



INSTITUTO FEDERAL DE EDUCAÇÃO CIÊNCIA E TECNOLOGIA - IFPE

CURSO DE BACHARELADO EM ENGENHARIA MECÂNICA

ACARCIO GOMES DE ALMEIDA JUNIOR

CONTROLE DE TRAJETÓRIA DO TIPO FUZZY MAMDANI PARA UM ROBÔ
PARALELO

CARUARU-PE

2021

ACARCIO GOMES DE ALMEIDA JUNIOR

CONTROLE DE TRAJETÓRIA DO TIPO FUZZY MAMDANI PARA UM ROBÔ
PARALELO

Trabalho de conclusão de curso apresentado ao Instituto de Educação, Ciência e Tecnologia de Pernambuco, como parte dos requisitos necessários para à obtenção do título de Engenheiro Mecânico.

Orientador: Professor Doutor Alexander Patrick Chaves de Sena

CARUARU-PE

2021

A447c

Almeida Junior, Acarcio Gomes de

Controle de trajetória do tipo Fuzzy Mamdani para um robô paralelo / Acarcio Gomes de Almeida Junior; orientação de Alexander Patrick Chaves de Sena. – Caruaru, 2021.

52 f.

Trabalho de Conclusão de Curso – (Graduação em Engenharia Mecânica) Instituto Federal de Educação, Ciência e Tecnologia de Pernambuco, Departamento de Engenharia Mecânica, Curso de Engenharia Mecânica, 2021.

1. Controle de máquinas. 2. Projeto de máquinas. 3. Robótica. 4. Automação de processos. 5. Robô delta. 6. Arquitetura paralela. 7. Modelagem Fuzzt Mamdani. 8. Protótipo robótico. 9. Moldeamento matemático. 10. Engenharia Mecânica - IFPE. I. Sena, Alexander Patrick Chaves de. II. Instituto Federal de Educação, Ciência e Tecnologia de Pernambuco. III. Trabalho de Conclusão de Curso. IV. Título.

CDD: 629.89

CDU: 621.313

AGRADECIMENTOS

A Deus por ter dado força e perseverança em todos os momentos de dificuldades.

Aos meus pais, Acarcio Gomes e Ezenilda Bezerra, por serem as maiores referências de pessoas em minha vida.

Ao meu tio Valdevino Bezerra, por todo suporte oferecido durante o período que estive com ele. A minhas irmãs Ávila e Águida. A toda minha família, que sempre me incentivaram.

Aos amigos Prof Cícero Correia, Evaldo Cavalcante, Gilmar Souto, Maycon Ferreira, Rafael Dantas e Wellis Eloi, por todo o apoio desde o ensino médio.

Ao amigo e orientador Prof. Dr. Alexander Patrick Chaves de Sena, o qual foi fundamental em toda minha vida acadêmica, me apresentando ao mundo da pesquisa e proporcionando experiências únicas de vida e acadêmicas, por meio de intercâmbios, participações de projetos e congressos internacionais.

Ao IIDV (Instituto Internacional Despertando Vocações) por ter me proporcionado uma experiência de vida única, por meio de um intercâmbio no Chile. Aos amigos Jonathan, Lucas e Rafael Lucena, por compartilharem comigo essa experiência. A Universidad Tecnológica de Chile – INACAP, por ter nos acolhido e ter dado todo o suporte durante este período.

Aos meus amigos Bruno, Cacio, Lucas Araújo, Maycon Ferreira, Thiago Duarte, Victor Cavalcanti e Vinicius Adriano, por todos os momentos compartilhados durante este período.

Aos amigos Asafe Santos, Cleyson, João e Jullyandryos, pelos momentos compartilhados no Laboratório de Robótica do IFPE.

A todas as pessoas que contribuíram direta ou indiretamente para a realização deste trabalho.

“The world ain’t all Sunshine and rainbows. It’s a very mean and nasty place and I don’t care how tough you are it will beat you to your knees and kepp you there permanently if you let it.

You, me, or nobody is gonna hit as hard as life.

*But it ain’t about how hard you hit,
it’s about how hard you can get hit and keep moving forward.*

How much you can take and keep moving forward.

That’s how winning is done”

Rocky Balboa

RESUMO

O presente trabalho apresenta o desenvolvimento de um controlador Fuzzy tipo Mamdani para a trajetória contínua no espaço de trabalho de um robô paralelo tipo delta linear. O sistema é composto por três atuadores posicionadores acionados por motores de corrente contínua com encoders incrementais para leitura de deslocamentos. Um módulo multifuncional de dados realizou a interface entre o computador e a máquina, captando os sinais provenientes dos encoders e fornecendo os sinais de controle aos circuitos de potência que acionam os motores. O software LabVIEW® foi empregado para a implementação do controlador, utilizando o toolkit de lógica Fuzzy, e o Matlab® para analisar e apresentar os resultados experimentais. A estratégia de controle apresenta três etapas de controle: a etapa de controle de velocidade, a etapa de controle de trajetória e a etapa de geração da trajetória. Para verificar o comportamento do sistema com os controladores, foram obtidos resultados experimentais em duas condições: apenas o controle de velocidade simples e o controle de velocidade a partir do acompanhamento de uma trajetória. Os resultados demonstraram que a estratégia de controle idealizada obteve bons índices de desempenho. Na análise da curva de resposta do sistema e dos dados obtidos, foi verificado um desempenho muito satisfatório do controlador atuando em 100 mm/s. O erro de regime permanente foi de -1,50mm para a trajetória em X, de -1,76mm para a trajetória em Y e de 1,02mm para a trajetória em Z, resultando em um erro de trajetória de 1,26%.

Palavra-Chave: Controle fuzzy Mamdani, Arquitetura paralela, Modelamento matemático, Robô delta, Controle de trajetória.

ABSTRACT

The present work presents the development of a Mamdani-type Fuzzy controller for the continuous trajectory in the workspace of a linear delta-type parallel robot. The system comprises three positioner actuators driven by direct current motors with incremental encoders for displacement reading. A multifunctional data module made the interface between the computer and the machine, capturing the signals coming from the encoders and supplying the control signals to the power circuits that drive the motors. LabVIEW[®] software was used to implement the controller, using the Fuzzy logic toolkit, and Matlab[®] to analyze and present the experimental results. The control strategy has three control steps: the speed control step, the trajectory control step, and the trajectory generation step. To verify the behavior of the system with the controllers, experimental results were obtained under two conditions: only the simple speed control and the speed control based on the tracking of a trajectory. The results showed that the idealized control strategy obtained good performance indices. In the analysis of the response curve of the system and the data obtained, a very satisfactory performance of the controller acting at 100 mm/s was verified. The steady state error was -1,50mm for the path in X, -1,76mm for the path in Y and 1,02mm for the path in Z, resulting in a path error of 1,26%.

Keyword: Mamdani fuzzy control, Parallel architecture, Mathematical modeling, Delta robot, Trajectory control.

LISTA DE FIGURAS

Figura 2.1: Elementos de um braço robótico. Fonte: Tartari Filho (2006).....	5
Figura 2.2. Cadeia cinemática: a) Aberta (robô serial) e b) fechada (robô paralelo). Adaptado de Carbone (2011).	6
Figura 2.3 Estrutura típica de robô delta linear: a) desenho em CAD e b) esquemático estrutural do conjunto. Adaptado de Li e Xu (2007).....	8
Figura 2.4. a) Representação da Cinemática Direta; b) Representação da Cinemática Inversa. Fonte: Autoria própria (2021).....	10
Figura 2.5. Representação Vetorial do Robô Paralelo. Fonte: Autoria própria (2021).....	11
Figura 2.6. Vetores base fixa. Fonte: Autoria própria (2021).	12
Figura 2.7. a) Robô paralelo, b) componentes utilizados na modelagem, b) representação vetorial da cadeia 1. Fonte: Autoria própria (2021).	12
Figura 2.8. Representação do Vetor D na máquina. Fonte: Autoria própria (2021).	13
Figura 2.9. Representação vetor E. Fonte: Autoria própria (2021).	14
Figura 2.10. Vetor da plataforma móvel. Fonte: Autoria própria (2021).....	14
Figura 2.11. Filosofia de controle clássico. Fonte: Autoria própria (2021).	19
Figura 2.12. Filosofia de controle fuzzy. Fonte: Silva, et al. (2019).....	20
Figura 2.13. Funções de pertinência tipo triangular para variável nível com as devidas variáveis linguísticas. Adaptado de Simões e Shaw (2007).	21
Figura 2.14. Representações gráficas: (a) conjunto A de função triangular, (b) conjunto B de função trapezoidal, (c) operação de união entre A e B, (d) operação de intersecção entre A e B, (e) operação de complemento de A e (f) operação de complemento de B. Fonte: Sena (2015).	23
Figura 2.15. Defuzzyficação pela média do máximo (MoM). Adaptado de Osiro (2013).	24
Figura 2.16. Defuzzyficação pelo centro de área (CoA). Adaptado de Osiro (2013).	25
Figura 3.1. Estrutura do robô delta controlado. Fonte: Silva et. al. (2021).....	26
Figura 3.2. Diagrama funcional de um sistema de aquisição por computador. Fonte: Silva Neto (2019).....	27
Figura 3.3. Encoder incremental com código Gray. Fonte: Silva Neto (2019).....	28
Figura 3.4. Detalhamento do hardware de aquisição montado. Fonte: Silva Neto (2019).....	29
Figura 3.5. Diagrama funcional do sistema de acionamento de acionamento. Fonte: Silva Neto (2019).	29
Figura 3.6. Detalhamento do hardware de acionamento montado. Fonte: Silva Neto (2019).	31

