

# FTL066–Programação de Sistemas de Tempo Real

## Laboratório 4

Prof. André Cavalcante  
andrecavalcante@ufam.edu.br

Outubro de 2019

### 1 Objetivos

- Criar uma simulação de um sistema simples
- Utilizar múltiplas *threads*

### 2 Introdução

Um robô móvel com acionamento diferencial pode ser descrito pelo modelo no espaço de estados 1:

$$\begin{aligned}\dot{x}(t) &= \begin{bmatrix} \sin(x_3) & 0 \\ \cos(x_3) & 0 \\ 0 & 1 \end{bmatrix} u(t) \\ y(t) &= \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} x(t)\end{aligned}\tag{1}$$

onde  $x(t) = [x_c \ y_c \ \theta]^T$ , sendo  $(x_c, y_c)$  a posição do centro de massa do robô e  $\theta$  a sua orientação.  $u(t) = [v \ \omega]^T$  é a entrada do sistema, sendo  $v$  a velocidade linear e  $\omega$  a velocidade angular do robô. A saída do sistema é  $y(t)$ .

A entrada do sistema é:

$$u(t) = \begin{cases} 0 & , \text{ para } t < 0 \\ \begin{bmatrix} 1 \\ 0.2\pi \end{bmatrix} & , \text{ para } 0 \leq t < 10 \\ \begin{bmatrix} 1 \\ -0.2\pi \end{bmatrix} & , \text{ para } t \geq 10 \end{cases}$$

### 3 Descrição

Neste laboratório o programa de simulação do robô móvel deverá ser dividido em duas tarefas. As tarefas podem ser implementadas utilizando processo ou *threads*.

Uma tarefa fará a simulação em si e outro processo fará a geração de  $u(t)$  e a amostragem de  $y_f(t)$ , sendo  $y_f(t)$  o ponto da frente do robô dado por:

$$y_f(t) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} x(t)$$

onde  $x(t) = [x_c \ y_c \ \theta]^T$  é o vetor de estados e  $u(t) = [v(t) \ w(t)]^T$  é a entrada do sistema.

O objetivo é fazer um programa em C para simular a resposta do sistema usando múltiplas tarefas:

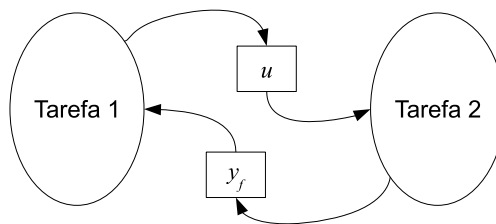


Diagrama do Programa em Tarefas

Portanto, há duas tarefas: uma fará a simulação, gerando o  $u(t)$  e a amostragem de  $y_f(x)$ , sendo  $y_f(x)$  o ponto da frente do robô, dado por:

$$y_f(t) = x(t) + \begin{bmatrix} 0.5 * D \cos(x_3) \\ 0 \\ 0 \end{bmatrix} x(t)$$

onde  $D$  é o diâmetro do robô. Ou seja, o valor da saída será agora  $y_f(t)$ , ao invés do  $y(t)$  original.

Deve-se assumir que  $x(t) = 0$  para  $t \leq 0$ .

## 4 Observações

- Utilize a estrutura gerada no exercício anterior e a expanda para este programa.
- O programa deverá fazer a simulação para  $t \in [0, 20]$ s e gerar como saída um arquivo ASCII contendo uma linha para cada valor de  $t$ . Os valores deverão ser separados pelo caractere *tab*.
- Cada linha deverá conter: o correspondente valor de  $t$ , o valor de  $u(t)$  e o valor de  $y_f(t)$ .
- Reestruture o programa do laboratório 3 de forma que existam as variáveis  $u$  e  $y_f$ , escalares e que estas sejam as únicas trocas de dados entre a simulação e a geração e armazenagem de valores.
- Separe o programa em dois processos ou duas *threads*, deixando em um deles a simulação e no outro a geração de  $u$ , a amostragem de  $y_f$  e a armazenagem dos valores.

- Altere a temporização da amostragem de  $y_f$  e da geração de  $u$  para 50ms. Note que a simulação deverá a ser feita com um período de 30ms.
- Faça o gráfico de  $y_f(k)$  amostrado para um horizonte de simulação de 20s.
- Faça uma tabela comparando os valores de média, variância, desvio padrão e valores máximos e mínimos de  $T(k)$  e  $J(k)$ , para o sistema sem carga e com carga.
- O relatório deverá descrever a hierarquia de diretórios utilizada para estruturar o programa e explicar porque esta hierarquia facilita a reutilização do código gerado. Além disso, deverá apresentar uma análise crítica dos resultados dos experimentos.

## 5 Entrega

- Data: 28 de outubro 2019 até às 23:59
- Fazer *upload* no Google Sala de Aula de:
  - Relatório em PDF
  - A pasta de desenvolvimento compactada
 OBS.: o professor irá baixar em sua máquina a pasta, descompactá-la, fazer um *make* e executar.
- Entrega em **duplas**.
- Não serão aceitos trabalhos iguais.
- Não serão aceitos trabalhos fora de prazo.
- Formato do relatório: Relatório Técnico da ABNT ou Conferência do IEEE (coluna simples).