



O SIMULADOR DE ROBÔS V-REP NO ENSINO DAS TÉCNICAS DE CONTROLE DE ROBÔS AUTÔNOMOS

Ricardo Kerschbaumer – ricardo@luzerna.ifc.edu.br
IFC - Instituto Federal de Educação Ciência e Tecnologia Catarinense
Rua São Roque, 41
89609-000 – Luzerna – SC

Carlos Raimundo Erig Lima – erig@utfpr.edu.br
UTFPR - Universidade Tecnológica Federal do Paraná
Av. Sete de Setembro, 3165
80230-901 – Curitiba – PR

Jean Marcelo Simão – jeansimao@utfpr.edu.br

UTFPR - Universidade Tecnológica Federal do Paraná
Av. Sete de Setembro, 3165
80230-901 – Curitiba – PR

Resumo: O ensino das técnicas de controle utilizadas em robôs autônomos é dificultado pela variedade de robôs existentes e pela diversidade de formatos que estes assumem. Construir dispositivos físicos onde se possa realizar experimentos requer tempo e recursos, o que normalmente resulta em experimentos limitados a poucos modelos de robôs, com poucos sensores. Alguns sensores costumam ter custo elevado e sua utilização em robôs autônomos nem sempre é simples. A utilização de um simulador de robôs é uma alternativa que contorna ambos os problemas. Em um ambiente simulado é possível utilizar uma grande diversidade de robôs sem custo. O número e a variedade de sensores utilizados em cada robô também pode ser muito maior em um ambiente virtual. Ilustrando esta ideia, este artigo descreve os experimentos e a análise dos resultados da implementação da lógica fuzzy no controle de um robô hexápode seguidor de linha. O robô foi simulado através do software V-Rep. Este controlador foi desenvolvido para a disciplina de Programação Avançada do Programa de Pós-Graduação em Engenharia Elétrica e Informática Industrial da Universidade Tecnológica Federal do Paraná. E foi também apresentado como projeto final na disciplina de Inteligência Artificial do mesmo programa e pode ser aplicado para o ensino de técnicas avançadas para controle de robôs móveis nos cursos de engenharia da UTFPR. O objetivo deste projeto foi desenvolver um controlador utilizando lógica fuzzy, capaz de controlar um robô hexápode seguidor de linhas, em um ambiente virtual. Após a implementação, foram realizados testes que demonstraram as funcionalidades do controlador, bem como do simulador e do robô virtual.



Palavras-chave: Robô hexápode, Controlador fuzzy, Simulador de robôs.

1. INTRODUÇÃO

A Universidade Tecnológica Federal do Paraná (UTFPR) fornece ao público brasileiro e a comunidade internacional vários cursos de engenharia. Hoje, são ofertados cursos de Engenharia Elétrica, Engenharia Mecânica, Engenharia de Controle e Automação e Engenharia da Computação. Nestes cursos há a demanda pelo ensino de sistemas robóticos como elementos centrais na formação de competências esperadas nos egressos das engenharias. Muitos dos experimentos propostos para exploração de todos os conhecimentos apresentados são abordados na forma de projeto baseados em plataformas de desenvolvimento baseadas em robôs móveis ou robôs autônomos, utilizando diferentes ferramentas de auxílio ao projeto. Exemplos de disciplinas que usam esta abordagem são: Circuitos Digitais, Microcontroladores, Arquitetura e Organização de Computadores, Controle 2, Lógica Reconfigurável e disciplinas optativas de robótica e inteligência artificial, entre outras. Por outro lado, os diversos programas de pós-graduação da UTFPR, oportunizam para seus alunos projetos de disciplinas que geram projetos perfeitamente aplicados como ferramentas de suporte ao ensino das engenharias. Dentro deste contexto, este trabalho apresenta a descrição de um simulador de robôs para avaliação de uma técnica particular de controle de trajetória.

A aplicação da lógica *fuzzy* no controle de um robô hexápode tem por objetivo demonstrar que é possível utilizar um ambiente virtual de simulação de robôs para realizar experimentos de controle. Para tal realizou-se o desenvolvimento de um software, utilizando linguagem C++, aplicando lógica fuzzy no controle de um robô hexápode seguidor de linhas. Tal programa permitiu comandar um robô virtual simulado no ambiente V-Rep. A interface do programa apresenta gráficos de performance do controlador e do robô.