Figura 3.7. Diagrama funcional do sistema de controle. Fonte: Silva Neto (2019).	31
Figura 3.8. Diagrama de blocos proposto. Fonte: Autoria Própria (2021).	32
Figura 4.1. Funções de pertinência da variável de entrada dos controladores de velocidade. .	34
Figura 4.2. Funções de pertinência da variável de Saída dos controladores de velocidade. Fonte: Autoria Própria (2021).	35
Figura 4.3. Curva de resposta do atuador 1 (100mm/s). Fonte: Autoria Própria (2021).	36
Figura 4.4. Curva de resposta do atuador 1 (-100mm/s). Fonte: Autoria Própria (2021).	36
Figura 4.5. Curva de resposta do atuador 2 (100mm/s). Fonte: Autoria Própria (2021).	37
Figura 4.6. Curva de resposta do atuador 2 (-100mm/s). Fonte: Autoria Própria (2021).	37
Figura 4.7. Curva de resposta do atuador 3 (100mm/s). Fonte: Autoria Própria (2021).	38
Figura 4.8. Curva de resposta do atuador 3 (-100mm/s). Fonte: Autoria Própria (2021).	38
Figura 4.9. Trajetória sem o controlador. Fonte: Autoria Própria (2021).	40
Figura 4.10. Funções de Pertinência da variável de entrada ErroX. Fonte: Autoria Própria (2021).	40
Figura 4.11. Funções de Pertinência da variável de entrada ErroY. Fonte: Autoria Própria (2021).	41
Figura 4.12. Funções de Pertinência da variável de entrada ErroZ. Fonte: Autoria Própria (2021).	41
Figura 4.13. Funções de Pertinência da variável de saída SaídaX. Fonte: Autoria Própria (2021).	42
Figura 4.14. Funções de Pertinência da variável de saída SaídaY. Fonte: Autoria Própria (2021).	42
Figura 4.15. Funções de Pertinência da variável de saída SaídaZ. Fonte: Autoria Própria (2021).	43
Figura 4.16. Trajetória XYZ com e sem controlador de trajetória. Fonte: Autoria Própria (2021).	44

LISTA DE TABELAS

Tabela 4.1. Parâmetros das funções de pertinência dos controladores fuzzy. Fonte: Autoria Própria (2021).....	35
Tabela 4.2. Regras Fuzzy. Fonte: Autoria Própria (2021).....	35
Tabela 4.3. Índices de desempenho dos controladores. Fonte: Autoria Própria (2021).....	39
Tabela 4.4. Parâmetros das funções do controlador de trajetória. Fonte: Autoria Própria (2021).	43
Tabela 4.5. Regras Fuzzy do controlador de trajetória. Fonte: Autoria Própria (2021).....	43
Tabela 4.6. Comparação de Trajetória Com e Sem controlador. Fonte: Autoria Própria (2021).	44

SUMÁRIO

CAPÍTULO I

INTRODUÇÃO	Erro! Indicador não definido.
1.1 CONSIDERAÇÕES INICIAIS	1
1.2 JUSTIFICATIVA	2
1.3 OBJETIVOS	3
1.4 ORGANIZAÇÃO DO TRABALHO	4

CAPÍTULO II

FUNDAMENTAÇÃO TEÓRICA.....	5
2.1 ROBÔS PARALELOS.....	5
2.1.1 Arquitetura e Cinemática de Mecanismos Paralelos	5
2.1.2 Modelamento Matemático	9
2.2 CONTROLE DE SISTEMAS MECÂNICOS.....	16
2.2.1 Introdução ao Controle de Máquinas.....	16
2.2.2 Introdução ao Controle Fuzzy Mamdani	20

CAPÍTULO III

INSTRUMENTAÇÃO E ESTRATÉGIA DE CONTROLE.....	26
3.1 INSTRUMENTAÇÃO DO PROTÓTIPO ROBÓTICO.....	26
3.1.1 Descrição do Sistema de Retroalimentação.....	27
3.1.2 Descrição do Sistema de Acionamento	29
3.2 DIAGRAMA DE CONTROLE EM MALHA FECHADA	31

CAPÍTULO IV

RESULTADOS EXPERIMENTAIS DO CONTROLE	33
4.1 CONTROLE DE VELOCIDADE.....	33
4.2 CONTROLE DE TRAJETÓRIA	39

CAPÍTULO V

CONCLUSÕES E TRABALHOS FUTUROS.....	45
5.1 CONSIDERAÇÕES FINAIS	45
5.2 TRABALHOS FUTUROS	45

CAPÍTULO I

INTRODUÇÃO

1.1. CONSIDERAÇÕES INICIAIS

Robôs paralelos são mecanismos com cadeias fechadas que apresentam vantagens frente aos robôs seriais quanto à rigidez, precisão/velocidade de posicionamento e razão peso/carga (SENA, SILVA e SILVA, 2017). Nos manipuladores paralelos, todos os atuadores são montados próximos à base, possibilitando assim uma possível redução da massa nas suas partes móveis, implicando que estes manipuladores apresentam características dinâmicas melhores em relação aos seriais (SILVA et. al., 2020). O interesse pelos robôs paralelos ocorreu como uma alternativa para os problemas de pouca capacidade de carga e rigidez que os robôs de configuração serial apresentam (SENA, SILVA e SILVA, 2017). Para controlar esse sistema de forma que o mesmo seja capaz de realizar movimentos específicos, é essencial conhecer quais são suas características cinemáticas e dinâmicas, suas limitações espaciais e finalmente de que forma podem ser controlados os diferentes atuadores para realizar as diferentes trajetórias planejadas. Recentemente algumas pesquisas foram publicadas descrevendo o uso da arquitetura paralela para concepção de robôs, tais como: Sena, Silva e Silva (2017), Almeida Júnior et al. (2018a); Almeida Júnior et al. (2018b); Silva et. al. (2019) e Silva et. al. (2020).

Para realização do controle de trajetória é necessário escolher uma estratégia para geração de pontos ao longo do percurso desejado. Os pontos que compõe uma trajetória podem ser gerados por meio de estratégias do tipo ponto-a-ponto ou contínuas. Na estratégia contínua por trajeto conhecido, o planejamento consiste no estudo e metodologias para definir o comportamento das variáveis do manipulador seguindo um percurso desejado (GONÇALVES, 2016). Sabendo que as principais variáveis do robô são as posições, velocidades e acelerações dos elos e juntas, este comportamento desejado pode ser entendido como uma predefinição dos valores destas variáveis em um intervalo de tempo com alguma parametrização (GONÇALVES, 2016). As parametrizações podem ser as limitações do espaço de trabalho da máquina ou as necessidades do comportamento, como uma velocidade constante ou posição específica. Na grande maioria das aplicações, definir uma trajetória significar definir que a ferramenta do manipulador siga um caminho determinado, tenha uma

velocidade específica ou que ela chegue a um determinado ponto no menor intervalo de tempo possível (GONÇALVES, 2016). O procedimento de geração de uma trajetória pode ser abordado por diversas metodologias matemáticas e analíticas ou computacionais e numéricas, mas todas com o mesmo objetivo de determinar um conjunto de pontos ou função descritiva de um caminho levando em conta as parametrizações (GONÇALVES, 2016). Analiticamente, uma das maneiras de gerar estas trajetórias é calcular os pontos desejados a partir das raízes de um polinômio de ordem n (GONÇALVES, 2016). Estes pontos servirão como uma sequência de entradas para o processo decisório realizado pelo algoritmo controlador.

O algoritmo controlador tipo fuzzy pertence ao grupo dos processos decisórios baseados em inteligência artificial (I.A.) por representar um novo paradigma dentro da engenharia de controle, baseado em transferência do conhecimento sobre um problema. Ao contrário dos controladores clássicos, o controlador fuzzy dispensa o conhecimento do modelo matemático da planta a ser controlada (XAVIER FILHO, 2008). O controlador fuzzy tem a capacidade de considerar vários critérios de desempenho simultaneamente, tanto na forma matemática como na forma linguística. Segundo Campos e Saito (2004), os objetivos destes controladores são: controlar e operar automaticamente processos complexos, não-lineares e multivariáveis, com desempenho pelo menos igual ao dos operadores; respeitar as especificações e restrições operacionais; ser simples, robusto e operar em tempo real. Ainda segundo Campos e Saito (2004), uma das grandes vantagens do controlador fuzzy é que ele pode reduzir consideravelmente o tempo de desenvolvimento de um processo decisório não-linear para um sistema complexo, além das vantagens decorrentes do fato de ser uma tecnologia fácil de ser implementada em tempo real e de ser comprovadamente robusta em aplicações práticas.

