

NAVEGAÇÃO GUIADA USANDO SISTEMA DE INFERÊNCIA *FUZZY* TIPO-2 INTERVALAR

WANDERSON ANTONIO DE SOUSA SILVA, JOÃO FERREIRA BORGES FILHO, JOSÉ MEDEIROS ARAÚJO JUNIOR, JOSÉ MARIA PIRES DE MENESES JUNIOR*

**Campus Universitário Ministro Petrônio Portella
Laboratório de Controle e Automação, Universidade Federal do Piauí
Teresina, PI, Brasil*

Email: wass120@hotmail.com, joaoffilho007@gmail.com, jmjuni@ufpi.edu.br,
josemenezesjr@ufpi.edu.br

Abstract— In autonomous robotics, autonomous navigation treats how a robot moves from a specific position to complete a task. In the meantime, the robot should handle internal and external constraints in an environment with dynamic changes. In the environment, uncertainties sources are common and decrease navigation performance of the robot. In many researches, the Type-1 Fuzzy Inference System (SIF1) is a typical solution to deal with uncertainties in real time navigation. However, SIF1 has limitations to handle uncertainties in unknown and dynamic environments. In this work, we develop an approach that a wall following robot, three and omnidirectional wheels, navigates in uncertainties environments using Interval Type-2 Fuzzy Inference System (SIF2I). To evaluate the approach, simulations and real experiments are performed in Robotino to analyze three metrics: total distance, power consumed and reference line distance. In the end of simulations and experiments, the metrics are studied and comparisons between SIF1 and SIF2I are performed. The results showed that SIF2I approach performed better robustness dealing with uncertainties in real environments.

Keywords— Autonomous Navigation, Type-1 *Fuzzy* Inference System, Intervalar Type-2 *Fuzzy* Inference System, Robotino.

Resumo— Navegação autônoma é um dos tópicos da robótica autônoma que trata como um robô se movimenta de uma posição específica para completar uma dada tarefa, ainda que sujeita a restrições internas e externas em ambientes dinâmicos. Em muitos trabalhos, o Sistema de Inferência *Fuzzy* Tipo-1 (SIF1) é oferecido como uma solução capaz de lidar com fontes de incertezas em navegação em tempo real. No entanto, o SIF1 apresenta limitações em lidar com incertezas associadas as mudanças apresentadas em ambientes dinâmicos e desconhecidos. Neste trabalho, propomos uma abordagem onde um robô seguidor de paredes, com três rodas e omnidirecional, navega em cenários incertos usando Sistema de Inferência *Fuzzy* Tipo-2 Intervalar (SIF2I). Para avaliação da abordagem, simulações e experimentos reais são realizados no Robotino para obter três métricas: distância percorrida, tensão consumida e distância entre a posição do robô e a linha de referência. Ao final das simulações e experimentos, os resultados mostraram que a abordagem usando SIF2I apresentou melhor robustez ao lidar com fontes de incertezas em ambientes reais que o SIF1.

Palavras-chave— Navegação Autônoma, Sistema de Inferência *Fuzzy* Tipo-1, Sistema de Inferência *Fuzzy* Tipo-2 Intervalar, Robotino.

1 Introdução

Em robótica móvel, um dos principais tópicos de pesquisa é a navegação autônoma (Siegwart et al., 2011). Nesse tópico, pesquisadores desenvolvem estratégias que permitem a um robô móvel movimentar-se a partir de uma posição específica para completar uma tarefa dada, mesmo com restrições internas e externas de ambientes dinâmicos. Tendo em sua disposição um conjunto de sensores, o robô deve reconhecer o ambiente ao qual está inserido e ser capaz de gerar um plano de navegação capaz de completar sua tarefa.

Há duas principais abordagens em planejamentos de navegação autônoma em ambientes reais (Kim and Chwa, 2015): deliberativa e reativa. Na abordagem deliberativa, o robô dispõe, tipicamente, de um modelo geométrico detalhado do ambiente a qual será submetido. Na prática, essa abordagem apresenta uma perda na robustez devido ao grau de incertezas encontrados no ambiente. Assim, causando falhas e uma constante requisição de um novo planejamento, o que leva a

um crescimento na complexidade computacional.

Por outro lado, a abordagem reativa exige pouco, ou nenhum, modelo do ambiente a ser navegado. Nessa abordagem, os sensores e atuadores do robô são regidos por uma função de transferência que produz um comportamento emergente a navegação. Dessa forma, a abordagem requer baixa complexidade computacional e submete o robô a uma ação específica em um breve período. Neste trabalho, iremos adotar um robô seguidor de paredes que adotará navegação reativa devido a sua baixa complexidade computacional.

Em navegação autônoma, fontes de incertezas são comuns (deslizamento de rodas, falha na leitura dos sensores, etc). Devido as fontes de incertezas serem um problema recorrente, o desempenho da navegação do robô é influenciado. Muitos trabalhos, como em (Hagras, 2004a) (Hagras, 2004b) (Kumbasar and Hagras, 2013) (Baklouti and Alimi, 2017) (Ayedi et al., 2017), relataram o Sistema de Inferência *Fuzzy* Tipo-1 (SIF1) como uma solução capaz de lidar com fon-

tes de incertezas em navegação em tempo real. No entanto, o SIF1 possui limitações em lidar com incertezas associadas as mudanças encontradas em ambientes dinâmicos e desconhecidos (Linda and Manic, 2010). Por trazer valores fixos, o sistema se torna ineficiente para redução, ou eliminação, dos ruídos encontrados no ambiente (Mendel et al., 2006)(Mbede et al., 2012).

