



VI CONGRESSO NACIONAL DE ENGENHARIA MECÂNICA
VI NATIONAL CONGRESS OF MECHANICAL ENGINEERING
18 a 21 de agosto de 2010 – Campina Grande – Paraíba - Brasil
August 18 – 21, 2010 – Campina Grande – Paraíba – Brazil

CONTROLE DE TRAJETÓRIAS PARA ROBÔS MÓVEIS UTILIZANDO LÓGICA FUZZY

Poliane Torres Megda, poliane@sc.usp.br
Luciano Cássio Lulio, lclulio@sc.usp.br
Mário Luiz Tronco, mariot@ibilce.unesp.br
Marcelo Becker, becker@sc.usp.br
Glauco A. P. Caurin, gcaurin@sc.usp.br

Universidade de São Paulo – EESC – Departamento de Engenharia Mecânica
Av. Trabalhador São-carlense, 400, Arnold Schimidt
São Carlos / SP – Brasil – CEP 13566-590

Resumo: Neste trabalho será apresentada a aplicação de Lógica Fuzzy como controlador à navegação autônoma de robôs móveis não-holonômicos em uma plataforma de simulação virtual (Webots) e posteriormente implementada no robô real e-Puck. Para tal robô, o controlador Fuzzy (código FIS Matlab) será responsável pelas decisões de desvio de obstáculos em um ambiente estático, com base nos sensores ópticos instalados na parte frontal do robô evitando os obstáculos encontrados em seu caminho, e pela trajetória ao melhor caminho estimado. Na implementação através da plataforma de simulação, o mesmo controlador aplicado, prevê a cada robô, controle de esterçamento e velocidade das rodas de forma independente, sendo possível a navegação em um ambiente conhecido, onde cada um responde satisfatoriamente a percepção sensorial de obstáculos estáticos e ao tráfego dinâmico dos robôs.

Palavras-chave: Planejamento de Trajetórias, Lógica Fuzzy, Robôs Móveis, Navegação autônoma, Desvio de obstáculos.

1. INTRODUÇÃO

O termo robô deriva da palavra *robota* que significa serviço compulsório ou atividade forçada e pode ser classificado segundo suas funções e aplicações (Capek, 1920). No desenvolvimento de robôs móveis, existe a preocupação contínua da menor interferência humana possível em suas funções, para que sejam capazes de se mover em ambientes reais e suprir a necessidade de manipulação em ambientes perigosos e complexos, bem como em tarefas clássicas, porém com a mesma necessidade do conhecimento das variáveis do sistema proposto, na determinação da modelagem dinâmica (Wolf *et al*, 2009 – Lulio and Bássora, 2008).

Para dotar estes sistemas de tais características, muitas áreas de conhecimento científico estão envolvidas ao alcance deste objetivo. São estudados e propostos modelos de sistemas de controle ao ajuste de parâmetros físicos necessários a determinada aplicação, e modelos computacionais ao processamento destes dados simulando o pensamento cognitivo humano, em áreas da Inteligência Artificial, Redes Neurais Artificiais e de Sistemas clássicos especialistas (e.g. análise probabilística, teoria dos conjuntos *Fuzzy*). Seguindo esta idéia para o trabalho proposto, cuja abordagem se concentra no desenvolvimento de um controlador ao planejamento de trajetórias para robôs móveis, há duas linhas de trabalho bem definidas em navegação: a trajetória a ser percorrida e o desvio dos obstáculos. Com a trajetória definida o veículo deve ser dotado da capacidade de desviar de obstáculos que surgem neste caminho. Vários métodos para controle de sistemas robóticos móveis têm sido desenvolvidos e geralmente são classificados em planejamento global e controle local (Benreguiieg *et al*, 1997).

No planejamento global, as informações são completas e os ambientes de navegação são estáticos e bem conhecidos. No planejamento local, os dados são obtidos através de sensores que captam dados do ambiente, dotando o navegador de conhecimento sobre distâncias, localizações e obstáculos. Com os dados coletados dos sensores, o navegador poderá tomar as devidas decisões sobre a melhor trajetória, controlando o esterçamento de rodas, o ajuste de velocidades ou até mesmo tais tarefas serem supervisionadas/guidadas (Becker, 2000).

