

Introdução à Robótica

Aula 06 – Controle Robótico



Professora: Danielle Casillo

Na aula de anterior ...

- **Sensores complexos**
 - Ultrassônicos
 - Lasers
 - Visão
- **Prática**
 - Livrando-se de obstáculos



Na aula de hoje ...

- **Controle Robótico**
 - Controle por realimentação
 - Os blocos Construtivos de Controle
 - Linguagem de Programação para robôs
 - Arquitetura de Controle
- **Prática**
 - Agarrar e liberar

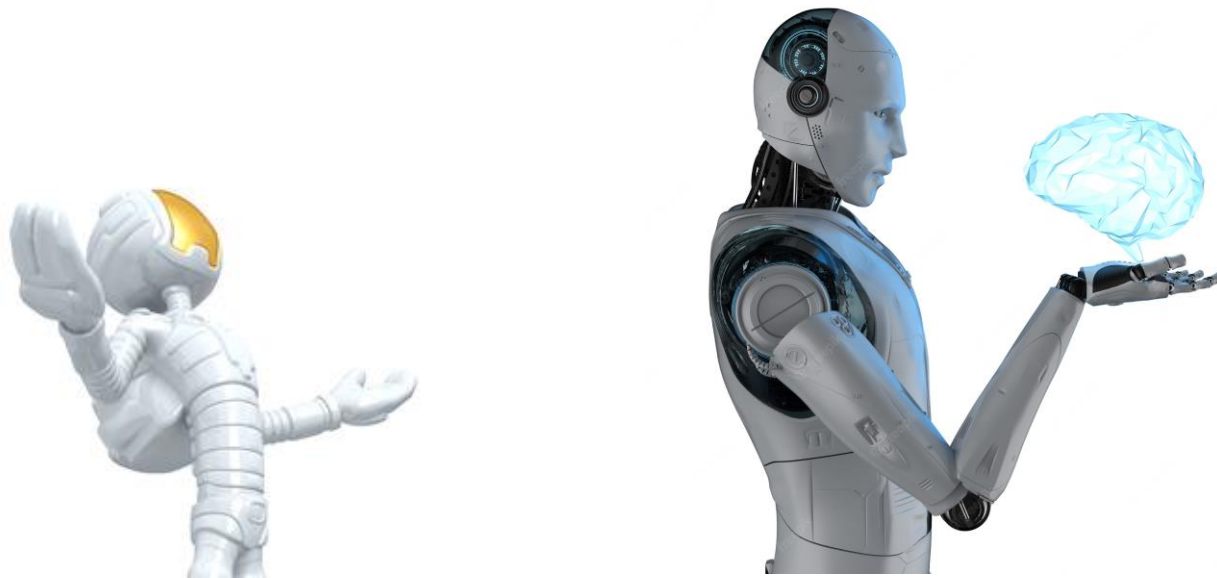


Capítulos 10, 11 e 12



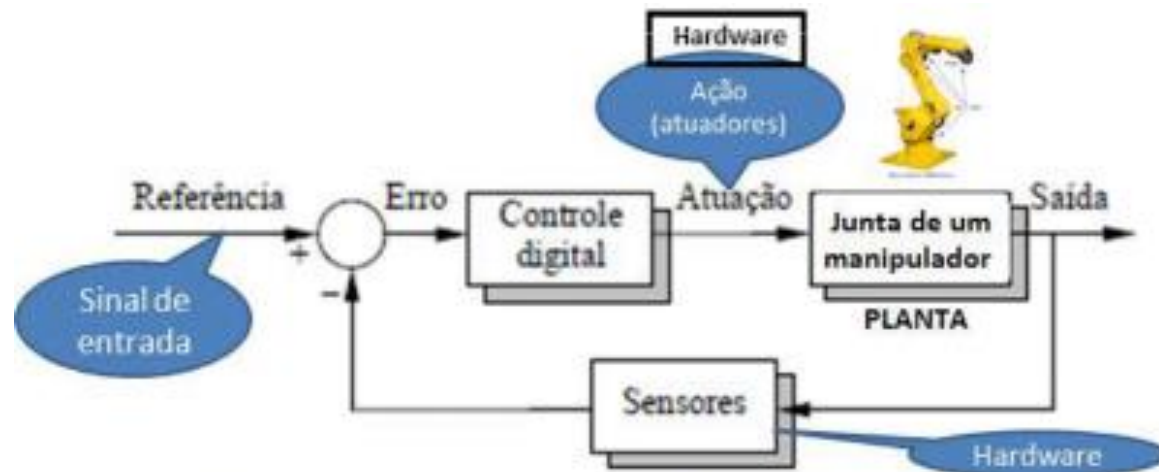
Mantenha o Controle!