1.2. JUSTIFICATIVA

Robôs paralelos são mecanismos com cadeias fechadas que apresenta vantagens frente aos robôs seriais quanto à rigidez, precisão/velocidade de posicionamento e razão peso/carga (SENA, SILVA e SILVA, 2017). Nos manipuladores paralelos, todos os atuadores são montados próximos à base, possibilitando assim uma possível redução da massa nas suas partes móveis, implicando que os manipuladores paralelos apresentam características dinâmicas melhores em relação aos seriais (SENA et. al., 2020). O interesse pelos robôs paralelos ocorreu como uma alternativa para os problemas de pouca capacidade de carga e rigidez que tinham os robôs de configuração serial (SENA, SILVA e SILVA, 2017). Para controlar esse sistema de forma que o mesmo seja capaz de imitar movimentos específicos, é

essencial conhecer quais são suas características cinemáticas e dinâmicas, suas limitações espaciais e finalmente de que forma podem ser controlados os diferentes atuadores para realizar as diferentes trajetórias planejadas. Cadeias cinemáticas fechadas possuem, em geral, rigidez mais elevada que as estruturas com cadeias abertas, e a deformação proveniente da complacência passiva é facilmente medida no efetuador (SENA et. al., 2020). Dentre as diferentes aplicações de sistemas de posicionamento modernos utilizando dispositivos robóticos paralelos é possível destacar: simuladores de voo, máquinas ferramentas, manipuladores cirúrgicos, sistemas de locomoção bípedes, entre outros.

Conhecendo-se a realidade da exploração dos robôs com arquitetura paralela no meio industrial e acadêmico – que embora pouco explorados, estão em crescimento – este trabalho de conclusão de curso visa o controle inteligente trajetória de um determinado tipo de robô paralelo, o robô Delta linear, para que seja descrita e demonstrada à viabilidade industrial destes dispositivos, bem como a importância do estudo dos mesmos no meio acadêmico. Existem várias questões a serem abordadas e discutidas sobre as aplicabilidades do mecanismo delta, o que embate com o pequeno número de trabalhos acadêmicos que se proponham a lidar com esse tipo de dispositivo.

1.3. OBJETIVOS

1.3.1. Objetivos Gerais

Este trabalho propõe o controle de trajetória baseado em lógica fuzzy para um protótipo com estrutura cinemática paralela tipo delta linear com 3 Graus de liberdade. O protótipo deverá ser capaz de realizar o papel de manipulador (XYZ) de alta velocidade, utilizando motores de corrente contínua (CC) e fusos de esferas recirculantes.

1.3.2. Objetivos específicos

- Montar um sistema de controle computacional com módulo multifuncional de dados;
- Desenvolver e simular um modelo cinemático para implementação do controle de trajetória;
- Implementar em software Matlab[®] integrado ao Labview[®] os controladores Fuzzy tipo Mamdani;
- Analisar o desempenho dos controladores em relação à velocidade, precisão e repetibilidade utilizando análise estatística.

1.4. ORGANIZAÇÃO DO TRABALHO

O presente trabalho está organizado em cinco capítulos mais as referências.

No **Capítulo 1**, é feita uma introdução e contextualização do trabalho indicando os principais objetivos.

No **Capítulo 2**, além da revisão bibliográfica como referência aos trabalhos científicos que dão sustentação teórica para o desenvolvimento dos controladores, apresenta-se todo o desenvolvimento da análise cinemática do robô.

No **Capítulo 3**, faz-se a revisão de alguns conceitos empregados para a instrumentação do mecanismo, apresenta-se a concepção e a instrumentação para o sistema de realimentação e acionamento, além da descrição do diagrama de controle utilizado.

No **Capítulo 4**, é demonstrado os resultados experimentais do controle de velocidade e trajetória do mecanismo, observando a resposta temporal e os erros de regime permanente.

No **Capítulo 5**, são apresentadas as considerações finais do trabalho.

E por fim, a apresentação das referências bibliográficas que foram utilizadas durante o desenvolvimento do trabalho.

CAPÍTULO II

FUNDAMENTAÇÃO TEÓRICA

2.1. ROBÔS PARALELOS

2.1.1. Arquitetura e Cinemática de Mecanismos Paralelos

De acordo com Tartari Filho (2006) os robôs são constituídos por seis partes: Ligamentos, atuadores, controladores, efetuador, sensores e juntas. Fazendo analogia ao ser humano, define-se:

- **Ligamentos:** Podem ser comparados a braços e antebraços. São elementos rígidos que se interligam através de juntas;
- **Atuadores:** Permite que o robô se movimente, são os seus músculos. Os atuadores mais utilizados são: motores elétricos, pistões e motores hidráulicos, redutores harmônicos, redutores planetários, guias lineares e muitos outros.
- **Controlador:** É parte pensante da máquina, recebe e processa comandos determinados e os transcreve em sinais eletrônicos que são enviados e entendidos pelos atuadores. Possuem hardware e software.
- **Efetuator:** Elemento que executa e define a aplicação do robô. O efetuador executa o trabalho e em geral cada robô trabalha com apenas um.
- **Sensores:** São equivalentes aos olhos, ouvidos e as células do tato dos humanos. Possibilita a localização e/ou interação do robô para com o ambiente.
- **Juntas:** As juntas são classificadas em dois tipos: as ativas e as passivas. Esta última, apresenta unicamente em robôs paralelos.



Figura 2.1: Elementos de um braço robótico. Fonte: Tartari Filho (2006)

Há diversas maneiras e particularidades para classificar um robô, como por exemplo: de acordo com a forma de atuação, estrutura, tipo de controlador, graus de liberdade e muitas outras formas (TARTARI FILHO, 2006).

Os manipuladores robóticos podem ser classificados em diferentes categorias dependendo de vários critérios, dentre eles, o número de graus de liberdade, estrutura cinemática, geometria do espaço de trabalho e características do movimento (SIMPLÍCIO, LIMA e JUNKES, 2016). Em relação à estrutura cinemática, pode-se classificar um manipulador como de arquitetura serial, paralela ou híbrida. Essa categorização leva em consideração a presença de cadeias cinemáticas abertas ou fechadas (ALMEIDA JUNIOR et al., 2018a).

A cinemática de robôs busca descrever o movimento espacial do manipulador como uma função do tempo, mais especificamente busca as relações analíticas de posição e orientação entre a extremidade da cadeia cinemática do manipulador e o sistema de referência definido (CASTRO, 2019). Mecanismo é definido como uma cadeia cinemática na qual pelo menos um elo foi fixado, ou conectado, à moldura de referência, a qual pode estar em movimento (MOSER, MALUCELLI e BUENO, 2010). A cadeia cinemática será denominada fechada quando suas extremidades estiverem conectadas, caso contrário, será denominada aberta (OLIVEIRA NETO, 2013). Na Fig. (2.2) é esquematizada a classificação da cadeia cinemática em aberta e fechada.

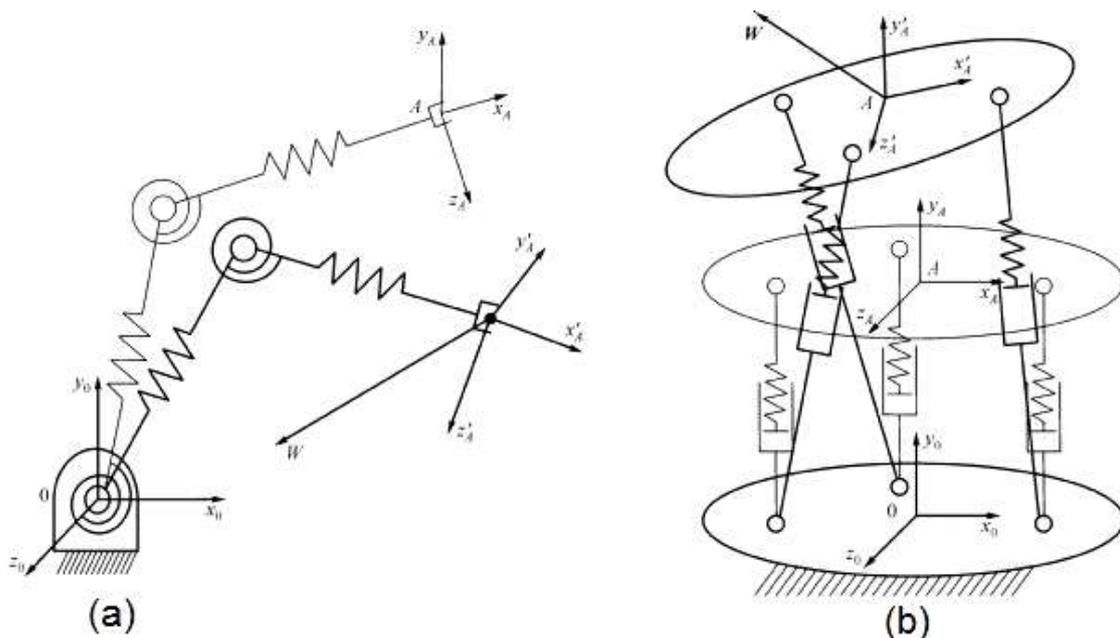


Figura 2.2. Cadeia cinemática: a) Aberta (robô serial) e b) fechada (robô paralelo). Adaptado de Carbone (2011).