Na área de inteligência artificial, a operação autônoma de um robô móvel em um ambiente real gera uma série de problemas como incertezas, dados desestruturados, ambiente completo desconhecido e sua dinâmica. Em casos gerais o conhecimento do local de trabalho é parcial e aproximado, a dinâmica do ambiente é predita parcialmente e os *hardwares* e *softwares* utilizados não são totalmente confiáveis. Um conjunto de métodos qualitativos ou formas

aproximadas de raciocínio têm sido desenvolvidas para construir um sistema capaz de tomar decisões em sistemas com incertezas que podem ser evitadas ou corrigidas. Estes métodos tentam captar alguns aspectos do comportamento humano no controle de sistemas. O objetivo é incorporar as incertezas inerentes ao recolhimento de informações e no processo de raciocínio ao invés de determiná-las explicitamente através de cálculos numéricos ou representações matemáticas (Benreguiet *et al*, 1997). Dentre estas teorias de argumentação qualitativa o mais utilizado para sistemas de controle é a teoria dos Conjuntos *Fuzzy*.

A Lógica *Fuzzy* (Nebulosa ou Difusa) é a lógica que suporta os modos de raciocínio que são aproximados ao invés de exatos. Modelagem e controle *fuzzy* de sistemas são técnicas para o tratamento de informações qualitativas de uma forma rigorosa. Derivada do conceito de conjuntos *fuzzy*, a lógica *fuzzy* constitui a base para o desenvolvimento de métodos e algoritmos de modelagem e controle de processos, permitindo a redução da complexidade de projeto e implementação, tornando-se a solução para problemas de controle até então intratáveis por técnicas clássicas. (Gomide e Gudwin, 1994).

Como no presente estudo o ambiente proposto é conhecido, a escolha pelo método de navegação local é o mais adequado. É utilizado o robô móvel educacional *e-Puck* (e-Puck, 2009), à aplicação e estudo do sistema de navegação autônomo (controlador *Fuzzy*), em um ambiente estático. No robô *e-Puck*, serão usados apenas os sensores ópticos frontais e laterais para a aquisição de dados sobre a distância relativa em relação aos obstáculos. O sistema de navegação em Lógica *Fuzzy* processa tais dados, condicionando as regras pertinentes a cada posição do robô no ambiente, e resultando no desvio eficiente de obstáculos, através do esterçamento em acionamento diferencial, sem colisões.

2. MATERIAIS E MÉTODOS

2.1. e-Puck

O mini robô móvel não-holonômico *e-Puck* possui acionamento diferencial, com centro de massa entre o eixo das rodas, sensores exteroceptivos para a interação com o ambiente, e microcontrolador dedicado a processamento de sinais, como características principais. Desenvolvido para fins educacionais pela EPFL (*Ecole Polytechnique Fédérale de Lausanne*), o projeto é aberto ao segmento didático (especificações técnicas de *hardware* e bibliotecas/lógica de *software*) cujos documentos são distribuídos e submetidos a uma licença permitindo que todos os usuários possam usá-lo e desenvolvê-lo (e-Puck, 2009).

Os sensores e atuadores presentes no e-Puck representam uma ampla gama de dispositivos que podem ser utilizados em várias áreas da engenharia:

- Sensores proprioceptivos (odometria) e exteroceptivos (distância) em diferentes modalidades: audio, visual, aos objetos, a gravidade;
- Dispositivos de entrada com diferentes larguras de banda de 10Hz a 10MHz;
- Atuadores eletromecânicos com diferentes ações sobre o meio ambiente (motores-de-passo);
- Dispositivos de comunicação sem fio (*Bluetooth*);
- Dois tipos de processadores: um de uso geral e outro dedicado (DSP – *Digital Signal Processor*).

A configuração básica de *hardware* eletrônico é formada por:

- Microcontrolador Microchip dsPIC30F6014A com 8KB de memória RAM e 144KB de memória flash;
- Oito sensores ópticos (infravermelhos) acoplados em torno da plataforma física que são utilizados para medidas de proximidades a obstáculos ou intensidade da luz ambiente. São sensores típicos de navegação;
- Acelerômetro 3D que fornece o vetor de aceleração do robô. Esse vetor pode ser usado para medir a inclinação e a aceleração produzida pelo movimento do robô;
- Três microfones de captação de som;
- Uma câmera CMOS com resolução de 640x480 pixels, para visão;

