclidean Setting ($\mathbb{R}^n$)
- Maximum Principle
- Sketch of Proof
- Example Problem

Data Smoothing in Matrix Lie-Group Setting ($G$)
- Maximum Principle
- Sketch of Proof
- Lie-Poisson Reduction
- Example Problem

Conclusion
Maximum Principle for Data Smoothing in a Matrix Lie-Group Setting

Data Smoothing as an Optimal Control Problem on Lie Group (G)

Minimize \( J(g(t_0), u) = \int_{t_0}^{t_N} L(u(t)) dt + \sum_{i=0}^{N} F(g(t_i), g_i) \) \( (7) \)

subject to: \( \dot{g}(t) = g(t)\xi_u(t) = TeLg(t) \cdot \xi_u(t), \quad g : [t_0, t_N] \to G, \quad u \in U = \{ u : [t_0, t_N] \to U \} \)

PMP for data smoothing (Theorem 3.2)

Let \( u^* \) be a solution for the optimal control problem (7). The corresponding state trajectory \( g^* \) is the base integral curve of a Hamiltonian vector field \( X_{H(g^*, p, u^*)} \) on \( T^*G \), where the pre-Hamiltonian is defined as

\[
H(g, p, u) = \langle p, TeLg \cdot \xi_u \rangle - L(u) \quad (8)
\]

and the optimal control input maximizes \( H \), i.e.

\[
H(g^*, p, u^*) = \text{Max}_{u \in U} H(g^*, p, u). \quad (9)
\]

(Observe that the pre-Hamiltonian is \( G \) invariant.) Moreover, data dependency of the cost functional causes jump discontinuities in \( p \), and the corresponding boundary values and jump conditions are given as

\[
p(t_0^-) = p(t_N^+) = 0 \quad (10)
\]

and,

\[
p(t_i^+) - p(t_i^-) = D_{g^*}(t_i)F, \quad i = 0, 1, \ldots, N \quad (11)
\]

where \( D_{g^*}(t_i)F \) represents the Frechet derivative of the fit-error at \( g^*(t_i) \in G \).
**Highlights of The Proof**

- We use a variational approach.
- Express Cost in terms of The Hamiltonian:

\[
J(g(t_0), u) = \int_{t_0}^{t_N} \left( \langle p(t), T_e L_{g(t)} \cdot \xi u(t) \rangle - H(g(t), p(t), u(t)) \right) dt + \sum_{i=0}^{N} F(g(t_i), g_i)
\]

- Perturbed Control:

\[
u_\epsilon = u^* + \epsilon \delta u, \quad \epsilon > 0 \quad \Longrightarrow \quad \xi_\epsilon = \xi_{u^*} + \epsilon \delta \xi u
\]

- Perturbation in State Trajectory:

\[
g_\epsilon = g^* + \epsilon \delta g + O(\epsilon^2), \quad \text{where} \quad \delta g = g^* \delta \xi u
\]

- Invoke first order necessary condition, i.e., \(\delta J(g^*(t_0), u^*) = 0\).

- Invoke second order necessary condition, i.e., \(\delta^2 J(g^*(t_0), u^*) \geq 0\).
A Quick Review of Lie-Poisson Reduction

- **Poincaré 1-form:**
  \[ \Theta_a \left( \xi'(0) \right) = \left\langle \pi(0), T\tilde{\pi}(a)L\tilde{\pi}(a)^{-1} \cdot (T_a\tilde{\pi} \cdot \xi'(0)) \right\rangle \]

- Define a Hamiltonian vector field on \( T^*G \) (\( H_\bullet \)), by exploiting the symplectic form associated with the Poincaré 1-form (\( \omega = -d\Theta \)).

- **Poisson Bracket:**
  \[ \phi, \psi \mapsto \{\phi, \psi\} = \omega(H_\phi, H_\psi) \]
  - \( \phi, \psi \) are smooth (\( C^\infty \)) functions on \( T^*G \)

- **Lie-Poisson Bracket:**
  \[ \pi^* \{ h_1, h_2 \}_g = \{ h_1, h_2 \}_g \circ \pi = \{ \pi^* h_1, \pi^* h_2 \} \]
  - \( \pi^* \): Pullback by \( \pi \)
  - \( h_1, h_2 \) are smooth (\( C^\infty \)) functions on \( g^* \)
Example Problem: Data Smoothing on $SE(2)$

**Dynamics**

\[
\begin{align*}
\dot{x} &= u_1 \cos \theta \\
\dot{y} &= u_1 \sin \theta \\
\dot{\theta} &= u_2
\end{align*}
\]

