



Robótica Espacial

Aula prática nº 9

Programação do robô UR3e - Exercícios suplementares

Vitor Santos

Universidade de Aveiro

Abril 2025

Sumário

- 1 Uso de IOs externos no robô UR3
- 2 Uso de scripts no robô UR3
- 3 Verificação de objetos na garra do robô UR3

Exercício 1 - Recolha e repaletização de objetos empilhados

Com base nos exercícios e ferramentas abordadas anteriormente, em particular:

- movimentos em juntas e em espaço linear,
- avanço até detetar colisão,
- fecho e abertura de garra,

fazer um programa que execute os seguintes passos:

- O robô parte de uma posição de repouso (por exemplo *home*);
- Coloca a ponta orientada verticalmente num ponto de aproximação a cerca de 10 cm da bancada sobre um grupo de 3 objetos sobrepostos de dimensões desconhecidas, mas que podem ser agarrados com a garra;
- O robô deve pegar nos objetos um a um e colocá-los numa outra zona da bancada pousados lado a lado.
- O robô deve voltar à posição de repouso.

A solução pode passar pela criação de um ciclo que executa 3 vezes (porque são 3 objetos) e termina quando o robô regressa à posição de repouso.



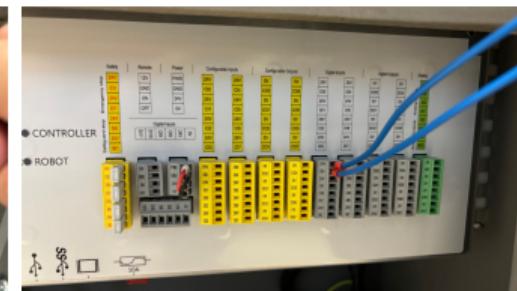
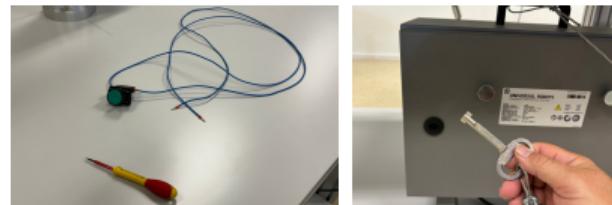
Exercício 2 - Ações condicionadas por IOs externos no UR3e

Descrição geral

- O objetivo do exercício é desenvolver um programa no robô UR3e utilizando o *teach pendant* (a consola de programação do robô) para executar diversos movimentos condicionados ao estado de IOs externos.
- Os passos mais relevantes em relação aos programas anteriores são:
 - Instalação de botões de pressão a simular IOs externos;
 - Instruções de espera ou de ação condicionada em função do estado dos IOs.

Ex. 2a) Instalação de um botão de pressão para simular um I/O

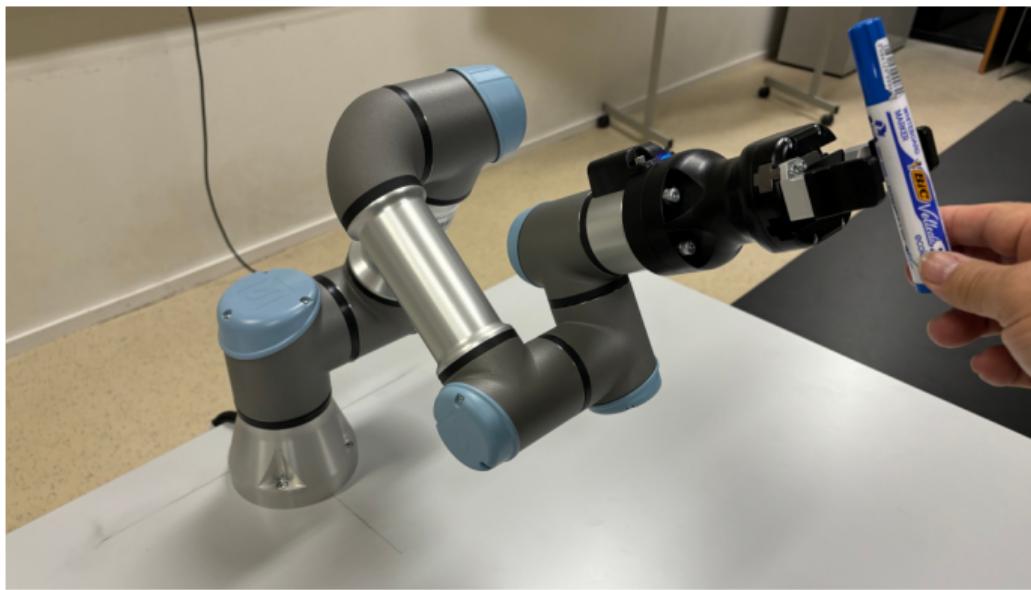
- Preparar um botão de pressão (normalmente aberto) com fios para ligar um I/O (DI0 - Digital In 0) do robô.
- Localizar e abrir o painel do controlador.
- Identificar o painel de terminais e retirar a primeira ficha dos IO cinzentos (a ficha de "Digital Inputs" cinzenta mais à esquerda).
- Nessa ficha, ligar os dois fios respetivamente a "24V" e a "DI0" com uma chave.
- Voltar a colocar o ficha de terminais no seu lugar.