Até aqui falamos a respeito dos corpos dos robôs, incluindo seus sensores e efetadores. Agora é hora de falar do cérebro do robô e dos controladores, que tomam decisões e comandam suas ações.



Controle por realimentação

- O controle por realimentação ou malha fechada ou *feedback* é uma forma pela qual um sistema (robô) atinge e mantém um estado desejado, geralmente chamado ponto de ajuste (*setpoint*), comparado continuamente o seu estado atual desejado.



- Realimentação (*feedback*) refere-se à **informação que é enviada de volta**, literalmente “retroalimentada”, ao controlador do sistema.

Controle por realimentação

O estado desejado do sistema, também chamado de estado-objetivo, é o estado ao qual o sistema deve chegar.



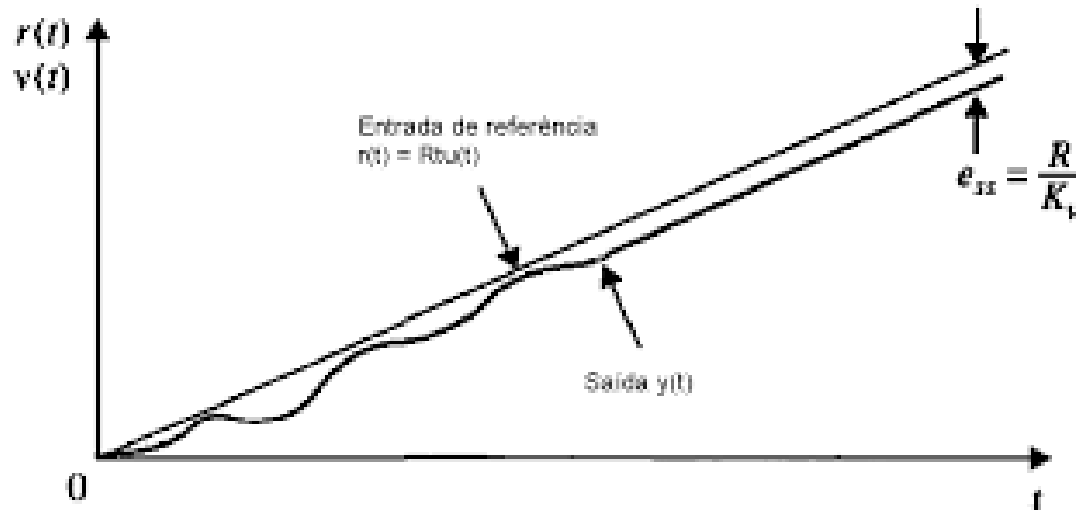
Objetivos de realização são os estados que o sistema tenta alcançar, tal como uma localização particular (saída de um labirinto).

Objetivos de manutenção, por outro lado, requerem a realização contínua de tarefas por parte do sistema (manter um robô bípede equilibrado e caminhando).



As diversas faces do erro

- A diferença entre os estados atual e desejado de um sistema é chamada **erro**, e o objetivo de qualquer sistema de controle é minimizar esse erro.



- O **controle por realimentação calcula o erro** e o informa explicitamente ao sistema, a fim de ajudá-lo a alcançar seu objetivo.



Exemplo de controle por realimentação

Como você projetaria o controlador de um robô para seguir uma parede usando o controle por realimentação?

- Para seguir uma parede, o estado-objetivo é estar a determinada distância ou dentro de um intervalo de distâncias de uma parede. Esse é um objetivo de manutenção (manter a distância ao longo do tempo).

Como determinar o erro neste caso?

- O erro é a diferença entre a distância desejada da parede e a distância real, em qualquer instante de tempo.



Exemplo de controle por realimentação

Qual(is) sensor(es) você usaria em um robô seguidor de parede e que informações ele(s) ofereceria(m)?

