

# Decoupling of a class of underactuated systems based on SMC Methogology

rev. 1

J.A. Ortega

October 20, 2022

## 1 Motivation

Consider the system

$$\{\text{eq:sys}\} \quad \dot{x} = Ax + Bu, \quad x \in \mathbb{R}^n, u \in \mathbb{R}^m \quad (\Sigma)$$

with one-degree of underactuation i.e.,  $n = m + 1$ , and  $A : \mathbb{R}^n \rightarrow \mathbb{R}^n$ ,  $B : \mathbb{R}^m \rightarrow \mathbb{R}^n$  are constants.

Objective: The goal is to design a sliding manifold  $\sigma = Cx$  to stabilize the system  $(\Sigma)$ . As requirement  $\sigma$  needs to be relative degree complete, i.e.,  $n$  with reference to the output  $u$ .

The motivation of the objective is to solve the stabilization of a underactuated system in a simpler manner. The methodology have two main steps: Design a  $n$ -RD sliding manifold  $\sigma$  then, propose a  $n$ -order Continuous Sliding Mode Controller to stabilize it. Note that, if the sliding-manifold has the same order that the system  $(\Sigma)$ , once  $\sigma \equiv 0$  does not exist zero dynamics.

## 2 Decoupling based on SMC

First we define an output of the  $(\Sigma)$  as  $s = Cx$ , s.t,

$$\dot{s} = CAx + CBu.$$

in order to annihilate  $CB$ ,

$$(CB)^\perp \dot{s} = (CB)^\perp CAx + (CB)^\perp CBu.$$

which can be written as

$$\underbrace{\frac{d}{dt} \begin{bmatrix} (CB)^\perp & 0 \\ 0 & 1 \end{bmatrix}}_{J_1} \underbrace{\begin{bmatrix} s \\ \int s d\tau \end{bmatrix}}_{Y_1(t)} = \underbrace{\begin{bmatrix} (CB)^\perp CA \\ C \end{bmatrix}}_{M_1} x(t)$$

We continue computing the derivative of  $\frac{d}{dt} J_1 Y_1(t) = M_1 x(t)$  in the same manner, until rank  $M_k = n$

$$\frac{d^2}{dt^2} J_1 Y_1(t) = M_1 Ax + M_1 Bu,$$

then,

$$\frac{d^2}{dt^2} (M_1 B)^\perp J_1 Y_1(t) = (M_1 B)^\perp M_1 Ax + (M_1 B)^\perp M_1 Bu.$$

$$\frac{d^2}{dt^2} \underbrace{\begin{bmatrix} (M_1 B)^\perp (CB)^\perp & 0 \\ 0 & (M_1 B)^\perp \end{bmatrix}}_{(M_1 B)^\perp J_1} \begin{bmatrix} s \\ \int s d\tau \end{bmatrix} = (M_1 B)^\perp M_1 Ax.$$

That can be rewritten as

$$\frac{d^2}{dt^2} \underbrace{\begin{bmatrix} (M_1 B)^\perp J_1 & 0 \\ 0 & 1 \end{bmatrix}}_{J_2} \begin{bmatrix} s \\ \int s d\tau \\ \int \int s d\tau dy \end{bmatrix} = \underbrace{\begin{bmatrix} (M_1 B)^\perp M_1 A \\ C \end{bmatrix}}_{M_2} x.$$

Following the same procedure we will arrive to a system of  $n$ -order,

$$\frac{d^n}{dt^n} J_{n-1} \underbrace{\begin{bmatrix} s \\ \int s d\tau \\ \vdots \\ \int \cdots \int s d\tau dy \end{bmatrix}}_{Y_{n-1}} = M_{n-1} Ax + M_{n-1} Bu \quad (1) \quad \{\text{eq:compS}\}$$

Notice that by construction  $J_k$ ,  $\forall k = 1, \dots, n-1$  is diagonal and therefore also invertible, so, we can write (1) as,

$$\frac{d}{dt} \begin{bmatrix} s^{(n-1)} \\ \vdots \\ \dot{s} \\ s \end{bmatrix} = J^{-1} M_{n-1} [Ax + Bu] \quad (2)$$

### 3 Furuta Pendulum Stabilization with LTI model

Consider the linearisation of the Furuta pendulum with the matrices

$$A = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 80.3 & -45.8 & -0.93 \\ 0 & 122 & -44.1 & -1.4 \end{bmatrix}, \quad B = \begin{bmatrix} 0 \\ 0 \\ 83.4 \\ 80.3 \end{bmatrix}.$$