



UNIVERSIDADE FEDERAL DE MINAS GERAIS
CURSO DE PÓS-GRADUAÇÃO EM ENGENHARIA ELÉTRICA

PRISCILA APARECIDA DIAS NICÁCIO

RELATÓRIO DE TÓPICOS ESPECIAIS EM SINAIS E SISTEMAS/F
CONTROLE USANDO SISTEMAS NEBULOSOS

PROFESSORES: LEONARDO A. MOZELLI E VÍCTOR COSTA DA SILVA CAMPOS

Belo Horizonte

2025

“Fuzzy logic is not fuzzy.
Basically, fuzzy logic is a precise
logic of imprecision and approximate
reasoning.”

— *Lotfi A. Zadeh, 2008.*

RESUMO

Este estudo investiga a simulação de um controlador fuzzy aplicado ao movimento de um robô diferencial, utilizando o ambiente "LabFuzzy/WebGL". Para isso, foram definidos os universos de discurso e os conjuntos fuzzy triangulares para as variáveis linguísticas essenciais, como Erro Angular (EA), Erro de Posição (EP), Velocidade Angular (VA) e Velocidade Linear (VEL). A partir dessas definições, elaboraram-se as regras de inferência, com o objetivo de regular o comportamento do robô, garantindo uma navegação suave e eficiente em direção ao alvo.

A construção do sistema fuzzy seguiu as etapas clássicas de fuzzificação, inferência e defuzzificação. A escolha dos parâmetros e funções de pertinência foi guiada por critérios de simetria, abrangência e consistência operacional, sendo ajustada com base em observações empíricas realizadas durante a simulação.

A explicação de como a linguagem natural foi utilizada para definir as regras de controle do sistema também é abordada e os resultados obtidos mostram que, mesmo com um conjunto simples de regras e funções, o sistema foi capaz de lidar eficientemente com diferentes cenários, destacando a eficácia da lógica fuzzy no tratamento da incerteza e na construção de controladores inteligentes.

A análise dos resultados válida a proposta do estudo e aponta direções promissoras para melhorias futuras, convidando o leitor a explorar em mais detalhes este sistema adaptativo e intuitivo.

SUMÁRIO

1. INTRODUÇÃO	6
2. MODELO CINEMÁTICO DO ROBÔ	6
3. VARIÁVEIS DE ENTRADA E SAÍDA DO CONTROLADOR	6
4. UNIVERSOS DE DISCURSO	7
5. CONTROLE UTILIZANDO LINGUAGEM NATURAL	7
6. REGRAS DE INFERÊNCIA DO CONTROLADOR FUZZY	8
7. DEFINIÇÃO E JUSTIFICATIVA DOS CONJUNTOS FUZZY	10
7.1 Funções De Pertinência Triangulares	12
8. SISTEMA DE INFERÊNCIA - MÉTODOS DE FUZZIFICAÇÃO E DEFUZZIFICAÇÃO	14
9. SIMULAÇÃO	15
10. ANÁLISE DOS RESULTADOS	16
10.1 Referências de Movimento na Simulação	17
11. CONCLUSÃO	18
12. REFERÊNCIAS BIBLIOGRÁFICAS	19

LISTA DE NOTAÇÕES

EA	Erro Angular — diferença entre a orientação atual do robô e a direção desejada
EP	Erro de Posição — distância entre o robô e o alvo
VA	Velocidade Angular — velocidade de rotação do robô
VEL	Velocidade Linear — velocidade de avanço do robô
NG	Negativo Grande — conjunto fuzzy para EA ou VA
NP	Negativo Pequeno — conjunto fuzzy para EA
Z	Zero — conjunto fuzzy para EA ou VA
PP	Positivo Pequeno — conjunto fuzzy para EA
PG	Positivo Grande — conjunto fuzzy para EA ou VA
μ	Grau de pertinência de um elemento a um conjunto fuzzy
Universo de Discurso	Intervalo que define os valores possíveis de uma variável linguística
θ (theta)	Ângulo de orientação do robô em relação ao referencial
ω (ômega)	Velocidade angular (rad/s) do robô diferencial
$\cos(\theta)$	Cosseno do ângulo θ , usado na equação de movimento do robô
$\sin(\theta)$	Seno do ângulo θ , usado na equação de movimento do robô
π (pi)	Constante matemática ($\approx 3,1416$), usada como limite do ângulo em radianos
rad	Radianos — unidade de medida angular no SI (Sistema Internacional)
Fuzzificação	Processo de conversão de valores numéricos em graus de pertinência
Inferência Fuzzy	Aplicação das regras fuzzy com base nas entradas
Defuzzificação	Conversão do resultado fuzzy em um valor numérico de controle
Regras IF-THEN	Regras linguísticas do tipo “Se... então...” usadas na inferência fuzzy

1. INTRODUÇÃO

A navegação de robôs móveis autônomos em ambientes dinâmicos e não estruturados representa um dos principais desafios da robótica moderna. A complexidade dos cenários reais e a necessidade de adaptação a situações incertas tornam os controladores tradicionais, baseados em modelos matemáticos precisos, muitas vezes insuficientes ou inflexíveis.

Nesse contexto, o controle fuzzy surge como uma alternativa eficaz, permitindo a modelagem de estratégias de controle através de regras linguísticas inspiradas no raciocínio humano, sem requerer um modelo exato do sistema. Este relatório tem como objetivo apresentar o desenvolvimento e a análise de um controlador fuzzy aplicado a um robô diferencial utilizando o simulador LabFuzzy/WebGL.

A partir da definição de variáveis linguísticas, conjuntos fuzzy e regras de inferência, buscou-se controlar o deslocamento do robô seguidor de forma suave, precisa e eficiente em relação ao robô líder. Ao longo deste estudo são detalhadas as etapas de modelagem do controlador, as configurações adotadas no simulador e as justificativas técnicas para cada decisão de projeto, com o intuito de demonstrar a efetividade da abordagem fuzzy no controle de sistemas dinâmicos sujeitos a incertezas e imprecisões.

2. MODELO CINEMÁTICO DO ROBÔ

O modelo cinemático bidimensional do robô é útil para determinar a movimentação do robô a partir dos sinais de controle gerados pelo sistema fuzzy e é descrito pelas equações diferenciais:

$$\begin{cases} \dot{x} = v \cos(\theta) \\ \dot{y} = v \sin(\theta) \\ \dot{\theta} = \omega \end{cases}$$

x, y : coordenadas da posição do robô no plano;

θ : orientação do robô em relação ao eixo x ;

v : velocidade linear;

ω : velocidade angular.

