

Controlador Fuzzy para robô seguidor de Linha

Gilmar de Alcantara¹, Laura de Bella²

Resumo

Trabalho desenvolvido na disciplina de Inteligência Artificial do curso de Engenharia de Computação como projeto de pesquisa com o objetivo de aprofundar o entendimento da lógica fuzzy e explorar uma aplicação prática da mesma. Nele, criaremos um controlador fuzzy para o robô seguidor de linha, a partir dos modelos Mamdani estudados em laboratório. Nosso principal objetivo é a implantação prática do controlador em colaboração com o Grupo de Computação Competitiva em Robótica (GCCR) do CEFET-MG.

Palavras-chave: Pêndulo Invertido. Inteligência Artificial. Controladores. Lógica Fuzzy.

1 – INTRODUÇÃO

Motivados pelos estudos da disciplina de Inteligência Artificial e atendendo a uma demanda do Grupo de Computação Competitiva em Robótica (GCCR) do CEFET-MG, decidimos aprofundar nossos estudos nos controladores fuzzy e em sua implementação prática.

Para isso, utilizaremos a lógica Mandani, aplicada a modelagem que será explanada em seguida. Temos como objetivo principal encontrar um controlador eficiente para o robô seguidor de linha e implementá-lo na prática.

2 – DESENVOLVIMENTO

2.1 – Modelagem

Temos na figura 1 esquema que descreve o nosso problema:

De modo prático, podemos entender o y_s como o erro que será captado pelos sensores e nosso controle fuzzy irá setar a velocidade dos motores de acordo com esse erro. Assim,

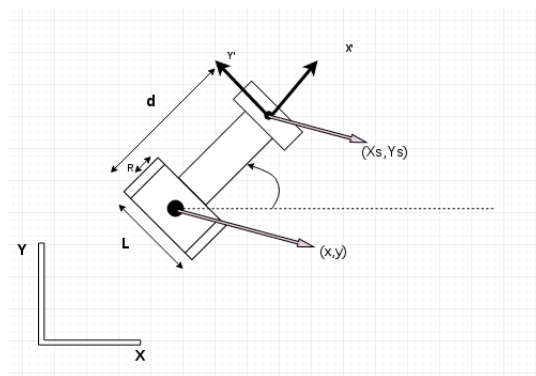


Figura 1 – Esquema do robô

poderemos inferir a variação da velocidade e sentido de rotação dos motores de modo a fazer o robô seguir a linha.

Tendo em vista um sistema fuzzy padrão na figura 2, aqui será proposto o seguinte sistema (figura 3), onde teremos 1 entrada de erro que é o deslocamento em metros do robô da linha, que pode ser positivo ou negativo (para esquerda ou para a direita), e a saída será a velocidade dos motores podendo ser para frente para trás ou nula com as devidas pertinências.

2.2 – Controle Fuzzy

Para detectar a linha foi usado um vetor de 8 sensores, e feito um cálculo de centroide para determinar a posição P da linha. O cálculo é

¹Centro Federal de Educação Tecnológica de Minas Gerais-CEGET MG

²Centro Federal de Educação Tecnológica de Minas Gerais-CEGET MG

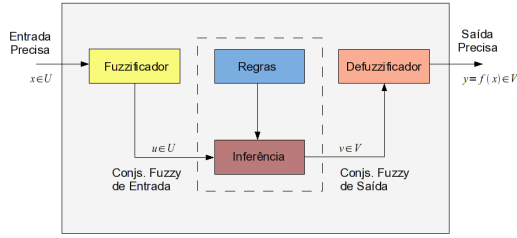


Figura 2 – Sistema fuzzy padrão

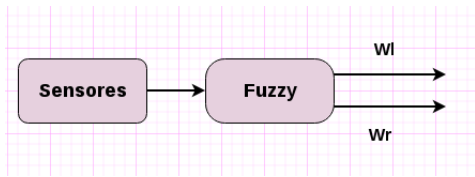


Figura 3 – Sistema proposto

Tabela 1 – Possibilidades das Distâncias do Robô à linha

dgp	distancia grande positiva
dpp	distancia pequena positiva
dz	distancia zero(ou muito proxima de zero)
dpn	distancia pequena negativa
dgn	distancia grande negativa

feito associando a leitura do sensor com a sua distância do centro do robô.