- Um **sensor de colisão** apenas diria ao robô que ele atingiu a parede, o robô poderia detectar a parede somente pelo contato.
- Um **sensor infravermelho** poderia fornecer informações de que existe uma parede, mas não a distância exata até ela.
- Um **sonar** proporcionaria a distância.
- Um **sistema de visão** além de propiciar a distância, saberia até mais sobre a parede.



Exemplo de controle por realimentação

Se **distância_da_parede** está no intervalo certo, então **continuar**.

Se **distância_da_parede** é maior que o desejado, então **virar para a parede**, senão, **afastar-se da parede**.

- Dado esse algoritmo de controle, como será o comportamento do robô?
 - Ele vai continuar ziguezagueando de um lado para outro enquanto se locomove beirando a parede

Quantos ziguezagues ele fará?

- Isso depende de 2 parâmetros: quantas vezes o erro é calculado e quantas correções (viradas) serão feitas.



Exemplo de controle por realimentação

Como podemos diminuir essa oscilação?

- A primeira delas é fazer a **análise do erro** com maior frequência, de modo que o robô possa virar muitas vezes, em vez de raramente.
- Outra é **ajustar o ângulo do giro**, de modo que o robô gire ângulos pequenos.
- Outra é encontrar um **intervalo de distâncias** que definam o objetivo do robô.



Os blocos construtivos do controle

- A maioria dos robôs tem muito mais coisas para fazer do que apenas seguir uma parede ou evitar obstáculos.
- A **arquitetura de controle** fornece princípios norteadores e limitações para organizar o sistema de controle de um robô (cérebro). A arquitetura ajuda o projetista a programar o robô de uma forma que produzirá o comportamento de saída desejado.



Os blocos construtivos do controle

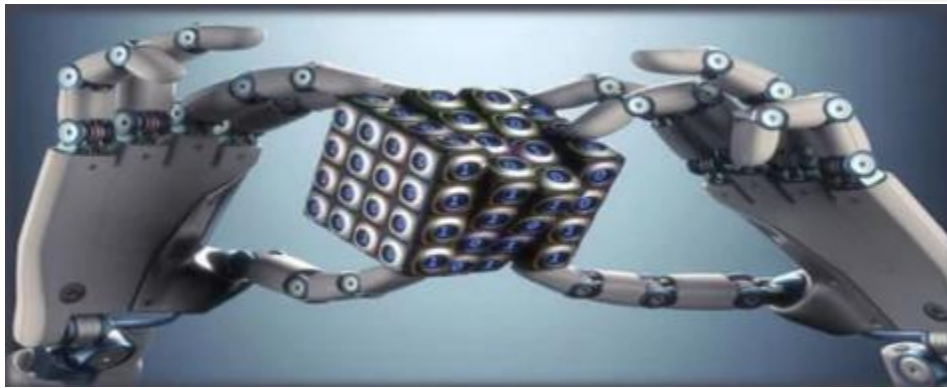
O controle de robôs pode ser executado em *hardware* e em *software*; porém, quanto mais complexo o controlador, é mais provável que ele seja implantado em *software*. Porque?

- O *hardware* é bom para aplicações rápidas e especializadas, e o *software* é bom para programas flexíveis, mais genéricos.
- Aí vem a Ciência da Computação que dedica grande parte de sua pesquisa ao desenvolvimento e análise de algoritmos e a robótica se preocupa com isso como a navegação, manipulação, aprendizagem, entre outros.



Linguagens de programação para robôs

Qual a melhor linguagem de programação de robôs?



- Os programadores de robôs usam uma variedade de linguagens que dependem da tarefa do robô, daquilo que estão acostumados a usar.
- Qualquer linguagem de programação que se preze é chamada **"Máquina de Turing universal"**, o que significa que, pelo menos em teoria, ela **pode ser usada para escrever qualquer programa.**



E as arquiteturas são...

- Seja qual for a linguagem usada para programar um robô, **o que importa é a arquitetura de controle** utilizada para implementar o controlador.
- As principais arquiteturas de controle de robôs são: **deliberativa** (não utilizada), **reativa**, **híbrida** e **baseada em comportamentos**.
- Antes de estudar as arquiteturas, vamos ver quais são as questões importantes a serem consideradas na decisão de qual arquitetura usar:



E as arquiteturas são...

- Existe muito ruído de sensor?
- O ambiente se modifica ou permanece estático?
- O robô pode sentir todas as informações de que necessita? Se não, quanto pode sentir?
- Com que rapidez o robô sente?
- Com que rapidez o robô age?