Mediante as limitações encontradas no SIF1, o Sistema de Inferência *Fuzzy* Tipo-2 (SIF2) passa ser uma solução viável para o tratamento de tais incertezas (Mendel et al., 2014). O SIF2 pode ser considerado como uma extensão do SIF1 (Kim and Chwa, 2015). Para obter uma melhor performance computacional (Hagras, 2004b)(Mendel et al., 2014), o SIF2 possui um caso particular, o Sistema de Inferência *Fuzzy* Tipo-2 Intervalar (SIF2I). Por ser um sistema amplamente utilizado (Hagras, 2004b)(Silva et al., 2014)(Ayedi et al., 2017)(Baklouti and Alimi, 2017), ele será utilizado neste trabalho.

No presente trabalho, pesquisas foram realizadas para destacar o estado da arte sobre a aplicação do SIF2I em navegação autônoma. Em (Ayedi et al., 2017), uma comparação entre o SIF1 e SIF2I é realizada com o objetivo de mostrar os benefícios da aplicação do SIF2I. Nesse trabalho, os autores realizam uma simulação onde um robô de rodas diferenciais é submetido à quatro cenários distintos. Em cada cenário, o robô parte de um ponto específico e navega até o ponto desejado. Para analisar o desempenho de cada controle *fuzzy*, os autores mediram as distâncias percorridas pelo robô e o erro relativo entre as distâncias percorridas por cada controlador *fuzzy*. Os resultados apresentados pelo trabalho mostram que o SIF2I sempre percorre menores distâncias que o SIF1. Portanto, levando os autores a concluir que o SIF2I mostra eficiência, se comparado ao SIF1, principalmente na presença de ruídos.

Já os autores em (Baklouti and Alimi, 2017) sugerem uma nova abordagem utilizando redes neurais artificiais no SIF2I para o processo de navegação autônoma. Para avaliar sua ideia, os autores submeteram o robô em dois cenários, com e sem ruído, onde o erro da saída desejada e da saída obtida pelo sistema proposto são analisados. Mediante os experimentos realizados e os resultados obtidos, os autores chegaram a conclusão que a abordagem oferecida para o desvio de obstáculos de um robô móvel apresentou bons resultados de navegação.

Por fim, o trabalho feito em (Kim and Chwa, 2015) apresenta um método de desvio de obstáculos, com SIF2I e redes neurais, para navegação de robôs móveis. Para medir o desempenho dos sistemas, os autores mediram o erro entre as distâncias percorridas e a distância total percorrida pelo robô, SIF1 e SIF2I. Os resultados simulados e experimentais mostraram que o método usando

SIF2I conduziram aos autores as seguintes análises: o robô conseguiu desviar de obstáculos, posicionamento estabilizado do robô, menor distância total percorrida e movimentos mais suaves ao desviar de obstáculos. Portanto, chegando à conclusão que o método usando SIF2I trabalha melhor que o método usando SIF1 para navegação autônoma.

Neste trabalho, propomos uma abordagem onde um robô seguidor de paredes, com três rodas e omnidirecional, navega, de forma reativa, em cenários *indoors* incertos usando SIF2I.

2 Sistemas Fuzzy

2.1 Sistemas fuzzy tipo-1

O sistema *fuzzy* tipo-1 pode ser entendido pelo mecanismo de quatro blocos (Mendel et al., 2014): o fuzzificador, a inferência, a base de regras e o defuzzificador, como podem ser vistos na Figura 1.

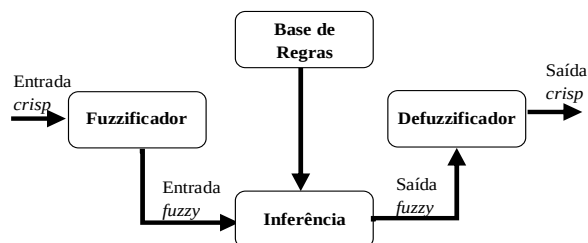


Figura 1: Arquitetura de um SIF1

2.2 Sistemas fuzzy tipo-2

O SIF2 é muito similar ao SIF1 (Mendel et al., 2006). A diferença é que o valor retornado por uma função de pertinência do SIF1 é um número real. Esse valor real torna o SIF1 inapropriado para modelagem direta de incertezas. Com a finalidade de superar esse problema, a lógica *fuzzy* tipo-2 oferece a possibilidade de modelar e manter estas incertezas. As incertezas de um conjunto *fuzzy* tipo-2 são representadas pelo *Footprint of Uncertainty* (FOU). O FOU fornece uma medida que é dispersa nas variáveis linguísticas que descrevem o conjunto *fuzzy* tipo-2 (Mendel et al., 2014).