O robô diferencial utilizado na simulação é composto por duas rodas paralelas acionadas de forma independente, permitindo o movimento através da variação de velocidade angular de cada roda. Esse tipo de configuração é amplamente utilizado devido a simplicidade mecânica e capacidade de manobra.

3. VARIÁVEIS DE ENTRADA E SAÍDA DO CONTROLADOR

As variáveis linguísticas utilizadas para o desenvolvimento do controlador fuzzy foram divididas em duas entradas e duas saídas pela necessidade de alinhar o robô com seu alvo (EA) e manter distância adequada (EP) ajustando proporcionalmente sua orientação e avanço:

- Entradas:

- Erro Angular (EA) é diferença entre a orientação do robô seguidor e a linha imaginária que conecta o seguidor ao líder (representa o desalinhamento angular). Universo de discurso: $[-\pi; \pi]$. Conjuntos:

Negativo Grande (NG), Negativo Pequeno (NP), Zero (Z), Positivo Pequeno (PP), Positivo Grande (PG). Funções de pertinência: triangulares simétricas, centradas nos valores típicos da faixa de erro.

- Erro de Posição (EP) é a distância euclidiana entre o robô seguidor e o robô líder. Universo de discurso: $[0;10]$. Conjuntos: Perto, Médio, Longe. Conjuntos triangulares justapostos, para manter transições suaves entre as regiões.

- Saídas:

- Velocidade Angular (VA) controla o giro do robô, ajustando sua direção.

- Universo de discurso: $[-2; 2]$. Conjuntos: Negativa, Zero, Positiva. Forma e amplitude ajustadas para limitar a rotação excessiva do robô.

- Velocidade Linear (VEL) define a velocidade de deslocamento do robô em linha reta. Universo de discurso: $[0; 5]$. Conjuntos: Lenta, Média, Rápida: Prioriza movimentos suaves, evitando oscilações abruptas.

4. UNIVERSOS DE DISCURSO

Os universos de discurso foram obtidos baseando-se em dados experimentais, simulações prévias com robôs, faixas operacionais observadas no simulador LabFuzzy/WebGL e considerando controle com respostas mais suaves, menor distância entre os robôs e melhor estabilidade. Desse modo, foram observados:

- O domínio físico do problema considerando que o intervalo deve refletir os limites reais ou operacionais da variável. Para o Erro Angular (EA), considerando um robô móvel diferencial, os ângulos normalmente variam entre $-\pi$ e π rad (-180° a 180°), pois é o intervalo padrão para representar orientação angular.
- As variáveis envolvidas, haja vista que, o universo deve conter todos os valores possíveis que a variável pode assumir durante o funcionamento do sistema. Para o Erro de Posição (EP) foi limitado o tamanho do ambiente, foi estabelecido que o robô opera no espaço de até 10 metros de distância do alvo $[0, 10]$.
- O histórico de dados ou simulações realizadas em outros estudos para definir intervalos mais realistas e evitar extrapolações. Diante disso, após simulações em estudos anteriores, foi verificado que a velocidade angular da maioria dos robôs não ultrapassa $\pm 2,0$ rad/s, então o universo de VA foi definido como $[-2, 2]$.
- A garantia de segurança e margem operacional adequada, pois em alguns casos, opta-se por ampliar o intervalo do universo de discurso ligeiramente para aumentar a robustez do sistema e evitar comportamentos extremos diante de imprecisões inesperadas nas variáveis de entrada. No caso desta simulação, optou-se pelo intervalo $[0, 5]$ para o universo de discurso da variável. Esse intervalo ligeiramente ampliado contribui para a robustez do controle fuzzy, especialmente em situações com imprecisões nos sensores evitando respostas abruptas ou inadequadas do sistema.

5. CONTROLE UTILIZANDO LINGUAGEM NATURAL

Considere a seguinte hipótese de controle baseada em regras linguísticas:

- Se o Erro de Posição (EP) é elevado, o robô encontra-se significativamente afastado do alvo;
- Se o Erro Angular (EA) é nulo, o robô está orientado em direção ao ponto de interesse corretamente;

Então, a estratégia mais eficiente consiste em aplicar uma velocidade linear máxima e manter a velocidade angular nula, possibilitando um deslocamento direto e retilíneo até a posição desejada. Essa configuração otimiza o tempo de resposta e reduz o consumo energético ao evitar correções desnecessárias na trajetória.

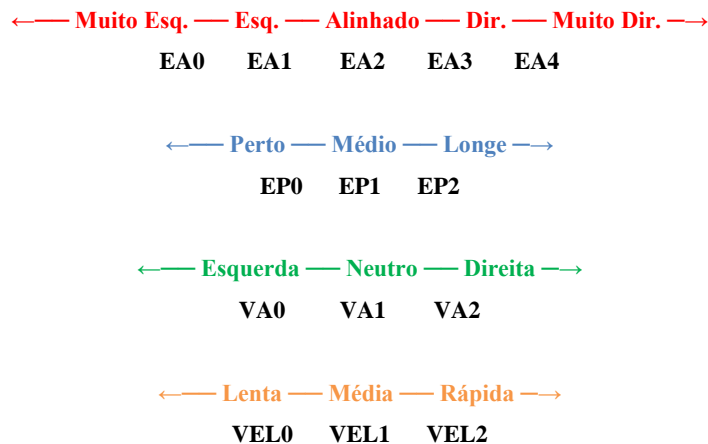
Por outro lado, quando o robô se encontra próximo do robô líder ($EP = \text{Perto}$), mas com um erro angular significativo ($EA \neq 0$), a prioridade do controle deve ser o ajuste da orientação. Nesse cenário, a velocidade linear deve ser reduzida, enquanto a velocidade angular é aumentada para permitir correção suave da direção. Essa abordagem evita oscilações, movimentos abruptos e ultrapassagens (overshoot), promovendo maior estabilidade no sistema e precisão no rastreamento do alvo.

Esse tipo de resposta ilustra uma lógica compatível com controladores especialistas baseados em regras heurísticas, nos quais as ações são tomadas com base em julgamentos linguísticos e condições contextuais.