$$P = \frac{\sum_{i=0}^7 sens(i) \times d(i)}{\sum_{i=0}^7 sens(i)}$$

Onde $sens(i)$ é a leitura do sensor i e $d(i)$ é a distância do sensor i da linha. Para essas distancia temos um range que vai de -0.035 metros a 0.035 metros. temos então as seguintes velocidades:

Tabela 2 – Curvas

dgp	(-0.0525, -0.035, -0.035, -0.0175)
dpp	(-0.035, -0.0175, -0.0175, -0.0)
dz	(-0.0175, 0.0, 0.0, 0.0175)
dpn	(0.0, 0.0175, 0.0175, 0.035)
dgn	(0.0175, 0.035, 0.035, 0.0525)

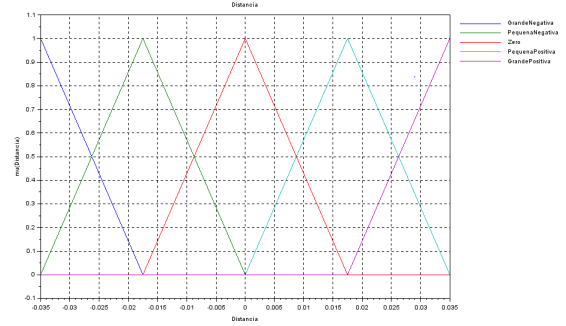


Figura 4 – Entrada do Controlador Fuzzy

Tabela 3 – Possibilidade Para as velocidades

vgf	velocidade grande para frente
vmf	velocidade média para frente
vp	velocidade pequena (próxima de zero)
vmt	velocidade media para traz
vgt	velocidade grande para traz(não usada)

Tabela 4 – Curvas

vgf	(-385, -255, -255, -127)
vmf	(-255, -127, -127, 0)
vp	(-127, 0.0, 0.0, 127)
vmt	(0.0, 127, 127, 255)
vgt	(127, 255, 255, 385)

As saídas são sinais PWM para os dois motores o esquerdo (1) e o direito(2). Variam de -255 a 255, sendo máxima para traz, zero e máxima para frente respectivamente.

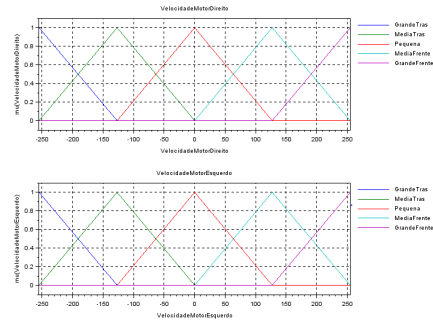


Figura 5 – Saída do Controlador Fuzzy

Teremos então a posição que será retornada é a distância em metros que o robô se encontra da linha, variando de $[-0.035, 0.035]$, como mostra a figura 2. Estipulamos 5 funções: grande

Tabela 5 – Regras do modelo

- Se a distancia for grande e positiva:
motor1: velocidade media para traz
motor2: velocidade grande para frente
- Se a distancia for pequena e positiva:
motor1: velocidade pequena (próxima de zero)
motor2: velocidade media para frente
- Se a distancia for igual a zero:
motor1: velocidade é grande para frente
motor2: velocidade é grande para frente
- Se a distancia for pequena e negativa:
motor1: velocidade media para frente
motor2: velocidade pequena(próxima de zero)
- Se a distancia for grande e negativa:
motor1: velocidade grande para frente
motor2: velocidade media para traz

positiva e negativa, pequena positiva e negativa e zero, que seria aquela próxima de zero. Estas funções foram espaçadas de maneira simétrica gerando a figura 4.

Para esse conjunto de valores foram criadas curva de inferência fuzzy (RUSSELL, 1995), que mapeiam esse erro para a velocidade do motor que varia de -255 a 255, valor ajustado as possibilidades do Arduíno que pode enviar até 5V de comando para os motores, e será a saída do controlador. Observe a figura 5. Assim como para a entrada, espaçamos as funções de saída em cinco, de maneira regular, variando entre grande ou média para frente ou para trás ou ainda pequena, que seria aquela próxima de zero.