A representação da inferência modelada pelo sistema *fuzzy* tipo-2 é descrito pelo mecanismo de cinco blocos da Figura 2. O sistema é iniciado com o fuzzificador, que mapeia os valores *crisp* em um conjunto *fuzzy* tipo-2. Em seguida, o motor de inferência computa as regras realizando combinações lógicas entre os antecedentes e consequentes do conjunto *fuzzy* tipo-2 que resulta em um saída *fuzzy* tipo-2. O próximo passo coleta todas as saídas *fuzzy* tipo-2, aplicando um método de redução, que permite converter o conjunto *fuzzy*

tipo-2 em um conjunto fuzzy tipo-1, esta etapa é chamada Tipo-Redutor. Finalmente, o conjunto reduzido é direcionado ao defuzzificador que então obtém o valor *crisp* de saída.

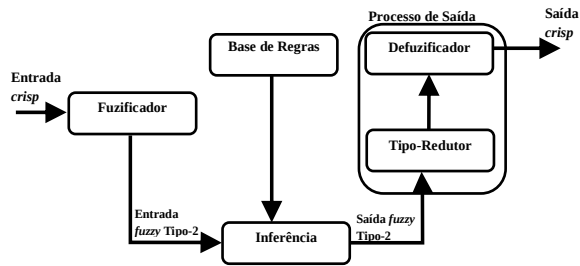


Figura 2: Arquitetura de um SIF2.

2.2.1 Fuzzificador

O bloco fuzzificador recebe os valores *crisp* de entrada e transforma-os em conjuntos *fuzzy* tipo-2. Assim como o SIF1, o fuzzificador pode ser singleton e não-singleton. Quando a incerteza desaparece, a função de pertinência do tipo-2 deve ser reduzida para uma função de pertinência do tipo-1. A FOU, que representa estas incertezas, pode ser representada de acordo com sua altura e posição (Mendel et al., 2006). A FOU é delimitada por uma função de pertinência superior e uma inferior, ambas do tipo-1. A função de pertinência superior pode ser representada por $\bar{u}(x)$. A função de pertinência inferior é representada na forma de $\underline{u}(x)$ para $\forall x \in X$, onde X é o universo de discurso e x uma variável base, como mostrado na Figura 3.

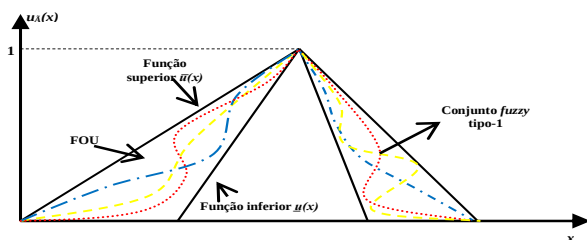


Figura 3: Mancha de incerteza (FOU). Função de pertinência Superior $\bar{u}(x)$, função de pertinência inferior $\underline{u}(x)$ e um conjunto intrínseco SIF1 para o SIF2.

2.2.2 Tipo-redutor

O bloco tipo-redutor tem como finalidade procurar o melhor conjunto *fuzzy* tipo-1 que represente o conjunto fuzzy tipo-2. Sendo a saída do bloco tipo-redutor, a transformação de um conjunto tipo-2 em conjunto tipo-1 (Mendel et al., 2014). Para obtenção do tipo-redutor, aplica-se o princípio da extensão proposta por Zadeh na defu-

zificação do tipo-1. Em razão de obter a saída do bloco tipo-redutor, existe um método do tipo-redutor equivalente para cada método de defuzzificação. Um método muito utilizado é o algoritmo proposto por Karnik e Mendel (KM) (Mendel et al., 2006).

O algoritmo KM é um procedimento iterativo para obtenção de um centro generalizado de um conjunto fuzzy tipo-2. Ele usa valores definidos estatisticamente para encontrar pontos que permitem a redução do conjunto fuzzy tipo-2 para conjunto tipo-1, permitindo somente iterações necessárias para a redução. Segundo o trabalho em (Mendel et al., 2006), esta característica permite que o algoritmo convirja monotonicamente e exponencialmente rápido.

2.3 Sistemas fuzzy tipo-2 Intervalares

Sistema *fuzzy* tipo-2, ou sistema *fuzzy* tipo-2 generalizado, que foi tratado até agora, possui um caso particular de conjunto *fuzzy* do tipo 2.

A função de pertinência do conjunto *fuzzy* tipo-2 generalizado é caracterizado pela não uniformidade de sua função de pertinência secundária. Nesta função de pertinência, há três dimensões que fornecem graus de liberdade adicionais que cria a possibilidade de modelar e manter as incertezas de forma direta (Mendel et al., 2014).

No entanto, esses graus de liberdade adicionais oferecem um custo computacional elevado. Para superar essa limitação, em (Mendel et al., 2006) é proposto um sistema *fuzzy* tipo-2 que torna a execução da abordagem viável computacionalmente. Daí surge o conceito de conjunto *fuzzy* tipo-2 intervalar, SIF2I.

O conjunto tipo-2 intervalar tem como princípio tornar o grau secundário sempre igual a 1. Desta maneira, a terceira dimensão da função de pertinência do conjunto *fuzzy* tipo-2 não mostra nenhuma informação adicional permitindo um custo computacional reduzido. Por esta razão, é a técnica usado neste trabalho.

