# An Introduction to Nonlinear Control

Shankar Sastry

University of California, Berkeley

From Sastry, Chapter 8, Nonlinear Systems

EECS 106B, 206B, Spring 2023

# Outline

# Outline

# Basics

Consider the following nonlinear single input single output systems

$$\begin{aligned} \dot{x} = & \quad f(x) + g(x)u \\ y = & \quad h(x) \end{aligned}$$

Here $x \in \Re^n$ is the state, $u \in \Re$, is the input.
$f(x), g(x) : \Re^n \rightarrow \Re^n$, represent the dynamics, that is they are each vector fields or directions of evolution of $x$, and $h(x) : \Re^n \rightarrow \Re$ is the output function and $y \in \Re$ is the output.
A canonical problem is to find a control law $u$ so that the output tracks a specified function of time $y_d(t), t \in [0, T]$. Let us think about this as a problem of inverting the control system, that is given a desired $y_d(t)$ find the desired $u_d(t)$. To this end differentiate the output $y$ with respect to time to get

$$\begin{aligned} \dot{y}(t) = & \quad \dot{h}(x(t)) \\ = & \quad Dh(x(t))(\dot{x}) \\ = & \quad Dh(x(t))(f(x(t)) + g(x(t))u(t) \\ = & \quad L_f h(x(t)) + L_g h(x(t))u(t) \end{aligned}$$

# Lie Derivatives

Recall that the derivatives of functions are row vectors:

$$Dh(x) = (\frac{dh}{dx_1}, \frac{dh}{dx_2}, \dots, \frac{dh}{dx_n})$$

Thus, $Dh(x)$ is the row vector of first derivatives of $h(x)$ with respect to $x$. $L_f h(x), L_g h(x)$ are called the *Lie derivatives* of the function $h(x)$ along the vector fields (differential equations, or directions) $f(x), g(x)$ respectively and are defined as

$$\begin{aligned} L_f h(x) &:= Dh(x)f(x) \\ L_g h(x) &:= Dh(x)g(x) \end{aligned}$$

Here $L_f h(x)$ refers to the rate of change of $h(x(t))$ along the direction of the flow of $f(x)$. Similarly for $L_g h(x)$. Note that $L_f h(x), L_g h(x)$ are both functions of $x$, that is from $\Re^n \to \Re$.

# Relative Degree One

Collecting the notation we have

$$\dot{y}(t) = L_f h(x(t)) = L_g h(x(t))u(t)$$

If the function $L_g h(x) \neq 0$ for $x$b in a set $U \in \Re^n$, then choosing the control $u(t)$ to be a state feedback control law of the form

$$u(t) = \frac{1}{L_g h(x(t)}(\dot{y}_d(t) - L_f h(x(t)))$$

yields

$$\dot{y}(t) = \dot{y}_d(t)$$

We are almost home then in terms of $y(t)$ tracking $y_d(t)$ except that we cannot guarantee that $y(0) = y_d(0)$, since the initial condition $x(0)$ of the control system may result in $y(0) \neq y_d(0)$. In this event, it is impossible to have $y(t) \equiv y_d(t)$ and the best we can do is to find a way to reduce the output error $e(t) := y(t) - y_d(t)$ to zero asymptotically as $t \to \infty$.

# Asymptotic Tracking for Relative Degree One

To this end we add an extra term to the control which is proportional to the output error $e(t)$ as follows with $\alpha_1 \in \Re$.

$$u(t) = \frac{1}{L_g h(x(t)}(\dot{y}_d(t) - L_f h(x(t)) - \alpha_1 e(t))$$

Using this control law gives us

$$\begin{aligned} \dot{y} = & \quad \dot{y}_d(t) - \alpha_1 e(t) \\ \dot{e} + \alpha_1 e = & \quad 0 \end{aligned}$$

Amazingly, so long as $\alpha_1 > 0$ this control law results in $e(t) \to 0$ as $t \to \infty$ regardless of the initial state. Better yet, you can control the rate of convergence to zero through the maginitude of $\alpha_1$.