E as arquiteturas são...

- Existe muito ruído no atuador?
- O robô precisa se lembrar do passado para fazer seu trabalho?
- O robô precisa pensar no futuro e prever as coisas para fazer seu trabalho?
- O robô precisa melhorar o seu comportamento ao longo do tempo e ser capaz de aprender coisas novas?



E as arquiteturas são...

- As arquiteturas de controle de robôs diferem substancialmente na forma como lidam com o tempo, a modularidade e a representação.

Modularidade: refere-se a maneira como o sistema de controle é quebrado em pedaços ou componentes (módulos), e como esses módulos interagem entre si para produzir o comportamento geral do robô.

Escala de tempo: refere-se a rapidez com que ele pode sentir e pensar.

Representação: é a forma pela qual a informação é armazenada ou codificada no robô.

Representação

O que se passa em sua cabeça?

- Representação é mais do que memória. É a forma como a informação é armazenada no robô.
- “O que é” e “Como é” representado tem um grande impacto sobre o controle do robôs.
- “O que está em seu cérebro influencia o que você pode fazer”.



Próxima aula....

- **Tipos de Controle**
 - Controle Deliberativo
 - Controle Reativo
 - Controle Híbrido
 - Controle Baseado em Comportamentos
- **Prática**
 - Ângulos e Padrões



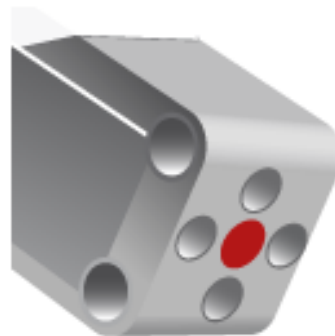
Prática

Agarrar e Liberar



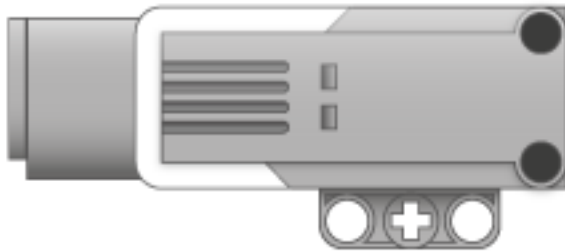
Bloco Motor Médio

- O motor médio pode ligar ou desligar, controlar seu nível de potência ou ligar o motor por um período de tempo ou rotações específicas.
- A entrada de potência aceita um número de -100 a 100.