Nos próximos capítulos serão tratados assuntos relacionados aos elementos utilizados no projeto, ao funcionamento do controlador e como ele se relaciona com o robô e seu simulador.

2. FUNDAMENTAÇÃO

Na realização deste projeto foram utilizados vários elementos distintos, tais como, o simulador de robôs, o controlador fuzzy, o robô hexápode, dentre outros. A seguir serão abordados alguns destes elementos.

2.1. Robô hexápode

O robô utilizado neste projeto é um robô hexápode, e como o próprio nome sugere este robô possui seis pernas, cada uma composta por três servomotores. Além das pernas o robô possui um corpo, ao qual estão conectados três sensores ópticos e um sensor ultrassônico. O sensor ultrassônico serve para evitar colisões e tem pouca relevância neste projeto. Já os sensores ópticos são utilizados para seguir a linha e tem grande importância no controle do robô. Estes sensores medem a intensidade de reflexão da luz em três pontos

distintos da superfície onde o robô se encontra, possibilitando assim a identificação da linha. A Figura 1 mostra a estrutura do robô utilizado.

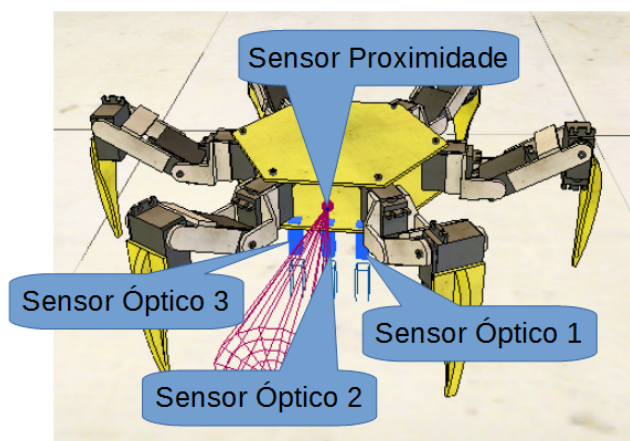


Figura 1 - Robô utilizado no projeto.

Cabe ao software de controle adquirir os sinais dos sensores e determinar a posição angular dos dezoito motores que compõem o robô de forma que o mesmo percorra a linha da forma mais rápida possível e com o mínimo de desvio.

2.2. Lógica fuzzy

A Lógica fuzzy é uma teoria matemática que tem como principal objetivo imitar a habilidade humana de tomar decisões em ambientes de incerteza e imprecisão. Com estes conceitos pode-se construir sistemas inteligentes de controle e suporte à decisão que lidem com informações imprecisas e subjetivas. Estas informações são expressões linguísticas cuja interpretação pode variar de um indivíduo para outro (GOLDSCHMIDT, 2010). Ou seja, a lógica *fuzzy* reflete a maneira como as pessoas pensam, tentando modelar o seu senso de palavras, tomada de decisão ou senso comum. O termo *fuzzy* na língua inglesa pode ter vários significados, e sua tradução ainda não é unanimidade: “nebuloso” e “difuso” são os exemplos mais populares de traduções para *fuzzy*.

O processo de inferência *fuzzy* consiste em receber os valores de entrada e apresentá-los aos conjuntos *fuzzy* de entrada através de um processo chamado de “fuzzificação”. Em seguida estes valores “fuzzificados” são mapeados dos conjuntos de entrada para os conjuntos de saídas através de um conjunto de regras. Os conjuntos de saída através de um processo chamado “defuzzificação” fornecem então os valores de saída. A Figura 2 apresenta um diagrama que ilustra o processo descrito.