3 Abordagem Proposta

A proposta deste trabalho é oferecer uma abordagem de navegação reativa para um robô com três rodas omnidirecionais usando SIF2I em ambientes *indoors* desconhecidos. Na Figura 4, é apresentado a aplicação da abordagem empregada neste trabalho. O robô inicia de um ponto específico e percorre todo o cenário procurando estar o mais próximo da linha de referência, dentro de um período de 200 segundos.

Para cada cenário, o robô realiza sua navegação em duas etapas. Na primeira, o robô navega sobre o controle do SIF1, enquanto na segunda, uma nova simulação, ou experimento, é iniciada e ele navega sobre o controle do SIF2I.

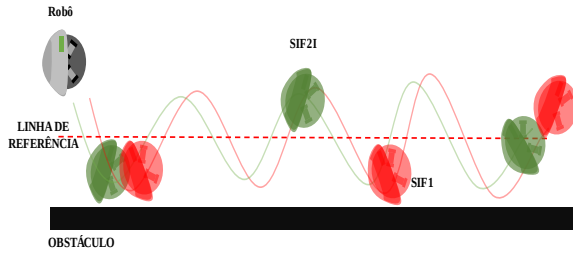
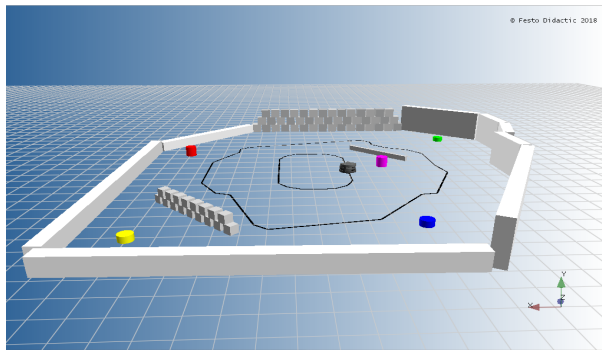


Figura 4: Ilustração da abordagem proposta. Robotino procura se aproximar da linha de referência usando o SIF1 e SIF2I.

Em todo seu trajeto, o robô adota o comportamento de um seguidor de paredes e evita colisões. Para avaliação da abordagem, simulações e experimentos reais são realizados no Robotino para obter três métricas: distância total percorrida, tensão total consumida e distância entre a posição do robô e a linha de referência (média e desvio padrão). Neste artigo, adotamos a linha de referência com uma distância de 30 cm da parede. Esse valor foi adotado levando em consideração as dimensões do Robotino e o alcance de seus infravermelhos.



(a) Cenário utilizado em simulação.



(b) Corredor onde os experimentos reais são realizados.

Figura 5: Experimentos realizados.

Em (Ayedi et al., 2017) e (Kim and Chwa, 2015), o robô navega usando o SIF1 e SIF2I com até duas simulações, ou experimentos, e para suas análises, até duas métricas (distância total percorrida ou erro entre as distâncias percorridas). Porém, neste trabalho, realizamos trinta simula-

ções e dez experimentos, de um mesmo cenário, coletamos dados de cada simulação para análise das métricas aqui propostas, além de realizarmos a análise de três métricas. Em simulação, o robô navega no ambiente apresentado na Figura 5a. Enquanto no experimento real, o robô navega em um corredor apresentado na Figura 5b.

4 Implementação

Robotino é o robô móvel holonômico desenvolvido pela Festo Didactic (Figura 6); com três rodas, e motores independentes, omnidirecionais montadas a um ângulo de 120° entre cada uma; nove sensores infravermelhos montados no chassi a um ângulo de 40° , um para o outro, onde medem distâncias que vão de 4 à 30 cm (expressa por uma função de tensão); giroscópio; encoders incrementais; entre outros sensores e atuadores (Festo-didactic, 2018). A fabricante fornece uma API (*Application Program Interface*) chamada *RobotinoCom* que oferece funções, via *wireless*, para acesso aos sensores e atuadores do robô. Essa API é a mesma utilizada nas simulações e experimentos reais.

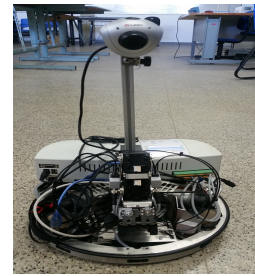


Figura 6: Robotino, robô utilizado nos experimentos deste trabalho.

As ferramentas utilizadas neste trabalho para alcançar os resultados obtidos são: JUZZY (Wagner, 2013), MATLAB R2017, Robotino SIM Demo 1.41 e Robotino (Festo-didactic, 2018). No MATLAB, o programa principal é executado. É por meio do MATLAB que as comunicações entre o Robotino SIM, o Robotino e o JUZZY são realizadas, Figura 7. O JUZZY é uma biblioteca Java utilizada para modelagem das configurações *fuzzy*. O Robotino SIM é o ambiente de simulação para experimentos com o Robotino. Por fim, Robotino é a plataforma robótica real utilizada nos experimentos.