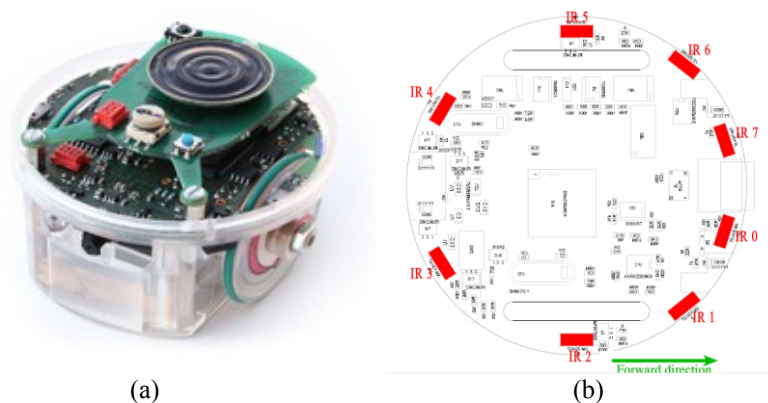


Figura 1. (a) *e-Puck*; (b) Disposição dos sensores infravermelhos (e-Puck, 2009).

O robô móvel, como ilustrado acima pela Figura 1, possui um sistema de acionamento mecânico simples, e flexibilidade quanto às atividades que podem ser desenvolvidas na exploração de aplicações usando os sensores proprioceptivos e exteroceptivos. Devido ao seu tamanho reduzido (75mm diâmetro), o robô móvel exige a mínima necessidade de cabeamento, podendo ser utilizado próximo ao computador que o controla, uma vez que sua comunicação é sem-fio através da estabilidade e robustez do protocolo *Bluetooth* (Mondada *et al.*, 2009).

Neste trabalho, o foco principal à aplicação do *e-Puck* será no uso dos sensores ópticos embarcados (quatro frontais e dois laterais). Os sensores ópticos frontais são posicionados com defasagem angular de 57° (1rad), e juntamente com os laterais transmitirão os dados de distância relativa do ambiente para o sistema de navegação.

A leitura dos sensores e configuração das velocidades de cada motor do robô para o desvio de obstáculos em cada distância relativa é implementada em linguagem C, com funções específicas da plataforma de simulação *Webots* (Cyberbotics, 2009) e do ambiente de programação *Mplab* (*Microchip*).

2.2. Controlador em Lógica Fuzzy

Conjuntos difusos (Nebulosos ou *Fuzzy*) foram propostos por Zadeh (Zadeh, 1965), e são uma generalização dos conjuntos clássicos. A diferença entre ambos consiste na variação de pertinência (0 ou 1) a determinado elemento para o modelo clássico, já nos nebulosos os graus de pertinência figuram no intervalo $\mu = [0,1]$. Para que um conjunto tenha pertinências variando neste intervalo, deve ser normalizado.

As funções de pertinência podem assumir várias formas. As mais utilizadas são em forma de sino, triangulares e trapezoidais. Para se expressar a função trapezoidal utilizam-se quatro parâmetros ($\alpha, \beta, \gamma, \delta$) que correspondem aos vértices do trapézio. Os parâmetros α e δ representam os valores mínimo e máximo da variável analisada, que têm pertinência dentro do conjunto. São definidas as operações de união e interseção para estes conjuntos (Becker, 2000).

Um número difuso é um valor numérico que representa um conjunto. As variáveis linguísticas representam vários conjuntos formados pelos números difusos, essas assumem valores dentro de um conjunto de termos linguísticos, ou seja, palavras ou frases, assim elas assumem instâncias linguísticas (Gomide e Gudwin, 1994). Uma variável pode assumir valor de um dos conjuntos linguísticos (e.g. alta, média, baixa).

As regras aplicadas são do tipo condição-ação, nestas um conjunto de condições de saída é ligada a ações que fazem com o processo estudado chegue às respostas esperadas. A idéia central é representar o conhecimento por meio de um conjunto de regras nas quais as condições dadas a partir de um conjunto de termos linguísticos associados às variáveis de saída/entrada do processo (Gomide e Gudwin, 1994). A relação entre as entradas e as regras para obter a saída é conhecida como inferência. Essa relação é uma avaliação das relações do processo estudado.

Quando um sistema é modelado, este deve ter seu comportamento bem descrito para que as análises e decisões tomadas pelo simulador correspondam com o desejado.

