UNIVERSIDADE FEDERAL DE PERNAMBUCO

Centro de Informática

Graduação em Engenharia da Computação

Métodos de Aprendizagem de Controle Local Aplicados para Robótica

**Aluno:**Jucemberg Urbano da Silva

**Orientador:**HansencleverBassani

13 de maio de 2015

Sumário

[1. Contexto 3](#_Toc418750134)

[2. Objetivo 4](#_Toc418750135)

[3. Metodologia 5](#_Toc418750136)

[4. Cronograma 8](#_Toc418750137)

[5. Possíveis Avaliadores 9](#_Toc418750164)

[6. Referências 10](#_Toc418750190)

[7. Assinaturas 11](#_Toc418750191)

# Contexto

Os robôs estão cada vez mais ganhando espaço em indústrias e no uso doméstico. Estes robôs podem ser divididos de acordo com a forma de programação que é utilizada para efetuar suas ações. Alguns robôs, principalmente os utilizados em indústrias, têm suas ações pré-definidas em sua programação. Assim, suas ações são repetitivas e suscetíveis a falhas devido a problemas como desgastes de peças ou alguma alteração no ambiente.

Por outro lado, existem os que possuem a capacidade de aprendizado continuo. Ou seja, eles conseguem perceber as mudanças no ambiente e alterar suas ações. Assim, estes robôs são chamados de autônomos e sua programação é feita com alguma técnica de Computação Inteligente.

Estas técnicas são de uso interessante para casos em que:

1. Os dados são abundantes;
2. Sistemas de usos contínuos, onde ocorre degradação no decorrer do tempo;
3. O sistema precisa ser robusto para mudanças nas distribuições das entradas;
4. Precisa detectar entre um numero grande de entradas características relevantes e excluir as redundantes e irrelevantes.

O ramo da Aprendizagem de Máquina proporciona técnicas para encontrar funções que descrevam o comportamento do sistema. Algumas técnicas utilizadas são: aproximador de funções, aprendizagem por recompensa, redes neurais, redes difusas (*fuzzy*), etc.

Outra característica importante é a forma do aprendizado, que pode ser supervisionado ou não supervisionado. Na primeira forma um analista ajuda de alguma forma na aquisição de conceitos ou no conhecimento estruturado do sistema, um exemplo é a logica *fuzzy*. Na segunda, o sistema precisa interagir com o ambiente e descobrir os padrões, gerando assim o seu próprio controle.

Segundo Schaae Atkeson[1]podemos descrever um sistema de controle típico com a seguinte equação:

Onde:

x= estado do robô

u = Sinais de controle

t = tempo

r = ruído

Onde:

t = tempo

= parâmetros de controle do sistema

Assim o problema consiste em descobrir a função u que é a função de controle do sistema, que produz os sinais de controle em função de , e .

Em sistemas reais, acabamos por encontrar uma não linearidade na função , o que dificulta a modelagem. Uma solução para este problema é reduzir e dividir o sistema em partes pequenas, até que cada parte se torne linear. Assim chegamos á um controle local,que serve como subsídio para gerar um controle global o qual desempenha funções de maior complexidade.

Neste trabalho utilizaremos um robô móvel real do tipo “*differential drive*”, que é um veículo com duas rodas com tração e uma roda livre, onde a diferença de velocidade entre as rodas com tração modifica a direção; e o simulador de robôs V-REP, para simular um braço robótico. O V-REP é uma ferramenta robusta de simulação. Ela traz uma abordagem visual e com suporte a várias linguagens de programação.

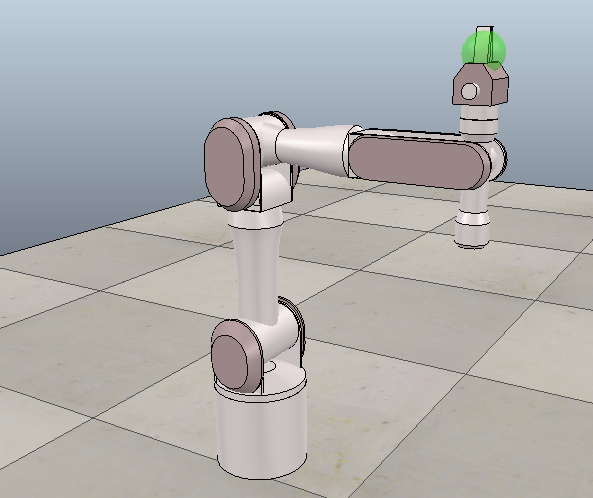


Figura 1- Exemplo de braço robótico do simulador V-REP

# Objetivo

O objetivo geral do presente trabalho é desenvolver um sistema aprendizagem de controle local que seja aplicável tanto para robôs móveis quando para manipuladores robóticos.

Os objetivos específicos são:

1. Estudar e descrever os principais métodos para aprendizagem de controle local;
2. Selecionar, Implementar e testar os métodos mais adequados para os problemas alvo;
3. Descrever os problemas encontrados;
4. Propor melhorias para os métodos atuais ou propor um novo método de aprendizagem de controle local,se for o caso.

# Metodologia

Na literatura há diversos métodos supervisionados e não supervisionados que podem solucionar o problema de encontrar o controlador local. Neste trabalho o foco será voltado para as redes neurais artificiais, sendo que serão considerados os seguintes modelos:Mapas Auto Organizáveis (SOM – do inglês *Self-OrganizingMaps*)[2][3], Redes PercepronMulti-Camadas (MLP)[4] e Redes Fuzzy[5].

