AER 1216 Ass I

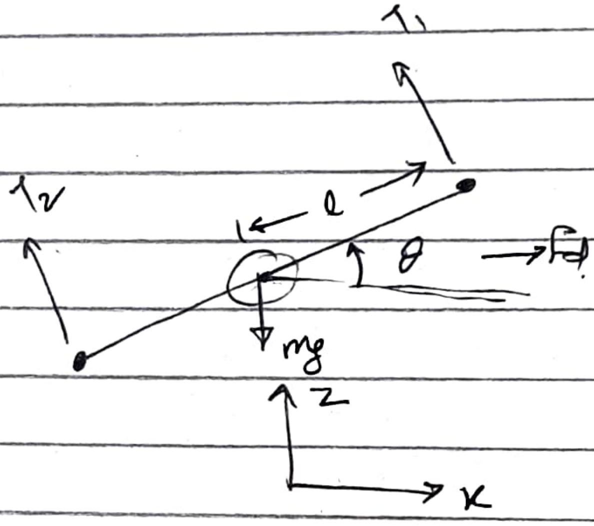
PAGE NO.:
DATE: / /

(a) inputs → $T_1(t)$ $T_2(t)$

outputs → $x(t)$, $z(t)$

state → $\{x(t), z(t), \theta(t), \dot{x}(t), \dot{z}(t), \dot{\theta}(t)\}$

∴ Governing Eqn



$m\ddot{x} \Rightarrow F_d - (T_1+T_2)\sin\theta$

$m\ddot{z} = -mg + (T_1+T_2)\cos\theta$

$J\ddot{\theta} = (T_1 - T_2)\,l$

$$\begin{bmatrix} \ddot{x} \\ \ddot{z} \\ \ddot{\theta} \end{bmatrix} = \begin{bmatrix} K(V_0-\dot{x})^2/m \\ -g \\ 0 \end{bmatrix} + \begin{bmatrix} -\frac{\sin\theta}{m} & -\frac{\sin\theta}{m} \\ \frac{\cos\theta}{m} & \frac{\cos\theta}{m} \\ -l/J & l/J \end{bmatrix} \begin{bmatrix} T_1 \\ T_2 \end{bmatrix}$$

SS → $\dot{X} \Rightarrow$

$$\begin{bmatrix} \dot{x} \\ \dot{z} \\ \dot{\theta} \\ \ddot{x} \\ \ddot{z} \\ \ddot{\theta} \end{bmatrix} = \begin{bmatrix} \dot{x} \\ \dot{z} \\ \dot{\theta} \\ K(V_0-\dot{x})^2/m \\ -g \\ 0 \end{bmatrix} + \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ -\sin\theta/m & -\sin\theta/m \\ \cos\theta/m & \cos\theta/m \\ -l/J & l/J \end{bmatrix} \begin{bmatrix} T_1 \\ T_2 \end{bmatrix}$$

$$y = \begin{bmatrix} x \\ z \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} x \\ z \\ \theta \\ \dot{x} \\ \dot{z} \\ \dot{\theta} \end{bmatrix}$$

PAGE NO.:
DATE: / /

(b) $1 \longrightarrow$ $F_d = (T_1 + T_2) \sin\theta$

$mg = (T_1 + T_2) \cos\theta$

$T_1 - T_2 = 0 \quad \Rightarrow \quad T_1 = T_2$

$$\left\{ \begin{array}{l} (2 T_1 \sin\theta = F_d)^2 \\ (2 T_1 \cos\theta = mg)^2 \end{array} \right. +$$

$\Rightarrow \quad 4 T_1^2 \Rightarrow F_d^2 + mg^2$

$T_{2e} = T_{1e} \Rightarrow \frac{1}{2}\sqrt{F_d^2 + mg^2}$

$\tan(\theta) \Rightarrow \left(\frac{F_d}{mg}\right) \Rightarrow \theta_e \Rightarrow \tan^{-1}\left(\frac{F_d}{mg}\right)$