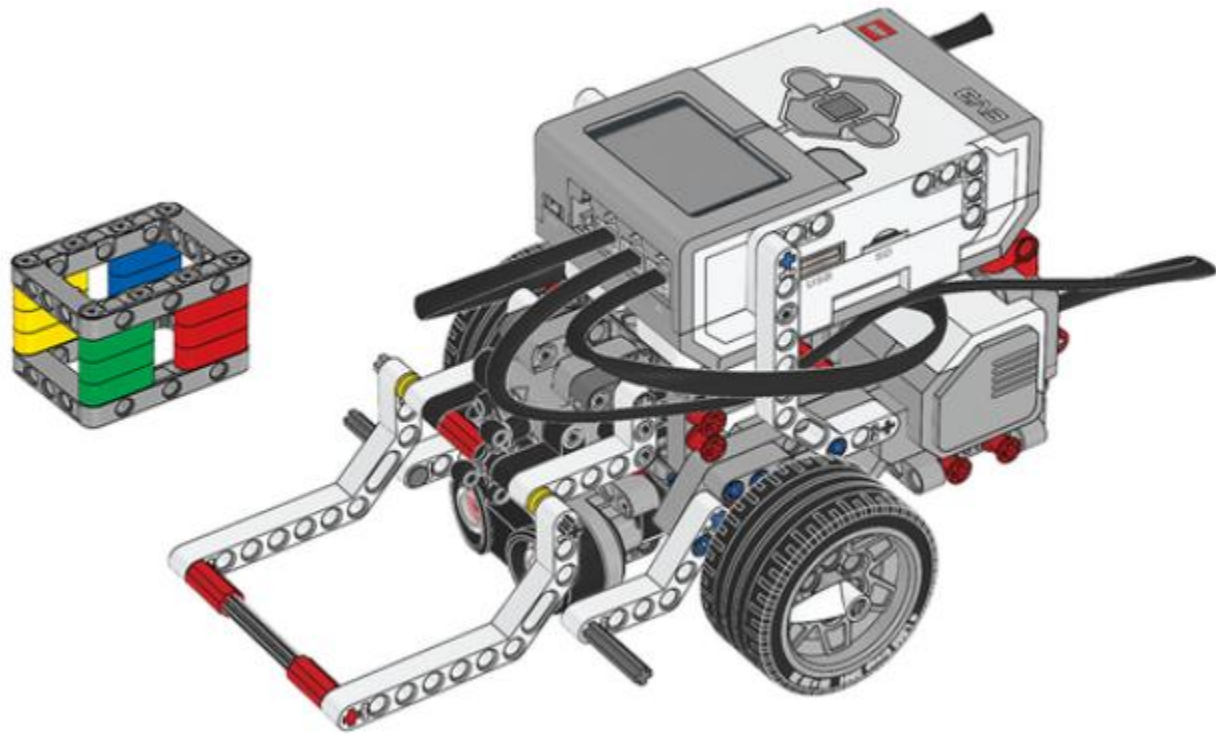
Bloco Motor Médio

- Um número positivo gira o motor médio no sentido horário e um número negativo o gira no sentido anti-horário.



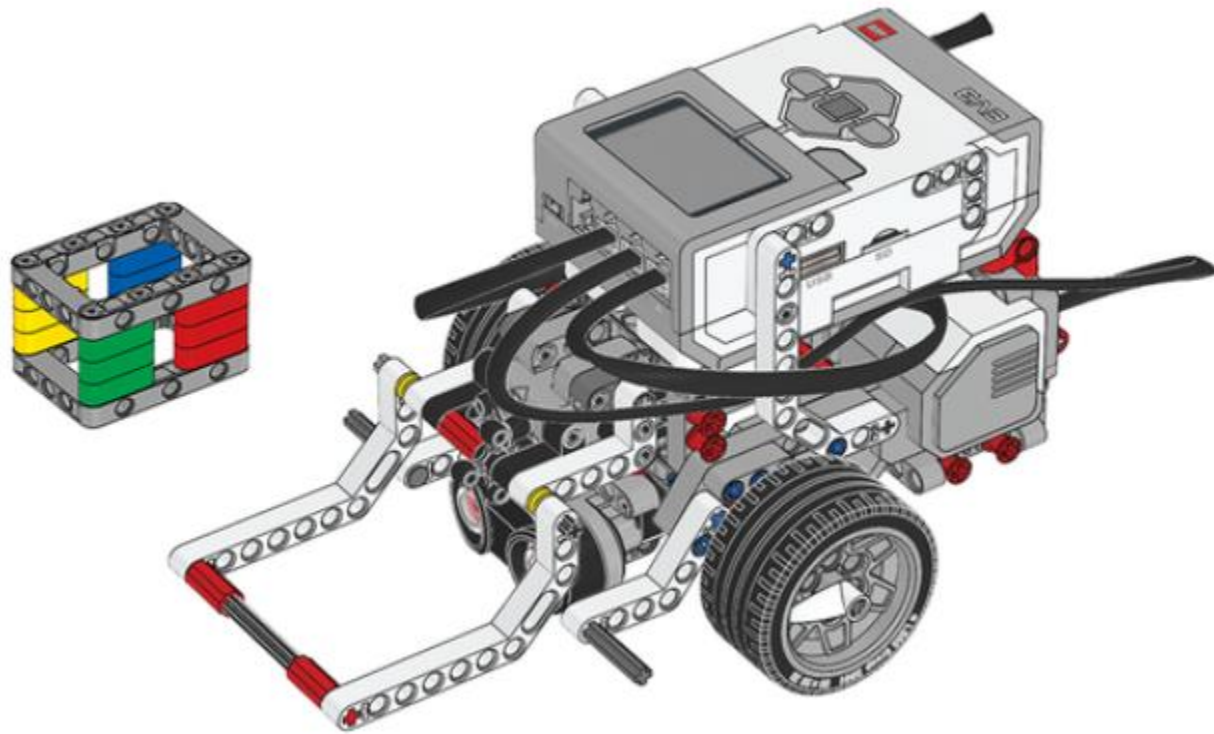
Bloco Motor Médio

- Montagem



Bloco Motor Médio

- Vídeo



Bloco Motor Médio

- Programação

