①SLS

ECE 664 Project
Dept. of ECE
University of Alberta
Due: See below for due dates

扔;投掷

The project considers the nonlinear motion control of a quadrotor UAV slung load system (SLS) as shown in Figures 1 and 2. This system is currently receiving attention in the literature and has a number of benefits for transporting payloads relative to other approaches such as grippers [12]. The objective of this project is to investigate feedback linearizability of the SLS. I am unaware of any references on verifying these conditions in the literature. However, you are welcome and encouraged to check the literature. The project should be performed in groups of one or two. Please let me know if you need help finding a group.

刚体
动力学

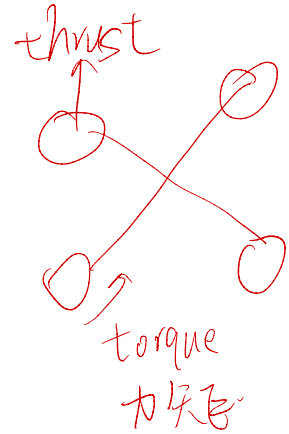
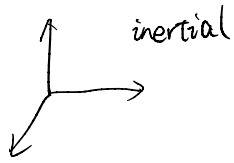


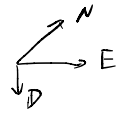
Figure 1: Hexrotor with slung load [6]. Note how the pendulum attaches to the UAV. This mechanism measures the angle of the load.

1 Modelling

1.1 Rotational Dynamics of the UAV

We choose a commonly used nonlinear rigid body model for the UAV and follows the presentation in [13]. (This model ignores drag forces on the vehicle, motor/propeller dynamics, and rotor gyroscopic effects.) Further details on modelling which includes such effects can be found in [1, 2] and other references. In particular, rotor drag forces 转子阻力 are recently of interest in the literature [1, 8] which could provide a way to extend this project.
The two reference frames used in modelling are the navigation frame $\mathcal{N}$ and the body frame $\mathcal{B}$. The navigation frame is assumed inertial and has its origin fixed (say) to the centre of





the lab. Its basis is the orthonormal set of vectors $\{n_1, n_2, n_3\}$ that are oriented north, east, and down, respectively. The body frame has its origin fixed to the quadrotor's CoM (centre of mass) and its basis $\{b_1, b_2, b_3\}$ is oriented forward, right, and down, respectively. Figure 2 shows frames $\mathcal{N}$ and $\mathcal{B}$. The configuration of the quadrotor belongs to the special Euclidean group SE(3), and includes the position $p \in \mathbb{R}^3$ of the origin of $\mathcal{B}$ relative to $\mathcal{N}$ and expressed in $\mathcal{N}$, and the orientation $R = R_b^n \in \mathrm{SO}(3) = \{R \in \mathbb{R}^{3\times 3} : RR^T = I, \det(R) = 1\}$ of $\mathcal{B}$ with respect to $\mathcal{N}$. As mentioned in Problem Set 1, the columns of $R = R_b^n$ are basis vectors of $\mathcal{B}$ expressed in $\mathcal{N}$:

$p^n = R_b^n \cdot p^b$

$$R = R_b^n = \begin{bmatrix} n_1 \cdot b_1 & n_1 \cdot b_2 & n_1 \cdot b_3 \\ n_2 \cdot b_1 & n_2 \cdot b_2 & n_2 \cdot b_3 \\ n_3 \cdot b_1 & n_3 \cdot b_2 & n_3 \cdot b_3 \end{bmatrix} = \begin{bmatrix} b_1^n & b_2^n & b_3^n \end{bmatrix}$$

where $a \cdot b, a, b \in \mathbb{R}^3$ is the dot product, and $b_k^n = \sum_{j=1}^{3} (n_j \cdot b_k) n_j$ denotes the representation of b_k in $\mathcal{N}$. We define the elementary rotation matrices by

$$R_1(\phi) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \mathrm{c}_\phi & -\mathrm{s}_\phi \\ 0 & \mathrm{s}_\phi & \mathrm{c}_\phi \end{bmatrix}, \quad R_2(\theta) = \begin{bmatrix} \mathrm{c}_\theta & 0 & \mathrm{s}_\theta \\ 0 & 1 & 0 \\ -\mathrm{s}_\theta & 0 & \mathrm{c}_\theta \end{bmatrix}, \quad R_3(\psi) = \begin{bmatrix} \mathrm{c}_\psi & -\mathrm{s}_\psi & 0 \\ \mathrm{s}_\psi & \mathrm{c}_\psi & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