Neste trabalho, as simulações e experimentos propostos foram adotados alguns parâmetros para o SIF1 e SIF2I, Figura 8. Nas entradas dos sistemas *fuzzy* foram utilizadas três informações. A primeira é a menor distância do lado direito (detectado pelos infravermelhos do robô IR9, IR8 e IR7). A segunda é a distância detectada pelo sensor infravermelho frontal (IR1) e, a terceira informação, o erro (diferença entre IR9 e IR7), ver

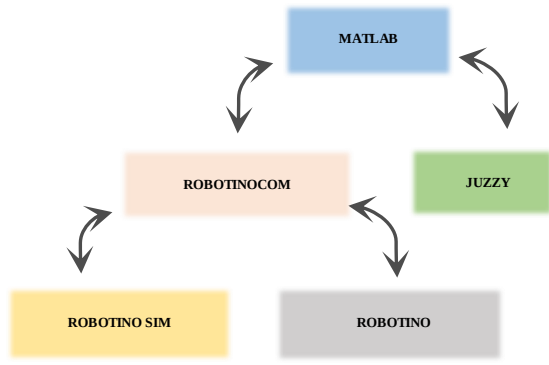
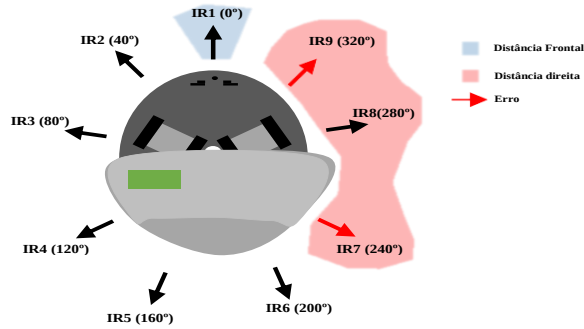
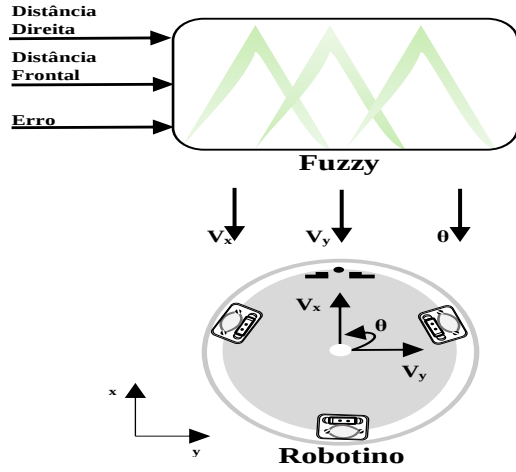


Figura 7: Diagrama da implementação dos experimentos realizados.

Figura 8a. O sistema de inferência utilizado é o operador mínimo de Mandani e o método centroide para obtenção das saídas. Para o SIF2I, o tipo-redutor utilizado é o KM.



(a) Entradas para os sistemas *fuzzy*: distância frontal (IR1), menor distância direita (IR9, IR8 e IR7) e erro (IR9 – IR7).



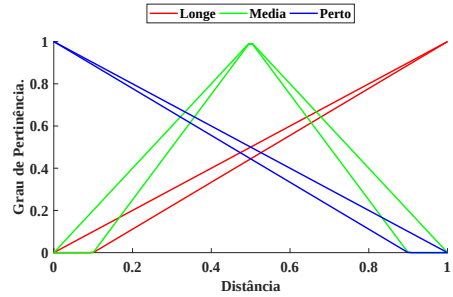
(b) Saídas *fuzzy* interagindo com o Robotino.

Figura 8: Interação sistema *fuzzy* e Robotino.

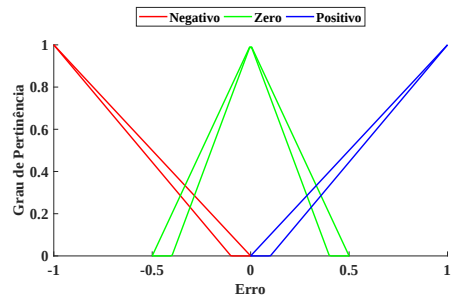
As funções de pertinências adotadas para as entradas foram as do tipo triangular, SIF1 e SIF2I. Neste trabalho, as configurações das funções de pertinências do SIF1 e SIF2I são bem semelhantes. Nas Figuras 9 e 10 são apresentadas as funções de pertinências do SIF2I. Nas entradas, as distâncias possuem os termos linguísticos *Off* (fora do al-

cance do infravermelho), *Longe*, *Média* e *Perto*. Já o erro apresenta os termos negativo, zero e positivo. As saídas dos sistemas são: velocidade V_x (velocidade para o robô no eixo x), V_y (velocidade para o robô no eixo y) e θ (orientação do robô), Figura 8b. As velocidades e o θ apresentam os termos linguísticos *rápida negativa* ($-R$), *media negativa* ($-M$), *baixa negativa* ($-B$), *zero* (Z), *baixa positiva* ($+B$), *media positiva* ($+M$) e *rápida positiva* ($+R$).

Aqui, a base de conhecimento *fuzzy* possui 24 regras (Tabela 1), as mesmas para ambos sistemas *fuzzy*. Essa tabela foi dividida em duas partes: a tabela primária de referência e a tabela secundária. Na tabela primária de referência, os termos linguísticos das distâncias (frontal e direita) são combinados para resultar em uma chave, com valor simbólico, que será associado aos termos linguísticos do erro. Uma vez que os antecedentes estejam configurados, a tabela secundária associa as chaves da tabela primária com os termos linguísticos do erro resultando nos consequentes.