Assim, quando o robô estiver muito distante e corretamente orientado, o controle privilegia o avanço em linha reta. Se estiver próximo, mas desalinhado, o foco passa a ser a correção angular, com deslocamento mais contido até que a direção adequada seja retomada.

6. REGRAS DE INFERÊNCIA DO CONTROLADOR FUZZY

O sistema fuzzy do robô seguidor foi modelado com base em 15 regras de inferência, estruturadas com duas entradas — Erro Angular (EA) e Erro de Posição (EP) — e duas saídas — Velocidade Angular (VA) e Velocidade Linear (VEL). Cada variável é dividida em conjuntos linguísticos triangulares:



Essa padronização (mapeamento dos rótulos) permitiu a criação das regras lógicas (ver tabela 1) que foram facilmente implementadas e registradas no simulador respeitando a limitação do número de conjuntos possíveis para cada variável:

"Se EP é EP2 e EA é EA2, então VA é VA1 e VEL é VEL2".

Tabela 1 – Regras do controlador Fuzzy

REGRAS DO CONTROLADOR FUZZY					
Regra	EA (Erro Angular)	EP (Erro de Posição)	VA (Velocidade Angular)	VEL (Velocidade Linear)	Justificativa
0	Zero	Longe	Moderada (VA1)	Rápida (VEL2)	Alinhado e longe → avança com velocidade
1	PP	Longe	Alta (VA2)	Rápida (VEL2)	Leve desalinhamento positivo, mas longe
2	NP	Longe	Zero (VA0)	Rápida (VEL2)	Leve desalinhamento negativo, não precisa girar
3	PG	Longe	Alta (VA2)	Média (VEL1)	Alto desalinhamento positivo → gira e reduz
4	NG	Longe	Zero (VA0)	Média (VEL1)	Muito desalinhado → reduz avanço
5	Zero	Médio	Moderada (VA1)	Rápida (VEL2)	Alinhado e a uma distância média
6	PP	Médio	Alta (VA2)	Média (VEL1)	Corrige e segue
7	NP	Médio	Zero (VA0)	Média (VEL1)	Leve desalinhamento à esquerda
8	PG	Médio	Alta (VA2)	Lenta (VEL0)	Muito desalinhado → gira e avança pouco
9	NG	Médio	Zero (VA0)	Lenta (VEL0)	Mesmo caso acima, para o outro lado
10	Zero	Perto	Moderada (VA1)	Média (VEL1)	Alinhado, mas próximo → reduz só um pouco
11	PP	Perto	Alta (VA2)	Lenta (VEL0)	Desalinhado e próximo → prioriza giro
12	NP	Perto	Zero (VA0)	Lenta (VEL0)	Leve desalinhamento e próximo → segue reto devagar
13	PG	Perto	Alta (VA2)	Lenta (VEL0)	Alto desalinhamento e muito perto
14	NG	Perto	Zero (VA0)	Lenta (VEL0)	Muito desalinhado e perto → não gira

Fonte: Elaborado pela autora

Na sequência, as justificativas foram convertidas em regras para lançamento no simulador:

Tabela 2: Sequência de Regras a serem lançadas no simulador LabFuzzy/WebGL.

Regra	EA	EP	VA	VEL
0	EA2	EP2	VA1	VEL2
1	EA3	EP2	VA2	VEL2
2	EA1	EP2	VA0	VEL2
3	EA4	EP2	VA2	VEL1
4	EA0	EP2	VA0	VEL1
5	EA2	EP1	VA1	VEL2
6	EA3	EP1	VA2	VEL1
7	EA1	EP1	VA0	VEL1
8	EA4	EP1	VA2	VEL0
9	EA0	EP1	VA0	VEL0
10	EA2	EP0	VA1	VEL1
11	EA3	EP0	VA2	VEL0
12	EA1	EP0	VA0	VEL0
13	EA4	EP0	VA2	VEL0
14	EA0	EP0	VA0	VEL0

Fonte: Elaborado pela autora

Desse modo, foram inseridas no simulador as regras lógicas “Se-Então” — da Tabela 2 — e realizadas as configurações restantes.