Um controlador *Fuzzy* trabalha as variáveis da seguinte forma: as variáveis de entrada são *fuzzyficadas*, onde é feito um escalonamento para normalização e os números se transformam em conjuntos ou variáveis linguísticas. Criando a base de regras o controlador já possui uma base de conhecimento, é esta que gera a estratégia de controle e os resultados. O processo de inferência é que traz as ações de controle para a saída. As ações de saída sofrem *defuzzyficação*, são transformadas de linguísticas para números. Um dos métodos mais utilizados em *defuzzyficação* é o método do centro de área (CDA), que retorna o centro da área do conjunto difuso de saída, ou conjunto inferido.

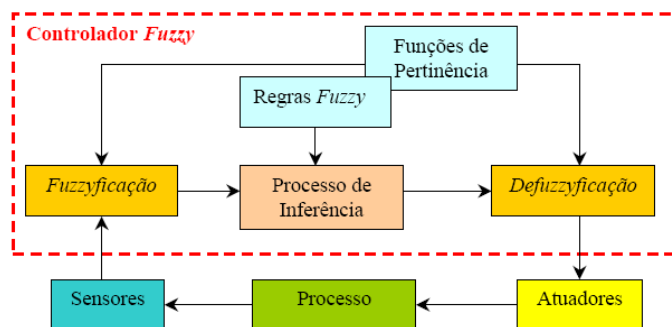


Figura 2. Estrutura básica de um controlador Fuzzy (Becker, 2000).

A Lógica *Fuzzy* foi escolhida devido a sua simplicidade, já que as decisões a serem tomadas serão enquadradas dentro de faixas, as respostas são obtidas em pouco tempo de processamento, o sistema é robusto para o propósito, e adequado para aplicações em tempo real (Figura 2), pois recebe um fluxo contínuo de informações dos sensores e de fácil manipulação com as incertezas envolvidas no caso, como a posição do robô e as distâncias relativas dos obstáculos.

O controlador desenvolvido considera a cinemática do *e-Puck*, uma vez que os dados de entrada do sistema (distância relativa) são obtidos através dos sensores ópticos, e os resultados esperados são o controle das velocidades das rodas e esterçamento das mesmas para que o robô possa desviar dos obstáculos encontrados.

Este trabalho apresenta aplicações dos conceitos *fuzzy* implementados em um algoritmo de desvio de obstáculos otimizado para robôs móveis. Tal algoritmo é desenvolvido através da plataforma computacional *Matlab (The Mathworks)* e incorporado à plataforma de simulação *Webots* em linguagem C.

2.2.1 Aplicação de Lógica Fuzzy usando o Matlab (etapa de controle)

Para que o desvio de obstáculos seja feito de forma eficiente, avaliando de forma sistemática as posições relativas de parte dos obstáculos em relação aos eixos de incidência dos sensores ópticos, são definidas três entradas e duas saídas desejadas para concepção do controlador *Fuzzy* (Suraj, 2008).

Entradas escolhidas (sensores de medidas de distâncias):

- Distância do Obstáculo Frontal (DOF) – *frontal obstacle distance* (FOD);
- Distância do Obstáculo à Esquerda (DOE) – *left obstacle distance* (LOD);
- Distância do Obstáculo à Direita (DOD) – *right obstacle distance* (ROD);

Na Figura 3, as funções-membro para cada valor de entrada são definidas em seus respectivos limiares de ação. Para as entradas escolhidas LOD e ROD o nível de variação de estado de pertinência estabelece o valor de 0,7 em padrão constante para a detecção de objetos “muito próximos” (*near at hand - NaH*), porém com início de inferência em 0,6, saindo do estado “próximo” (*near - N*). Para o estado de inferência “longe” (*far - F*), o valor em padrão linear decrescente para ambas as funções-membro é entre 0 à 0,25, uma vez que nesta instância, a distância relativa do sensor óptico ao objeto não é passível de colisão.

Ainda na mesma figura, para a função-membro FOD, o nível de variação de estado de pertinência está entre 0,8 à 1 em padrão constante para a detecção de objetos *NaH*. Já para o processo de inferência à objetos no estado *F*, o valor em padrão linear decrescente está entre 0 à 2,5, para o mesmo propósito de desvio de obstáculos supracitado.



Figura 3. Conjuntos Fuzzy para as entradas do controlador.