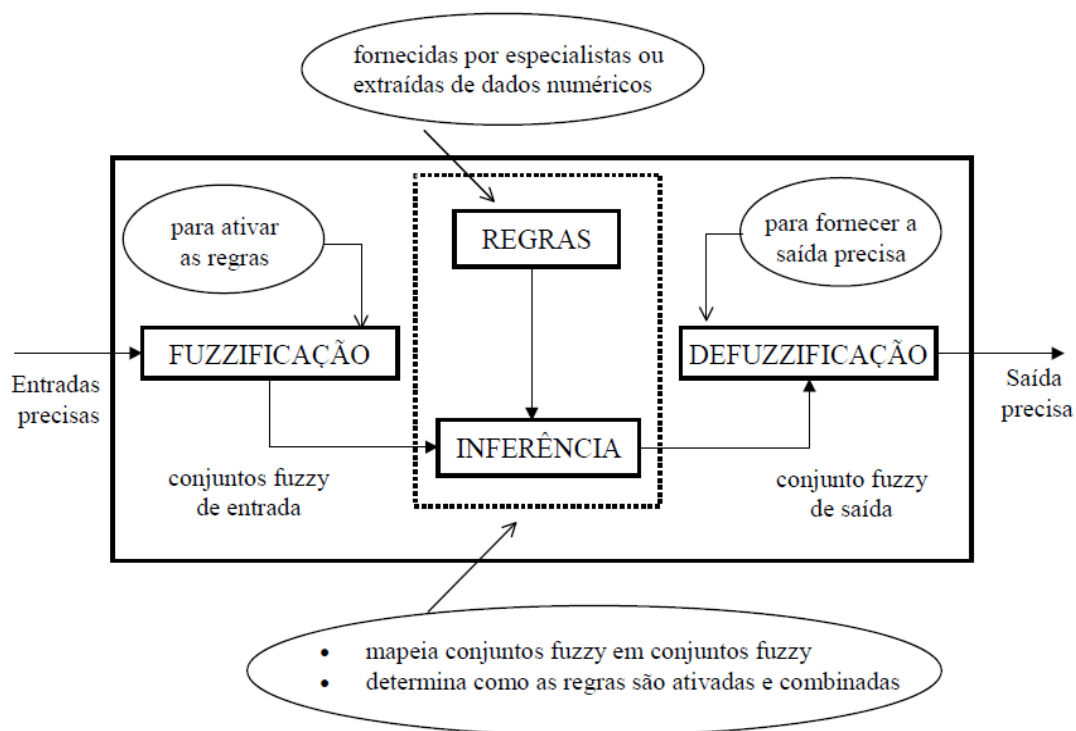


Figura 2 - Processo de inferência *fuzzy*.

2.3. O simulador de robôs V-Rep

O robô foi simulado no ambiente de simulação de robôs V-rep. Este ambiente foi escolhido por ter as características necessárias ao projeto e por possibilitar o desenvolvimento de vários outros projetos no futuro. A Figura 3 apresenta a interface deste programa. É importante salientar que o V-Rep é gratuito para fins educacionais.