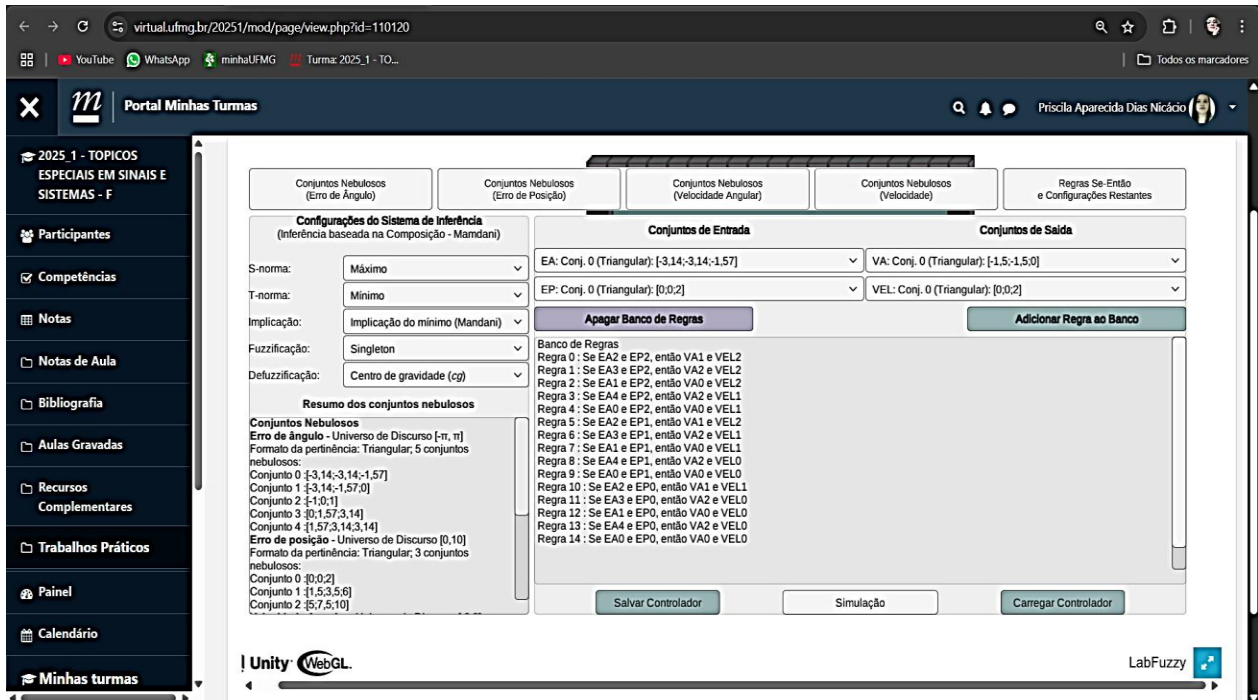


Figura 1 –Regras lógicas no simulador LabFuzzy/WebGL

7. DEFINIÇÃO E JUSTIFICATIVA DOS CONJUNTOS FUZZY

A definição das posições dos conjuntos fuzzy foi feita com base no contexto do problema de controle, nas fórmulas de cálculo fornecidas e nos critérios de simetria e cobertura do universo de discurso. Também foram considerados os limites físicos das variáveis, conforme a cinemática diferencial do robô. A lógica fuzzy oferece flexibilidade na modelagem de variáveis contínuas, mas o número de conjuntos definidos impacta diretamente a granularidade do controle. Um número excessivo de conjuntos dificulta a interpretação do sistema e aumenta a complexidade computacional, além de exigir uma base de regras mais extensa e propensa a inconsistências. Por outro lado, um número reduzido de conjuntos pode simplificar o controle, mas pode gerar respostas abruptas ou imprecisas. Para cada variável, foi determinada a quantidade mínima de conjuntos necessária para garantir boa sensibilidade e interpretabilidade no controle:

- Erro Angular (EA) – 5 conjuntos fuzzy: Universo de Discurso: $[-\pi, \pi] \approx [-3.14, 3.14]$.

Conjuntos definidos: NG (Negativo Grande), NP (Negativo Pequeno), Z (Zero), PP (Positivo Pequeno), PG (Positivo Grande). O erro angular é a variável mais sensível no sistema, pois determina o desalinhamento entre a orientação atual e a desejada do robô. Ao utilizar cinco conjuntos, representa-se com precisão tanto desalinhamentos significativos quanto pequenos, em ambas as direções (esquerda e direita), além do alinhamento perfeito (Z). Se

usássemos apenas três conjuntos (Negativo, Zero e Positivo), o controle ficaria mais brusco, dificultando ajustes finos na trajetória. Por outro lado, com sete conjuntos, o sistema seria mais refinado, mas exigiria uma base de regras excessivamente complexa, o que não traria ganhos significativos para um problema com movimentação contínua como este.

- Erro de Posição (EP) – 3 conjuntos fuzzy: Universo de Discurso: $[0, 10]$.

Conjuntos definidos: Perto, Médio, Longe. O erro de posição reflete a distância escalar até o robô líder. Três categorias são suficientes para controlar a aproximação de maneira eficaz: maior velocidade quando o robô está distante do alvo e desaceleração progressiva à medida que se aproxima. Com apenas dois conjuntos (Perto e Longe), a transição seria abrupta. Já com quatro ou mais conjuntos, a complexidade aumentaria sem trazer um benefício real na precisão do controle. O conjunto "Perto" foi definido com vértices em $(0, 0, 2)$, promovendo respostas mais cautelosas quando o robô está perto do destino. O conjunto "Médio" $(1.5, 3.5, 6)$ cobre a zona de transição de forma eficiente, sobrepondo-se aos conjuntos "Perto" e "Longe" para um controle mais adaptativo. Logo, o conjunto "Longe" $(5, 7.5, 10)$ lida com distâncias maiores e ativa comportamentos mais agressivos de aproximação.

- Velocidade Angular (VA) – 3 conjuntos fuzzy: Universo de Discurso: $[-2, 2]$.

Conjuntos definidos: Negativa, Zero, Positiva. A velocidade angular, como resposta ao erro angular, deve indicar claramente a direção do giro (horário ou anti-horário) ou a ausência de rotação. Três conjuntos são ideais para capturar essas situações: os extremos $(-1.5, 1.5)$ permitem manobras corretivas rápidas, enquanto o conjunto "Zero" $(-0.5, 0, 0.5)$ proporciona estabilidade quando o alinhamento está adequado. Essa distribuição oferece um equilíbrio entre agilidade e suavidade nas correções de direção. A utilização de mais conjuntos, como "Negativa Grande" ou "Positiva Grande", traria maior precisão, mas exigiria uma base de regras mais detalhada, o que poderia tornar o sistema mais difícil de ajustar e entender.

- Velocidade Linear (VEL) – 3 conjuntos fuzzy: Universo de Discurso: $[0, 5]$.

Conjuntos definidos: Lenta, Média, Rápida. A velocidade linear está diretamente relacionada à distância até o destino e deve refletir um controle de aproximação suave. Três níveis são suficientes para controlar a aceleração, a desaceleração e o deslocamento em velocidade constante. Se usássemos apenas dois conjuntos (Lenta e Rápida), a transição seria muito brusca. Já incluir mais de três (como "Muito Lenta" e "Muito Rápida") não traria benefícios significativos nesse contexto. Os conjuntos "Lenta", "Média" e "Rápida" foram posicionados ao longo do intervalo $[0, 5]$ de forma progressiva: "Lenta" $(0, 0, 2)$ controla o movimento em proximidade com o alvo, "Média" $(1, 2, 3)$ proporciona uma transição suave, e "Rápida" $(2, 4, 4)$ favorece a aceleração em distâncias maiores, garantindo um comportamento dinâmico e eficiente.

Os conjuntos fuzzy foram definidos com base nos intervalos de variação relevantes de cada variável, garantindo sobreposição adequada entre eles para uma interpolação eficaz e buscando simplicidade lógica. Todos os conjuntos adotados têm forma triangular, o que facilita sua implementação e interpretação. Os conjuntos extremos

foram posicionados nas extremidades dos universos de discurso, enquanto os intermediários foram distribuídos de forma simétrica ou proporcional.

A escolha dos vértices dos conjuntos foi realizada a partir de observações da simulação, visando otimizar o comportamento do robô em diferentes cenários de aproximação e alinhamento. Os testes preliminares confirmaram que esses ajustes melhoraram a fluidez e a estabilidade do movimento. Embora os vértices tenham sido definidos heurísticamente, com base na análise do sistema e em dados empíricos, métodos formais, como análise estatística ou otimização, poderiam ser aplicados para aprimorar o processo, especialmente em sistemas reais com maior variabilidade.