- Verificar no menu dos I/Os na consola que o botão está funcional.

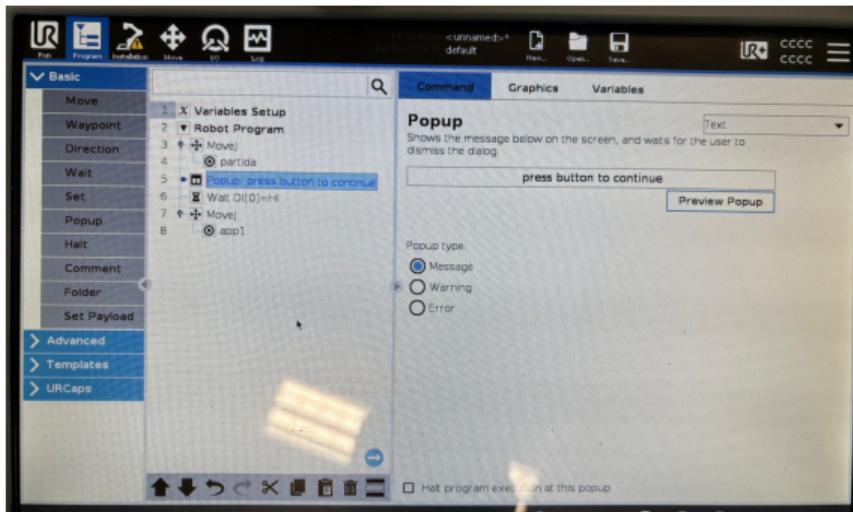
Ex. 2b) Programa de handover

- Criar um programa que se inicia na posição "Home" e se coloca numa posição pronta a receber do operador um objeto (e.g. uma caneta)
- Garantir que a garra fica aberta para a ação seguinte que é a de "dar" a caneta ao robô. Recomenda-se uma posição similar à ilustrada com um *waypoint* definido por *freedrive*.



Ex. 2c) O programa espera até haver sinal no DI0

- Inserir uma instrução que espera (Wait) até o Digital Input 0 (DI0) ficar ativo.
- Em opção pode-se usar uma caixa de mensagem a lembrar o operador para ativar a DI0
- Enquanto segura a caneta entre os dedos do gripper, o operador dá a ordem de preensão (fecho da garra) atuando o botão de pressão.
- O robô fecha a garra mas fica à espera que o operador volte a premir o botão.
- Em seguida, o robô continua e pousa a caneta noutro ponto junto ao tampo da mesa.



Exercício 3 - Múltiplos IOs e ações alternativas

- Este exercício tem a mesma base do exercício anterior mas...
- ... deve-se acrescentar um segundo botão (ligado ao DI1) para o robô executar tarefas alternativas:
 - 1 Quando se prime o botão 1 o robô larga peça num local A.
 - 2 Quando se prime o botão 2 o robô larga a peça num local B diferente de A.
 - 3 E ainda se poderia pensar numa 3^a ação que seria se os dois botões fossem atuados ao mesmo tempo, mas essa variante pode ser ignorada numa primeira fase.
- Os locais A e B devem ser *waypoints* definidos antecipadamente.

Exercício 4 - Programa com scripts para ações especiais no UR3

Desenvolver um programa que vai a um dado ponto com diferentes redundâncias de cotovelo mas dando apenas um waypoint.

- ① Definir manualmente (por juntas ou por freedrive) esse *waypoint* com cotovelo em baixo e dar-lhe um nome.
- ② Escrever um *script* que vai buscar o TCP que fica guardado como uma pose.
- ③ Ir para outro ponto (e.g. Home).
- ④ Regressar ao ponto original mas com cotovelo em cima.
 - Escrever um novo *script* para obter a cinemática inversa da pose guardada antes com o cotovelo em baixo, e guardar numa variável (qalt).
- ⑤ Criar outro *script* que faz um movej para qalt.
- ⑥ Mover para outro ponto (e.g. Home) e terminar.