where $\mathrm{c}_\theta = \cos\theta$ and $\mathrm{s}_\theta = \sin\theta$. We use roll-pitch-yaw (ϕ-θ-ψ), ZYX, or RPY Euler angles to parametrize the rotation matrix:

sai theta phi

$$\begin{aligned} R(\eta) = R_b^n(\eta) &= R_3(\psi) R_2(\theta) R_1(\phi) \\ &= \begin{bmatrix} c_\psi & -s_\psi & 0 \\ s_\psi & c_\psi & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} c_\theta & 0 & s_\theta \\ 0 & 1 & 0 \\ -s_\theta & 0 & c_\theta \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 \\ 0 & c_\phi & -s_\phi \\ 0 & s_\phi & c_\phi \end{bmatrix} \\ &= \begin{bmatrix} \mathrm{c}_\theta \mathrm{c}_\psi & \mathrm{s}_\phi \mathrm{s}_\theta \mathrm{c}_\psi - \mathrm{c}_\phi \mathrm{s}_\psi & \mathrm{c}_\phi \mathrm{s}_\theta \mathrm{c}_\psi + \mathrm{s}_\phi \mathrm{s}_\psi \\ \mathrm{c}_\theta \mathrm{s}_\psi & \mathrm{s}_\phi \mathrm{s}_\theta \mathrm{s}_\psi + \mathrm{c}_\psi \mathrm{c}_\phi & \mathrm{c}_\phi \mathrm{s}_\theta \mathrm{s}_\psi - \mathrm{s}_\phi \mathrm{c}_\psi \\ -\mathrm{s}_\theta & \mathrm{s}_\phi \mathrm{c}_\theta & \mathrm{c}_\phi \mathrm{c}_\theta \end{bmatrix} \end{aligned}$$

→ Simulation

where $\eta = [\phi, \theta, \psi]^T$. A limitation of the ZYX Euler angle parametrization is its singularity when $\theta = \pi/2 + k\pi$, $k \in \mathbb{Z}$. At these angles, ~~there is a many-to-one relation between η and R.~~

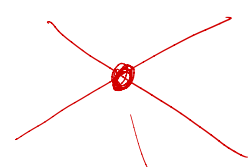
SE(3) × SO(3) → multi body

The rotational dynamics for the quadrotor is

角加速度

惯性坐标

$$\boxed{\begin{aligned} \dot{R} &= R\,\mathrm{sk}(\omega) && \text{(1a)} \\ J\dot{\omega} &= -\mathrm{sk}(\omega) J \omega + \tau && \text{(1b)} \end{aligned}}$$



where $\omega \in \mathbb{R}^3$ is angular velocity in $\mathcal{B}$, and $J \in \mathbb{R}^{3\times 3}$ is the inertia of the vehicle. The quadrotor's mass is assumed distributed symmetrically about its body frame axes, and so we take $J = \mathrm{diag}([J_1, J_2, J_3])$, i.e., a diagonal matrix. The vector $\tau = [\tau_1, \tau_2, \tau_3]^T \in \mathbb{R}^3$ is the applied torque due to all propeller thrusts and expressed in $\mathcal{B}$, the skew operator $\mathrm{sk}(\cdot) : \mathbb{R}^3 \to so(3)$ is → 力矩

螺旋桨推力

Body frame.

$$\cancel{S(x)}\ \mathrm{sk}(x) = \begin{bmatrix} 0 & -x_3 & x_2 \\ x_3 & 0 & -x_1 \\ -x_2 & x_1 & 0 \end{bmatrix}, \quad \text{where } x = \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix}.$$

$\omega \in \mathbb{R}^3$, 每个方向的 angular velocity

and $so(3) = \{M \in \mathbb{R}^{3\times 3} : M^T = -M\}$ is a 3-dimensional space of 3×3 skew symmetric matrices. One (simple way) way to model the slung load cable and payload uses a spherical 球铰 pendulum. A simple modelling assumption is to attach this pendulum to the CoM of the UAV. Hence, the pendulum does not exert a torque on the UAV. Hence, the rotational dynamics (1) is decoupled from the pendulum dynamics. Attaching the pendulum to the UAV CoM is a common assumption which is unlikely to hold in practice. The work in [10] removes this assumption and shows the increase in modelling complexity that results and lack of rigour in the control law proposed. Control design or even simulation which relaxes the CoM assumption provides an excellent opportunity for for extending the project. Further, generalizing the single spherical pendulum to multiple pendulum is another good opportunity.