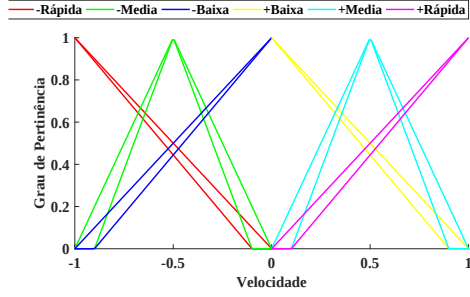


(a) Entrada distâncias (Frontal e Direita).

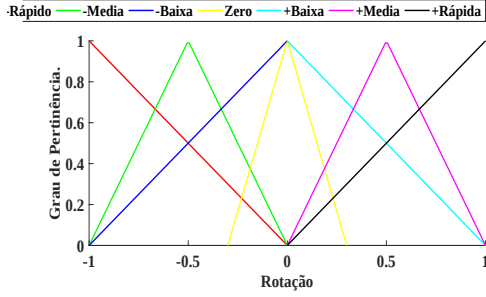


(b) Entrada Erro.

Figura 9: Funções de Pertinências das entradas do SIF2I.



(a) Saída velocidades.



(b) Saída rotação (θ).

Figura 10: Funções de Pertinências das saídas do SIF2I.

	Direito				
Frontal	Off	Perto	Media	Longe	
Off	A	A1	A2	A3	
Longe	B	B1	B2	B3	

(a) Tabela primária de referência. Associa os termos linguísticos das distâncias em uma chave de referência.

	Erro		
	Negativo	Zero	Positivo
A	$V_x + M/V_y - R/\theta Z$	$V_x + M/V_y - R/\theta Z$	$V_x + M/V_y - R/\theta Z$
A1	$V_x + M/V_y + M/\theta + B$	$V_x + M/V_y + M/\theta Z$	$V_x + M/V_y + M/\theta + B$
A2	$V_x + M/V_y + M/\theta - B$	$V_x + M/V_y + M/\theta Z$	$V_x + M/V_y + M/\theta + B$
A3	$V_x + M/V_y - M/\theta - B$	$V_x + M/V_y - M/\theta Z$	$V_x + M/V_y - M/\theta + B$
B	$V_x + M/V_y + M/\theta + B$	$V_x + M/V_y + M/\theta Z$	$V_x + M/V_y + M/\theta + B$
B1	$V_x + M/V_y + M/\theta + M$	$V_x + M/V_y + M/\theta + M$	$V_x + M/V_y + M/\theta + M$
B2	$V_x + M/V_y + M/\theta + M$	$V_x + M/V_y + M/\theta + M$	$V_x + M/V_y + M/\theta + M$
B3	$V_x + M/V_y + M/\theta + M$	$V_x + M/V_y + M/\theta Z$	$V_x + M/V_y + M/\theta + M$

(b) Tabela secundária. Associa as chaves de referências ao erro e resultando nos consequentes.

Tabela 1: Base de regras SIF1 e SIF2I.

Na Tabela 1b, os consequentes apresentam a seguinte expressão: $\langle V_x \rangle \langle Sinal \rangle \langle Letra\ inicial\ do\ termo\ linguístico\ de\ V_x \rangle / \langle V_y \rangle \langle Sinal \rangle \langle Letra\ inicial\ do\ termo\ linguístico\ de\ V_y \rangle / \langle \theta \rangle \langle Sinal \rangle \langle Letra\ inicial\ do\ termo\ linguístico\ de\ \theta \rangle$. Por exemplo, o consequente $V_x + M/V_y - R/\theta Z$ significa que se frontal está desligada e a direita está desligada e o erro é negativo então a velocidade no eixo x

é media positiva e velocidade no eixo y é rápida negativa e a orientação é zero.

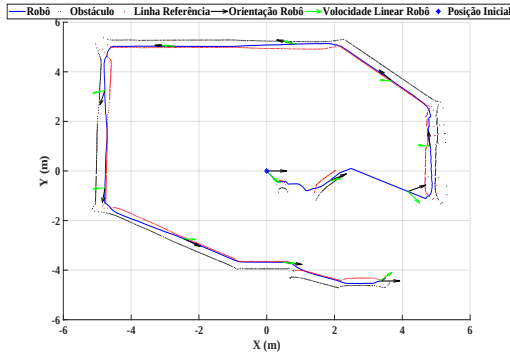
5 Resultados

Para avaliar a eficiência da abordagem proposta empregamos as seguintes métricas: distância total percorrida, tensão consumida e o erro. Aqui, a distância total percorrida é utilizada para analisar a distância total, em metros, navegada pelo robô em cada etapa. Essa métrica foi escolhida para mostrar o quão próximo da linha de referência o robô navega, pois quanto mais próximo da linha referência menor a distância percorrida pelo robô. Na tensão consumida, mostramos a tensão, em volts, consumida na bateria do robô durante a realização de sua tarefa, essa tensão é obtida por meio do *RobotinoCom*.