# Relative Degree Two

In the event that $L_g h(x) \equiv 0$ for $x \in U$, we see that $\dot{y}(t) = L_f h(x(t))$, does not depend on the input $u$. In this case, we keep going with the differentiation as follows:

$$\begin{aligned} \ddot{y}(t) = & \quad \frac{d}{dt} L_f h(x(t)) \\ = & \quad L_f(L_f h(x(t)) + L_g L_f h(x(t)) u(t) \\ = & \quad L_f^2 h(x(t)) + L_g L_f h(x(t)) u(t) \end{aligned}$$

Here we have introduced the notation $L_f^2 h(x) := L_f(L_f h(x)) : \Re^n \to \Re$. Now if $L_g L_f h(x) \neq 0$ for $x \in U$ it follows that the control law,

$$u(t) = \frac{1}{L_g L_f h(x(t)}(-L_f^2 h(x(t)) + \ddot{y}_d(t))$$

yields

$$\ddot{y}(t) = \ddot{y}_d(t)$$

# Tracking for Relative Degree Two

To allow for the possibility that the initial state $x(0)$ does not yield $y(0) = y_d(0), \dot{y}(0) = \dot{y}_d(0)$, we modify the control law above to

$$u(t) = \frac{1}{L_g L_f h(x(t)}(-L_f^2 h(x(t)) + \ddot{y}_d(t) - \alpha_2 \dot{e} - \alpha_1 e)$$

to yield

$$\ddot{e} + \alpha_2 \dot{e} + \alpha_1 \dot{e} = 0$$

Once again, we can choose the proportional and derivative feedback gains $\alpha_1, \alpha_2$ respectively to drive $e, \dot{e}$ to 0 at a desired rate. It is worth emphasizing that the control law is a state feedback law by rewriting it as

$$\begin{aligned} u(t) = \quad & \tfrac{1}{L_g L_f h(x(t)}(-L_f^2 h(x(t)) \\ & + \ddot{y}_d(t) - \alpha_2(h(x(t)) - \dot{y}_d) - \alpha_1(h(x(t)) - y_d)) \end{aligned}$$

# Higher Relative Degree

If in addition to $L_g h(x) \equiv 0$, we also have $L_g L_f h(x) \equiv 0$, keep differentiating the output till the input appears on the right hand side. Thus, let $r$ be the smallest integer such that for $x \in U$

$$L_g h(x) = L_g L_f h(x) = \ldots = L_g L_f^{r-2} h(x) \equiv 0, L_g L_f^{(r-1)} h(x) \neq 0$$

Then, it follows that

$$y^{(r)}(t) = L_f^r h(x(t)) + L_g L_f^{(r-1)} h(x) u$$

and the control law

$$\begin{aligned} u(t) = \quad & \frac{1}{L_g L_f^{r-1} hx(t)} (y_d^r(t) - L_f^r h(x) \\ & -\alpha_r (L_f^{(r-1)} h(x(t)) - y_d^{(r-1)}(t)) - \cdots - \alpha_1 (h(x(t)) - y_d(t))) \end{aligned}$$

yields the following equation for the output error

$$e^r(t) + \alpha_r e^{(r-1)}(t) + \cdots + \alpha_1 e(t) = 0$$

Once again, it is possible to control the speed of convergence to 0 of the output error regardless of the initial condition.

# Outline

Control of single input single output systems

Input Output Linearization

Full State Feedback Linearization

Multi-Input Multi-Output Systems

Control of Quadrotor UAVs

# Input Output Feedback Linearization

It is indeed quite miraculous, that if a system has relative degree $r$ that it is possible to make the output of the control system track a(ny) desired trajectory. The only proviso is that the tracking is asymptotic in $t$, though the rate of convergence can be sped up through choice of the constants $\alpha_i, i = 1, \ldots, r$ ($r \leqslant n$ under some modest technical conditions on $f, g$). This is referred to as input output feedback linearization since the closed loop system is linearized. For instance if $y_d(t) \equiv 0$, then we have

$$y^r(t) + \alpha_r y^{r-1}(t) + \cdots + \alpha_1 y(t) = 0$$

# Input Output Linearization

This input output linearization is "exact" meaning to say that if $f, g, h$ are known as functions of $x$ then we can make the output equation exactly linear! This is different from several other options such as Jacobian linearization, Poincare linearization, Carlemann linearization, etc. For example if $x_0 \in U$ is an equilibrium point, that is $f(x_0) = 0$, then the Jacobian linearization is

$$\begin{aligned} \dot{z} = &\quad Az + bu \\ y = &\quad cz \end{aligned}$$

with $A = Df(x_0) \in \Re^{n \times n}, b = g(x(0)) \in \Re^n, c = Dh(x(0))$.
Two issues still need to be answered:

- What about the remaining $(n - r)$ states?
- What if $f, g, h$ were not known exactly?

We will start with the first question.

# Normal Form

Let $\xi_i, i = 1, \ldots, r$ be the output and its $r-1$ derivatives:

$$\xi_1 = h(x), \xi_2 = L_f h(x), \ldots, \xi_r = L_f^{r-1} h(x)$$

It may be shown (Chapter 8, Sastry 1999) that the $\xi_i, i = 1, \ldots, r)$ are independent, and further $(n-r)$ additional coordinates $\eta_i, i = 1, \ldots (n-r)$ can be chosen to be a smooth, invertible transformation of coordinates $\Phi : x \in \Re^n \to (\xi, \eta) \in \Re^n$. In these coordinates we have

$$\begin{aligned}
\dot{\xi}_1 = & \quad \xi_2 \\
\dot{\xi}_2 = & \quad \xi_3 \\
& \vdots \\
\dot{\xi}_r = & \quad L_f^r h(\Phi^{-1}(\xi, \eta)) + L_g L_f^h (\Phi^{-1}(\xi, \eta)) u \\
\dot{\eta}_1 = & \quad L_f \eta_1 (\Phi^{-1}(\xi, \eta)) + L_g \eta_1 (\Phi_{-1}(\xi \eta)) u \\
& \vdots \\
\dot{\eta}_{n-r} = & \quad L_f \eta_{n-r} (\Phi^{-1}(\xi, \eta)) + L_g \eta_{n-r} (\Phi_{-1}(\xi \eta)) u
\end{aligned}$$

# Output Zeroing Control

This equation in more succinct form is

$$
\begin{aligned}
\dot{\xi}_1 = &\quad \xi_2 \\
\dot{\xi}_2 = &\quad \xi_3 \\
&\vdots \\
\dot{\xi}_r = &\quad b(\xi,\eta) + a(\xi,\eta)u \\
\dot{\eta}_1 = &\quad q_1(\xi,\eta) + p_1(\xi\eta))u \\
&\vdots \\
\dot{\eta}_{n-r} = &\quad q_{n-r}(\xi,\eta)) + p_{n-r}(\xi\eta))u
\end{aligned}
$$

Choose the feedback linearizing control law for zeroing the output $y_d(t) \equiv 0$, we have the particularly pleasing form

$$u(t) = -\frac{1}{a(\xi,\eta)} b(\xi,\eta)$$

# Output Zeroing Dynamics

Under the output zeroing control the closed loop system has the form

$$
\begin{aligned}
\dot{\xi}_1 &= \xi_2 \\
\dot{\xi}_2 &= \xi_3 \\
&\vdots \\
\dot{\xi}_r &= 0 \\
\dot{\eta} &= q(\xi,\eta) - p(\xi,\eta)\frac{b(\xi,\eta)}{a(\xi,\eta)} \\
y &= \xi_1
\end{aligned}
$$

This has a chain of integrators in $\xi_1$ variables. If they are start at $\xi_i(0) = 0$ they will continue as $\xi_i(t) \equiv 0$. While the $\eta$ variables are influenced by the $\xi$ variables, the converse is not true. The output $y = \xi_1$ in particular is unaffected by the $\eta_i$.