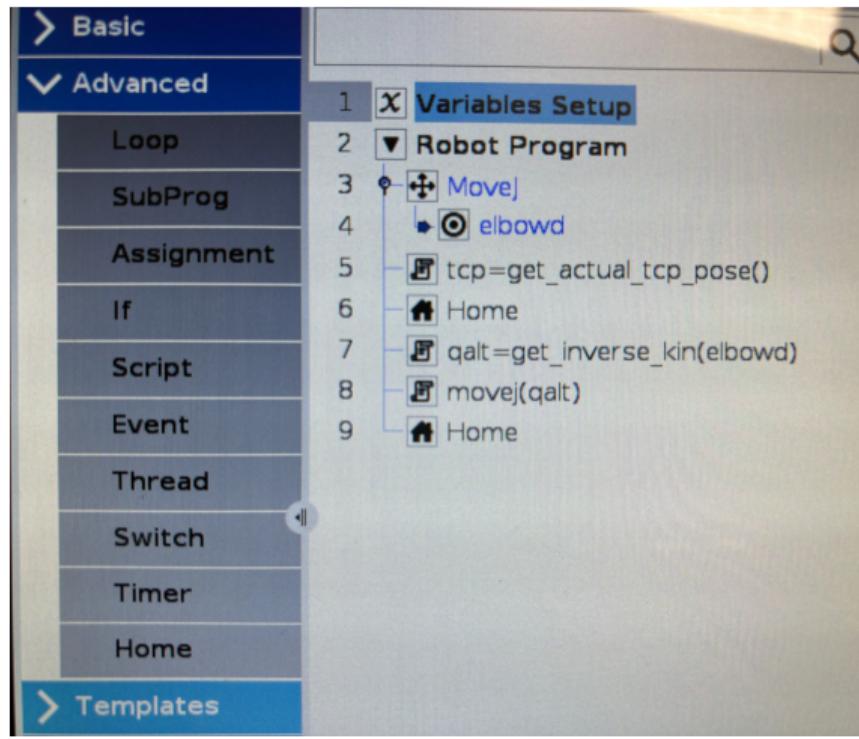
Exemplo de cotovelo em baixo



Exemplo de cotovelo em cima

Ex .4 (cont.) Excertos de um programa base para o exercício

Ilustração dos passos anteriores



Exercício 5 - Um programa integrador mais completo

Escrever um programa que inclua o maior número possível de funcionalidades estudadas até ao momento nos UR3e, nomeadamente:

- ① Inserção de dois botões como I/Os de entrada;
- ② Definir uma região da bancada onde será colocado um objeto.
- ③ Criar um ponto de aproximação para agarrar esse objeto da mesa mediante deteção de contacto com a mesa; sugere-se avançar com a garra aberta e detetar o contacto com a mesa (atenção à velocidade de avanço!)
- ④ Em estado de espera, e conforme o botão de pressão que o operador atuar, facultar ao operador o objeto numa posição de cotovelo em cima (botão 1) ou de cotovelo em baixo (botão 2).
- ⑤ Quando o operador tiver o objeto na mão, deve premir um botão de I/O (à escolha) para o robô abrir a garra e libertar o objeto.
- ⑥ Entretanto o operador coloca o objeto de novo na mesa e o processo repete-se em continuo.



Exercício 6 - Programa no UR3 que verifica o tipo de objetos colocado na garra e geras ações distintas conforme o objeto

- Desenvolver um programa que recebe um objeto do operador e conforme o curso da garra ao pegar nele faz ações diferentes.
- Uma sugestão é ter dois objetos de dimensões diferentes (que se possam distinguir pelo curso de fecho da garra): por exemplo uma caneta (marcador) e um objeto de dimensões diferentes (uma pequena caixa ou bloco, etc.)
- O bloco deve ser pousado num qualquer ponto da mesa
- O marcador deve ser usado para desenhar uma figura numa folha de papel e depois também pousado sobre a mesa

Nota: Se houver alguma dificuldade inesperada em distinguir os objetos pela garra, pode-se simplificar o exercício e a decisão é de dar o marcador ou de não dar objeto nenhum e o robô verifica apenas se tem objeto sem o "medir". Se a garra tiver o marcador)faz o que está descrito adiante, se não tiver, pode ir a um outro ponto pré-definido ou fazer outra tarefa ao critério do programador...



Descrição do programa geral da tarefa

- ① Robô começa num ponto inicial seguro (e.g. Home).
- ② Posiciona-se com a garra aberta para receber um objeto do operador.
- ③ O operador fornece-lhe um de dois objetos: o bloco ou o marcador, e este último deve ser-lhe dado de forma a que a ponta que escreve fique alinhada para vir a escrever.
- ④ O robô aguarda até o operador atuar um IO.
- ⑤ Quando o IO é atuado, o robô fecha a garra e verifica o objeto pelo curso da garra.
- ⑥ Se o objeto for o bloco, deve colocá-lo sobre a bancada num qualquer ponto a definir.
- ⑦ Se o objeto for o marcador, deve aproximar-se na vertical de uma folha de papel e prepara-se para desenhar uma linha.
- ⑧ Avança lentamente com o marcador até tocar no papel e aí faz movimentos lineares (por exemplo desenhar um quadrado).
- ⑨ No fim de desenhar deve ir pousar a caneta sobre a mesa mas numa posição horizontal para não escrever accidentalmente na mesa.
- ⑩ Depois desta tarefa (pousar o bloco ou o marcador), pode recomeçar o ciclo ou então terminar numa posição de repouso.



Ilustração de instruções para verificação da garra

Programa com elementos principais para verificar se a garra tem objeto (*Grip Check*)