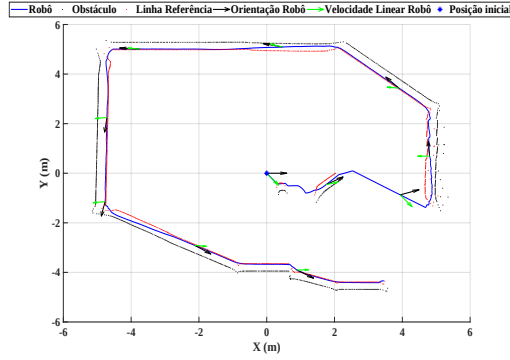
Já a métrica da distância entre a posição do robô e a linha de referência, em milímetros, foi escolhida para analisar a proximidade do robô da linha proposta usando o SIF1 e SIF2I. Para cada simulação, e experimento, a média e o desvio padrão dessa métrica é apresentada nos resultados. O erro é considerado pela diferença entre os IR9 e IR7 ($IR9 - IR7$). Por meio desse erro, o *fuzzy* verifica o sentido da orientação do robô: se a diferença é positiva, a orientação do robô está no sentido anti-horário; se negativo, orientação no sentido horário.

Durante as simulações e experimentos, os dados de posição do robô, da linha de referência e dos obstáculos detectados foram coletados e plotados nas Figuras 11a e 11b. Nessas figuras são exibidas as navegações realizadas pelo robô usando SIF1 e SIF2I na primeira simulação.

Todas as métricas são obtidas nas simulações e experimentos realizados e exibidas nas Tabelas 2a, 2b, 3a e 3b. Nas Tabelas 2a e 2b, as análises dos resultados simulados no cenário da Figura 5a são descritos. Na primeira coluna, a sequência das simulações, as estatísticas e *scores* são apresentados. As sequências de simulações trazem a ordem da simulação, 1 à 30. As estatísticas indicam as médias (*AVG*) e desvio padrão (*STD*) de todas as simulações realizadas para o SIF1 e SIF2I. Os *scores* mostram em quantas simulações SIF2I vence (destacado em vermelho) o SIF1 ($SIF2I < SIF1$), ou empata ($SIF2I == SIF1$) nas métricas das colunas correspondentes, o melhor é destacado por meio do simbolo asterisco (*). Nas Tabelas 3a e 3b, os resultados dos experimentos reais realizados no cenário indicado na Figura 5b são analisadas. Na primeira coluna, a sequência de experimentos, as estatísticas e *scores* são apresentados.



(a) SIF1.



(b) SIF2I.

Figura 11: Navegação robô em simulação no cenário da Figura 5a.

Simulação	Distância Total (m)		Tensão Consumida (V)	
	*SIF1	SIF2I	*SIF1	SIF2I
1	35,5093	35,7691	0,1385	0,1385
2	35,7513	35,9114	0,1385	0,1385
3	35,225	36,3558	0,1385	0,1662
4	35,6301	35,2691	0,1385	0,1385
5	35,6833	35,5982	0,1385	0,1385
...	...	...	...	...
30	45,7174	45,9029	0,1939	0,1662
Estatísticas				
AVG	38,66840	39,0062	0,15328	0,15328
STD	4,81270	4,9218	0,021498	0,024919
Scores				
SIF2I<SIF1	13		8	
SIF2I==SIF1	0		10	

(a) Distância total e tensão consumida.

Simulação	Distância de Referência AVG (mm)		Distância de Referência STD (mm)	
	SIF1	*SIF2I	SIF1	*SIF2I
1	88,2941	66,1906	68,4928	58,5741
2	100,567	56,5557	72,5564	52,1157
3	84,7627	56,4505	68,6813	56,0199
4	90,3559	72,5726	64,0747	60,9752
5	89,7882	71,3608	65,9862	60,6768
...	...	...	...	...
30	85,737	69,8121	67,3831	62,2925
Scores				
SIF2I<SIF1	30		28	
SIF2I==SIF1	0		0	

(b) Distância de referência.

Tabela 2: Resultados obtidos em simulação.

Na Tabela 2a, apenas em 13 simulações, o SIF2I relatou menor distância percorrida do que o SIF1. No entanto, nas estatísticas, ambos são bem semelhantes. Com relação a tensão consumida, na maioria das simulações o SIF1 apresenta menor consumo, porém, com uma pequena diferença do SIF2I, como mostrado nas estatísticas. Em paralelo com a Tabela 3a, o SIF2I desenvolveu menor distância, em 70% dos experimentos. Mas, assim como nos resultados simulados, a média e desvio padrão bem similares. Com relação a tensão consumida, as Tabelas 2a e 3a exibem valores bem próximos um do outro. Esses resultados revelam que, nas métricas de distância total e tensão consumida, o SIF1 e SIF2I evidenciam desempenho próximos um do outro.

Experimento	Distância Total (m)		Tensão Consumida (V)	
	SIF1	*SIF2I	*SIF1	SIF2I
1	17,88600	18,0492	0,3324	0,4432
2	17,88590	17,632	0,1108	0,1108
3	17,83150	17,4742	0,1108	0,1108
4	17,57160	17,3355	0,1108	0,1108
5	17,55890	17,1678	0,3324	0,9972
...	...	...	...	...
10	17,62250	17,5732	1,4404	0,2216
Estatísticas				
AVG	17,67840	17,5406	0,40997	0,40997
STD	0,13477	0,23024	0,49567	0,39452
Scores				
SIF2I<SIF1	7		2	
SIF2I==SIF1	0		4	

(a) Distância total e tensão consumida.