# Zero Dynamics: Minimum Phase

The zero dynamics are the dynamics consistent with the output held identically zero. Thus $y(t) = \xi_1(t) \equiv 0$ implies that $\xi_i(t) \equiv 0, i = 2, \ldots, r$, and the residual dynamics are

$$\dot{\eta} = q(0,\eta) - p(0,\eta)\frac{b(0,\eta)}{a(0,\eta)}$$

$\eta = 0 \in \Re^{n-r}$ is an equilibrium point of these dynamics. If $\eta = 0$ is a stable equilibrium point of this nonlinear system, the control system is said to be minimum phase. If the equilibrium point is unstable the system is said to be non-minimum phase.

# Bounded tracking

Back to tracking a signal $y_d(t)$ which along with its derivatives $\dot{y}_d(t), \ddot{y}_d(t), \ldots, y_d^{(r)}(t)$ is bounded. The control law

$$
\begin{aligned}
u(t) = \quad & \tfrac{1}{L_g L_f^{r-1} h x(t)} (y_d^r(t) - L_f^r h(x) \\
& - \alpha_r (L_f^{(r-1)} h(x(t)) - y_d^{(r-1)}(t)) - \cdots - \alpha_1 (h(x(t)) - y_d(t)))
\end{aligned}
$$

It is surprising that even if the zero dynamics, that is the $\eta$ variables are unstable, that the output tracks $y_d(t)$ asymptotically. Using some Lyapunov analysis (see Ch 8 of Sastry 1999) it can be seen that if the equilibrium 0 of the zero dynamics is *exponentially stable*, then if $y_d(t)$ and its first $r$ derivatives are bounded, that all the state variables $x(t)$ are bounded. If one is unconcerned about the $\eta$ part of the state space, no assumptions on minimum phase are needed.

# Inversion and Feedback Linearization

We set ourselves the control objective of tracking a desired output $y_d(t)$. Thus, it should come as no surprise that the control law we have derived "inverts" the model. Thus, if the model were non-minimum phase, the inverse would be unstable. These are the zero dynamics. However, by our artifact of rendering these variables unobservable they do not affect the output tracking. However, from a practical standpoint, having unstable zero dynamics can be undesirable. Chapter 9 of Sastry 1999 gives examples of the behavior of non-minimum phase control systems and how to modify the controller in some instances of ”slightly non-minimum phase” systems.

For general non-minimum phase systems there are limitations on how accurate tracking can be even asymptotically! We will explore this later!

# Outline

# Exact Feedback Linearization

If the relative degree $r = n$, the dimension of the state space $x$, then the control law

$$\begin{aligned} u(t) = \; & \frac{1}{L_g L_f^{n-1} h x(t)} (y_d^n(t) - L_f^n h(x) \\ & -\alpha_n (L_f^{(n-1)} h(x(t)) - y_d^{(n-1)}(t)) - \cdots - \alpha_1 (h(x(t)) - y_d(t))) \end{aligned}$$

results in a closed loop system

$$y^n(t) - y_d^n(t) + \alpha_n (y^{n-1}(t) - y_d^{n-1}(t) + \cdots + \alpha_1 (y(t) - y_d(t)) = 0$$

In terms of the normal form thus there are no $\eta$ variables. This is referred to as full state feedback linearization. There exists a state feedback $u = \frac{1}{L_g L_f^{n-1} h(x)} (-L_f^n h(x) + v)$ and a change of coordinates $\xi = \Phi(x)$, so that the closed loop system is completely linear!

$$\begin{aligned} \dot{\xi}_1 = \; & \xi_2 \\ \dot{\xi}_2 = \; & \xi_3 \\ & \vdots \\ \dot{\xi}_n = \; & v \\ y = \; & \xi_1 \end{aligned}$$

# Outline

# Two-Input Two-Output (TITO) Systems

Consider the TITO system

$$\begin{aligned} \dot{x} = &\quad f(x) + g_1(x)u_1 + g_2(x)u_2 \\ y_1 = &\quad h_1(x) \\ y_2 = &\quad h_2(x) \end{aligned}$$

Let $r_1$ be the smallest integer such that $y_1(t)$ needs to be differentiated $r_1$ times before one of the inputs appears. That is for $x \in U$

$$\begin{aligned} & L_{g_1} h_1(x) = L_{g_1} L_f h_1(x) = \cdots = L_{g_1} L_f^{r_1-2} h_1 \equiv 0 \\ & L_{g_2} h_1(x) = L_{g_2} L_f h_1(x) = \cdots = L_{g_2} L_f^{r_1-2} h_1 \equiv 0 \\ & L_{g_1} L_f^{r_1-1} h_1(x) \neq 0 \text{or} L_{g_2} L_f^{r_1-1} h_1(x) \neq 0 \end{aligned}$$

A similar definition holds for $r_2$ for the second output $h_2$.