2.3 – Aplicação do Controle Mamdani

O motor1 é o esquerdo e o motor 2 é o direito Ao aplicar o lógica fuzzy tivemos as saídas mostradas nas figuras 6 e 8 para os motores dois.

Do modo proposto que com erro zero a velocidade é máxima para os dois motores e para frente e no pior dos casos quando o erro é muito

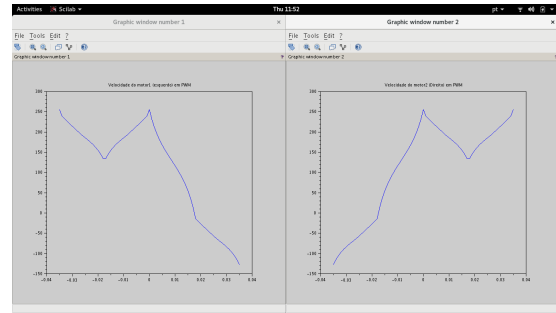


Figura 6 – Relação erro, velocidade dos motores

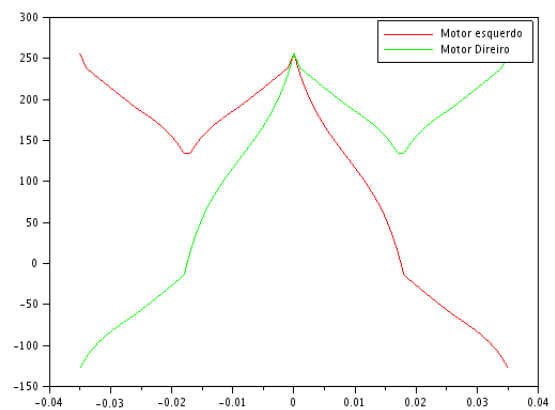


Figura 7 – Relação erro, velocidade do motor esquerdo

grande ele vai fazer um roda gira para frente e a outra para traz girando assim em torno do seu eixo, para o erro pequeno o fuzzy reduz a velocidade e faz um ação mais sutil de controle possibilitando mais liberdade de movimento e um desempenho melhor.

3 – IMPLEMENTAÇÃO

A implementação foi feita em Arduíno, foi usado um chassi da robocore, e sensores da Pololu a na figura x temos uma imagem do robô, como relação ao código foi usado uma biblioteca fuzzy para Arduino (ALVES MSC. MARVIN LEMOS; CO AUTHORS: DOUGLAS S. KRIDI, 2016).

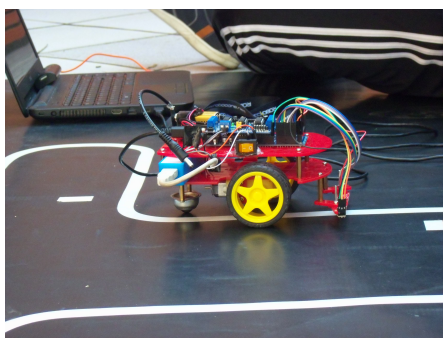


Figura 8 – *Relação erro, velocidade do motor esquerdo*

AUTHORS: DOUGLAS S. KRIDI, K. L. A. *Robotic Research Group(UESPI-Teresina)*. 2016. Disponível em: <http://www.zerokol.com/2012/09/arduino-fuzzy-uma-biblioteca-fuzzy-para.html>. Acesso em: 1 de dezembro de 2016.

RUSSELL, S. N. *Artificial Intelligence*. 1995.

4 – CONCLUSÃO

O experimento foi bem sucedido, obtendo um resultado proveitoso e incrementando a eficiência do robô seguidor de linha.

Pudemos observar que o controle fuzzy pode ser extremamente bem sucedido quando projetado de maneira cuidadosa e bem modelado matematicamente. Esperamos que os resultados obtidos sejam suficientes e que o nosso controlador possa ser utilizado pelo robô seguidor de linha em competições.

AGRADECIMENTOS

Agradecemos ao professor Rogério, da disciplina de Inteligência Artificial, pelos ensinamentos, pela orientação e por ter viabilizado o desenvolvimento deste trabalho.

Agradecemos também ao GCCR (Grupo de Computação Competitiva em Robótica) por ter disponibilizado material para a implantação prática do nosso estudo.

REFERÊNCIAS

ALVES MSC. MARVIN LEMOS; CO