

Aula Prática 1: Movimento Longitudinal

AB-722

Flávio Luiz Cardoso Ribeiro

<http://flavioluiz.github.io>

flaviocr@ita.br

Departamento de Mecânica do Voo
Divisão de Engenharia Aeronáutica e Aeroespacial
Instituto Tecnológico de Aeronáutica



2018

Equações do Movimento

As seguintes equações representam o movimento longitudinal da aeronave:

$$\begin{aligned}\dot{V} &= \frac{T \cos(\alpha + \alpha_F) - D}{m} - g \sin(\gamma) \\ \dot{\gamma} &= \frac{L + T \sin(\alpha + \alpha_F)}{mV} - \frac{g \cos(\gamma)}{V} \\ \dot{q} &= \frac{1}{I_{yy}} (m_A + m_F) \\ \dot{x} &= V \cos \gamma \\ \dot{H} &= V \sin \gamma \\ \dot{\alpha} &= q - \dot{\gamma}\end{aligned}$$

Modelo Aerodinâmico

Forças e momentos aerodinâmicos:

$$L = \frac{1}{2}\rho V^2 S C_L \qquad D = \frac{1}{2}\rho V^2 S C_D \qquad m = \frac{1}{2}\rho V^2 S c C_m$$

Modelo aerodinâmico linear:

$$C_L = C_{L_0} + C_{L_\alpha} \alpha + C_{L_{\delta p}} \delta p + C_{L_q} \left(q \frac{c_{ref}}{V_{ref}} \right)$$

$$C_m = C_{m_0} + C_{m_\alpha} \alpha + C_{m_{\delta p}} \delta p + C_{m_q} \left(q \frac{c_{ref}}{V_{ref}} \right) + C_{m_{\dot{\alpha}}} \left(\dot{\alpha} \frac{c_{ref}}{V_{ref}} \right)$$

$$C_D = C_{D_0} + k_1 C_L + k_2 C_L^2$$

Modelo Propulsivo

Força de Tração:

$$T = \pi T_{max,i} \left(\frac{\rho}{\rho_i} \right)^{n_\rho} \left(\frac{V}{V_i} \right)^{n_V}$$

onde $T_{max,i}$ é a tração máxima obtida nas condições ρ_i, V_i .

- Pistão: $n_V = -1$
- Turbofan ou jato subsônico: $n_V = 0$
- Jato supersônico: $n_V = 1$

Momento devido à tração:

$$m_F = z_F \cos(\alpha_F) T + x_F \sin(\alpha_F) T$$

Modelo Atmosférico

Varição da temperatura com altitude:

$$T = T_n + A_n(H - H_n)$$
$$\left(\frac{\rho}{\rho_n}\right) = \left(\frac{T}{T_n}\right)^{-\left(1 + \frac{g_0}{A_n R}\right)}$$

Para a troposfera (até 11 km):

$$A_0 = -6.510^{-3} \text{ K/m}$$
$$H_0 = 0 \text{ m}$$
$$\rho_n = 1.225 \text{ kg/m}^3$$

Acima de 11 km: $A_{11} = 0 \text{ K/m}$ (temperatura constante)

A solução fica: $\left(\frac{\rho}{\rho_{11}}\right) = \exp\left(\frac{-g * (H - 11000)}{(R * (T_{11}))}\right)$

Equações do movimento

Estados e controles:

$$X = [V \quad \gamma \quad q \quad \alpha \quad H \quad x]^T$$

$$U = [\delta p \quad \pi]^T$$

As equações do movimento consistem em um sistema de equações da forma:

$$\dot{X} = f(X, U)$$

Para uma posição inicial $X(0)$, e dada a entrada U , podemos encontrar $X(t)$ resolvendo a equação numericamente (Euler, Runge-Kutta, etc.). Resolveremos utilizando a função ODE45 do Matlab.

Equilíbrio

Dada uma condição permanente de vôo reto: H_{eq} , V_{eq} e γ_{eq} . Em situação de equilíbrio, temos:

$$\begin{aligned}\dot{V} &= \frac{T \cos(\alpha + \alpha_F) - D}{m} - g \sin(\gamma) = 0 \\ \dot{\gamma} &= \frac{L + T \sin(\alpha + \alpha_F)}{mV} - \frac{g \cos(\gamma)}{V} = 0 \\ \dot{q} &= \frac{1}{I_{yy}} (m_A + m_F) = 0\end{aligned}$$

E as seguintes incógnitas: α_{eq} , π_{eq} e δp_{eq} .

Podemos resolver esse sistema utilizando a função FSOLVE do MATLAB.

Tarefas da Aula Prática

- Fazer uma função MATLAB implementando as equações do movimento para a aeronave sugerida: $\dot{X} = f(t, X, U)$
- Encontrar α_{eq} , π_{eq} e δp_{eq} , utilizando a função FSOLVE;
- Simular o sistema $\dot{X} = f(t, X, U)$ usando a função ODE45;

Ver os seguintes arquivos:

- carregaaeronave.m: dados da aeronave-exemplo;
- atmosfera.m: função que calcula massa específica e temperatura do ar;

Tarefas da Aula Prática

Entregar um programa com as seguintes características:

- Acesse o `carregaaeronave.m` para inicializar os dados de uma aeronave;
- Pergunte ao usuário as condições de voo (velocidade, altitude e ângulo de trajetória);
- Calcule e apresente as condições de equilíbrio (deflexão de profundor, manete e ângulo de ataque);
- Pergunte ao usuário a perturbação de ângulo de ataque, altitude e velocidade desejada para simulação;
- Simule o voo da aeronave nas condições perturbadas e apresente os gráficos das variações de todos os estados (usar `subplot!`).

Tarefas da Aula Prática

Sugestões de programação:

- Não utilizar variáveis globais (utilize *structs* ou vetores como argumentos das funções!);
- Procure não repetir código! Use a mesma função de dinâmica tanto para simular quanto para calcular o equilíbrio;
- Procure deixar o código claro: **use indentação e faça comentários!**