



ENG 169617 – Tópicos em Engenharia (Cinemática de robôs)

Trabalho 02

O prazo de entrega deste trabalho é 19/01/2015 até 23:59h. Você deverá enviá-lo em formato digital para o e-mail bernardes@unb.br com o campo de assunto preenchido da seguinte forma: “ENG 169617 – T2 (matrícula)”, onde o termo *matrícula* deve ser substituído pelo seu número de matrícula na UnB.

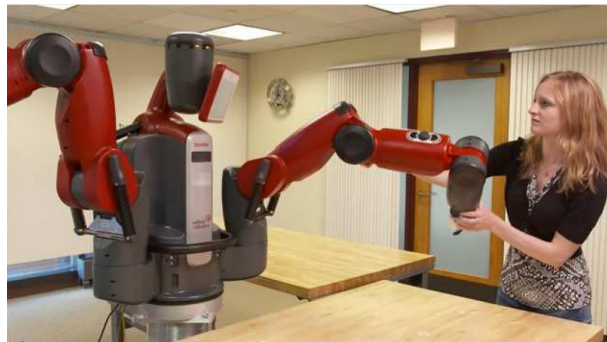
Caso haja mais de um arquivo a ser enviado, compacte-os em uma única pasta zipada.

Você pode conversar com os outros alunos sobre este trabalho, fazer perguntas no Piazza, usar calculadora ou outras ferramentas e consultar fontes externas como a Internet. Entretanto, você deve entregar seu próprio trabalho. Trabalhos copiados serão anulados.

PARTE I - Exercícios escritos:

A primeira parte consiste em um conjunto de problemas escritos, sendo dois retirados do livro texto, *Robot Modeling and Control*, Spong, Hutchinson e Vidyasagar. Siga as instruções e explicações extras quando houverem. As questões devem ser respondidas à mão de forma organizada e clara.

1: A empresa Rethink Robotics comercializa um robô chamado Baxter. Assista a vídeos do robô Baxter no YouTube (exemplo: www.youtube.com/watch?v=gXOkWuSCkRI) para aprender sobre sua cinemática. Desenhe um esquemático do cadeia cinemática serial do braço esquerdo do Baxter (o que a mulher está tocando na figura abaixo). Use a convenção do livro para representar graficamente juntas de revolução e prismáticas em 3D.



“3-7: Consider the three-link Cartesian manipulator of Figure 3.28. Derive the forward kinematic equations using the DH convention.”

Sua solução deve incluir um esquemático do manipulador com frames posicionados apropriadamente, uma tabela com os parâmetros DH, e uma matriz de transformação final. Então, responda a seguinte questão: quais são as coordenadas x , y e z da ponta do efetuador expressas no frame da base (em função dos parâmetros do robô e das coordenadas de juntas)?

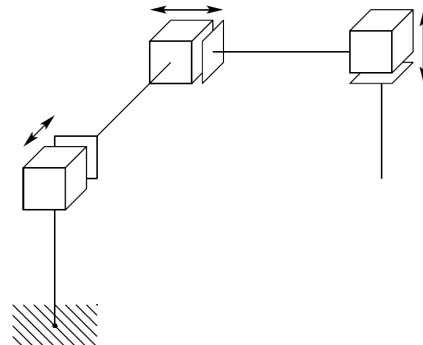


Figure 3.28 (Spong)

“3-7: Consider the three-link articulated robot of Figure 3.27. Derive the forward kinematic equations using the DH convention.”



Sua solução deve incluir um esquemático do manipulador com frames posicionados apropriadamente, uma tabela com os parâmetros DH, e uma matriz de transformação final. Então, responda a seguinte questão: quais são os componentes x , y e z expressos no frame da base, de um vetor unitário apontando na direção do último link do robô (da terceira junta para a ponta, em função dos parâmetros do robô e das coordenadas de juntas)?

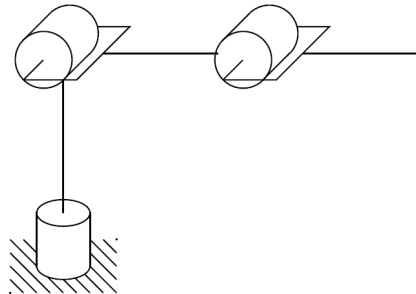


Figure 3.27 (Spong)

PARTE II - Simulação no MATLAB:

Para este trabalho, a tarefa em simulação consiste em modificar o script de MATLAB fornecido para que ele faça a animação de um robô SCARA se movendo sob várias condições.

1) Leia o Exemplo 3.6 do livro texto (Spong). Ele deriva o modelo cinemático direto de um manipulador SCARA usando a convenção DH. A solução fornecida inclui um esquemático, uma tabela de parâmetros DH, as matrizes A , e a transformação completa T^0_4 . Você não precisa refazer a derivação de nenhum desses itens, mas você deve entender a estrutura do manipulador antes de continuar. Note que o exemplo mostra quatro juntas, mas este problema irá considerar apenas as três primeiras juntas (ignorando a junta de revolução no punho, que é parametrizada por θ^*_4).