7.1 Funções De Pertinência Triangulares

Erro Angular (EA): Universo $[-\pi, \pi]$

NG (Negativo Grande): (-3.14, -3.14, -1.57)

NP (Negativo Pequeno): (-3.14, -1.57, 0)

Z (Zero): (-1, 0, 1)

PP (Positivo Pequeno): (0, 1.57, 3.14)

PG (Positivo Grande): (1.57, 3.14, 3.14)

Erro de Posição (EP): Universo $[0, 10]$

Perto: (0, 0, 2)

Médio: (1,5, 3,5, 6)

Longe: (5, 7,5, 10)

Velocidade Angular (VA): Universo $[-2, 2]$

Negativa: (-1.5, -1.5, 0)

Zero: (-0.5, 0, 0.5)

Positiva: (0, 1.5, 1.5)

Velocidade Linear (VEL): Universo $[0, 5]$

Lenta: (0, 0, 2)

Média: (1, 2, 3)

Rápida: (2, 4, 4)

Em seguida, os parâmetros das funções de pertinência foram inseridos no simulador, respeitando as posições previamente definidas para os conjuntos fuzzy e os universos de discurso atribuídos a cada variável.

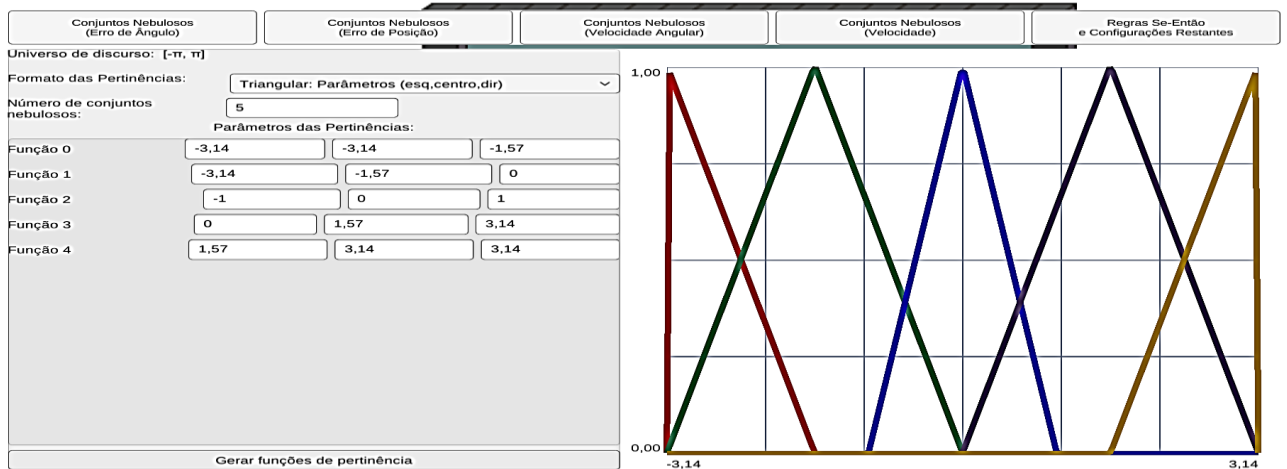


Figura 2 – Lançamento de dados do conjuntos nebuloso EA.

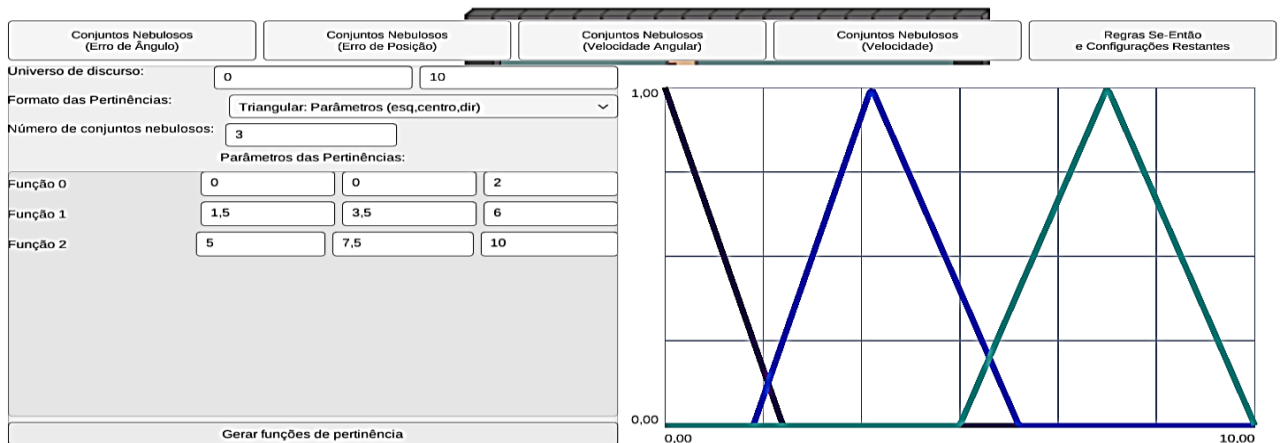


Figura 3 – Lançamento de dados do conjuntos nebuloso EP.

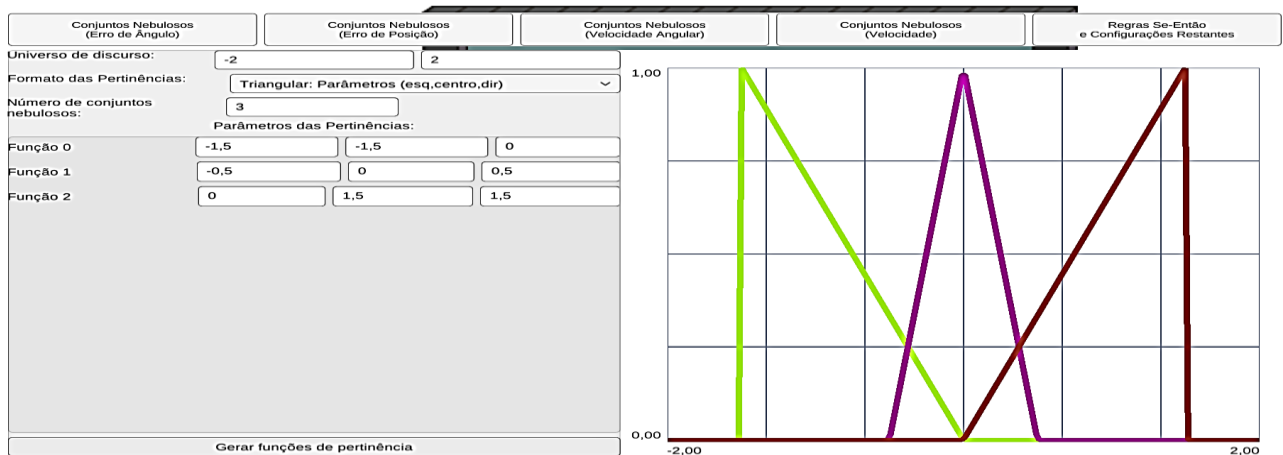


Figura 4 – Lançamento de dados do conjuntos nebuloso VA.

