3. INTRODUÇÃO À REPRESENTAÇÃO DE ESPAÇO DE ESTADOS

A representação de Espaço de Estados modela a dinâmica do sistema de suspensão a partir do conjunto de equações diferenciais de primeira ordem estabelecidos nas equações 2.1 e 2.2. É útil para:

- Representar sistemas com múltiplas entradas e múltiplas saídas;
- Descreve o comportamento do sistema no domínio tempo em uma forma matricial compacta.

A forma geral é dada pelas equações 3.1 e 3.2:

$$\dot{x} = \mathbf{A}x + \mathbf{B}u \tag{3.1}$$

$$y = \mathbf{C}x + \mathbf{D}u \tag{3.2}$$

Onde:

- x: Vetor de estado (variáveis do sistema que capturam o comportamento dinâmico do sistema);
- u: Vetor de entrada (forças externas);
- y: Vetor de saída (quantidades de interesse);
- **A**, **B**, **C**, **D**: Matrizes que definem as dinâmicas do sistema.

Após derivação**(1)** das equações de movimento (2.1 e 2.2), reformulando em equações de primeira ordem e combinando na forma vetorial**(1)**, temos as matrizes do Espaço de Estados da seguinte forma:

$$\mathbf{A} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ -\dfrac{k_s}{m_s} & -\dfrac{c_s}{m_s} & \dfrac{k_s}{m_s} & \dfrac{c_s}{m_s} \\ 0 & 0 & 0 & 1 \\ \dfrac{k_s}{m_u} & \dfrac{c_s}{m_u} & -\dfrac{k_s+k_t}{m_u} & -\dfrac{c_s}{m_u} \end{bmatrix},\ \mathbf{B} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ \dfrac{k_t}{m_u} \end{bmatrix},\ \mathbf{C} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix},\ \mathbf{D} = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$$

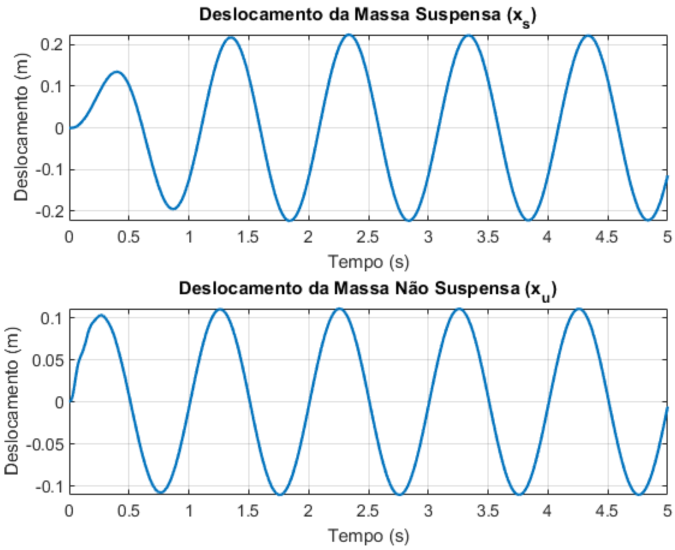
3.1 Código no MATLAB

Abaixo está o código, comentado, sobre como é feito a simulação via MATLAB.

(1) Este procedimento não será mostrado para fins de objetividade.

```
% Parâmetros
m_s = 250;      % Massa suspensa (kg) - 250 a 500 kg
m_u = 50;       % Massa não suspensa (kg) - 25 a 75 kg
k_s = 15000;    % Rigidez da suspensão (N/m) - 10 000 a 50 000 N/m
k_t = 200000;   % Rigidez do pneu (N/m) - 150 000 a 250 000 N/m
c_s = 1000;     % Amortecimento da suspensão (Ns/m) - 1 000 a 5 000 Ns/m

% Matrizes do Espaço de Estados
A = [0, 1, 0, 0;
    -k_s/m_s, -c_s/m_s, k_s/m_s, c_s/m_s;
     0, 0, 0, 1;
     k_s/m_u, c_s/m_u, -(k_s+k_t)/m_u, -c_s/m_u];
B = [0; 0; 0; k_t/m_u];
C = [1, 0, 0, 0; 0, 0, 1, 0];
D = [0; 0];

% Função do MATLAB (sys) para definir o Sistema das matrizes do Espaço de Estados
sys = ss(A, B, C, D);

% Parâmetros para Simulação
A = 0.1;        % Amplitude do solavanco, valor arbitrário
t = 0:0.01:5;   % Período/Tempo de 0 a 5 segundo com degrau (step) de 0.01 s

% Simulação
u = A * sin(2 * pi * t);        % Excitação da rua de 0.1 m (altura do solavanco)
[y, t, x] = lsim(sys, u, t);     % lsim é uma função linear do MATLAB para simular
sistemas no domínio tempo

% Resultados - basicamente aqui são funções e operações do MATLAB, deve ter seme-
lhante no JAVA
figure;
subplot(2,1,1);
plot(t, y(:,1), 'LineWidth', 1.5);
title('Deslocamento da Massa Suspensa (x_s)');
xlabel('Tempo (s)');
ylabel('Deslocamento (m)');
grid on;

subplot(2,1,2);
plot(t, y(:,2), 'LineWidth', 1.5);
title('Deslocamento da Massa Não Suspensa (x_u)');
xlabel('Tempo (s)');
ylabel('Deslocamento (m)');
grid on;
```

Os resultados extraídos são:



Deslocamento da Massa Suspensa (x_s)
Deslocamento (m)
Tempo (s)
Deslocamento da Massa Não Suspensa (x_u)
Deslocamento (m)
Tempo (s)