The rotational kinematics (1a) expressed in η are

运动学

$$\dot{\eta} = W(\eta)\omega \rightarrow \text{design}$$

$\eta = [\text{phi}, \text{theta}, \text{psi}]$
$\phi \quad \theta \quad \psi$

with

$$W(\eta) = \begin{bmatrix} 1 & s_\phi t_\theta & c_\phi t_\theta \\ 0 & c_\phi & -s_\phi \\ 0 & s_\phi / c_\theta & c_\phi / c_\theta \end{bmatrix} \tag{2}$$

and where $t_\theta = \tan\theta$. Note that (2) has a singularity at $\theta = (2k+1)\pi/2, k \in \mathbb{Z}$ due to the Euler angle singularity.

rank≠3

Note: You are expected to use (1b) in your simulation of the SLS. (You are expected to use (2) in your control design.)

1.2 Physical inputs

We assume each propeller generates thrust $f_i \in \mathbb{R}^3$ in the $-b_3$ direction and denote the total thrust due to all propellers by the scalar input $\bar{u} \geq 0$, i.e., the total thrust vector is $F = -\bar{u}b_3 = -R\bar{u}n_3$ and $\|F\| = \bar{u}$, where $\|\cdot\|$ is a vector 2-norm. Hence, $\bar{u}$ can be thought of as one of the system inputs. (The other three inputs are often taken as $\tau \in \mathbb{R}^3$.)

Although not required, to make your model closer to the real thing, inputs $\bar{u}, \tau$ can be mapped to the physical input of the UAV which are PWM signals to the Electronic Speed Controller (ESC) and denoted $W_i, i = 1, 2, 3, 4$. Controlling individual propeller speeds creates an input torque τ. Following the usual thrust modelling assumptions [2, 13], there is a one-to-one relation between $(\bar{u}, \tau)$ and the physical inputs $W_i, i = 1, 2, 3, 4$. For a UAV in "cross" configuration we have

constant

thrust

$$\begin{matrix} 1\times 1 \\ \\ 3\times 1 \end{matrix} \begin{bmatrix} \bar{u} \\ \tau \end{bmatrix} = \begin{bmatrix} K_u & K_u & K_u & K_u \\ -c_\Theta K_u \ell & c_\Theta K_u \ell & c_\Theta K_u \ell & -c_\Theta K_u \ell \\ s_\Theta K_u \ell & -s_\Theta K_u \ell & s_\Theta K_u \ell & -s_\Theta K_u \ell \\ -K_\tau & -K_\tau & K_\tau & K_\tau \end{bmatrix} \begin{bmatrix} \tilde{W}_1^2 \\ \tilde{W}_2^2 \\ \tilde{W}_3^2 \\ \tilde{W}_4^2 \end{bmatrix} \tag{3}$$

input torque

yaw

4个 motor

hardware

where K_u is a thrust constant so that the ith rotor has a thrust vector $f_i = -K_u \tilde{W}_i^2 b_3$, ℓ is arm length, K_τ is a counter torque constant, $\tilde{W}_i$ is a normalized PWM signal defined by $\tilde{W}_i = (W_i - W_{\min})/(W_{\max} - W_{\min})$ where $W_{\min}$ and $W_{\max}$ are the minimum and maximum

$0 < \quad < 1$

modulation signal

反扭矩常数

归一化PWM信号

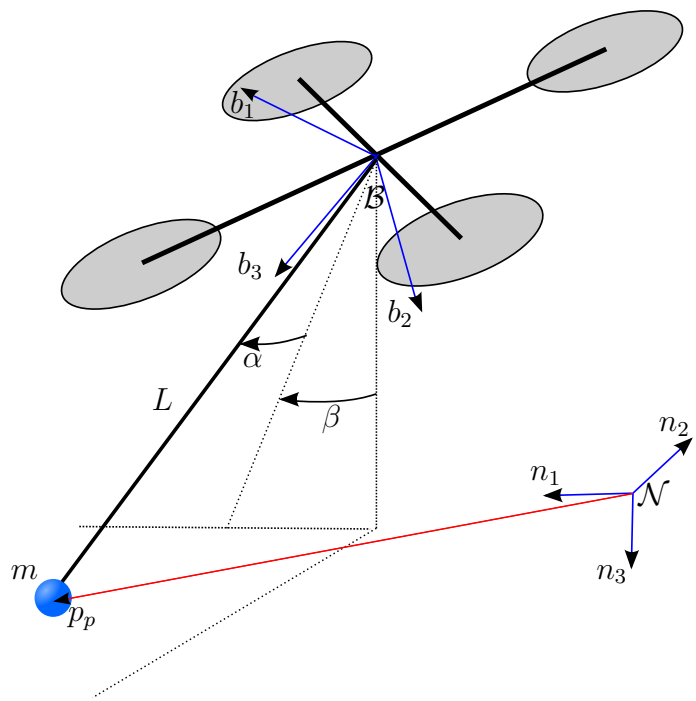


Figure 2: The SLS model. The suspended load is a spherical pendulum attached to the UAV CoM.

values of W_i, respectively, and Θ is the angle the arm makes with the b_1 axis. Therefore, the input $\tilde{W}_i \in [0, 1]$ is normalized. Thrust model (3) clearly shows that u and τ are bounded. Relation (3) can be thought of as a "mixer" and can be found in autopilot firmware such as PX4 [7].