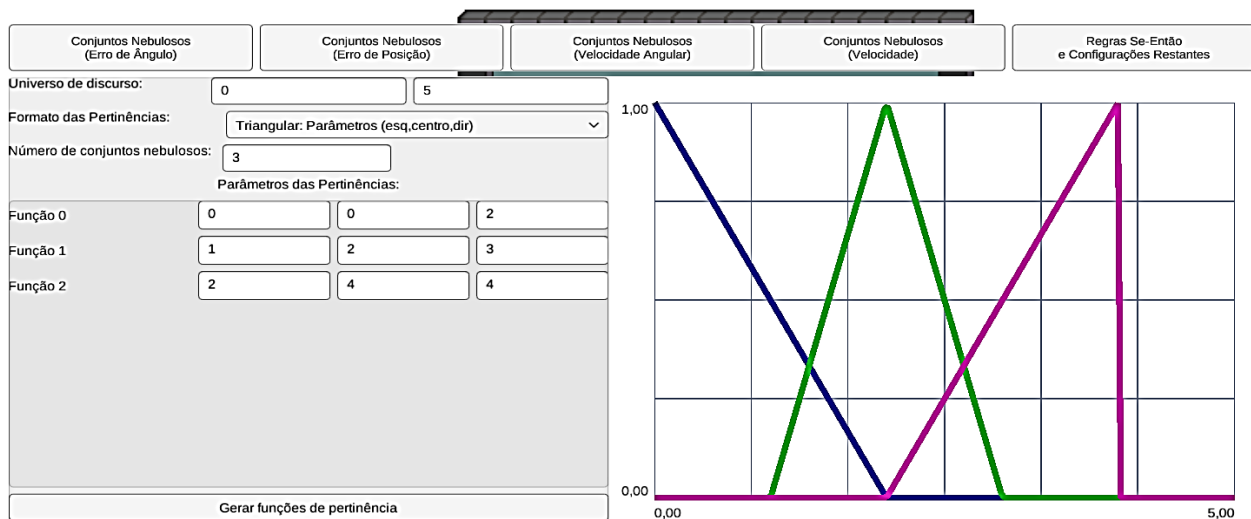


Figura 5 – Lançamento de dados do conjuntos nebuloso VEL.

8. SISTEMA DE INFERÊNCIA - MÉTODOS DE FUZZIFICAÇÃO E DEFUZZIFICAÇÃO

O controlador fuzzy desenvolvido para o robô seguidor foi estruturado com base na arquitetura clássica composta pelas etapas de fuzzificação, inferência e defuzzificação, integradas de forma a garantir transições suaves e contínuas entre os estados do sistema. Essa abordagem mostrou-se eficaz na simulação realizada no ambiente LabFuzzy/WebGL, onde a estabilidade da trajetória e a resposta fluida do robô às variações de posição e orientação do robô líder foram requisitos fundamentais. As configurações adotadas para o sistema de inferência fuzzy foram definidas com foco em simplicidade computacional, boa interpretabilidade e desempenho eficaz na simulação:

- **Fuzzificação Singleton:** As entradas do sistema — erro angular (EA) e erro de posição (EP) — foram tratadas como valores pontuais, com grau de pertinência 1 apenas na posição exata do valor de entrada. Essa abordagem (estabelecida pelos professores da disciplina), simplifica o processo de fuzzificação, reduzindo o custo computacional sem comprometer a qualidade do controle, ideal para simulações interativas como a realizada.
- **T-norma (conjunção):** Para a avaliação dos antecedentes das regras fuzzy, foi utilizada a operação de mínimo, que exige que todas as condições sejam satisfeitas simultaneamente. Esse operador garante uma interpretação intuitiva e conservadora das regras, condizente com as exigências do controle do movimento.
- **Implicação:** A implicação fuzzy também foi baseada na operação de mínimo, o que limita a ativação da saída ao menor grau de verdade do antecedente. Essa escolha é coerente com a t-norma adotada e contribui para evitar respostas desproporcionais nas ações de controle, o que foi observado como positivo para manter a suavidade dos movimentos do robô seguidor.
- **S-norma (agregação):** Para a combinação das contribuições das diferentes regras ativadas, utilizou-se o operador máximo. Dessa forma, a maior ativação entre as regras é preservada, garantindo que a regra mais relevante tenha maior influência no resultado final.

- Defuzzificação: A defuzzificação foi realizada pelo método do centro de gravidade, que calcula a média ponderada dos valores de saída fuzzy. Esse método foi decisivo para gerar valores contínuos e suaves, o que se refletiu diretamente na estabilidade da trajetória e na minimização de oscilações no movimento do robô seguidor.

Essa configuração permitiu ao controlador fuzzy gerar ações de controle contínuas e realistas, evitando mudanças abruptas na velocidade linear e angular do robô seguidor. Como resultado, o robô seguidor manteve uma distância adequada e constante em relação ao robô líder, com movimentos suaves e bem ajustados às variações do ambiente simulado.

9. SIMULAÇÃO

🔗 Links para visualização das simulações:

- Simulação no LabFuzzy/WebGL: <https://youtu.be/3C-k2cj7LVI>
- Simulação no MATLAB: <https://youtu.be/2IBIxBWELxs>

Observação: Os vídeos demonstram o comportamento do controlador fuzzy em resposta às três referências disponíveis no ambiente de simulação: circular, retangular e manual tanto no Matlab quanto no LabFuzzy/WebGL.