Saídas desejadas:

- Velocidade da Roda Esquerda (VRE) – *left wheel velocity* (LWV);
- Velocidade da Roda Direita (VRD) – *right wheel velocity* (RWV);

Na Figura 4, o processo de inferência que resulta no estado de *Defuzzificação* por *Centróide* (Simon and Mohan, 1990 – Fatmi *et al.*, 1995 – Malhotra and Sarkar, 2003) para as funções-membro LWV e RWV estabelece a variação de estado de pertinência entre 0 à 0,1 em padrão constante para o estado 'f' (*fast*), entre 0,25 à 0,4 (em padrão linear crescente), entre 0,4 à 0,6 (em padrão constante) e entre 0,6 à 0,7 (em padrão linear decrescente) para o estado 's' (*slow*). Já para o estado 'fcurve' (*fast curve*), os valores de inferência são entre 0,7 à 0,8 (em padrão linear crescente) e entre 0,8 à 1 (em padrão constante).

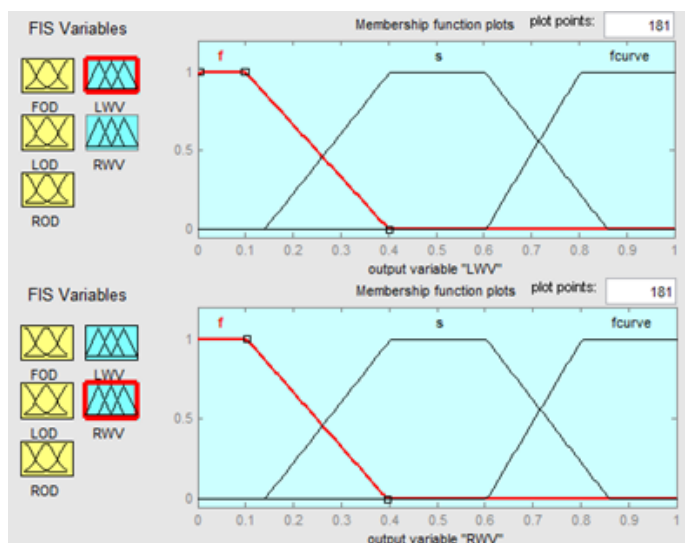


Figura 4. Conjuntos Fuzzy para as saídas do controlador.

A implementação do sistema Fuzzy é realizada através do toolbox de Lógica Fuzzy (Matlab FIS) e posteriormente adaptada ao problema proposto como código-fonte para o interpretador de regras em *cross-compiler* na linguagem de programação C (Harmeet and Arora, 2006 – Harisha *et al.*, 2003 – Castillo *et al.*, 2003).

Através do editor *Anfis* é realizada a adição das variáveis de entrada e saída, funções-membro definidas como gaussiana, e a *defuzzificação* feita pelo método do *Centróide* (centro de gravidade do conjunto Fuzzy).

Tabela 1. Definição dos dados de entrada e saída e regras de operação.

Regra	Condição	DOF	Condição	DOE	Condição	DOD	Condição	VRE	Condição	VRD
1	IF	L	AND	L	AND	L	THEN	r	AND	r
2	IF	L	AND	L	AND	P	THEN	l	AND	r
3	IF	L	AND	P	AND	L	THEN	r	AND	l
4	IF	L	AND	P	AND	P	THEN	r	AND	r
5	IF	P	AND	L	AND	L	THEN	l	AND	r
6	IF	P	AND	L	AND	P	THEN	l	AND	r
7	IF	P	AND	P	AND	L	THEN	r	AND	l
8	IF	P	AND	P	AND	P	THEN	l	AND	l
9	IF	P	AND	MP	AND	L	THEN	cr	AND	l
10	IF	P	AND	L	AND	MP	THEN	l	AND	cr

Contextualizando a Tabela 1, as entradas DOF, DOE, DOD podem ser: Longe (L), Perto (P) ou Muito Perto (MP) - (*Far – F, Near – N, Near at Hand – NaH*). As saídas: VRE, VRD podem ser: Rápido (r), Lento (l) ou Curva Rápida (cr) - (*fast – f, slow – s, fast curve – fcurve*)

O código em Matlab incorpora a Lógica Difusa ao Programa de controle Principal do *Webots* – os valores de distância dos sensores são adicionados ao código (*cross-compiler* – tradução do arquivo em módulo FIS à linguagem C), cujos gráficos de superfície são gerados pelo Editor FIS de Lógica Difusa, configurado como Mamdani.