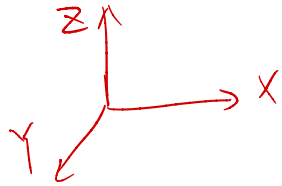
$\mathbb{R}^3$ 位置所有可达到的位置，

$\mathbb{S}^2$

1.3 Pendulum and Translational UAV Variable (PT) Dynamics

The PT (pendulum and translational UAV variable) dynamics form a coupled subsystem whose configuration space is $\mathbb{R}^3 \times \mathbb{S}^2$ which consists of the UAV position p and pendulum orientation which we parameterize by angles α, β. Note that other choices are possible for the parameterization of pendulum angle [9] and provide possible extensions to this project. The angle of the pendulum relative to n_3 is α and β is the angle relative to n_2 (see Figure 2). Hence, we take the configuration variable $q = [p^T, \alpha, \beta]^T$.
Following [3, 4], we use the Euler-Lagrange equations to derive the system dynamics. We

qua position pendulum configuration



Z-Y-X

define the position of the pendulum relative to the navigation frame as

$$\boxed{p_p = p + R_x(\alpha)R_y(\beta)[0,0,L]^T = p + L[\mathrm{s}_\beta, -\mathrm{s}_\alpha\mathrm{c}_\beta, \mathrm{c}_\alpha\mathrm{c}_\beta]^T} \tag{4}$$

where L denotes pendulum length, and position of the pendulum relative to the origin of $\mathcal{N}$ is denoted $p_p \in \mathbb{R}^3$. For simplicity the mass of the pendulum m_p is concentrated at its endpoint and the rod is assumed massless. In (4) we can think of rotating a frame attached to the pendulum $\mathcal{P}$ which is initially aligned with $\mathcal{N}$, then rotating about n_2 by an angle β and then followed by a rotation about n_1 by an angle α. Hence, the rotation between $\mathcal{N}$ and $\mathcal{P}$ is

inertial frame 固定坐标系，顺序 order

order of rotation is not commuted

$$R_p^n = R_x(\alpha)R_y(\beta) \tag{5}$$

where R_p^n maps points expressed in $\mathcal{P}$ to $\mathcal{N}$. Note that since the rotations are performed relative to the fixed frame $\mathcal{N}$, we perform successive rotations by premultiplying in (5). In particular, R_p^n maps $[0,0,L]^T$ expressed in $\mathcal{P}$ into $L[\mathrm{s}_\beta, -\mathrm{s}_\alpha\mathrm{c}_\beta, \mathrm{c}_\alpha\mathrm{c}_\beta]^T$ in frame $\mathcal{N}$. Note that the subscript/superscript convention used for the rotation matrix $\boxed{R_p^n}$ is very useful in keeping track of the frame used to represent a vector.

We drop the UAV rotational kinetic energy in the derivation of the PT dynamics as the UAV rotational dynamics (1) are decoupled from the PT dynamics. The kinetic energy of the PT system is

$$K(q,\dot{q}) = \frac{1}{2}(m\,\|\dot{p}\|^2 + m_p\,\|\dot{p}_p\|^2) \tag{6}$$

where m is UAV mass. The potential energy of the system is given by

$$V(q) = -mgp_3 - m_p g(p_3 + L\mathrm{c}_\alpha\mathrm{c}_\beta) \tag{7}$$

Combining (6) and (7) we obtain the Lagrangian

$$L(q,\dot{q}) = K(q,\dot{q}) - V(q) \tag{8}$$

Substituting L into the Euler-Lagrange equations

$$\frac{\mathrm{d}}{\mathrm{d}t}\frac{\partial L}{\partial \dot{q}_j} - \frac{\partial L}{\partial q_j} = \bar{\tau}_j, \quad 1 \le j \le 5 \tag{9}$$

where $\bar{\tau}_j$ are generalized external torques to be specified later, we obtain the standard 2nd order mechanical system form