Em termos mecânicos, os manipuladores seriais são constituídos por atuadores nas suas partes móveis, implicando em massas e momentos de inércia relativamente altos (ABREU et al., 2020). Na sua maioria um atuador é fixo à base, sendo que os demais são embarcados, ocasionando no aumento de inércia das partes móveis. Além disso, cada junta suporta a massa das demais e dos segmentos subsequentes (ABREU et al., 2020). Nessas condições a rigidez do sistema é comprometida, acarretando em uma menor precisão e capacidade de posicionamento (SILVA et al., 2020). Por isso, para garantir um mínimo de rigidez ao sistema, os elementos devem ser volumosos e pesados, comprometendo o comportamento dinâmico da estrutura serial (ABREU et al., 2020). Um conceito importante é a complacência de um robô. Em geral o efetuador de um robô serial quando submetido a forças e torques externos sofre pequenas modificações em sua posição devido a uma folga dos atuadores, vibração de seus componentes ou deformação em seus links, entre outros (ABREU et al., 2020). Essas grandezas não podem ser quantificadas por sensores internos do robô e por isso não podem ser corrigidas pela malha de controle, daí o nome complacência passiva (SILVA, et al., 2019). Em diversas aplicações na indústria como, por exemplo, em máquinas ferramentas, essa complacência é uma característica indesejada. Erros nas juntas até o órgão terminal, somados à flexibilidade da estrutura, comprometem à precisão do conjunto (SILVA, et al., 2019).

Robôs paralelos são mecanismos constituídos de um efetuador com n graus de liberdade, equipados com duas ou mais cadeias cinemáticas fechadas que unem a base fixa com o efetuador final (GASCA, 2018). Os robôs paralelos são frequentemente usados para aplicações industriais, tais como operações de pick and place, embalagem, soldagem e operações CNC (GASCA, 2018), necessárias em indústrias de componentes eletrônicos, alimentícias, farmacêuticas, cosméticos, dentre outros.

As arquiteturas paralelas despertam grande interesse por apresentar uma boa rigidez associada à exatidão do posicionamento, possuindo capacidade de carga maior que as tradicionais arquiteturas seriais (ALMEIDA JUNIOR et al., 2018a). Estes dispositivos podem trabalhar com altas velocidades com menores níveis de problemas inerciais, se comparados aos mecanismos seriais. Quando comparadas às estruturas seriais, as paralelas possuem espaço de trabalho menor em relação ao volume total da estrutura, entretanto, ela pode ser instalada sobre o posto de trabalho, não ocupando espaço no chão de fábrica (ALMEIDA JUNIOR et al., 2018a). Recentemente algumas pesquisas foram publicadas descrevendo o uso da arquitetura paralela para concepção de robôs específicos, tais como: Sena, Silva e Silva

(2017), Almeida Júnior et al. (2018a), Almeida Júnior et al. (2018b), Gasca (2018) e Silva et al. (2020).

Ao longo dos anos, diferentes robôs paralelos surgiram e muitas dessas arquiteturas paralelas são genuinamente inovadoras; no entanto, o mais famoso foi o robô Delta proposto por Clavel (1988), que tem sido objeto de consideráveis estudos para muitas aplicações desde seu desenvolvimento inicial (GASCA, 2018).

Devido à crescente necessidade de aumentar a produção, as indústrias farmacêuticas e de embalagens em geral adotaram os robôs Delta em suas linhas devido a sua característica de elevada eficiência nas tarefas de “pegar-e-posicionar” (pick-and-place) (ALMEIDA JÚNIOR et al., 2018b). A plataforma móvel do delta tem três graus de liberdade, movendo-se sempre paralelamente à base, devido aos três elos se conectarem com a base fixa (ALMEIDA JÚNIOR et al., 2018b). A configuração Delta Linear (também conhecido como Linapod) é uma evolução do projeto original do robô Delta, onde os braços que atuavam em movimento rotacional são substituídos por atuadores lineares (GASCA, 2018), aumentando o espaço de trabalho em uma direção (GONÇALVES, 2009).

A Fig.(2.3) apresenta uma estrutura delta linear e seu respectivo esquemático estrutural.

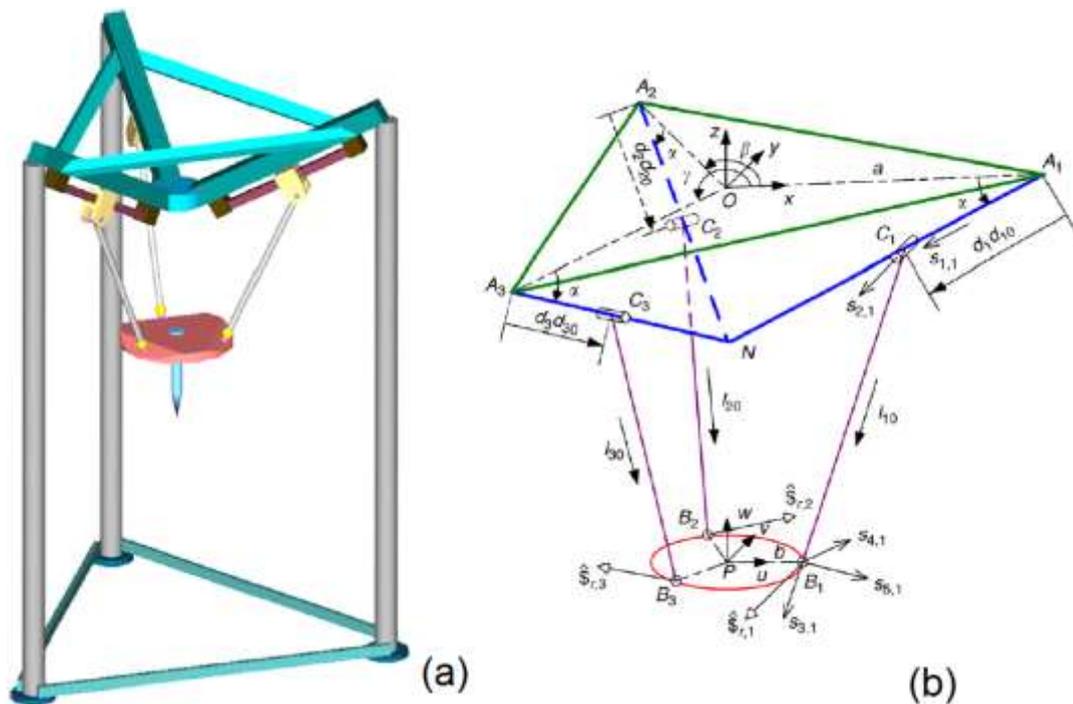


Figura 2.3 Estrutura típica de robô delta linear: a) desenho em CAD e b) esquemático estrutural do conjunto. Adaptado de Li e Xu (2007).

A cinemática de mecanismos paralelos é fundamentalmente diferente da cinemática dos mecanismos seriados. Nos mecanismos seriados cada ligamento carrega todas as forças

dos ligamentos que se seguem e é possível utilizar a sobreposição de efeitos para cálculos de movimentos (KNAPP e COBET, 2000; SILVA NETO, 2019). Cada arquitetura paralela possui características cinemáticas particulares dificultando a aplicação de equações generalistas. A obtenção das equações cinemáticas dos mecanismos paralelos requer a análise de cada cadeia, relacionando os ligamentos com a plataforma móvel e a base fixa (FILHO, 2006; SILVA NETO, 2019).

A cinemática inversa é uma tarefa essencial para o controle de posição, e consiste em calcular as coordenadas das articulações do mecanismo correspondentes a uma configuração específica do efetuador (posição e orientação) (FILHO, 2006; ALMEIDA JÚNIOR et al., 2018c). Para uma arquitetura paralela, a cinemática inversa pode ser realizada pela observação geométrica (FILHO, 2006; ALMEIDA JÚNIOR et al., 2018c). Em contraste com a cinemática inversa, o problema cinemático direto consiste em encontrar a localização do elemento atuador final do robô relativo à base, partindo de conhecer as coordenadas das juntas atuadas. A solução do problema cinemático direto do robô Delta Linear não é tão trivial, sendo que os procedimentos para achar a solução envolvem cálculos matemáticos mais complexos.