# Vector Relative Degree

The TITO system is said to have *vector relative degree* $r_1, r_2$ if the matrix multiplying the two inputs is invertible:

$$\begin{bmatrix} y_1^{r_1} \\ y_2^{r_2} \end{bmatrix} = \begin{bmatrix} L_f^{r_1} h_1 \\ L_f^{r_2} h_2 \end{bmatrix} + \begin{bmatrix} L_{g_1} L_f^{r_1-1} h_1 & L_{g_2} L_f^{r_1-1} h_1 \\ L_{g_1} L_f^{r_2-1} h_2 & L_{g_2} L_f^{r_2-1} h_2 \end{bmatrix} \begin{bmatrix} u_1 \\ u_2 \end{bmatrix}$$

The matrix multiplying the control inputs $A(x) \in \Re^{2\times 2}$ is referred to as the decoupling matrix. Rewriting the preceding equation as

$$\begin{bmatrix} y_1^{r_1} \\ y_2^{r_2} \end{bmatrix} = b(x) + A(x) \begin{bmatrix} u_1 \\ u_2 \end{bmatrix}$$

shows that the control law

$$u = A^{-1}(x)(-b(x) + v)$$

decouples and linearizes the system from the new inputs $v$ to $y$ resulting in the closed loop system

$$\begin{bmatrix} y_1^{r_1} \\ y_2^{r_2} \end{bmatrix} = \begin{bmatrix} v_1 \\ v_2 \end{bmatrix}$$

# Normal Form for a TITO system

It can be verified that the coordinates

$$\xi_1^1 = h_1(x), \xi_2^1 = L_f h_1(x), \ldots, \xi_{r_1}^1 = L_f^{r_1-1} h_1(x),$$
$$\xi_1^2 = h_2(x), \xi_2^2 = L_f h_2(x), \ldots, \xi_{r_2}^2 = L_f^{r_2-1} h_2(x)$$

are independent and can be completed with $\eta \in \Re^{n-r_1-r_2}$ to yield

$$\begin{array}{lll} \dot{\xi}_1^1 = \xi_2^1 & \dot{x}i_2^1 = \xi_3^1 & \ldots \dot{\xi}_{r_1}^1 = v_1 \\ \dot{\xi}_1^2 = \xi_2^2 & \dot{\xi}_2^2 = \xi_3^2 & \ldots \dot{\xi}_{r_1}^2 = v_2 \\ \dot{\eta} = q(\xi, \eta) + P(\xi, \eta) v & & \end{array}$$

Here $q(\xi, \eta) \in \Re^{n-r_1-r_2}, P(\xi, \eta) \in \Re^{n-r_1-r_2 \times 2}$ and the zero dynamics are

$$\dot{\eta} = q(0, \eta)$$

# Dynamic Extension

One additional case that needs to be considered is when the so-called decoupling matrix $A(x)$ is singular, that is, it has rank 1. When this happens we have to resort to a trick called dynamic extension: Choose matrix $\beta(x) \in \Re^{2\times 2}$ so as to compress the columns, that is

$$A^1(x) = A(x)\beta(x) = [a_1^1(x)0]$$

with $a_1^1(x) \in \Re^2$. Define new inputs

$$u^1 = \beta^{-1}(x)u$$

It is easy to see that the decoupling matrix for the TITO from $v$ to $y$ is $A_1^1(x)$. Now make $u_1^1$ a state variable $x_{n+1}$ and augment the state space equations by

$$\dot{x}_{n+1} = \dot{u}_i^1 = v_1$$

and define $v_2 := u_2^1$

# Dynamic Extension Algorithm

With the extended control system with state space $x \in \Re^{n+1}$ and new inputs $v_1, v_2$, we have

$$
\begin{aligned}
\dot{x} = & \quad f(x) + g(x)\beta(x) \begin{bmatrix} x_{n+1} \\ v_2 \end{bmatrix} \\
\dot{x}_{n+1} = & \quad v_1 \\
y_1 = & \quad h_1(x) \\
y_2 = & \quad h_2(x)
\end{aligned}
$$