**Lie-Group Formulation**

\[
\dot{g} = g \xi_u = g(u_2 X_1 + u_1 X_2), \quad g \in SE(2); \quad X_1, X_2 \in \mathfrak{se}(2)
\]

where,

\[
g = \begin{bmatrix}
\cos \theta & -\sin \theta & x \\
\sin \theta & \cos \theta & y \\
0 & 0 & 1
\end{bmatrix}
\]

and,

\[
\begin{align*}
X_1 &= [e_2, -e_1, 0_{3 \times 1}] \\
X_2 &= [0_{3 \times 2}, e_1] \\
X_3 &= [0_{3 \times 2}, e_2]
\end{align*}
\]
Example Problem: Data Smoothing on \( SE(2) \)

**DYNAMICS**

\[
\begin{align*}
\dot{x} &= u_1 \cos \theta \\
\dot{y} &= u_1 \sin \theta \\
\dot{\theta} &= u_2
\end{align*}
\]

\( r = (x, y) \)

**LIE-GROUP FORMULATION**

\[
\dot{g} = g \xi_u = g(u_2 X_1 + u_1 X_2), \quad g \in SE(2); \quad X_1, X_2 \in \mathfrak{se}(2)
\]

where,

\[
g = \begin{bmatrix}
\cos \theta & -\sin \theta & x \\
\sin \theta & \cos \theta & y \\
0 & 0 & 1
\end{bmatrix}
\]

and,

\[
\begin{align*}
X_1 &= \begin{bmatrix} e_2, -e_1, 0_{3 \times 1} \end{bmatrix} \\
X_2 &= \begin{bmatrix} 0_{3 \times 2}, e_1 \end{bmatrix} \\
X_3 &= \begin{bmatrix} 0_{3 \times 2}, e_2 \end{bmatrix}
\end{align*}
\]

- Find a curve \( g : [t_0, t_N] \rightarrow SE(2) \), to traverse through targeted positions \( r_0 \rightarrow r_1 \rightarrow \cdots \rightarrow r_N \)

\[
\text{Minimize} \quad \sum_{i=0}^{N} \left\| r(t_i) - r_i \right\|^2 + \lambda \int_{t_0}^{t_N} (u_1^2 + u_2^2) \, dt
\]

subject to \( g(t_0) \in SE(2), \quad u_1, u_2 \in U, \quad \dot{g} = g(u_2 X_1 + u_1 X_2) \),

\[
L(u) = \lambda(u_1^2 + u_2^2) = \lambda \langle \xi_u, \xi_u \rangle_{\mathfrak{se}(2)}, \quad \text{where} \quad \langle v_1, v_2 \rangle_{\mathfrak{se}(2)} = \text{Tr}(v_1 M v_2^T), \quad M = \text{diag}\{\frac{1}{2}, \frac{1}{2}, 1\}
\]

**Lagrangian**

- **Intermediate State-Cost:** \( F(g(t_i), r_i) = \|Ag(t_i)e_3 - r_i\|^2, \text{ where } A = \begin{bmatrix} e_1 & e_2 \end{bmatrix}^T \)
**Example Problem: Data Smoothing on $SE(2)$**

- **Introduce:**
  \[ \mu = \sum_{i=1}^{3} \mu_i X_i^b \in \mathfrak{se}^* (2), \text{ where } \langle X_i^b, X_j^b \rangle = \begin{cases} 1 & \text{if } i = j \\ 0 & \text{otherwise} \end{cases} \text{ for } i, j \in \{1, 2, 3\} \]

- **pre-Hamiltonian:**
  \[ H(g, p, u) = \langle p, T_e L_g \cdot \xi_u \rangle - L(u) = \langle T_e L^*_g \cdot p, \xi_u \rangle - L(u) = u_2 \mu_1 + u_1 \mu_2 - \lambda(u_1^2 + u_2^2). \]
Example Problem: Data Smoothing on \(SE(2)\) [Contd.]

- **Introduce:** \(\mu = \sum_{i=1}^{3} \mu_i X_i^b \in \mathfrak{se}^*(2)\), where \(\langle X_i, X_j^b \rangle = \begin{cases} 1 & \text{if } i = j \\ 0 & \text{otherwise} \end{cases} \) for \(i, j \in \{1, 2, 3\}\).

- **pre-Hamiltonian:** \(H(g, p, u) = \langle p, T_e L_g \cdot \xi u \rangle - L(u) = \langle T_e L_g^* \cdot p, \xi u \rangle - L(u) = u_2 \mu_1 + u_1 \mu_2 - \lambda (u_1^2 + u_2^2)\).

