The Input-State Linearization Challenge

Mathematical Details

John Y. Hung

Department of Electrical & Computer Engineering
Auburn University

30 March 2017

Nomenclature

x, y, Z	scalars
$x(y), F(z)$	scalar-valued functions with scalar argument
$\mathbf{x}, \mathbf{y}, \mathbf{z}$	vectors
$T(\mathbf{x})$	scalar-valued function with vector argument
$\mathbf{f}(\mathbf{x}), \mathbf{T}(\mathbf{x})$	vector-fields: vector-valued functions with vector argument
$\mathcal{D}$	a set of vector fields (its span is called a *distribution*)
$\mathbf{A}, \mathbf{F}$	matrices
R^n	vector space of real-valued variables in n dimensions
$\mathcal{L}_{\mathbf{f}} T(\mathbf{x})$	Lie derivation of scalar $T(\mathbf{x})$ with respect to vector $\mathbf{f}$

What is Desired?

Given the process dynamics in terms of state $\mathbf{x} \in \mathrm{R}^n$:

$$\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}) + \mathbf{g}(\mathbf{x})u \tag{1}$$

one desires a coordinate transformation:

$$\mathbf{z} = \mathbf{T}(\mathbf{x})$$

so that process dynamics in terms of new state $\mathbf{z}$ take this form:

$$\dot{\mathbf{z}} = \mathbf{A}\mathbf{z} + \mathbf{b}\beta(\mathbf{z})[u + \alpha(\mathbf{z})]. \tag{2}$$

- functions $\alpha(\mathbf{z}), \beta(\mathbf{z})$ are scalar-valued nonlinearities
- pair $\mathbf{A}, \mathbf{b}\beta(\mathbf{z})$ is a type of *Brunovsky form*

What is a Brunovsky form?

$$\mathbf{A} = \begin{bmatrix} 0 & 1 & 0 & \cdots & 0 \\ 0 & 0 & 1 & \cdots & 0 \\ & & \ddots & \ddots & \\ 0 & & \cdots & 0 & 1 \\ 0 & & \cdots & & 0 \end{bmatrix}, \quad \mathbf{b}\beta(\mathbf{z}) = \begin{bmatrix} 0 \\ 0 \\ \vdots \\ 0 \\ \beta(\mathbf{z}) \end{bmatrix} \tag{3}$$

- Brunovsky form (2), (3) models the process nonlinearities in terms of the *scalar*-valued nonlinear functions $\alpha(\mathbf{z}), \beta(\mathbf{z})$
- Original model (1) has *vector*-valued nonlinearities $\mathbf{f}(\mathbf{x}), \mathbf{g}(\mathbf{x})$
- Brunovsky form is very close to linear controllable canonic form.

Advantage of the Brunovsky form

Easier to choose process input that:

- cancels the process nonlinearities
- introduces a new input variable v

Choose:

$$u = -\alpha(\mathbf{z}) + \frac{1}{\beta(\mathbf{z})} v.$$

Then the closed loop dynamics are in linear controllable canonic form:

$$\dot{\mathbf{z}} = \begin{bmatrix} 0 & 1 & 0 & \cdots & 0 \\ 0 & 0 & 1 & \cdots & 0 \\ & & \ddots & \ddots & \\ 0 & & \cdots & 0 & 1 \\ 0 & & \cdots & & 0 \end{bmatrix} \mathbf{z} + \begin{bmatrix} 0 \\ 0 \\ \vdots \\ 0 \\ 1 \end{bmatrix} v.$$

Studying the transformation

$$\mathbf{z} = \mathbf{T}(\mathbf{x})$$

Expanded:

$$\begin{aligned} z_1 &= T_1(\mathbf{x}) \\ z_2 &= T_2(\mathbf{x}) \\ &\vdots \\ z_n &= T_n(\mathbf{x}) \end{aligned}$$

Structural implications of **A** and **b**

Examining the first row of **A** and **b**:

$$\dot{z}_1 = z_2 \quad \rightarrow \quad z_2 = \dot{z}_1$$

or

$$T_2(\mathbf{x}) = \dot{T}_1(\mathbf{x})\dot{\mathbf{x}}$$

Substituting model for $\dot{\mathbf{x}}$:

$$T_2(\mathbf{x}) = \mathcal{L}_{\mathbf{f}}\, T_1(\mathbf{x}) + \mathcal{L}_{\mathbf{g}}\, T_1(\mathbf{x}) u$$

But $T_2(\mathbf{x})$ is not a function of input u, so $\mathcal{L}_{\mathbf{g}}\, T_1 = 0$, which yields:

$$T_2(\mathbf{x}) = \mathcal{L}_{\mathbf{f}}\, T_1(\mathbf{x}).$$

Examining first $n-1$ rows of $\mathbf{A}$ and $\mathbf{b}$

The first row yields:

$$T_2(\mathbf{x}) = \mathcal{L}_{\mathbf{f}} T_1(\mathbf{x}).$$

By similar argument:

$$\begin{aligned} T_3(\mathbf{x}) &= \mathcal{L}_{\mathbf{f}} T_2(\mathbf{x}) \\ &\vdots \\ T_n(\mathbf{x}) &= \mathcal{L}_{\mathbf{f}} T_{n-1}(\mathbf{x}) \end{aligned}$$