Repeat the procedure of differentiating till the inputs show up at integers $\tilde{r}_1, \tilde{r}_2^1$, and checking if the new decopling matrix is non-singular. If it is not continue with the column compression and dynamic extension procedure. Under some mild conditions (roughly the "invertibility of the original nonlinear control system" – see Sastry 1993) the procedure will converge and the augmented nonlinear system has vector relative degree and the normal form for that system can be derived. You will see this in action in a UAV example soon.

# MIMO Systems

There are no changes in generalizing the TITO discussion to a Mutii-Input Multi-Output System provided they are *square*, that is the number of inputs $n_i = n_o$. Then the MIMO system is said to have vector relative degree $r_1, r_2, \ldots, r_{n_i}$ if the decoupling matrix $A \in \Re^{n_i \times n_i}$ with entries

$$A_{ij}(x) = L_{g_i} L_f^{r_j - 1} h_j$$

is invertible. As in the TITO case, if $A(x)$ is singular, we proceed with the dynamic extension algorithm till the augmented control system gets vector relative degree.

Also, if the sum of the relative degrees (or extended relative degrees)

$$n = r_1 + \cdots + r_{n_i}$$

then the system is full state linearizable by state feedback.

# Outline

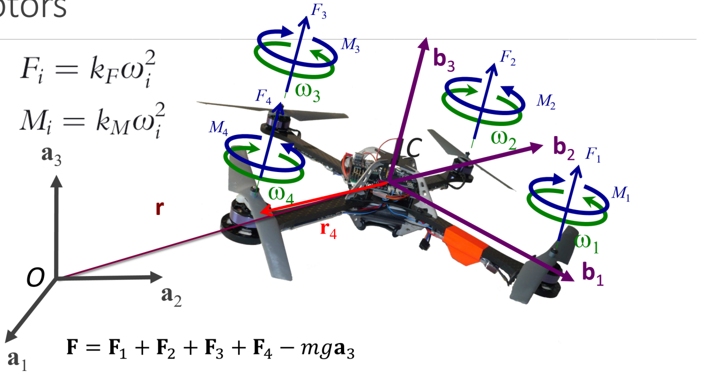
# A Simple Quadrotor

From **Prof. Vijay Kumar and Dr. James Paulos**
Lecture notes MEAM 620, University of Pennsylvania, Spring Term 2020.

## Quadrotors

$$F_i = k_F \omega_i^2$$

$$M_i = k_M \omega_i^2$$




$$\mathbf{F} = \mathbf{F}_1 + \mathbf{F}_2 + \mathbf{F}_3 + \mathbf{F}_4 - mg\mathbf{a}_3$$

$$\mathbf{M} = (\mathbf{r}_1 \times \mathbf{F}_1 + \mathbf{r}_2 \times \mathbf{F}_2 + \mathbf{r}_3 \times \mathbf{F}_3 + \mathbf{r}_4 \times \mathbf{F}_4) + (\mathbf{M}_1 + \mathbf{M}_2 + \mathbf{M}_3 + \mathbf{M}_4)$$

# Quadrotor Dynamics

From **Prof. Vijay Kumar and Dr. James Paulos**
Lecture notes MEAM 620, University of Pennsylvania, Spring Term 2020.

$$ {}^{A}\boldsymbol{\omega}^{B} = p\,\mathbf{b}_1 + q\,\mathbf{b}_2 + r\,\mathbf{b}_3 $$

$$ m\ddot{\mathbf{r}} = \begin{bmatrix} 0 \\ 0 \\ -mg \end{bmatrix} + R \begin{bmatrix} 0 \\ 0 \\ \underbrace{\boxed{F_1 + F_2 + F_3 + F_4}}_{u_1} \end{bmatrix} $$