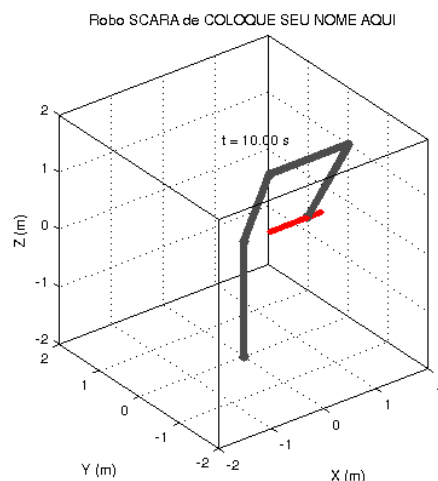
1) Além deste roteiro, você deverá usar outro arquivo disponibilizado junto ao .zip:

Script inicial MATLAB: *robo_scara_inicial.m*

2) Renomeie o script inicial para *robo_scara_matricula.m*, substituindo por sua matrícula da UnB, de forma que cada aluno tenha um arquivo único.

3) Escreva seu nome no topo do arquivo .m e no título do gráfico como indicado nos comentários do código.

4) Execute o código e veja o que acontece. Na forma como está o script, a animação mostrará um robô arbitrário e estático, com uma ponta que se move para frente e para trás em uma linha reta, deixando um rastro vermelho. O último frame da animação deverá se parecer com o plot da figura abaixo. Note que o tempo virtual de simulação decorrido é mostrado sobre o robô.





Você pode rotacionar o plot para visualizar melhor sua natureza tridimensional usando “Rotate 3D”. Essa ferramenta é a seta arredondada em sentido anti-horário localizada no topo da janela de plot:



5) Sua primeira tarefa é construir três históricos de coordenadas de juntas para o robô seguir. O código inicial inclui 5 modos de movimento, ditados pelo valor da variável *motion_mode*, cujo valor você pode setar na linha 44.

- *motion_mode*=0: o robô permanece parado com coordenadas de juntas em zero.
- *motion_mode*=1: apenas a junta 1 se move. Esse histórico de coordenadas de juntas está pronto para você.
- *motion_mode*=2: apenas a junta 2 se move. Você deve programar esse histórico de coordenadas de juntas.
- *motion_mode*=3: apenas a junta 3 se move. Você deve programar esse histórico de coordenadas de juntas.
- *motion_mode*=4: todas as juntas se movem em um padrão interessante de sua escolha. Você deve programar esse histórico de coordenadas de juntas.

Siga os exemplos fornecidos para preencher os históricos de coordenadas de juntas para os modos de movimento 2, 3 e 4. Você pode usar a variável de tempo *t* para fazer as juntas ativas se moverem como você quiser ao longo do tempo. Por enquanto, a posição da ponta do robô está setada para $\theta_1 = \text{theta1}$, $\theta_2 = \text{theta2}$ e $d_3 = d3$. Para criar essa animação, você tem que decidir quais pontos ao longo do robô você quer mostrar e então descobrir como calcular suas posições dada a atual configuração do robô. Eu recomendo que você comece plotando apenas um ponto, como a ponta do efetuador, e então acrescente os outros pontos do robô.

Você deve colocar a saída dos seus cálculos na matriz *points_to_plot*. Essa matriz possui 3 linhas, correspondendo a coordenadas *x*, *y* e *z* no frame da base. Cada coluna representa um ponto que você quer plotar e a ordem da coluna é a ordem em que os pontos são plotados. A última coluna deve ser a posição da ponta do robô; é o único ponto cuja trajetória é rastreada usando um histórico de pontos vermelhos.

Importante: Seus cálculos não podem utilizar nenhuma função do MATLAB, biblioteca ou código externo que trabalhe com matrizes de rotação, transformadas homogêneas, parâmetros de Denavit-Hartenberg, animação de robôs ou tópicos relacionados. Ao invés disso, você deverá codificar todos os cálculos você mesmo, usando apenas funções de baixo nível como *sin*, *cos*, e matemática de vetor/matriz.

Quando seus cálculos estiverem corretos, a figura do robô deve se mover conforme o especificado para o modo de movimento selecionado. Teste todos os cinco modos de movimento e tente deixar todos funcionando corretamente. Observe o movimento resultante da ponta do robô em cada caso.

6) Volte ao histórico de coordenadas do modo de movimento 4 e modifique-o para produzir uma trajetória de ponta particularmente interessante. Gaste um pouco de tempo brincando com diferentes funções para melhorar seu entendimento da cinemática deste robô.