Recentemente, alguns pesquisadores realizaram o modelamento cinemático em arquiteturas paralelas, tais como: Almeida Junior et al. (2018b); Araujo-Gómez et al. (2019); Castro (2019); Thomas, Joy e Sudheer (2020).

Há três abordagens tradicionais utilizadas no cálculo da dinâmica dos robôs de arquitetura paralela: Formulação de Newton-Euler, de Lagrange e de D'Alembert (ou principio do trabalho virtual) (FILHO, 2006; SILVA NETO, 2019). Algumas pesquisas são dedicadas à análise dinâmica de estruturas paralelas, tais como: Knapp e Cobet (2000); Filho (2006) e Stan et. al. (2011).

A possibilidade dos robôs paralelos de operarem com altas acelerações e alta exatidão gerou a necessidade do desenvolvimento de sistemas de controle de alto desempenho (FILHO, 2006; SILVA NETO, 2019). Esses controladores exigem a solução das equações dinâmicas da máquina em tempo real (SILVA NETO, 2019). Os controles de posição e trajetória são fundamentais em aplicações industriais para garantir a produção, qualidade dos produtos e segurança dos processos.

2.1.2. Modelamento Matemático

Existem dois problemas fundamentais para resolver a cinemática dos robôs. O primeiro é conhecido como o problema da cinemática direta, que consiste em determinar a

posição e a orientação do efetuador do robô de acordo com um sistema de coordenadas que se toma como referência, como consequência, é possível determinar os valores das articulações e dos parâmetros geométricos dos elementos do robô. O segundo, denominado cinemática inversa, resolve a configuração que deve se adotar para uma posição e orientação do efetuador conhecidas (BARRIENTOS et al, 1997). Na Fig.(2.4) estão representadas as cinemáticas direta e inversa.

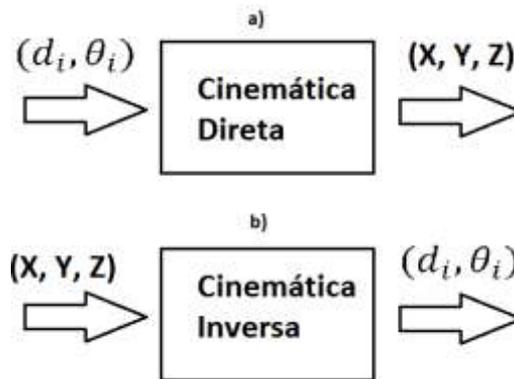


Figura 2.4. a) Representação da Cinemática Direta; b) Representação da Cinemática Inversa. Fonte: Autoria própria (2021).

Além de encontrar a relação de posicionamento dos robôs, a análise cinemática também trata de encontrar as relações entre as velocidades e acelerações do movimento das articulações e do efetuador.

Dado que um robô pode se considerar como uma cadeia cinemática formada por objetos rígidos (os ligamentos) unidos entre si mediante articulações, é possível estabelecer um sistema de referência situado em um ponto fixo do robô e descrever a localização de cada um dos ligamentos com respeito à referência. Desta forma, os problemas de cinemática direta e inversa se reduzem a encontrar um sistema de equações que relacionam a posição e a orientação do efetuador do robô com respeito ao sistema de referência fixo (BARRIENTOS et al, 1997).

Como os robôs formam cadeias cinemáticas, o sistema de equações pode ser encontrado por meio de uma soma vetorial das cadeias. Vários métodos são encontrados na literatura para solução do problema, tais como em: Li e Xu (2005) e Almeida Junior et al. (2018b).

As técnicas de análise cinemática de manipuladores de robôs seguem diversas metodologias distintas. Neste trabalho a metodologia seguida será de acordo com a metodologia de análise cinemática presente em Norton (2013).

A análise cinemática tem como objetivo definir um conjunto de equações que relaciona o deslocamento dos atuadores lineares (C_1 , C_2 e C_3) e o deslocamento do efetuador (X , Y e Z). A Fig.(2.5) apresenta uma representação vetorial do robô.

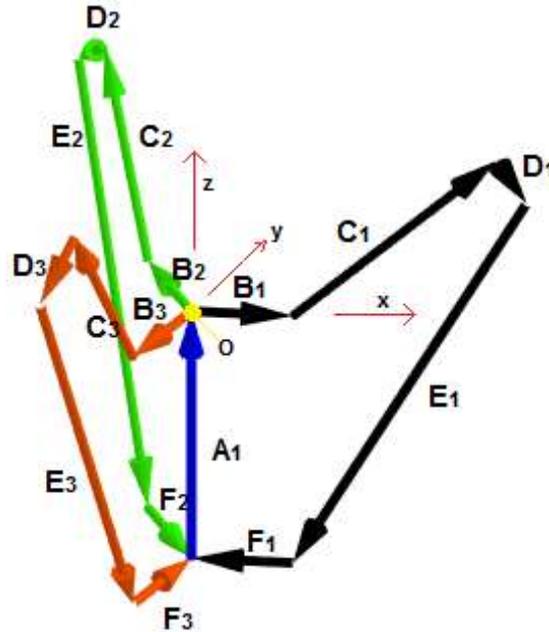


Figura 2.5. Representação Vetorial do Robô Paralelo. Fonte: Autoria própria (2021).

A cadeia cinemática respectiva a cada atuador do robô está representada pelos subíndices 1, 2 e 3, onde cada atuador está defasado em 120° em relação aos outros. Assim, é possível observar que o robô é composto por três cadeias cinemáticas que compartilham o mesmo vetor posição (A_1). Além disso, é possível observar que a plataforma móvel é formada pelos vetores F_1 , F_2 e F_3 , e a fixa por B_1 , B_2 e B_3 . Os vetores C_1 , C_2 e C_3 representam o deslocamento dos atuadores, os vetores E_1 , E_2 e E_3 representam as hastes de acionamento e os vetores D_1 , D_2 e D_3 representam uma distância fixa da conexão da haste com o elemento de transmissão de movimento do atuador. O vetor A_1 representa o posicionamento do efetuador, e o ponto O é a origem do referencial fixo do sistema.

Ao analisar a cadeia cinemática, pode-se observar que:

$$\vec{A}_1 + \vec{B}_1 + \vec{C}_1 + \vec{D}_1 + \vec{E}_1 + \vec{F}_1 = 0 \quad (2.1)$$

$$\vec{A}_2 + \vec{B}_2 + \vec{C}_2 + \vec{D}_2 + \vec{E}_2 + \vec{F}_2 = 0 \quad (2.2)$$

$$\vec{A}_3 + \vec{B}_3 + \vec{C}_3 + \vec{D}_3 + \vec{E}_3 + \vec{F}_3 = 0 \quad (2.3)$$

Onde \vec{A}_1 , \vec{A}_2 e \vec{A}_3 são iguais e representam o vetor posição, logo:

$$\vec{A}_1 = \vec{A}_2 = \vec{A}_3 = -X \vec{i} - Y \vec{j} - Z \vec{k} \quad (2.4)$$

A base fixa é representada pelos vetores B e pode ser observada na Fig.(2.6).

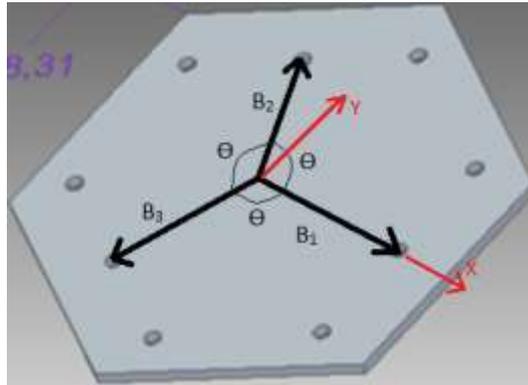


Figura 2.6. Vetores base fixa. Fonte: Autoria própria (2021).

O módulo dos vetores B é representado pela distância OE, e o θ é a defasagem angular dos atuadores. Assim:

$$\vec{B}_1 = OE \vec{i} \quad (2.5)$$

$$\vec{B}_2 = OE * \cos(\theta) \vec{i} + OE * \sin(\theta) \vec{j} \quad (2.6)$$

$$\vec{B}_3 = OE * \cos(2\theta) \vec{i} + OE * \sin(2\theta) \vec{j} \quad (2.7)$$

Os vetores C representam o deslocamento dos atuadores. A direção deste vetor é fixa e está numa angulação γ com o plano XY. A Fig.(2.7) apresenta a constituição mecânica do robô, os componentes importantes para o modelamento e a representação vetorial da cadeia cinemática pertencente ao atuador 1.