An (∞) number of Equilib^n points Exist! as the Copter Can hover anywhere

(c) $\Rightarrow \quad T_1 + T_2 = \frac{mg}{\cos(\theta)}$

For first Eq^n in x- dynamics

state $\rightarrow \begin{Bmatrix} x \\ \dot{x} \end{Bmatrix}$

$m\ddot{x} = F_d - (T_1 + T_2) \sin\theta$

$\dot{x} \rightarrow \begin{Bmatrix} \dot{x} \\ \ddot{x} \end{Bmatrix}$

$m\ddot{x} \Rightarrow F_d - mg \tan\theta$

Input $\Rightarrow \{\theta\}$

$\therefore$ SS $\Rightarrow \begin{Bmatrix} \dot{x} \\ \ddot{x} \end{Bmatrix} \Rightarrow \begin{Bmatrix} \dot{x} \\ \left(\frac{F_d}{m}\right) \end{Bmatrix} - \begin{bmatrix} 0 \\ g \tan\theta \end{bmatrix}$

$\therefore$ Linearizing $\rightarrow$ $f_1 \rightarrow \dot{x}$

$$\left.\frac{\partial f_1}{\partial x}\right|_{x_e} (x - x_e) + \cancel{\frac{\partial f}{\partial y}(y - y_e)}$$

$\rightarrow \left.\ddot{x}\right|_{x_e} \Rightarrow 0\,!$

PAGE NO. :
DATE : / /

$f_2 \rightarrow \left(\frac{F_d}{m}\right) - g\tan\theta$

Linearizing the E^n

$\frac{\partial f_2}{\partial X}$

↳ state → x → $\frac{\delta f_2}{\delta x} = 0$

→ $\dot{x}$ → $\frac{\delta f_2}{\delta \dot{x}} = \left.\frac{2K(V_\infty - \dot{x})(-1)}{m}\right|_{x_e} (\dot{x} - \dot{x}_0)$

$= \frac{-2KV_\infty}{m}$

$\frac{\delta f_2}{\delta u} \Rightarrow \left. -g\sec^2\theta \right|_{\theta_e} (\theta - \theta_0)$

$\Rightarrow -g\sec^2\theta_e$

$\Rightarrow -g\left(1 + \left(\frac{KV_\infty}{m}\right)^2\right)\delta\theta$

$$\therefore \begin{bmatrix} \delta\dot{x} \\ \delta\ddot{x} \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ 0 & \frac{-2KV_\infty}{m} \end{bmatrix} \begin{bmatrix} \delta x \\ \delta\dot{x} \end{bmatrix} + \begin{bmatrix} 0 \\ -g\left(1 + \left(\frac{KV_\infty}{mg}\right)\right)^2 \end{bmatrix} \delta\theta$$

$$\delta y = \begin{bmatrix} 1 & 0 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} \delta x \\ \delta\dot{x} \end{bmatrix}$$

In part c, we have linearized our equations and hence, we will be looking to use a linear controller.
There are two controllers talked about in the presentation.
1. Pole Placement
2. Linear Quadratic Regulator
The Pole Placement method has certain uncertaininty regarding eigen values, and given that I don't have much
experience in designing control systems, I will not prefer the Pole Placement Controller
The Linear Quadratic Regulator is a more regularized with a standard procedure and hence, I will prefer the
Linear Quadratic Regulator.
The Linear Quadratic Regulator works on the concept of optimal control by minizing a predefined cost function.
The Cost here is a linear system of equations where we penalize both the control effort and the error.
We have the Linear Dynamical control system for State-Space Equation, we have the reference equilibrium points
and the cost function which needs to be optimized. We apply the principles of dynamic programming to go from the
final state to the intial state. The recursive relation in the equations gives us the net value which include current cost, Action Cost,
and the Closed loop dynamics - (transpose(state-space)*Future Value matrix*state-space).