3. RESULTADOS EXPERIMENTAIS – SIMULAÇÕES DINÂMICAS

Como resultados iniciais à aplicação do controlador Fuzzy, têm-se os modelos de variação de inferência para cada estado de entrada, correspondente aos valores de saída desejados, como pode ser observado na Figura 5.

Observa-se o comportamento dos estados de entrada e respectivos valores desejados nas saídas para o ajuste de velocidade das rodas direita e esquerda. Como a variação de inferência está relacionada com a média de probabilidades na detecção de obstáculos, várias situações podem ser simuladas para valores diferentes de fator inicial, e correspondência com os processos de saída esperados.

Como a variação no ajuste de velocidade depende da forma que é realizada o esterçamento, a Tabela 2 demonstra os testes realizados durante a detecção de obstáculos e os possíveis caminhos de saída (ângulos de esterçamento), considerando as posições dos sensores vistos na Figura 1b e as regras formuladas pelo toolbox *Anfis*.

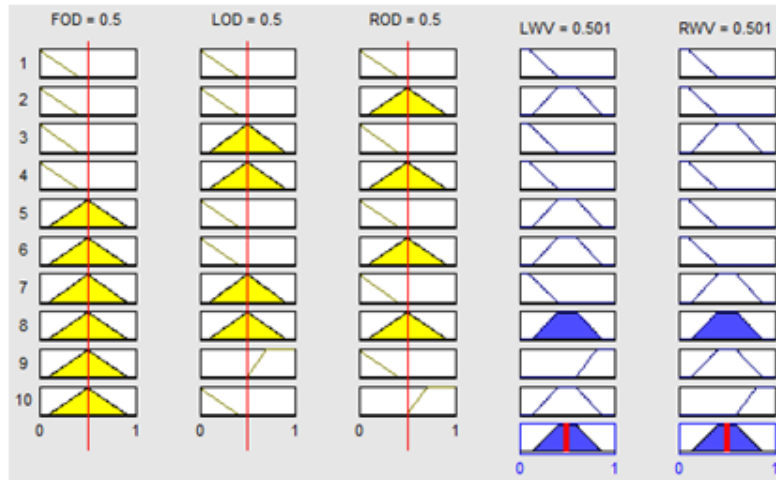


Figura 5. Processos de variação de inferência.

Tabela 2. Resultados dos testes com o robô e-Puck.

Sensores ativos	Ângulo de esterçamento
IR0/IR1	90° à esquerda
IR6/IR7	90° à direita
IR1/IR2	135° à esquerda
IR5/IR6	135° à direita
IR0/IR7	90° à esquerda/direita (relativo à posição do obstáculo seguinte)
IR6/IR7/IR0/IR1	180° à esquerda/direita (relativo à posição do obstáculo seguinte)
IR0/IR1/IR2	180° à esquerda
IR7/IR6/IR5	180° à direita
IR0/IR7/IR6/IR5	225° à esquerda
IR7/IR0/IR1/IR2	225° à direita

Graficamente, as funções de superfície denotam a relação de condição das regras formuladas quanto aos possíveis valores de saída, ou seja, os ângulos de esterçamento em relação à velocidade das rodas, dependendo dos obstáculos encontrados.