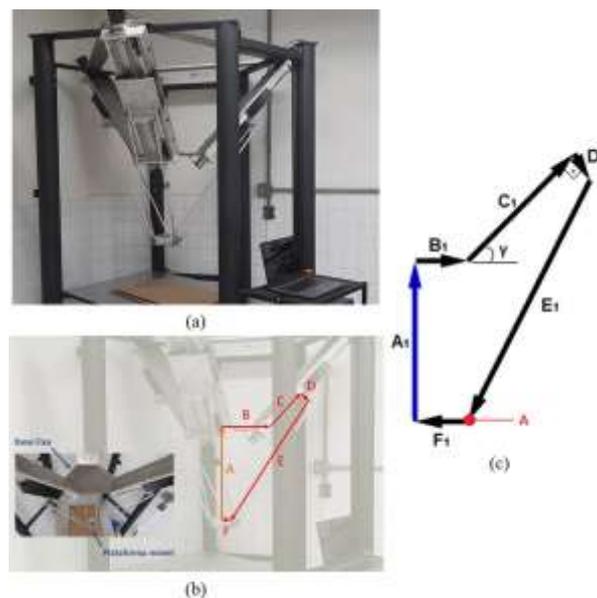


Figura 2.7. a) Robô paralelo, b) componentes utilizados na modelagem, c) representação vetorial da cadeia 1. Fonte: Autoria própria (2021).

$$\vec{C}_1 = C_1 * \cos(\gamma) \vec{i} + C_1 * \sin(\gamma) \vec{k} \quad (2.8)$$

$$\vec{C}_2 = C_2 * \cos(\gamma) * \cos(\theta) \vec{i} + C_2 * \cos(\gamma) * \sin(\theta) \vec{j} + C_2 * \sin(\gamma) \vec{k} \quad (2.9)$$

$$\vec{C}_3 = C_3 * \cos(\gamma) * \cos(2\theta) \vec{i} + C_3 * \cos(\gamma) * \sin(2\theta) \vec{j} + C_3 * \sin(\gamma) \vec{k} \quad (2.10)$$

Onde C_1 , C_2 e C_3 representam as posições de cada atuador.

Os vetores D_i das Eq.(2.11), (2.12) e (2.13) representam a conexão entre os elementos de transmissão do acionamento e as hastes, estando fixado a 90° em relação à transmissão, possuindo um comprimento fixo D como podem ser observados na Fig.(2.8).



Figura 2.8. Representação do Vetor D na máquina. Fonte: Autoria própria (2021).

$$\vec{D}_1 = D * \sin(\gamma) \vec{i} - D * \cos(\gamma) \vec{k} \quad (2.11)$$

$$\vec{D}_2 = D * \sin(\gamma) * \cos(\theta) \vec{i} + D * \sin(\gamma) * \sin(\theta) \vec{j} - D * \cos(\gamma) \vec{k} \quad (2.12)$$

$$\vec{D}_3 = D * \sin(\gamma) * \cos(2\theta) \vec{i} + D * \sin(\gamma) * \sin(2\theta) \vec{j} - D * \cos(\gamma) \vec{k} \quad (2.13)$$

Os vetores E_i representam as hastes do robô, e terão movimentos em todas as direções do espaço. O comprimento das hastes é representado por L , assim como as componentes angulares. A representação de um vetor no espaço pode ser observada na Fig.(2.9).

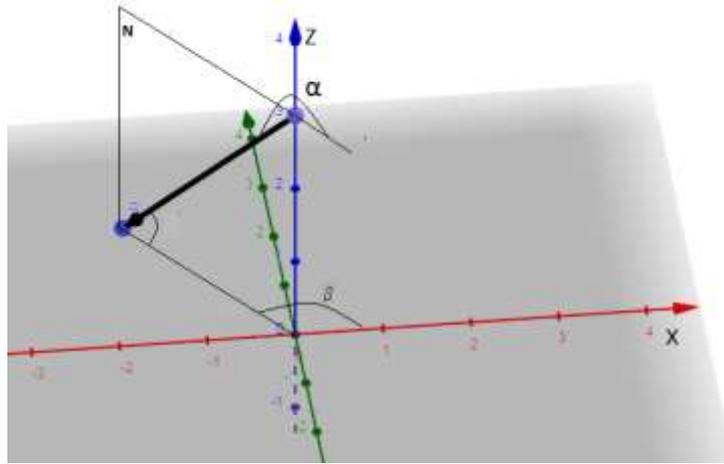


Figura 2.9. Representação vetor E. Fonte: Autoria própria (2021).

$$\vec{E}_1 = L * \cos(\alpha_1) * \cos(\beta_1) \vec{i} + L * \cos(\alpha_1) * \sin(\beta_1) \vec{j} + L * \sin(\alpha_1) \vec{k} \quad (2.14)$$

$$\vec{E}_2 = L * \cos(\alpha_2) * \cos(\beta_2) \vec{i} + L * \cos(\alpha_2) * \sin(\beta_2) \vec{j} + L * \sin(\alpha_2) \vec{k} \quad (2.15)$$

$$\vec{E}_3 = L * \cos(\alpha_3) * \cos(\beta_3) \vec{i} + L * \cos(\alpha_3) * \sin(\beta_3) \vec{j} + L * \sin(\alpha_3) \vec{k} \quad (2.16)$$

Os vetores F_i representam a plataforma móvel (peça conhecida como trizeta), e podem ser observados na Fig.(2.10) onde OF representa o comprimento fixo do vetor.



Figura 2.10. Vetor da plataforma móvel. Fonte: Autoria própria (2021).

$$\vec{F}_1 = -OF \vec{i} \quad (2.17)$$

$$\vec{F}_2 = -OF * \cos(\theta) \vec{i} - OF * \sin(\theta) \vec{j} \quad (2.18)$$

$$\vec{F}_3 = -OF * \cos(2\theta) \vec{i} - OF * \sin(2\theta) \vec{j} \quad (2.19)$$

Substituindo as equações de (2.4) a (2.19), em (2.1), (2.2) e (2.3), temos:

Em \vec{i} :

$$-X + OE + C_1 * \cos(\gamma) + D * \sin(\gamma) + L * \cos(\alpha_1) * \cos(\beta_1) - OF = 0 \quad (2.20)$$

$$-X + OE * \cos(\theta) + C_2 * \cos(\gamma) * \cos(\theta) + D * \sin(\gamma) * \cos(\theta) + L * \cos(\alpha_2) * \cos(\beta_2) - OF * \cos(\theta) = 0 \quad (2.21)$$

$$-X + OE * \cos(2\theta) + C_3 * \cos(\gamma) * \cos(2\theta) + D * \sin(\gamma) * \cos(2\theta) + L * \cos(\alpha_3) * \cos(\beta_3) - OF * \cos(2\theta) = 0 \quad (2.22)$$

Em \vec{j} :

$$-Y + L * \cos(\alpha_1) * \sin(\beta_1) = 0 \quad (2.23)$$

$$-Y + OE * \sin(\theta) + C_2 * \cos(\gamma) * \sin(\theta) + D * \sin(\gamma) * \sin(\theta) + L * \cos(\alpha_2) * \sin(\beta_2) - OF * \sin(\theta) = 0 \quad (2.24)$$

$$-Y + OE * \sin(2\theta) + C_3 * \cos(\gamma) * \sin(2\theta) + D * \sin(\gamma) * \sin(2\theta) + L * \cos(\alpha_3) * \sin(\beta_3) - OF * \sin(2\theta) = 0 \quad (2.25)$$

Em \vec{k} :

$$-Z + C_1 * \sin(\gamma) - D * \cos(\gamma) + L * \sin(\alpha_1) = 0 \quad (2.26)$$

$$-Z + C_2 * \sin(\gamma) - D * \cos(\gamma) + L * \sin(\alpha_2) = 0 \quad (2.27)$$

$$-Z + C_3 * \sin(\gamma) - D * \cos(\gamma) + L * \sin(\alpha_3) = 0 \quad (2.28)$$

A solução do sistema formado pelas equações de (2.20) à (2.28), irá retornar todas as posições dos elementos do robô, assim ao derivar essas equações em relação ao tempo, teremos um sistema que terá como solução as velocidades de cada elemento, assim derivando em relação ao tempo ($\partial/\partial t$).

$$\left\{ \begin{array}{l} -\dot{X} + \dot{C}_1 * \cos(\gamma) + L * (-\sin(\alpha_1) * \cos(\beta_1) * \dot{\alpha}_1 - \cos(\alpha_1) * \sin(\beta_1) * \dot{\beta}_1) = 0 \\ -\dot{Y} + L * (-\sin(\alpha_1) * \sin(\beta_1) * \dot{\alpha}_1 + \cos(\alpha_1) * \cos(\beta_1) * \dot{\beta}_1) = 0 \\ -\dot{Z} + \dot{C}_1 * \sin(\gamma) + L * \cos(\alpha_1) * \dot{\alpha}_1 = 0 \\ -\dot{X} + \dot{C}_2 * \cos(\gamma) * \cos(\theta) + L * (-\sin(\alpha_2) * \cos(\beta_2) * \dot{\alpha}_2 - \cos(\alpha_2) * \sin(\beta_2) * \dot{\beta}_2) = 0 \\ -\dot{Y} + \dot{C}_2 * \cos(\gamma) * \sin(\theta) + L * (-\sin(\alpha_2) * \sin(\beta_2) * \dot{\alpha}_2 + \cos(\alpha_2) * \cos(\beta_2) * \dot{\beta}_2) = 0 \\ -\dot{Z} + \dot{C}_2 * \sin(\gamma) + L * \cos(\alpha_2) * \dot{\alpha}_2 = 0 \\ -\dot{X} + \dot{C}_3 * \cos(\gamma) * \cos(2\theta) + L * (-\sin(\alpha_3) * \cos(\beta_3) * \dot{\alpha}_3 - \cos(\alpha_3) * \sin(\beta_3) * \dot{\beta}_3) = 0 \\ -\dot{Y} + \dot{C}_3 * \cos(\gamma) * \sin(2\theta) + L * (-\sin(\alpha_3) * \sin(\beta_3) * \dot{\alpha}_3 + \cos(\alpha_3) * \cos(\beta_3) * \dot{\beta}_3) = 0 \\ -\dot{Z} + \dot{C}_3 * \sin(\gamma) + L * \cos(\alpha_3) * \dot{\alpha}_3 = 0 \end{array} \right. \quad (2.29)$$