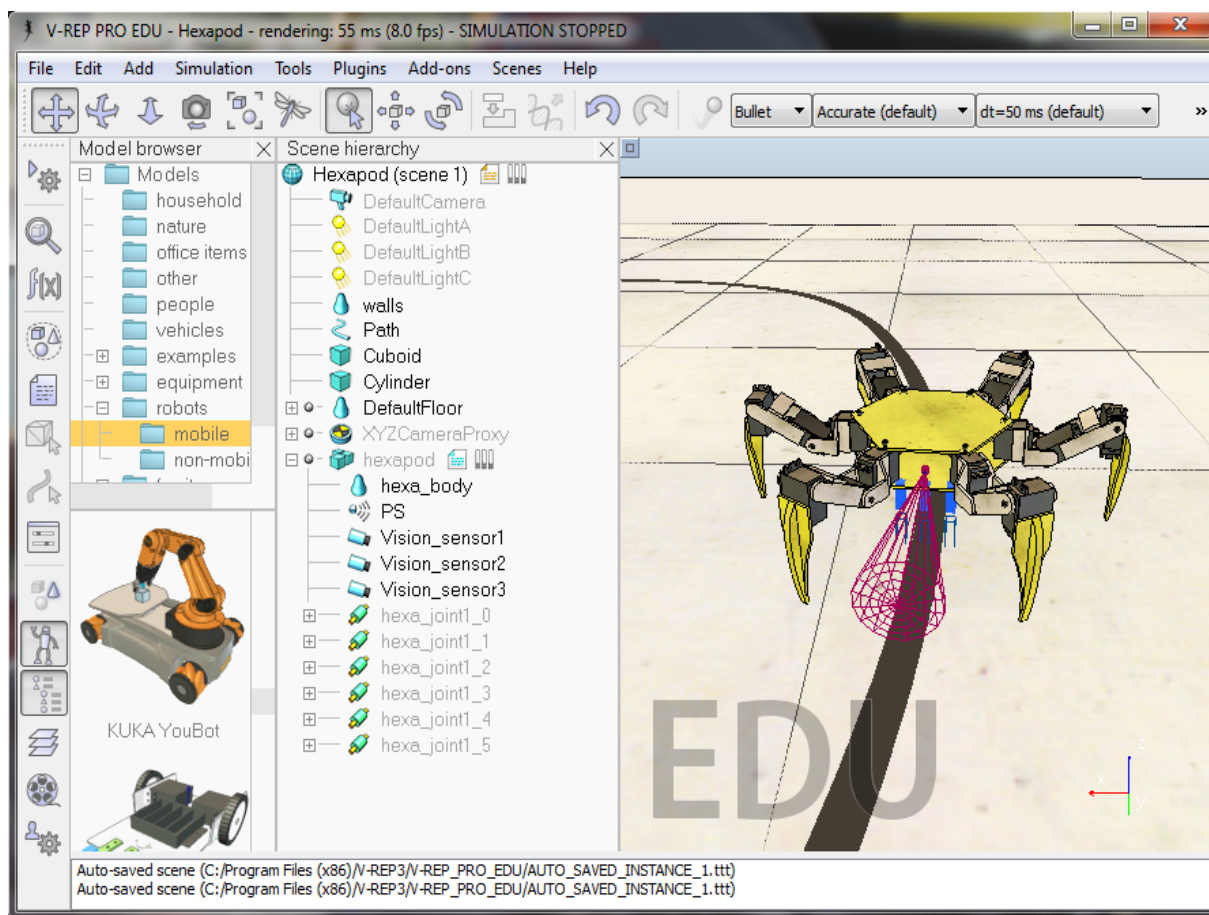


Figura 3 - A interface do V-Rep.

O V-Rep apresenta várias formas de controle dos robôs, seja via programação de scripts em linguagem LUA no próprio ambiente ou via APIs para várias linguagens de programação. (COPPELIA ROBOTICS, 2014).

3. OBJETIVOS

O principal objetivo deste projeto foi controlar um robô hexápode seguidor de linhas em um ambiente virtual utilizando lógica *fuzzy*. Para tal outros objetivos foram traçados, dentre eles pode-se citar: adaptar um robô hexápode do simulador de robôs V-Rep para ser controlado externamente, criar uma interface serial que conecte o simulador de robôs ao software controlador e desenvolver o software de controle utilizando lógica *fuzzy*.

Robôs seguidores de linha devem percorrer um circuito determinado por uma linha escura em uma superfície clara no menor tempo possível com o menor desvio possível da linha. (GLOWICKI, BUTKIEWICZ, 2013).

4. IMPLEMENTAÇÃO

4.1. O controlador *fuzzy*

O software desenvolvido implementou a lógica de controle utilizando três conjuntos *fuzzy* para cada um dos sensores de entrada, com os termos linguísticos, Baixo, Médio e Alto representando a intensidade de luz refletida pela superfície em cada um dos sensores ópticos. A Figura 4 apresenta estes conjuntos.