$\ddot{q} =$

$$M(q)\ddot{q} + C(q,\dot{q})\dot{q} + G(q) = B\bar{u} \tag{10}$$

where $M(q)$ is the positive definite mass matrix

$$M(q) = \begin{bmatrix} m+m_p & 0 & 0 & 0 & Lm_p\mathrm{c}_\beta \\ * & m+m_p & 0 & -Lm_p\mathrm{c}_\beta\mathrm{c}_\alpha & Lm_p\mathrm{s}_\beta\mathrm{s}_\alpha \\ * & * & m+m_p & -Lm_p\mathrm{s}_\alpha\mathrm{c}_\beta & -Lm_p\mathrm{s}_\beta\mathrm{c}_\alpha \\ * & * & * & L^2m_p\mathrm{c}_\beta^2 & 0 \\ * & * & * & * & L^2m_p \end{bmatrix}$$

not position.

gamma: 加速度.

The entries with * can be determined from $M(q) = M(q)^T$.

$$\downarrow \quad \frac{d}{dt}\frac{\partial L}{\partial \dot{q}} - \frac{\partial L}{\partial q}$$

```
> eqn:= [diff(diff(Lg, diff(p__1(t), t)), t) - diff(Lg, p__1(t)), diff(diff(Lg, diff(p__2(t), t)), t) - diff
  (Lg, p__2(t)), diff(diff(Lg, diff(p__3(t), t)), t) - diff(Lg, p__3(t)), diff(diff(Lg, diff(alpha(t), t)),
  t) - diff(Lg, alpha(t)), diff(diff(Lg, diff(beta(t), t)), t) - diff(Lg, beta(t))];
```

$$eqn := \Big[-L m_p \beta_t^2 \sin(\beta) + L m_p \cos(\beta)\, \beta_{t,t} + (m + m_p)\, p_{1_{t,t}},\ L m_p \beta_t^2 \cos(\beta) \sin(\alpha) + 2 L m_p \sin(\beta)\, \alpha_t \cos(\alpha)\, \beta_t + L m_p \sin(\beta) \sin(\alpha)\, \beta_{t,t} + (m + m_p)\, p_{2_{t,t}} - L \cos(\beta)\, \alpha_{t,t} \cos(\alpha)\, m_p + L \cos(\beta)\, \alpha_t^2 \sin(\alpha)\, m_p,\ -L m_p \beta_t^2 \cos(\beta) \cos(\alpha) + 2 L m_p \sin(\beta)\, \alpha_t \sin(\alpha)\, \beta_t - L m_p \sin(\beta) \cos(\alpha)\, \beta_{t,t} + (m + m_p)\, p_{3_{t,t}} - L \cos(\beta)\, \alpha_{t,t} \sin(\alpha)\, m_p - L \cos(\beta)\, \alpha_t^2 \cos(\alpha)\, m_p - (m + m_p)\, g,\ -L p_{2_{t,t}} \cos(\beta) \cos(\alpha)\, m_p + L p_{2_t} \beta_t \sin(\beta) \cos(\alpha)\, m_p - L p_{3_{t,t}} \cos(\beta) \sin(\alpha)\, m_p + L p_{3_t} \beta_t \sin(\beta) \sin(\alpha)\, m_p - 2 L^2 \cos(\beta)\, \alpha_t m_p \beta_t \sin(\beta) + L^2 \cos(\beta)^2 \alpha_{t,t} m_p - L m_p \big(\sin(\beta) \sin(\alpha)\, p_{3_t} + \sin(\beta) \cos(\alpha)\, p_{2_t}\big) \beta_t + L \cos(\beta) \sin(\alpha)\, m_p g,\ L^2 \beta_{t,t} m_p + L m_p \big(-\beta_t \cos(\beta) \cos(\alpha)\, p_{3_t} + \sin(\beta)\, \alpha_t \sin(\alpha)\, p_{3_t} - \sin(\beta) \cos(\alpha)\, p_{3_{t,t}} + \beta_t \cos(\beta) \sin(\alpha)\, p_{2_t} + \sin(\beta)\, \alpha_t \cos(\alpha)\, p_{2_t} + \sin(\beta) \sin(\alpha)\, p_{2_{t,t}} + p_{1_{t,t}} \cos(\beta) - p_{1_t} \beta_t \sin(\beta)\big) - L m_p \big(-\cos(\beta) \cos(\alpha)\, p_{3_t} + \cos(\beta) \sin(\alpha)\, p_{2_t} - p_{1_t} \sin(\beta)\big) \beta_t - L p_{2_t} \sin(\beta)\, \alpha_t \cos(\alpha)\, m_p - L p_{3_t} \sin(\beta)\, \alpha_t \sin(\alpha)\, m_p + L^2 \cos(\beta)\, \alpha_t^2 m_p \sin(\beta) + L \sin(\beta) \cos(\alpha)\, m_p g \Big] \tag{7}$$