Uma vez que a análise de velocidade tenha sido feita, o próximo passo é determinar a aceleração dos pontos de interesse do robô. Sabe-se que a aceleração é a derivada de 2ª ordem da posição, e neste sentido, ao aplicar a 2ª derivada em relação ao tempo no sistema que determina as posições, tem-se:

$$\left\{ \begin{array}{l} -\ddot{X} + \ddot{C}_1 \cdot \cos(\gamma) + L \cdot \left(\begin{array}{l} -\cos(\alpha_1) \cdot \cos(\beta_1) \cdot \ddot{\alpha}_1^2 + \sin(\alpha_1) \cdot \sin(\beta_1) \cdot \ddot{\alpha}_1 \cdot \dot{\beta}_1 - \sin(\alpha_1) \cdot \cos(\beta_1) \cdot \ddot{\alpha}_1 + \\ \sin(\alpha_1) \cdot \sin(\beta_1) \cdot \ddot{\alpha}_1 \cdot \dot{\beta}_1 - \cos(\alpha_1) \cdot \cos(\beta_1) \cdot \dot{\beta}_1^2 - \cos(\alpha_1) \cdot \sin(\beta_1) \cdot \dot{\beta}_1 \end{array} \right) = 0 \\ -\ddot{Y} + L \cdot \left(\begin{array}{l} -\cos(\alpha_1) \cdot \sin(\beta_1) \cdot \ddot{\alpha}_1^2 - \sin(\alpha_1) \cdot \cos(\beta_1) \cdot \ddot{\alpha}_1 \cdot \dot{\beta}_1 - \sin(\alpha_1) \cdot \sin(\beta_1) \cdot \ddot{\alpha}_1 - \\ \sin(\alpha_1) \cdot \cos(\beta_1) \cdot \ddot{\alpha}_1 \cdot \dot{\beta}_1 - \cos(\alpha_1) \cdot \sin(\beta_1) \cdot \dot{\beta}_1^2 + \cos(\alpha_1) \cdot \cos(\beta_1) \cdot \dot{\beta}_1 \end{array} \right) = 0 \\ -\ddot{Z} + \ddot{C}_1 \cdot \sin(\gamma) + L \cdot (-\sin(\alpha_1) \cdot \ddot{\alpha}_1^2 + \cos(\alpha_1) \cdot \ddot{\alpha}_1) = 0 \\ -\ddot{X} + \ddot{C}_2 \cdot \cos(\gamma) \cdot \cos(\theta) + L \cdot \left(\begin{array}{l} -\cos(\alpha_2) \cdot \cos(\beta_2) \cdot \ddot{\alpha}_2^2 + \sin(\alpha_2) \cdot \sin(\beta_2) \cdot \ddot{\alpha}_2 \cdot \dot{\beta}_2 - \sin(\alpha_2) \cdot \cos(\beta_2) \cdot \ddot{\alpha}_2 + \\ \sin(\alpha_2) \cdot \sin(\beta_2) \cdot \ddot{\alpha}_2 \cdot \dot{\beta}_2 - \cos(\alpha_2) \cdot \cos(\beta_2) \cdot \dot{\beta}_2^2 - \cos(\alpha_2) \cdot \sin(\beta_2) \cdot \dot{\beta}_2 \end{array} \right) = 0 \\ -\ddot{Y} + \ddot{C}_2 \cdot \cos(\gamma) \cdot \sin(\theta) + L \cdot \left(\begin{array}{l} -\cos(\alpha_2) \cdot \sin(\beta_2) \cdot \ddot{\alpha}_2^2 - \sin(\alpha_2) \cdot \cos(\beta_2) \cdot \ddot{\alpha}_2 \cdot \dot{\beta}_2 - \sin(\alpha_2) \cdot \sin(\beta_2) \cdot \ddot{\alpha}_2 - \\ \sin(\alpha_2) \cdot \cos(\beta_2) \cdot \ddot{\alpha}_2 \cdot \dot{\beta}_2 - \cos(\alpha_2) \cdot \sin(\beta_2) \cdot \dot{\beta}_2^2 + \cos(\alpha_2) \cdot \cos(\beta_2) \cdot \dot{\beta}_2 \end{array} \right) = 0 \\ -\ddot{Z} + \ddot{C}_2 \cdot \sin(\gamma) + L \cdot (-\sin(\alpha_2) \cdot \ddot{\alpha}_2^2 + \cos(\alpha_2) \cdot \ddot{\alpha}_2) = 0 \\ -\ddot{X} + \ddot{C}_3 \cdot \cos(\gamma) \cdot \cos(2\theta) + L \cdot \left(\begin{array}{l} -\cos(\alpha_3) \cdot \cos(\beta_3) \cdot \ddot{\alpha}_3^2 + \sin(\alpha_3) \cdot \sin(\beta_3) \cdot \ddot{\alpha}_3 \cdot \dot{\beta}_3 - \sin(\alpha_3) \cdot \cos(\beta_3) \cdot \ddot{\alpha}_3 + \\ \sin(\alpha_3) \cdot \sin(\beta_3) \cdot \ddot{\alpha}_3 \cdot \dot{\beta}_3 - \cos(\alpha_3) \cdot \cos(\beta_3) \cdot \dot{\beta}_3^2 - \cos(\alpha_3) \cdot \sin(\beta_3) \cdot \dot{\beta}_3 \end{array} \right) = 0 \\ -\ddot{Y} + \ddot{C}_3 \cdot \cos(\gamma) \cdot \sin(\theta) + L \cdot \left(\begin{array}{l} -\cos(\alpha_3) \cdot \sin(\beta_3) \cdot \ddot{\alpha}_3^2 - \sin(\alpha_3) \cdot \cos(\beta_3) \cdot \ddot{\alpha}_3 \cdot \dot{\beta}_3 - \sin(\alpha_3) \cdot \sin(\beta_3) \cdot \ddot{\alpha}_3 - \\ \sin(\alpha_3) \cdot \cos(\beta_3) \cdot \ddot{\alpha}_3 \cdot \dot{\beta}_3 - \cos(\alpha_3) \cdot \sin(\beta_3) \cdot \dot{\beta}_3^2 + \cos(\alpha_3) \cdot \cos(\beta_3) \cdot \dot{\beta}_3 \end{array} \right) = 0 \\ -\ddot{Z} + \ddot{C}_3 \cdot \sin(\gamma) + L \cdot (-\sin(\alpha_3) \cdot \ddot{\alpha}_3^2 + \cos(\alpha_3) \cdot \ddot{\alpha}_3) = 0 \end{array} \right. \quad (2.30)$$

Nos sistemas acima os valores de L, OE, D, γ , θ e OF são conhecidos. Dessa forma, existem três sistemas com nove equações e doze incógnitas. No entanto, aplicando os conceitos da cinemática direta e inversa, os três sistemas ficam com nove equações e nove incógnitas, tendo assim uma solução possível. Para cinemática direta, os valores de C_1 , C_2 , C_3 , \dot{C}_1 , \dot{C}_2 , \dot{C}_3 , \ddot{C}_1 , \ddot{C}_2 e \ddot{C}_3 são conhecidos, e para cinemática inversa os valores conhecidos são X , Y , Z , \dot{X} , \dot{Y} , \dot{Z} , \ddot{X} , \ddot{Y} e \ddot{Z} . Para solucionar os sistemas é necessária a utilização de métodos numéricos para sistemas não lineares.

O método mais amplamente estudado e conhecido para resolver sistemas de equações não lineares é o método de Newton. Por meio deste método, dada uma aproximação inicial x_0 da solução, calcula-se a aproximação utilizando a Eq.(2.31).

$$x_{k+1} = x_k - J(x_k)^{-1} * F(x_k) \quad (2.31)$$

Aplicando os conceitos da Eq.(2.31) na Eq.(2.29) e Eq.(2.30), é possível solucionar os sistemas, e obter os valores das velocidades e acelerações dos atuadores.