As redes SOM foram propostas por Kohonen[3]. Está técnica propõe um mapeamento de um sistema de ordem elevada eu uma estrutura de dimensão inferior a original. Este mapeamento guarda a relação de vizinhança dos dados originais.

A Figura 2abaixo mostra uma arquitetura típica de mapa auto-organizável:

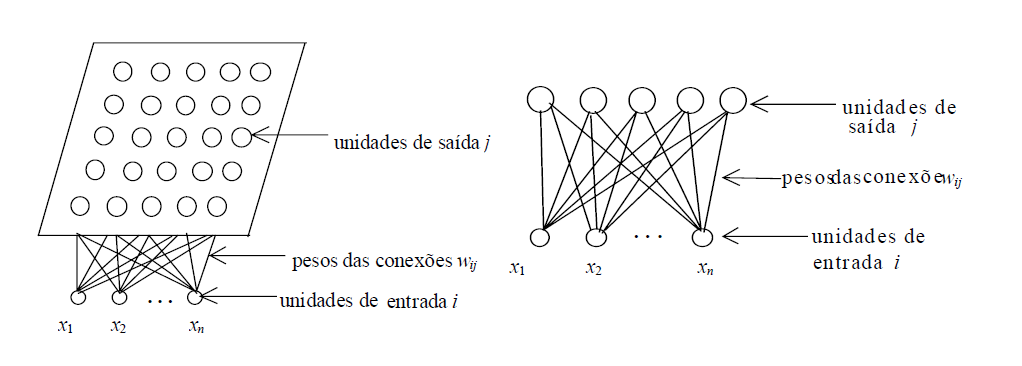


Figura 2 – Arquitetura de uma Rede SOM.

As Redes MLP [4]consistem em uma variação de rede neural clássica.Nela existem mais de uma camada no nível intermediário. Esta característica faz com que a rede consiga trabalhar com sistemas mais complexos que uma rede neural simples. Seu treinamento mais comum é com o algoritmo de *backpropagation*. Neste algoritmo um conjunto de sinais é imposto na entrada e segue o percurso até a saída, no final é comparado com o valor esperado, e o erro resultante é utilizado para ajustar os pesos das conexões (Figura 3).

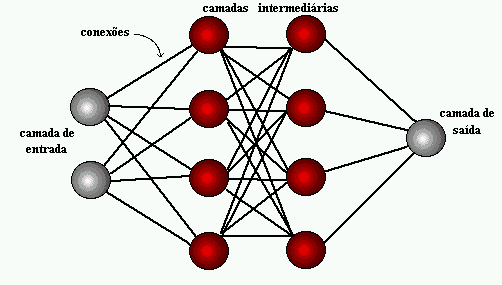


Figura 3 – Arquitetura de uma Rede MLP.

As Redes Fuzzy [5] têm como base a logica difusa. Este tipo de lógica utiliza níveis intermediários entre verdadeiro e falso da logica clássica. Assim, uma dada informação é dita verdadeira com um peso associado.

Figura 4 – Exemplo de Sistema Fuzzy.

Um sistema inteligente fuzzy pode ser visto como a Figura 4. Podemos notar três camadas: fuzzificação, inferência e defuzzificação. Na fuzzificação os dados de entrada são transformados na lógica difusa, na inferência são aplicadas as regras difusas e o resultado é defuzzificado pra podermos obter a saída.

Nos três casos descritos acima, os sinais de saída destas redes neurais serão utilizados como sinais de controle do sistema () e os sinais de entrada serão informações do ambiente, e parâmetros de controle do robô. Como uma simplificação, neste trabalho não será considerada o tempo.

# Cronograma

O cronograma abaixo descreve as atividades que serão realizadas durante este trabalho ao longo do tempo:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Atividade** | **Março** | **Abril** | **Maio** | **Junho** | **Julho** |
| Pesquisa do estado da arte |  |  |  |  |  |
| Implementação e testes dos métodos |  |  |  |  |  |
| Proposição de melhorias |  |  |  |  |  |
| Escrita do Relatório |  |  |  |  |  |
| Preparação da apresentação Oral |  |  |  |  |  |

# Possíveis Avaliadores

* Aluizio Fausto Ribeiro Araujo
* Carlos Alexandre Barros Mello

# Referências

1. Schaa e Atkeson, 2010 , Learning Control in Robotics, IEEERobotics& Automation Magazine
2. Chi-Hsu Wang, 2012,Adaptive SOM-BasedFuzzyNeural Network Controller Design for Multi-Agent System DispatchingandPath ,IEEE World CongressonComputationalIntelligence
3. Kohonen, Teuvo,1982. "Self-OrganizedFormationofTopologicallyCorrectFeatureMaps". *BiologicalCybernetics*
4. Rosenblatt, Frank. X, 1961. PrinciplesofNeurodynamics: PerceptronsandtheTheoryofBrainMechanisms. Spartan Books, Washington DC
5. Marcelo Godoy Simões, lanS.Shaw, Controle e Modelagem Fuzzy,1 Edição, 2007

# Assinaturas

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

HansencleverBassani

**Orientador**

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Jucemberg Urbano da Silva

**Aluno**