From [11, Chapter 6] the (k, j) element of C is determined by

$$\boxed{C_{kj} = \sum_{i=1}^{5} \left(\frac{\partial M_{kj}}{\partial q_i} + \frac{\partial M_{ki}}{\partial q_j} + \frac{\partial M_{ij}}{\partial q_k}\right)\dot{q}_i}$$

where d_{kj} is the (k, j) element of M. This yields

$$C(q, \dot{q}) = \begin{bmatrix} 0 & 0 & 0 & 0 & -Lm_p s_\beta \dot{\beta} \\ 0 & 0 & 0 & Lm_p(c_\alpha s_\beta \dot{\beta} + c_\beta s_\alpha \dot{\alpha}) & Lm_p(s_\beta c_\alpha \dot{\alpha} + c_\beta s_\alpha \dot{\beta}) \\ 0 & 0 & 0 & Lm_p(s_\beta s_\alpha \dot{\beta} - c_\beta c_\alpha \dot{\alpha}) & Lm_p(s_\beta s_\alpha \dot{\alpha} - c_\beta c_\alpha \dot{\beta}) \\ 0 & 0 & 0 & -L^2 m_p s_\beta c_\beta \dot{\beta} & -L^2 m_p s_\beta c_\beta \dot{\alpha} \\ 0 & 0 & 0 & L^2 m_p c_\beta s_\beta \dot{\alpha} & 0 \end{bmatrix}$$

5

5×5

We have

$$G(q) = \begin{bmatrix} 0 & 0 & -g(m + m_p) & gLm_p c_\beta s_\alpha & gm_p L s_\beta c_\alpha \end{bmatrix}^T$$

and

$$B\bar{u} = -\underbrace{\begin{bmatrix} c_\phi s_\theta c_\psi + s_\phi s_\psi \\ c_\phi s_\theta s_\psi - s_\phi c_\psi \\ c_\phi c_\theta \\ 0 \\ 0 \end{bmatrix}}_{B} \bar{u} = \begin{bmatrix} F \\ 0 \\ 0 \end{bmatrix} \tag{11}$$

方向

Hence, for the PT dynamics $\bar{\tau} = [\bar{\tau}_1, \ldots, \bar{\tau}_5]^T = B\bar{u}$. Note that the first three elements of $B\bar{u}$ is the thrust vector expressed in $\mathcal{N}$, i.e., $F = -\bar{u}Rn_3$. The most reliable (i.e., typo-free) source for the matrices M, C, G is the Maple code
`https://eclass.srv.ualberta.ca/mod/resource/view.php?id=5035014`

PDE把t隐藏了
(diff((t), t), t)

2 Project Requirements

2.1 Part 1: Initial Project Tasks: due Nov. 15

Please provide a written report for Part 1. Include all Maple/Matlab code. Feel free to use LaTeX for documentation. The Latex to generate this document is at `https://eclass.srv.ualberta.ca/mod/resource/view.php?id=5492204`

1. In this part you obtain a number of control-affine state space models of the form

$$\dot{x} = f(x) + \sum_{j=1}^{m} g_i(x) u_i \tag{12}$$

$Mq + C(q,\dot{q})\dot{q} + G(q) =$

Note: it is generally better to present your dynamics in control-affine form (12) using physical variable notation such as ϕ, α instead of x_k. This makes work a bit easier. Also, be sure to respect the ordering in the state variable specified below.

$P_p(P, \alpha, \beta)$ pendulum.

$\dot{x}_q = \dot{q} =$

$\dot{x}_{\dot{q}} = \ddot{q} = -[MC\dot{q} + G]\begin{bmatrix} F \\ 0 \end{bmatrix} = f + gu$

SFBL

$x \in R^{10}$ $u = F$

$q \leftarrow \begin{bmatrix} P_p \\ \alpha \\ \beta \end{bmatrix}$

$\dot{q} \leftarrow \begin{bmatrix} v = \dot{P}_p \\ x_\alpha = \dot{\alpha} \quad x_\beta = \dot{\beta} \end{bmatrix}$

$\dot{x} = f(x) + g(x)\cdot u$

$\cdot\ (x, y, z)$

解耦

能分为2个

姿机动力学

(a) **Inner loop or UAV rotational dynamics** consisting of (1) and input $u = \tau \in \mathbb{R}^3$. Here you should use Euler angles η and angular velocity ω in your system state. Use $x = [\eta^T, \omega^T]^T \in \mathbb{R}^6$.