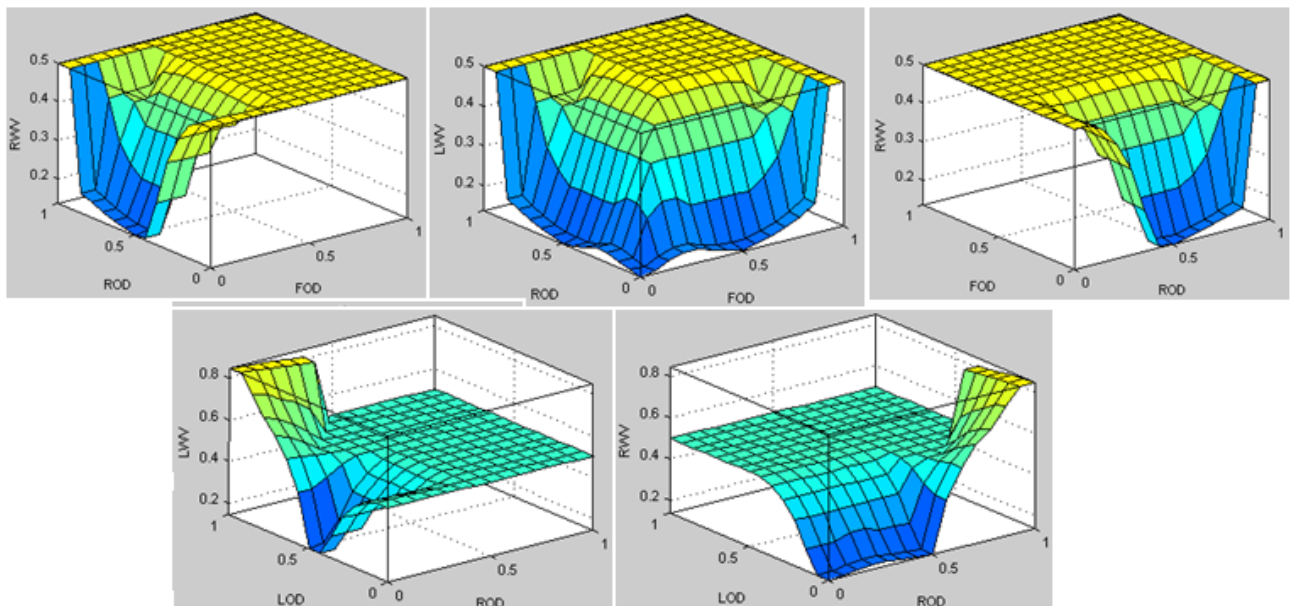


Figura 6. Gráficos de superfície da relação entrada/saída.

4. CONCLUSÕES

O controlador *Fuzzy*, quando configurado em suas entradas com variação trapezoidal nas instâncias de maior proximidade do sensor óptico com os obstáculos, responde de forma significativa aos esterçamentos à direita e esquerda, procurando o melhor caminho de rota livre para o robô *e-Puck* navegar, como registrado no item 3. Da mesma forma, foi possível aplicar na plataforma de simulação os conceitos de teoria *Fuzzy* a um controlador de navegação em um ambiente de múltiplos robôs, cada qual respondendo ao nível de percepção sensorial à obstáculos estáticos e ao tráfego dinâmico dos demais robôs.

Como sequência de trabalho, as etapas que serão implementadas para reforçar o treinamento e aprendizagem das regras, bem como aprimorar a predição de obstáculos, realizando manobras mais simples e efetivas, além de realizar a auto-localização de cada robô são: detectar linhas e cantos do ambiente, parametrizar e calcular ponto mais próximo numa dada linha (obstáculo e ambiente), realizar regressão de linhas ortogonais (análise de componentes do mapa), algoritmo para atualização de posições, algoritmo de estimação (filtro gaussiano – *FK/FKE*), obter gráficos de grids de ocupação estimados e posições mantidas

5. AGRADECIMENTOS

Os autores agradecem ao Laboratório de Mecatrônica do Departamento de Engenharia Mecânica (EESC/USP) pela colaboração na aquisição do robô-móvel *e-Puck* para os devidos testes experimentais.