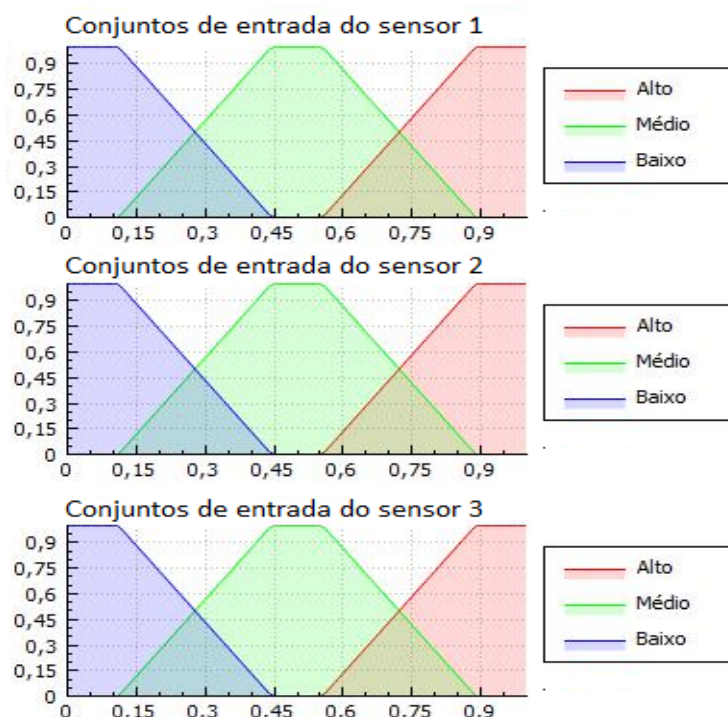


Figura 4 - Conjuntos *fuzzy* das entradas

Na saída foram utilizados os termos linguísticos, Gira a esquerda, Vira a esquerda, Anda, Vira a direita e Gira a direita representando a direção que o robô deve tomar para permanecer sobre a linha. A Figura 5 apresenta os conjuntos utilizados na saída.

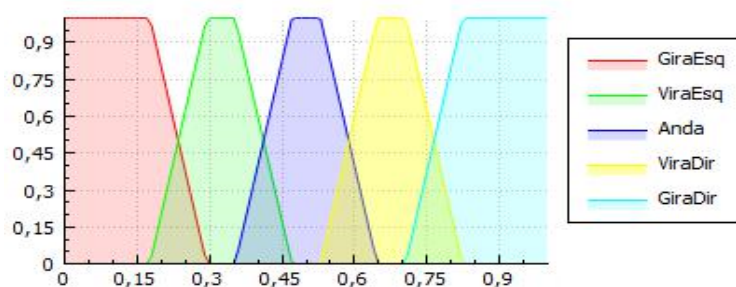


Figura 5 - Conjuntos *fuzzy* de saída.

É importante salientar que o controlador *fuzzy* implementado não é diretamente responsável pelo acionamento dos servomotores. O controlador tem como saída um sinal que determina qual a rota que o robô deve seguir.

A determinação do ângulo de cada um dos dezoito motores bem como o controle das posições das pernas são determinadas por um algoritmo especialmente desenvolvido para tal fim.

O algoritmo de controle dos servomotores das pernas recebe como entrada a direção que o robô deve seguir. Este sinal vem do controlador e diz se o robô deve andar em frente ou virar para um dos lados, bem como qual o raio da curva a ser realizada. O algoritmo então ajusta a amplitude dos passos em cada um dos lados do robô de forma a seguir a ordem recebida.

4.2. A interface entre o controlador e o simulador

O software de controle é um programa totalmente independente do simulador, assim foi necessário criar uma forma de conecta-lo ao V-Rep. Como foi citado anteriormente o V-Rep permite que se desenvolva scripts em linguagem LUA para controlar os robôs e a simulação como um todo.

Também são disponibilizados pelo fabricante do V-Rep várias APIs que permitem desenvolver programas externos de controle em linguagens C/C++, Python, Java, Lua, Matlab, dentre outras. Optou-se neste projeto em implementar uma conexão serial entre o controlador e o simulador.

A interface foi desenvolvida utilizando-se uma conexão serial por dois motivos. Primeiro, se o controlador desenvolvido é capaz de controlar um robô virtual recebendo sinais dos sensores e enviando sinais aos motores pela porta serial, ele também pode controlar um robô real que forneça as mesmas conectividades em uma interface serial. Segundo, se o robô utilizado no simulador pode interagir com um controlador em software pela porta serial, ele também pode interagir com um controlador desenvolvido em hardware que possua uma interface serial. Assim é possível testar controladores em diversas plataformas, como por exemplo, programas de computador, sistemas microcontrolados, sistemas em lógica programável entre outros. A Figura 6 ilustra a interface desenvolvida.

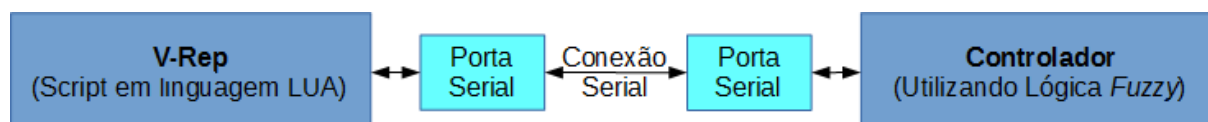


Figura 6 - Interface entre o controlador e o simulador.