- **Optimal Control Input**
  \[
  \begin{pmatrix}
  u_1^* \\
  u_2^*
  \end{pmatrix} = \frac{1}{2\lambda} \begin{pmatrix}
  \mu_2 \\
  \mu_1
  \end{pmatrix}
  \]

- **Reduced Costate Dynamics**
  \[
  \begin{pmatrix}
  \dot{\mu}_1 \\
  \dot{\mu}_2 \\
  \dot{\mu}_3
  \end{pmatrix} = \frac{1}{2\lambda} \begin{pmatrix}
  -\mu_2 \mu_3 \\
  \mu_3 \mu_1 \\
  -\mu_1 \mu_2
  \end{pmatrix} \quad t \in (t_k, t_{k+1})
  \]

- **Jump Conditions**
  \(\mu_i(t_k^+) - \mu_i(t_k^-) = \text{Tr} \left( 2g(t_k)^T A T^T [Ag(t_k)e_3 - r_k] e_3^T X_i^T \right) \quad k \in \{0, 1, \cdots, N - 1\}\)

- **Boundary Values**
  \(\mu_i(t_0^-) = \mu_i(t_N^+) = 0\)
Example Problem: Data Smoothing on $SE(2)$  

- **Introduce:** $\mu = \sum_{i=1}^{3} \mu_i X_i^b \in se^*(2)$, where $\langle X_i, X_j^b \rangle = \begin{cases} 1 & \text{if, } i = j \\ 0 & \text{otherwise} \end{cases} \ i, j \in \{1, 2, 3\}$

- **pre-Hamiltonian:** $H(g,p,u) = \langle p, T_e L_g \cdot \xi_u \rangle - L(u) = \langle T_e L_g^* \cdot p, \xi_u \rangle - L(u) = u_2 \mu_1 + u_1 \mu_2 - \lambda (u_1^2 + u_2^2)$.

- **Optimal Control Input**
  \[
  \begin{pmatrix} u_1^* \\ u_2^* \end{pmatrix} = \frac{1}{2\lambda} \begin{pmatrix} \mu_2 \\ \mu_1 \end{pmatrix}
  \]

- **Reduced Costate Dynamics**
  \[
  \begin{pmatrix} \dot{\mu}_1 \\ \dot{\mu}_2 \\ \dot{\mu}_3 \end{pmatrix} = \frac{1}{2\lambda} \begin{pmatrix} -\mu_2 \mu_3 \\ \mu_3 \mu_1 \\ -\mu_1 \mu_2 \end{pmatrix} \quad t \in (t_k, t_{k+1})
  \]

- **Jump Conditions**
  \[
  \mu_i(t_k^+) - \mu_i(t_k^-) = \text{Tr} \left( 2g(t_k)^T A^T [Ag(t_k)e_3 - r_k] e_3^T X_i^T \right) \quad k \in \{0, 1, \cdots, N-1\}
  \]

- **Boundary Values**
  \[
  \mu_i(t_0^-) = \mu_i(t_N^+) = 0
  \]

- **Conserved Quantities**
  \[
  \begin{align*}
  \text{Hamiltonian:} & \quad h = \frac{1}{4\lambda} (\mu_1^2 + \mu_2^2) \\
  \text{Casimir:} & \quad C = \frac{1}{4\lambda} (\mu_2^2 + \mu_3^2)
  \end{align*}
  \]

  \[
  \begin{align*}
  \mu_1(t) & = \pm 2\sqrt{\lambda h} \ C \left( \sqrt{\frac{C}{\lambda}} (t + \phi_k), \sqrt{\frac{h}{C}} \right) \\
  \mu_2(t) & = 2\sqrt{\lambda h} \ S \left( \sqrt{\frac{C}{\lambda}} (t + \phi_k), \sqrt{\frac{h}{C}} \right) \quad t \in (t_k, t_{k+1}) \\
  \mu_3(t) & = \pm 2\sqrt{\lambda C} \ D \left( \sqrt{\frac{C}{\lambda}} (t + \phi_k), \sqrt{\frac{h}{C}} \right)
  \end{align*}
  \]
Data Smoothing in a Euclidean Setting ($\mathbb{R}^n$)
- Maximum Principle
- Sketch of Proof
- Example Problem

Data Smoothing in Matrix Lie-Group Setting ($G$)
- Maximum Principle
- Sketch of Proof
- Lie-Poisson Reduction
- Example Problem

Conclusion
**Conclusion**

**Summary**
- Developed an extended version of the maximum principle to address data smoothing using generative models.
- Results are applicable to problems in both Euclidean and finite dimensional matrix Lie group settings.
- This approach yields solution in a semi-analytical way.

**Future Directions**
- To consider Lagrangians involving higher derivatives of control input.


Thank You !!!