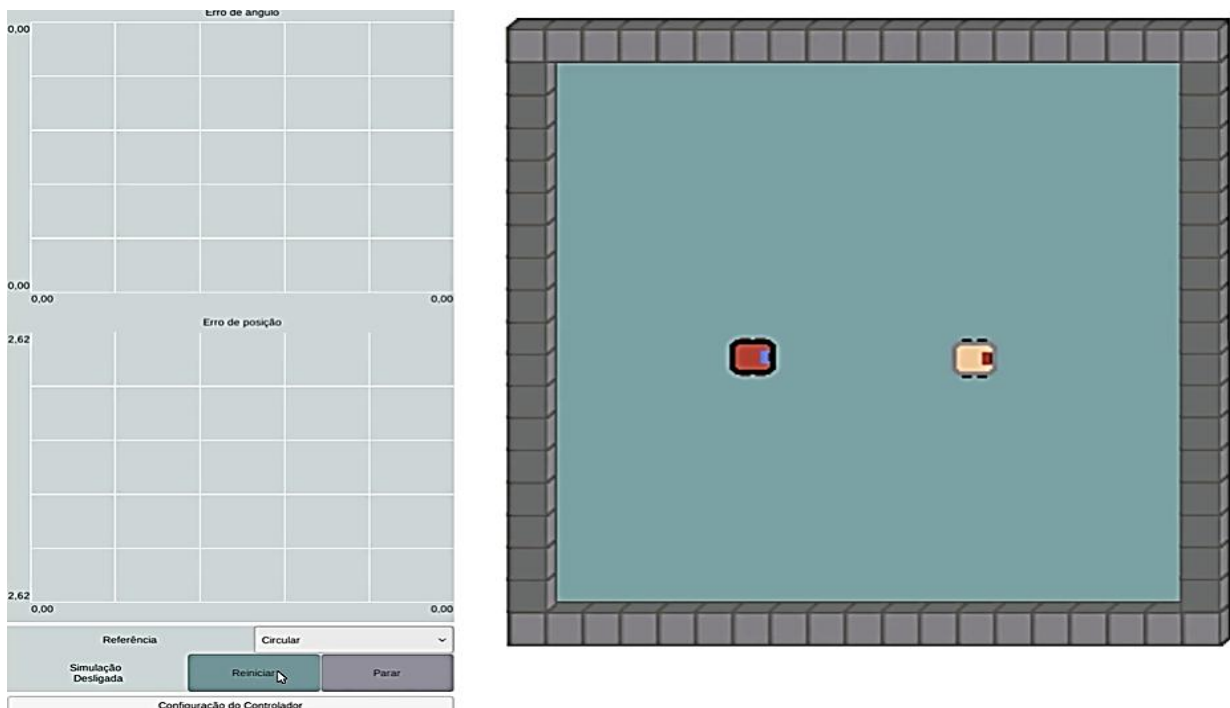


Figura 6 – Robô seguidor (em vermelho) em relação ao robô líder no simulador LabFuzzy/WebGL.

10. ANÁLISE DOS RESULTADOS

A estrutura adotada no controlador fuzzy foi desenvolvida com base em funções de pertinência triangulares, aplicadas às variáveis linguísticas do sistema e buscou representar, de forma intuitiva e eficiente, os estados de entrada e saída relevantes para o controle do robô diferencial seguidor. Foram estabelecidas 15 regras fuzzy que correlacionam as variáveis de entrada (EA e EP) com as saídas (VA e VEL), com o objetivo de promover um comportamento de controle suave, responsivo e seguro, respeitando as exigências dinâmicas de um sistema robótico realista.

A simulação realizada no ambiente LabFuzzy/WebGL demonstrou o comportamento suave, estável e coerente para o robô diferencial seguidor (representado pelo carro vermelho). Os conjuntos fuzzy utilizados foram obtidos com base em critérios técnicos que garantem transições contínuas entre os diferentes estados do sistema, permitindo respostas adequadas a distintas condições de erro angular e de posição. O robô seguidor apresentou desempenho satisfatório, com trajetória fluida, mantendo a orientação em direção ao robô líder e reduzindo gradualmente a distância relativa. A lógica fuzzy mostrou-se capaz de realizar ajustes dinâmicos tanto na velocidade linear quanto na velocidade angular:

- Pequenos erros angulares ($EA \approx 0$): o robô mantém a trajetória com velocidade angular próxima de zero ($VA \approx 0$), priorizando o avanço retilíneo.
- Grandes erros angulares: são corrigidos por meio de ajustes apropriados em VA, com valores positivos ou negativos, dependendo da direção do desalinhamento.
- Variação da distância (EP): a velocidade linear (VEL) é ajustada proporcionalmente, reduzindo conforme o robô se aproxima do robô líder, o que contribui para evitar aproximações bruscas.

Vale ressaltar que, nesse estudo inicialmente, os conjuntos fuzzy associados à variável Velocidade Linear (VEL) foram definidos considerando um universo de discurso reduzido, conforme segue:

- Velocidade Linear (VEL): Universo de discurso [0,2.5]

Lenta: (0,0,1)

Média: (0.5,1,1.5)

Rápida: (1,2,2)

Contudo, observou-se que, com esse intervalo mais restrito, a resposta do robô seguidor apresentava uma velocidade inferior à desejada, resultando em um distanciamento excessivo em relação ao robô líder.

Para melhorar a performance do controle e reduzir essa defasagem, o universo de discurso da variável VEL foi expandido, passando a considerar o intervalo completo sugerido pelo ambiente de simulação, conforme abaixo:

- Velocidade Linear (VEL): Universo de discurso [0,5]

Lenta: (0,0,2)

Média: (1,2,3)

Rápida: (2,4,4)

Esse ajuste permitiu um aumento mais expressivo na velocidade do robô seguidor, contribuindo para uma trajetória mais fluida e com menor distância em relação ao robô líder, sem comprometer a estabilidade do sistema nem afetar negativamente as demais variáveis da simulação.

O comportamento observado na simulação reforça que os conjuntos fuzzy e as regras de inferência estão bem balanceados: não foram identificados oscilações, travamentos ou comportamentos instáveis, o que valida a consistência do modelo. Durante a simulação, o robô diferencial apresentou:

- Movimento contínuo e sem oscilações bruscas, mesmo nas transições entre erros angulares elevados e baixos, evidenciando interpolação eficiente entre os conjuntos fuzzy.
- Aproximação progressiva do robô líder, com redução gradual da velocidade linear conforme o erro de posição diminui, respeitando os limites definidos para aproximação segura.
- Correção angular adequada, com maior intensidade quando EA era elevado (NG ou PG) e menor intensidade à medida que se aproximava do alinhamento ($EA \approx Z$).
- Correspondência entre as regras fuzzy definidas e o comportamento prático.