与位置解耦

(b) **for the PT system or outer loop** Here you should take your input to be $u = F \in \mathbb{R}^3$. Use state $x = [p_p^T, \alpha, \beta, v^T, \gamma_\alpha, \gamma_\beta]^T \in \mathbb{R}^{10}$ where $\gamma_\alpha = \dot{\alpha}$ and $\gamma_\beta = \dot{\beta}$ denote angular rates of the pendulum, and $v = \dot{p}_p$ denotes linear velocity of the pendulum.

位置

位置坐标、速度坐标

$\mathbb{R}^{2n}$

(c) **The entire system** including the rotational dynamics (1) and the PT dynamics from step. Use state definition $x = [p_p^T, \alpha, \beta, \eta^T, v^T, \gamma_\alpha, \gamma_\beta, \omega^T]^T \in \mathbb{R}^{16}$. The system input is $u = (\bar{u}, \tau^T)^T \in \mathbb{R}^4$.

$\eta = [\phi, \theta, \psi]$

切丛 tangent bundle.

To help define your models, use the maple code at
`https://eclass.srv.ualberta.ca/mod/resource/view.php?id=5492205`
which defines the form (10). Note that the rotational dynamics were not included in the Maple file.

Important Note: We use the pendulum position p_p and not the UAV position p in all the models above.

2. (Optional) Can you derive the Euler-Lagrange equations and corresponding model (10) for the PT system using p_p instead of p in the configuration variable q? That is, take $q = [p_p^T, \alpha, \beta] \in \mathbb{R}^5$. Modify the above Maple file to do this and present your M, C, G. Check your results against those obtained in Step (1b).

min

$y = Cx$

3. Linearize your models from Step (1b) and (1c) above using a logical operating point. For your linearizations provide the A, B matrices for the LTI system. Jacobi

$u = -kx$ Linear state feedback control

$x = [0, 0, 0 \ldots]$

4. for Step (1b) and (1c) design a linear state feedback control to stabilize the system in hover for $\boxed{p_p = 0}$. Validate your control by simulation using the appropriate nonlinear model which was used to obtain the linearization. Use a range of suitable initial to test the stabiliztion law. For the case of (d) you can also regulate $\psi = 0$.

$-kx$

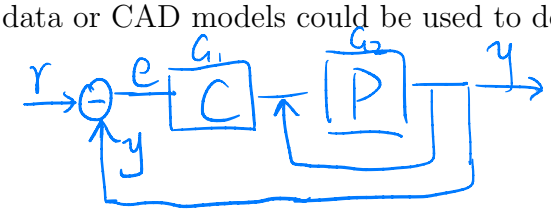
5. (Optional) Can you design a linear controller for a time-varying reference for p_p and ψ. For your reference output trajectory use

$u = -kx$

$$y_d(t) = \begin{bmatrix} p_p(t) \\ \psi_d(t) \end{bmatrix} = \begin{bmatrix} (-1.5\sin(\pi t/4) + 1)\ \text{m} \\ 0.75\sin(\pi t/2)\ \text{m} \\ (2\sin(\pi t/4) - 3)\ \text{m} \\ 0.01t\ \text{rad} \end{bmatrix} \tag{13}$$

$K = \{\lambda_1, \lambda_2 \ldots \lambda_i\}$ $\lambda_i < 0$

Matlab's `ODE45` is a good choice for simulation. This is because it is simplest and best to perform all your calculations in Maple and then write a simulation file for the model and controller (i.e., the RHS of the closed-loop ODE) and run simulation in Matlab. For your simulation use the model parameters in Table 1. These values are approximate and flight data or CAD models could be used to determine more accurate values.



x

$(0, 0, 0)$

$\dot{x} = Ax + Bu$

$x = 0$

R, Q = eye Q, R = diagon. Constant Matrix

2

Ractr.

S R R

S = R (H, Q, R)

m	1.6 kg
J_1	0.03 kg m^2
J_2	0.03 kg m^2
J_3	0.05 kg m^2
Θ	33.4°
ℓ	0.28 m
$W_{\min}$	1 ms
$W_{\max}$	2 ms
K_u	8 N
K_τ	0.8 N m
L	1 m
m_p	0.16 kg

Table 1: Nominal SLS parameters.

K = (A^T S A)^T ( )

$\lim_{t\to\infty} S = \text{const}$

2.2 Part 2: Project Tasks: due Dec 8

On Dec. 8 you should be able to present the results from the following three designs. Perform the following

1. Check the conditions for static SFBL (static state feedback linearization) for each of the systems you computed in Part 1. If you find any of your systems meet the conditions, then design and simulate an SFBL output tracking control for p_p and ψ. Use (13) for your reference trajectory. Describe in detail how you check your SFBL conditions. If SFBL is possible, then what is the domain on which the controller is well-defined.