Para que o simulador interpretasse os comandos enviados pelo controlador foi desenvolvido um script em linguagem LUA. O script não forneceu ao robô nenhum tipo de controle, ele apenas foi responsável por receber e tratar os sinais enviados pelo controle aos motores e fornecer ao controle os sinais oriundos dos sensores.

O script desenvolvido pode ser facilmente alterado para ser utilizado em outros robôs, bem como para permitir acesso a vários robôs simultaneamente.

5. RESULTADOS

O robô adaptado para ser utilizado no projeto apresentou todas as características de um robô real, permitindo assim que se avalie o desempenho do controlador.

O script desenvolvido para conectar o simulador a porta serial funcionou perfeitamente, permitindo que o controlador acessasse o robô no simulador sem dificuldades.

O software desenvolvido para o controle do robô possui interface gráfica onde são apresentados dinamicamente os valores das entradas, os conjuntos ativados pelas regras, o valor de saída entre outras informações. A Figura 7 apresenta a interface do software.

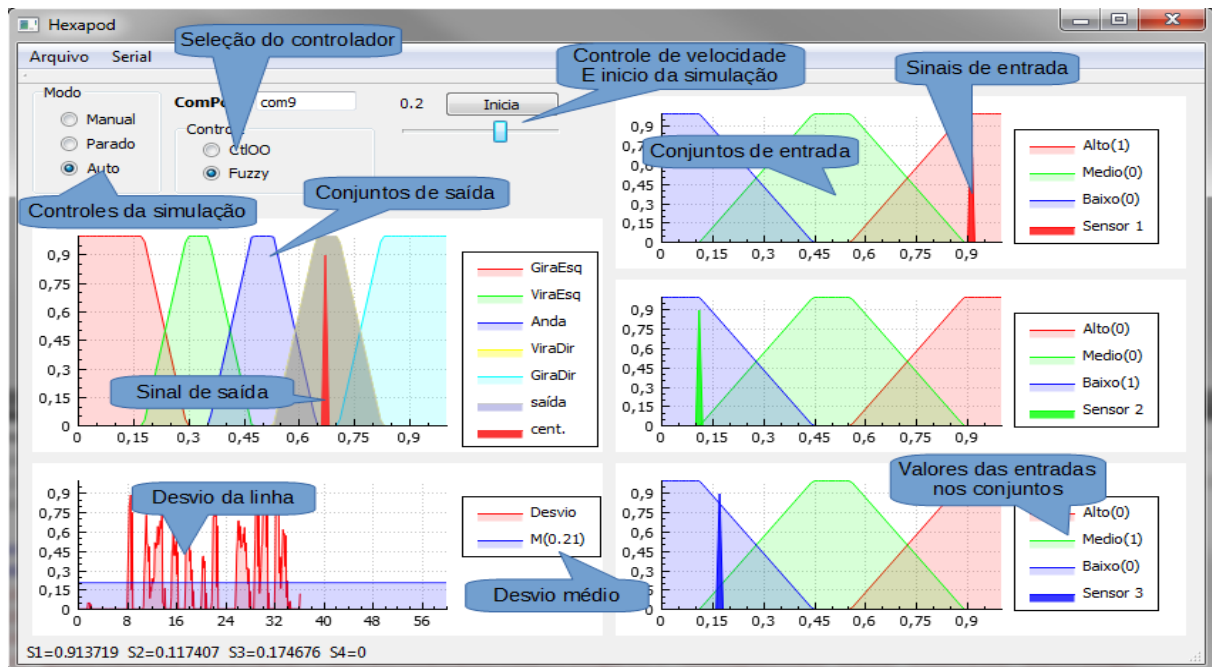


Figura 7 - Interface do software de controle

Na Figura 7 são apresentados vários balões descrevendo os principais componentes da interface.

O controle *fuzzy* foi eficiente em manter o robô sobre a linha, cumprindo assim seu objetivo. A Figura 8 apresenta o desvio da linha em um período de um minuto.