10.1 Referências de Movimento na Simulação

Ao final do vídeo da simulação, o ambiente exibe três comportamentos de referência — circular, retangular e manual — os quais servem como base para avaliação qualitativa do desempenho do controlador:

- Movimento Circular: Na simulação, o controle fuzzy favorece o movimento circular com curvas amplas, ajustando a trajetória de forma contínua, com boa fluidez e respostas suaves a variações de erro angular. O robô seguidor realizou correções gradativas e curvas suaves durante a aproximação ao robô líder.
- Movimento Retangular: O comportamento do robô seguidor na simulação é compatível com a movimentação em linha reta e curvas ortogonais — mais quadrado, com paradas e realinhamentos visíveis. O robô seguidor tende a reduzir bruscamente a velocidade linear (VEL) conforme se aproxima — coerente com um controle mais "reto e travado", típico da referência retangular.
- Movimento Manual (Controle Ideal): À medida que o robô seguidor se aproxima do robô líder, sua trajetória inicialmente reta evolui para um movimento lento e contínuo ao redor do alvo. Esse comportamento decorre da redução do erro de posição (EP0) e do erro angular ($EA \approx 0$), ativando regras fuzzy que priorizam segurança e estabilidade — com velocidade linear reduzida (VEL0 ou VEL1) e praticamente nenhuma rotação (VA0). Como os conjuntos são suavemente sobrepostos, o robô seguidor realiza pequenos ajustes mesmo em estados estáveis, o que resulta em uma leve órbita em torno do líder, reproduzindo com fidelidade o comportamento esperado para um controle fuzzy bem calibrado.

Os resultados obtidos indicam que o controlador fuzzy lidou de forma eficiente com diferentes condições de erro e distância, promovendo um comportamento natural e eficaz ao robô seguidor. A inferência fuzzy se mostrou apropriada para a tarefa de navegação baseada em controle reativo, contribuindo para um acompanhamento contínuo e seguro do robô líder.

11. CONCLUSÃO

Este estudo apresentou o projeto e a implementação de um controlador baseado em lógica fuzzy para um robô diferencial, com o objetivo de seguir um robô líder mantendo uma distância adequada e promovendo movimentos suaves e estáveis.

A abordagem fuzzy mostrou-se eficaz para lidar com a natureza contínua e incerta do ambiente, possibilitando que o robô seguidor se adaptasse dinamicamente às variações de posição e orientação do líder. A definição criteriosa das variáveis linguísticas, o uso de conjuntos fuzzy triangulares e a formulação de regras heurísticas intuitivas resultaram em respostas realistas e consistentes.

A estrutura do sistema de inferência — composta por fuzzificação singleton, t-norma do mínimo, s-norma do máximo, implicação pelo mínimo e defuzzificação por centro de gravidade — proporcionou decisões de controle fluidas com baixo custo computacional.

Os resultados das simulações confirmaram que o comportamento do robô esteve em consonância com os padrões esperados, validando tanto o desenho dos conjuntos fuzzy quanto a lógica das regras estabelecidas.

O robô foi capaz de tomar decisões de navegação de maneira contínua, adaptativa e eficiente, sem a necessidade de modelos matemáticos complexos ou técnicas convencionais de controle. A partir da análise dos testes, foram identificadas oportunidades relevantes de aprimoramento, capazes de elevar ainda mais a robustez e a performance do controlador desenvolvido:

- Expandir o conjunto "Longe" e revisar suas regras para tornar o sistema mais ágil, reduzindo o tempo de aproximação ao alvo, mesmo com grandes desvios angulares.
- Intensificar a velocidade angular em casos de desalinhamento acentuado ($EA = NG$ ou PG) para corrigir a trajetória mais rapidamente.
- Refinar a sobreposição entre conjuntos fuzzy para garantir transições suaves e movimento fluido.
- Implementar adaptação dinâmica nas regras ou pesos fuzzy, ajustando-os conforme o erro de posição ao longo da trajetória.
- Incorporar uma lógica preditiva simples para antecipar tendências de erro, aumentando a eficiência e agilidade do controle.
- Testar o sistema em condições de ruído sensorial para avaliar sua robustez em cenários mais realistas e desafiadores.
- Extensão do controlador fuzzy para contextos com múltiplos alvos e obstáculos, favorecendo comportamentos cooperativos ou evasivos. Essa evolução ampliaria significativamente a aplicabilidade do sistema em ambientes robóticos reais, dinâmicos e de maior complexidade.

12. REFERÊNCIAS BIBLIOGRÁFICAS

1. PASSINO, Kevin M.; YURKOVICH, Stephen. *Fuzzy control*. Menlo Park: Addison Wesley Longman, 1998.
2. ZADEH, Lotfi A. Fuzzy sets. *Information and Control*, v. 8, n. 3, p. 338–353, 1965.
3. SILVA, Wagner F. da; ALMEIDA, Leandro dos S.; RODRIGUES, Douglas T. Desenvolvimento de um sistema de controle fuzzy para robô seguidor de linha. *Revista Eletrônica Científica da FATEC*, v. 3, n. 2, p. 14-23, 2017.
4. SIMULADOR: *TPI – Robô móvel (LabFuzzy/WebGL)*. Ambiente de simulação disponibilizado via Moodle. Universidade Federal de Minas Gerais (UFMG), 2025.
5. ROSS, Timothy J. *Fuzzy logic with engineering applications*. 4. ed. New York: Wiley, 2016.
6. DRIANKOV, Dimiter; HELLENDORF, Hans; REINFRANK, Michael. *An introduction to fuzzy control*. 2. ed. London: Springer, 1996.
7. TANAKA, Kazuo. *Fuzzy control systems design and analysis: a linear matrix inequality approach*. New York: Wiley, 2001.
8. SANTOS, R. T. dos; VELLOSO, A. G. D. Controle de robôs móveis: uma abordagem baseada em lógica fuzzy. *Revista Controle & Automação*, v. 18, n. 1, p. 15-22, 2007.