In inertial frame

$$ I \begin{bmatrix} \dot{p} \\ \dot{q} \\ \dot{r} \end{bmatrix} = \underbrace{\boxed{\begin{bmatrix} L(F_2 - F_4) \\ L(F_3 - F_1) \\ M_1 - M_2 + M_3 - M_4 \end{bmatrix}}}_{\mathbf{u}_2} - \begin{bmatrix} p \\ q \\ r \end{bmatrix} \times I \begin{bmatrix} p \\ q \\ r \end{bmatrix} $$

In body frame

# Planar Quadrotor Control System

With state variables $y = x_1, \dot{y} = x_2, z = x_3, \dot{z} = x_4, \phi = x_5, \dot{\phi} = x_6$, the quadrotor control system is given by

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \\ \dot{x}_3 \\ \dot{x}_4 \\ \dot{x}_5 \\ \dot{x}_6 \end{bmatrix} = \begin{bmatrix} x_2 \\ 0 \\ x_4 \\ -g \\ 0 \\ 0 \end{bmatrix} + \begin{bmatrix} 0 & 0 \\ -\frac{1}{m}\sin x_5 & 0 \\ 0 & 0 \\ \frac{1}{m}\cos x_5 & 0 \\ 0 & 0 \\ 0 & \frac{1}{I_{xx}} \end{bmatrix} \begin{bmatrix} u_1 \\ u_2 \end{bmatrix}$$

with outputs

$$\begin{aligned} y_1 &= x_1 \\ y_2 &= x_3 \end{aligned}$$

# Planar Quadrotor Linearization

Diffferentiating the outputs till the inputs appear yields

$$\begin{bmatrix} \ddot{y}_1 \\ \ddot{y}_2 \end{bmatrix} = \begin{bmatrix} 0 \\ -g \end{bmatrix} + \begin{bmatrix} -\frac{1}{m}\sin x_5 & 0 \\ \frac{1}{m}\cos x_5 & 0 \end{bmatrix} \begin{bmatrix} u_1 \\ u_2 \end{bmatrix}$$

The matrix multilying the inputs is singular: the second column is all zeros. It appears that $u_1$ has shown up before $u_2$ (too soon!). A trick to slow down the appearance of $u_1$ is to first set $u_1 = x_7, \dot{u}_1 = v_1$, with the new input $v_1$. This now yields

$$\begin{bmatrix} y_1^{(3)} \\ y_2^{(3)} \end{bmatrix} = \begin{bmatrix} -\frac{1}{m}\cos x_5 x_6 \\ -\frac{1}{m}\cos x_5 x_6 \end{bmatrix} = \begin{bmatrix} -\frac{1}{m}\sin x_5 & 0 \\ \frac{1}{m}\cos x_5 & 0 \end{bmatrix} \begin{bmatrix} v_1 \\ u_2 \end{bmatrix}$$

# Linearization with dynamic extension

The matrix multiplying the inouts $v_1, u_2$ is still singular, so we set $v_1 = x_8, \dot{v}_1 = w_1$, the new input. Now we get

$$\begin{bmatrix} y_1^{(4)} \\ y_2^{(4)} \end{bmatrix} = \begin{bmatrix} \frac{1}{m}\sin x_5 x_6^2 \\ \frac{1}{m}\sin x_5 x_6 \end{bmatrix} = \begin{bmatrix} -\frac{1}{m}\sin x_5 & -\frac{1}{m}\cos x_5 \\ \frac{1}{m}\cos x_5 & -\frac{1}{m}\sin x_5 \end{bmatrix} \begin{bmatrix} w_1 \\ u_2 \end{bmatrix}$$

With respect to the augmented system with $x \in \Re^8$ and the new inputs $w_1, u_2$ the quadrotor control system has vector relative degree of (4,4). Thus, the augmented control system can be full state linearized and decoupled from $w_1, u_2$ to $y_1, y_2$! The only price to be paid is slowing down the input $u_1$ by two integrators.

Thank you for your attention. Questions?

Shankar Sastry
sastry@coe.berkeley.edu