6. REFERÊNCIAS

- Becker, M., 2000. “Aplicação de Tecnologias Assistivas e Técnicas de Controle em Cadeiras de Rodas Inteligentes”, Tese de doutorado, Campinas, Brasil, pp. 103-109.
- Benreguiég, M., Hoppenot, P., H Maaref, H., Colle, E., Barret, C., 1997. “Fuzzy navigation strategy : application to two distinct autonomous mobile robots”, *Robotica* VI. 15 , pp 609 – 615, Printed in the United Kingdom, Cambridge University Press.
- Capek, K; R.U.R. (Rossum's Universal Robots), 1920.
- Castillo, O., Aguilar L., Cardinas S., “Fuzzy Logic Tracking Control For Unicycle Mobile Robots”, *Engineering Letters*, pp. 73-82, 2003.
- E-puck, 2009. “Welcome on e-puck web site”. 3 Out. 2009, <<http://www.e-puck.org>>.
- Fatmi, A., Yahmadi, A.A., Khriji, L., Masmoudi, N., “A Fuzzy Logic Based Navigation of a Mobile Robot”, *Proceedings of World Academy of Science, Engineering and Technology*, pp. 159-177, 1995.
- Franklin, G. F.; Powell, J. D.; Workman, M. *Digital control of dynamic systems*. 3 ed., Menlo Park: Addison Wesley Longman, Inc., 1998.
- Fuzzy Logic Toolbox MatLab (Fuzzy Inference System) - Building Systems with Fuzzy Logic Toolbox Software. 26 Out. 2009, < <http://www.mathworks.com/access/helpdesk/help/toolbox/fuzzy/fp243dup9.html> >.
- Galvão, C.O., 1999. “Introdução à Teoria dos Conjuntos Difusos”, *Sistemas Inteligentes: Aplicações a Recursos Hídricos e Sistemas Ambientais*, VI. 1, pp. 167-191, Porto Alegre, Brasil.
- Gomide, F.A.C., Gudwin, R.R, 1994. “Modelagem, Controle, Sistemas e Lógica *Fuzzy*”, *SBA Controle & Automação*, Vol. 4,Nº 3, Campinas, Brasil.
- Harisha, S.K., Ramkanth Kumar P., Sharma S.C., “Fuzzy Logic Reasoning To Control Mobile Robot On Pre-Defined Strip Path”, *Proceedings of World Academy of Science, Engineering and Technology*, Vol. 4, pp.539-546,2003.
- Harmeet, S., Arora S., “Using Fuzzy Logic for Mobile Robot Control”, IIT-D manuals, India, 2006.
- Ibrahim, A. M. *Fuzzy Logic for Embedded Systems Applications*. USA: Elsevier, 2003.
- Lulio, L.C, Bássora, L.A., “Estudo e Aplicação de Técnicas de Controle a Robôs Móveis”. Trabalho de Conclusão de Curso, LER – UNORP, São José do Rio Preto – SP, Brasil, 2008.
- Malhotra, R., Sarkar, A., “Development of a Fuzzy Logic Based Mobile Robot for Dynamic Obstacle Avoidance and Goal Acquisition in an Unstructured Environment”, *Proceedings of IEEE/ASME, International Conference on Advanced Intelligent Mechatronics*, pp.235-247, 2003.
- Mondada, F., Bonani, M., Raemy, X., Pugh, J., Cianci, C., Klapcoz, A., Megnenat, S., Zufferey, J., Floreano, D., Maqrtinoli, A., 2009. “The e-puck, a robot designed for Education in engineering”, *Proceedings of the 9th Conference on Autonomous Robot Systems and Competitions*, Vol. 1(1) pp. 59-65.
- Mutambara, A. G. O, *Design and analysis of control systems*. Florida: CRC, 1999.
- Suraj, N. J., “Mobile Platform Control using Fuzzy Logic and Vehicle Balancing”. Research Project. Mechanical Engineering Department, National Institute of Technology, Rourkela, India, 2008
- Simon, D. & Peri V. M., “Fuzzy Logic Control For An Autonomous Robot”, *IEEE Competition*, Vol. 6, No.4, pp.445-454, 1990.
- Webots teaching material, Cyberbotics Corporation, 2009.

- Wolf, D.F., Simões, E.V., Osório, F.S., Junior, O.T., 2009. "Robótica Móvel Inteligente: da Simulação às Aplicações no Mundo Real", Vol. 1, São Carlos, Brasil.
- Zadeh, L. A. "Fuzzy Sets". *Information and Control*, V. 8: 338-353.1965.
- Zavanglas, P.G., Tzafestas, S., 2003. "Motion control of mobile robot Obstacle Avoidance and Navigation: A Fuzzy Logic-Based Approach", *Systems Analysis Modelling Simulation*, Vol. 43, N° 12, pp. 1625-1637.

7. DIREITOS AUTORAIS

Os autores são os únicos responsáveis pelo conteúdo do material impresso incluído no seu trabalho.

FUZZY LOGIC-BASED TRAJECTORY CONTROL FOR MOBILE ROBOTS

Abstract. *This work aims the application of Fuzzy Logic as autonomous navigation controller for non-holonomic mobile robots, implemented on a virtual computing platform (Webots), and thereafter applied into a real mini mobile robot e-Puck. For such robot, the Fuzzy controller (Matlab FIS cross-compiled source code) will handle all possible obstacle avoidance in a static environment, based on frontal optical sensors, bypassing all obstacles on way movement and update suitable trajectories for new landmark course. With embedded tests through simulation platform, the same applied controller, provides for each robot, steering control and independent wheel speed, making navigation through known environments possible, where each one performs the appropriate sensory perception of static obstacles and dynamic robots traffic.*

Keywords. *Trajectories Planning, Fuzzy Logic, Mobile Robots, Autonomous Navigation, Obstacle Avoidance.*