4.6

Theo 4.6.1

2. For models in Part 1, Step (1b) and (1c) check conditions for input-output SFBL using output $[p_p, \psi]$ and $[p, \psi]$. Use static state feedback only. If any designs are successful, then simulate them. $\boxed{4.6.2}$

3. For models in Part 1, Step (1b) and (1c) if you found conditions for static SFBL did not hold, can you dynamically extend the state to achieve dynamic state feedback linearization? To do this, use the DEA (Dynamic Extension Algorithm) described in [5, Ch. 5.4]. Use pendulum position p_p and ψ as candidate outputs. Describe your DEA steps in detail and explain the choice of any design parameters available. If your DEA is able to achieve relative degree, but not able to achieve full relative degree of your extended system, then use input-output state feedback linearization if the conditions for for this design hold. If you are able to achieve a dynamic SFBL with full relative degree, then use static SFBL on the extended dynamics. Simulate any feasible designs using the time-varying reference (13).

2.3 Part 3: Report TBD

to be determined.

Please submit your final report and code after the final exam on a date TBD.

References

[1] Moses Bangura. *Aerodynamics and Control of Quadrotors*. PhD thesis, College of Engineering and Computer Science, The Australian National University, 2017.

[2] Samir Bouabdallah. *Design and control of quadrotors with application to autonomous flying*. PhD thesis, Ecole Polytechnique Federale de Luasanne, Lausanne, Switzerland, 2007.

[3] M. E. Guerrero, D. A. Mercado, R. Lozano, and C. D. García. Passivity based control for a quadrotor uav transporting a cable-suspended payload with minimum swing. In *Proceedings of the 54th IEEE Conference on Decision and Control*, pages 6718–6723, 2015.

[4] M. E. Guerrero, D.A. Mercado, R. Lozano, and C.D. García. Swing-attenuation for a quadrotor tranporting a cable suspended payload. *ISA Transactions*, 68:433–449, 2017.

[5] Alberto Isidori. *Nonlinear Control Systems*. Springer-Verlag, London, 3 edition, 1995.

[6] Kristian Klausen, Thor I. Fossen, and Tor Arne Johansen. Nonlinear control with swing damping of a multirotoruav with suspended load. *Journal of Intelligent Robotic Systems*, 88:379–394, 2017.

[7] Lorenz Meier. PX4 autopilot. `http://pixhawk.org/` [accessed 01 Jan 2018], 2018.

[8] Amir Moeini, Alan F. Lynch, and Qing Zhao. Nonlinear control with swing damping of a multirotor uav with suspended load. *Journal of Intelligent Robotic Systems*, 2021. submitted.

[9] L. Qian and H. H. T. Liu. Path-following control of a quadrotor uav with a cable-suspended payload under wind disturbances. *IEEE Transactions on Industrial Electronics*, 67(3):2021–2029, 2020.

[10] Longhao Qian and Hugh H.T. Liu. Dynamics and control of a quadrotor with a cable suspended payload. In *Proceedings of the Canadian Conference on Electrical and Computer Engineering*, pages 1–4, Windsor, ON, April 2017.

[11] Mark W. Spong, Seth Hutchinson, and M. Vidyasagar. *Robot Modelling and Control*. Wiley, New York, NY, 2006.

[12] Daniel K. D. Villa, Alexandre S. Brand ao, and Mário Sarcinelli-Filho. Survey on load transportation using multirotor uavs. *Journal of Intelligent Robotic Systems*, 98:267–296, 2020.

[13] Hui Xie. *Dynamic Visual Servoing of Rotary Wing Unmanned Aerial Vehicles*. PhD thesis, Dept. of Electrical and Computer Engineering, University of Alberta, Edmonton, AB, February 2016.

1. $J\dot{w} + \tilde{w} J w = \tau$ 刚体转动动力学方程

torque(力矩，局部坐标)

$$J = \begin{bmatrix} J_1 & & \\ & J_2 & \\ & & J_3 \end{bmatrix}$$

转动变量

$$\tilde{w} = \mathrm{skew}(w) = \begin{pmatrix} 0 & -w_3 & w_2 \\ w_3 & 0 & -w_1 \\ - & w_1 & \end{pmatrix}$$

$$\boxed{\tilde{w} \times \begin{pmatrix} d_1 \\ d_2 \\ d_3 \end{pmatrix} = w \times d}$$

$$① \begin{cases} \dot{r} = w(r)\, w \\ J\dot{w} = -\tilde{w} J w + \tau \end{cases}$$