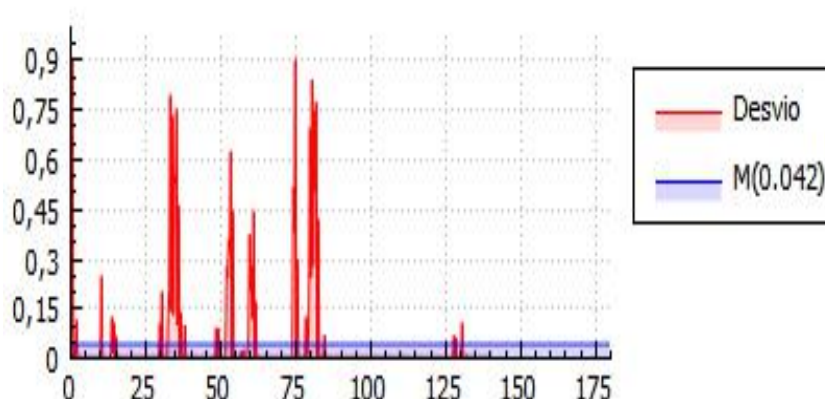


Figura 8 - Desvio do robô com relação a linha

O desvio do robô é medido em uma faixa de 0 a 1, onde 0 representa o robô no centro da linha e 1 representa o robô completamente fora da linha.

6. CONSIDERAÇÕES FINAIS

Após realizar vários testes com o controlador *fuzzy* e com o robô hexápode no ambiente virtual é possível concluir que o simulador de robôs V-Rep é uma ferramenta poderosa capaz de simular com realismo vários tipos de robôs, com vários tipos de sensores e todo o ambiente onde estes robôs podem interagir.

A utilização de robôs virtuais se mostrou viável, permitindo que se utilize uma grande variedade de robôs interagindo entre si e com outros objetos em um ambiente totalmente controlado.

Com relação a interface serial que interliga o controlador ao simulador pode-se afirmar que cumpriu adequadamente o seu papel, permitindo assim que se realizem uma série de outros experimentos no futuro, facilitando a interação do aluno com a ferramenta.

O controlador demonstrou que a utilização de lógica *fuzzy* no controle de robôs hexápodes seguidores de linha funciona adequadamente.

Por fim pode-se concluir que a utilização do simulador de robôs V-Rep pode ser uma ferramenta importante na realização de experimentos em várias disciplinas nos cursos de engenharia e em outras áreas afins.

Agradecimentos

É importante agradecer aos professores Dr. Myriam R. de Biase da Silva Delgado e Dr. Cesar Augusto Tacla pela importante colaboração no desenvolvimento da lógica de controle *fuzzy*.



7. REFERÊNCIAS BIBLIOGRÁFICAS

COPPELIA ROBOTICS. **Coppelia Robotics v-rep Create. Compose. Simulate. Any Robot.** Disponível em: <<http://www.coppeliarobotics.com/>> Acesso em: 22 mai. 2014.

GOLDSCHMIDT, Ronaldo Ribeiro. Uma introdução à Inteligência Computacional: fundamentos, ferramentas e aplicações, Rio de Janeiro: IST-Rio, 2010.

GLOWICKI, M.; BUTKIEWICZ, B.S., Autonomous line-follower with fuzzy control. Signal processing symposium (sps) - IEEE. Serock: p.1-6. 07 jul. 2013.

THE V-REP ROBOT SIMULATOR AT TEACHING AUTONOMOUS ROBOT CONTROL TECHNIQUES

Abstract: *The teaching of control techniques used in autonomous robots is complicated by the variety of existing robots and diversity of formats they take on. Build physical devices where they can conduct experiments requires time and resources, which usually results in experiments limited to a few models of robots with a few sensors. Some sensors usually have high cost and their use in autonomous robots is not always simple. The use of a robot simulator is an alternative that bypasses both problems. In a simulated environment is possible to use a wide variety of robots with no cost. The number and variety of sensors used on each robot can also be much greater in a virtual environment. Illustrating this idea, this article describes the experiments and the results analyzes of the implementation of fuzzy logic in control of a line follower hexapod robot. The robot was simulated through the software V-Rep. This controller was developed for the discipline of Advanced Programming of the Graduate Program in Electrical and Computer Engineering of the Technological Federal University of Paraná. It was also submitted as a final project in the discipline of Artificial Intelligence of the same program. The objective of this project was to develop a controller using fuzzy logic, able to control a hexapod robot follower of lines in a virtual environment. After implementation, tests have shown the functionality of the controller, the simulator and the virtual robot.*

Key-words: *Hexapod Robot, Fuzzy controller, Robot simulator.*