Experimento	Distância de Referência AVG (mm)		Distância de Referência STD (mm)	
	SIF1	*SIF2I	SIF1	*SIF2I
1	35,1425	25,1486	32,2834	27,4999
2	30,9536	21,0615	23,7667	23,1342
3	29,0887	16,6643	28,4484	13,6202
4	23,788	22,0771	15,7108	29,1935
5	20,1435	14,5176	14,9548	15,1331
...	...	...	...	...
10	85,737	69,8121	67,3831	62,2925
Scores				
SIF2I<SIF1	10		6	
SIF2I==SIF1	0		0	

(b) Distância de referência.

Tabela 3: Resultados obtidos em experimentos reais.

Já na Tabela 2b, as 30 simulações apontam que a média da distância entre a posição do robô e a linha de referência usando SIF2I são menores. Na Tabela 3b, os experimentos com o robô real confirmaram os resultados relatados pelas simulações. Os resultados descritos nas Tabelas 2b e 3b mostram que a abordagem proposta com SIF2I permite ao robô navegar mais próximo da linha de referência.

6 Conclusão

Neste trabalho, propomos uma abordagem de navegação reativa para um robô com três rodas omnidirecionais usando SIF2I em ambientes *indoors* desconhecidos. Para validar a proposta, o robô foi submetido a simulações e experimentos reais onde ele navega como um seguidor de paredes. Ao final das simulações e experimentos, as métricas foram analisadas e viabilizaram a comparação entre o SIF1 e SIF2I. Os resultados mostraram que a abordagem usando SIF2I apresentou melhor desempenho ao lidar com fontes de incertezas em ambientes reais que o SIF1, permitindo assim, uma abordagem de navegação autônoma satisfatória.

Como trabalhos futuros, devemos aplicar técnicas evolucionárias para encontrar a configuração ótima dos sistemas *fuzzy*. Assim como implementar uma abordagem mais precisas para localização do robô.

Referências

- Ayedi, D., Boujelben, M. and Rekik, C. (2017). Interval type-2 TSK fuzzy approach for autonomous mobile robot control in presence of uncertainties, *2017 14th International Multi-Conference on Systems, Signals & Devices (SSD)*, IEEE, pp. 280–286.
- Baklouti, N. and Alimi, A. M. (2017). Interval type-2 beta fuzzy neural network for wheeled mobile robots obstacles avoidance, *2017 International Conference on Advanced Systems and Electric Technologies (IC-ASET)*, IEEE, pp. 481–486.
- Festo-didactic (2018). Disponível em: <<http://www.festo-didactic.com/>>. Acesso em: 27 Fev. 2018.
- Hagras, H. (2004a). A Hierarchical Type-2 Fuzzy Logic Control Architecture for Autonomous Mobile Robots, *IEEE Transactions on Fuzzy Systems* **12**(4): 524–539.
- Hagras, H. (2004b). A type-2 fuzzy logic controller for autonomous mobile robots, *2004 IEEE International Conference on Fuzzy Systems (IEEE Cat. No.04CH37542)* **2**: 965–970.
- Kim, C.-J. and Chwa, D. (2015). Obstacle Avoidance Method for Wheeled Mobile Robots Using Interval Type-2 Fuzzy Neural Network, *IEEE Transactions on Fuzzy Systems* **23**(3): 677–687.
- Kumbasar, T. and Hagras, H. (2013). A Type-2 Fuzzy Cascade Control Architecture for Mobile Robots, *2013 IEEE International Conference on Systems, Man, and Cybernetics* pp. 3226–3231.
- Linda, O. and Manic, M. (2010). Comparative analysis of Type-1 and Type-2 fuzzy control in context of learning behaviors for mobile robotics, *IECON 2010 - 36th Annual Conference on IEEE Industrial Electronics Society*, IEEE, pp. 1092–1098.
- Mbede, J. B., Melingui, A., Zobo, B. E., Merzouki, R. and Bouamama, B. O. (2012). zSlices based type-2 fuzzy motion control for autonomous robotino mobile robot, *Proceedings of 2012 IEEE/ASME 8th IEEE/ASME International Conference on Mechatronic and Embedded Systems and Applications*, IEEE, pp. 63–68.
- Mendel, J. M., Hagras, H., Tan, W.-W., Melek, W. W. and Hao Ying (2014). *Introduction To Type-2 Fuzzy Logic Control Theory And Applications*, Wiley, New Jersey.
- Mendel, J. M., John, R. I. and Liu, F. (2006). Interval Type-2 Fuzzy Logic Systems Made Simple, *IEEE Transactions on Fuzzy Systems* **14**(6): 808–821.
- Siegiwart, R., Nourbakhsh, I. R. and Scaramuzza, D. (2011). *Introduction to autonomous mobile robots*, second edn, MIT Press.
- Silva, W. A. S., Rabelo, R. A. L. and Santana, A. M. (2014). Interval Type-2 Fuzzy System for a safe autonomous robot navigation in uncertain environment, *2014 XL Latin American Computing Conference (CLEI)*, IEEE, pp. 1–8.
- Wagner, C. (2013). Juzzy - A Java based toolkit for Type-2 Fuzzy Logic, *2013 IEEE Symposium on Advances in Type-2 Fuzzy Logic Systems (T2FUZZ)*, IEEE, pp. 45–52.