Structure of $\mathbf{A}, \mathbf{b}$ also implies recursion is possible

See the relationship between T_2 and T_1:

$$T_2(\mathbf{x}) = \mathcal{L}_{\mathbf{f}} T_1(\mathbf{x}).$$

Therefore:

$$\begin{aligned} T_3(\mathbf{x}) &= \mathcal{L}_{\mathbf{f}} T_2(\mathbf{x}) = \mathcal{L}_{\mathbf{f}}^2 T_1(\mathbf{x}) \\ &\vdots \\ T_n(\mathbf{x}) &= \mathcal{L}_{\mathbf{f}} T_{n-1}(\mathbf{x}) = \ldots = \mathcal{L}_{\mathbf{f}}^{n-1} T_1(\mathbf{x}) \end{aligned}$$

Summary: Transformation $\mathbf{T}(\mathbf{x})$ is recursively computed from the first function $T_1(\mathbf{x})$!

The Lie derivatives with respect to vector $\mathbf{g}(\mathbf{x})$

Recall this (studying the first row of $\mathbf{A}$ and $\mathbf{b}$):

$$\mathcal{L}_{\mathbf{g}} T_1(\mathbf{x}) = 0.$$

The next $n-2$ rows yield similar results:

$$\begin{aligned} \mathcal{L}_{\mathbf{g}} T_2(\mathbf{x}) &= 0 \\ &\vdots \\ \mathcal{L}_{\mathbf{g}} T_{n-1}(\mathbf{x}) &= 0. \end{aligned}$$

But the last row of $\mathbf{b}\beta(\mathbf{z})$ is nonzero, which implies:

$$\mathcal{L}_{\mathbf{g}} T_n(\mathbf{x}) \neq 0$$

Apply recursion again...

Conditions on $\mathcal{L}_g$:

$$\begin{aligned} \mathcal{L}_{\mathbf{g}} T_1(\mathbf{x}) &= 0 \\ \mathcal{L}_{\mathbf{g}} T_2(\mathbf{x}) &= 0 \\ &\vdots \\ \mathcal{L}_{\mathbf{g}} T_{n-1}(\mathbf{x}) &= 0 \\ \mathcal{L}_{\mathbf{g}} T_n(\mathbf{x}) &\neq 0 \end{aligned}$$

$+$

Recursive algorithm:

$$\begin{aligned} T_2(\mathbf{x}) &= \mathcal{L}_{\mathbf{f}} T_1(\mathbf{x}) \\ T_3(\mathbf{x}) &= \mathcal{L}_{\mathbf{f}}^2 T_1(\mathbf{x}) \\ &\vdots \\ T_n(\mathbf{x}) &= \mathcal{L}_{\mathbf{f}}^{n-1} T_1(\mathbf{x}) \end{aligned}$$

$\Rightarrow$

Conditions on $T_1(\mathbf{x})$:

$$\begin{aligned} \mathcal{L}_{\mathbf{g}} T_1(\mathbf{x}) &= 0 \\ \mathcal{L}_{\mathbf{g}} \mathcal{L}_{\mathbf{f}} T_1(\mathbf{x}) &= 0 \\ &\vdots \\ \mathcal{L}_{\mathbf{g}} \mathcal{L}_{\mathbf{f}}^{n-2} T_1(\mathbf{x}) &= 0 \\ \mathcal{L}_{\mathbf{g}} \mathcal{L}_{\mathbf{f}}^{n-1} T_1(\mathbf{x}) &\neq 0 \end{aligned}$$

Summary of the input-state linearization problem

First, find a $T_1(\mathbf{x})$ satisfying:

$$\begin{aligned} \mathcal{L}_{\mathbf{g}} T_1(\mathbf{x}) &= 0 \\ \mathcal{L}_{\mathbf{g}}\mathcal{L}_{\mathbf{f}} T_1(\mathbf{x}) &= 0 \\ &\vdots \\ \mathcal{L}_{\mathbf{g}}\mathcal{L}_{\mathbf{f}}^{n-2} T_1(\mathbf{x}) &= 0 \\ \mathcal{L}_{\mathbf{g}}\mathcal{L}_{\mathbf{f}}^{n-1} T_1(\mathbf{x}) &\neq 0 \end{aligned}$$

Then, recursively compute $\mathbf{T}(\mathbf{x})$:

$$\begin{aligned} T_2(\mathbf{x}) &= \mathcal{L}_{\mathbf{f}} T_1(\mathbf{x}) \\ T_3(\mathbf{x}) &= \mathcal{L}_{\mathbf{f}}^2 T_1(\mathbf{x}) \\ &\vdots \\ T_n(\mathbf{x}) &= \mathcal{L}_{\mathbf{f}}^{n-1} T_1(\mathbf{x}) \end{aligned}$$

Q: Can $T_1(\mathbf{x})$ be found?
A: Depends on $\mathbf{f}(\mathbf{x})$ and $\mathbf{g}(\mathbf{x})$!