2.2. CONTROLE DE SISTEMAS MECÂNICOS

2.2.1. Introdução ao Controle de Máquinas

O objetivo básico de um sistema de controle consiste em aplicar sinais adequados na entrada do processo de modo a que o sinal de saída satisfaça certas especificações e/ou apresente um comportamento particular (ABREU et al., 2020). O sistema a ser controlado

geralmente chamado de processo ou planta é o dispositivo ou fenômeno que se deseja atuar com alguma finalidade, sendo este representado por uma variável de entrada, uma variável de saída e uma relação (função de transferência) entre as duas variáveis (ABREU et al., 2020). A entrada do processo é chamada de variável de controle ou variável manipulada, e a saída do processo é chamada de variável controlada ou variável de processo (ABREU et al., 2020). Um distúrbio ou perturbação é caracterizado por um sinal que tende a afetar de modo adverso o valor da variável de saída de um sistema (ABREU et al., 2020).

Neste contexto, um Sistema de Controle em Malha Aberta (SCMA) consiste em aplicar um sinal de controle pré-determinado, esperando-se que ao final de um determinado tempo a variável controlada atinja um determinado valor ou apresente um determinado comportamento, sendo que neste tipo de sistema não são utilizadas informações sobre evolução do processo para determinar o sinal de controle a ser aplicado em um determinado instante (ABREU et al., 2020). Por outro lado, no controle em malha fechada, informações sobre como a saída de controle está evoluindo a partir de uma realimentação, são utilizadas para determinar o sinal de controle que deve ser aplicado ao processo em um instante específico (ABREU et al., 2020). O sinal de saída é comparado com um sinal de referência e o desvio (erro) entre estes dois sinais é utilizado para determinar o sinal de controle que deve efetivamente ser aplicado ao processo (ABREU et al., 2020). Assim, o sinal de controle é determinado de forma a corrigir este desvio entre a saída e o sinal de referência. O dispositivo que utiliza o sinal de erro para determinar ou calcular o sinal de controle a ser aplicado à planta é chamado de controlador ou compensador (DORF, 2018).

Uma vantagem dos sistemas de controle a malha fechada é o fato de que o uso da retroação torna a resposta do sistema relativamente insensível a perturbações externas e a variações internas de parâmetros do sistema, sendo possível à utilização de componentes baratos e sem muita exatidão para obter o controle preciso de um determinado processo, o que não é possível com o controle à malha aberta (ABREU et al., 2020). Entretanto, do ponto de vista da estabilidade, é mais fácil construir sistemas a malha aberta porque a estabilidade destes sistemas é menos problemática (ABREU et al., 2020). Por outro lado, a estabilidade em sistemas de controle a malha fechada é sempre um grande problema pela tendência em corrigir erros além do necessário, o que pode ocasionar oscilações de amplitude constante ou crescente com o tempo (ABREU et al., 2020). Deve-se enfatizar que, para sistemas onde as entradas são conhecidas antecipadamente no tempo e não há distúrbios é aconselhável o uso de controle a malha aberta (ABREU et al., 2020).

Inserido na malha de controle fechada, o controlador determina um sinal de erro comparando o valor de saída da planta com uma referência de entrada. A partir desse erro, determina um sinal de controle para reduzir o erro à zero ou a um valor aceitável, do ponto de vista da utilização do sistema controlado (ABREU et al., 2020).

As malhas convencionais de controle têm sido utilizadas na maioria dos processos automatizados no meio industrial (ABREU et al., 2020). Entre os motivos pode-se mencionar: estrutura simples, robustez, reduzido número de parâmetros configurados, conhecimento intuitivo sobre o desempenho destas técnicas de controle, e requerimento mínimo de conhecimento matemático e técnico (ABREU et al., 2020). Entretanto, na teoria de controle convencional, os objetivos de projeto dos controladores são fixos e definidos pelo projetista (ABREU et al., 2020). Consequentemente, a abordagem convencional de controle é limitada para muitas situações complexas, devido ao fato que estas requerem algumas formas de autonomia e as estratégias convencionais de controle oferecem um número reduzido de graus de liberdade (ABREU et al., 2020). Estas complexidades caracterizam processos com modelos matemáticos "pobres", elevados níveis de ruído, rigorosos índices de desempenho, presença de não-linearidades, sensores distribuídos e a necessidade da tomada de decisões (ABREU et al., 2020).

Para obtenção de modelo matemático da planta, com o objetivo de se projetar o controlador, inicialmente é realizado o processo de identificação da planta, podendo ser usados métodos como caixa preta, branca e cinza (ABREU et al., 2020). Desse modo, o projeto do controlador é realizado a partir das especificações desejadas para um bom desempenho do sistema (ABREU et al., 2020). Apesar do sistema de retroalimentação manter estabilidade para futuras variações e incertezas, algumas variações são tão amplas e com efeitos significativos sobre o comportamento dinâmico que um ganho de realimentação linear e com coeficientes constantes é incapaz de fornecer a flexibilidade necessária para atender às especificações de desempenho (ABREU et al., 2020). Sendo assim, passa a ser necessário medir continuamente estas variações e então ajustar devidamente os parâmetros de controle (“ganhos não-lineares”). A Fig. (2.11) apresenta o diagrama de blocos típico de uma estrutura de controle clássico.

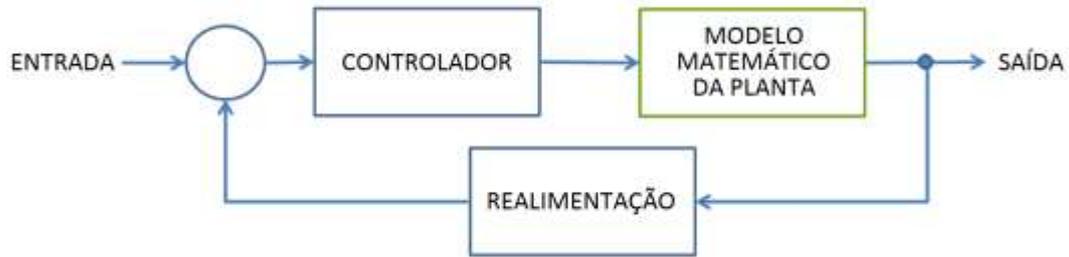


Figura 2.11. Filosofia de controle clássico. Fonte: Autoria própria (2021).

Com o objetivo de aperfeiçoar o desempenho dos controladores convencionais, diversas técnicas de controle moderno vêm sendo desenvolvidas nos últimos anos. O desenvolvimento da teoria do controle ótimo, robusto e adaptativo, tem expandido de forma significativa o conceito de automação e ampliado à factibilidade de controle automático na prática (CRUZ, 2018). No entanto, as técnicas envolvidas geralmente dependem da existência de um modelo matemático para a planta e para o controlador, além de requerer formas estruturadas de representação e lógica clássica (CRUZ, 2018). Técnicas de controle inteligente, por sua vez, buscam gerar ações de controle efetivas sem recorrer a tais hipóteses (CRUZ, 2018). Neste contexto, o controle fuzzy tem se mostrado uma alternativa viável ao controle clássico com parâmetros variantes no tempo, não-lineares e com informações imprecisas (PEREIRA JÚNIOR, 2018).

A lógica fuzzy é uma técnica que tenta emular sistematicamente e matematicamente o raciocínio humano e as tomadas de decisão. A lógica fuzzy permite que os conhecimentos empíricos e heurísticos, sejam representados por regras e então transferidos para um bloco de funções. Assim, a lógica fuzzy fornece uma maneira clara e intuitiva de implementar sistemas de controle, tomada de decisão e sistemas de diagnóstico em vários ramos da indústria (PEREIRA JÚNIOR, 2018). Uma apresentação detalhada pode ser encontrada em Simões e Shaw (2007) e Yen e Langari (1999).

A literatura classifica os sistemas fuzzy considerando as características do método de tomada de decisão que eles empregam (SILVA, et al., 2019). Nos modelos linguísticos as equações diferenciais, que normalmente caracterizam os sistemas dinâmicos, são substituídas por um conjunto de regras cuja construção abrange predicados e, muitas vezes, conhecimentos heurísticos (SILVA, et al., 2019). O método de inferência determina a forma operacional do modelo linguístico (SILVA, et al., 2019). Um dos métodos mais divulgados, pela sua simplicidade e por se adaptar muito bem aos controladores fuzzy, é o método de Mamdani, que tem como base a regra de composição de inferência máx-min (SILVA, et al., 2019). Este método considera uma função de pertinência na saída com certo perfil, de modo

que uma saída dita defuzzyficada pode ser obtida por meio das propriedades de área das saídas agregadas de todas as regras (FUJIMOTO, 2005; SILVA, et al., 2019).

De acordo com Simões e Shaw (2007) pode-se construir um sistema fuzzy Mamdani utilizando os seguintes blocos básicos: fuzzyficação, base de regras, tomada de decisões e defuzzyficação. Na Fig.(2.12) é apresentada a estrutura de um sistema de inferência Fuzzy.

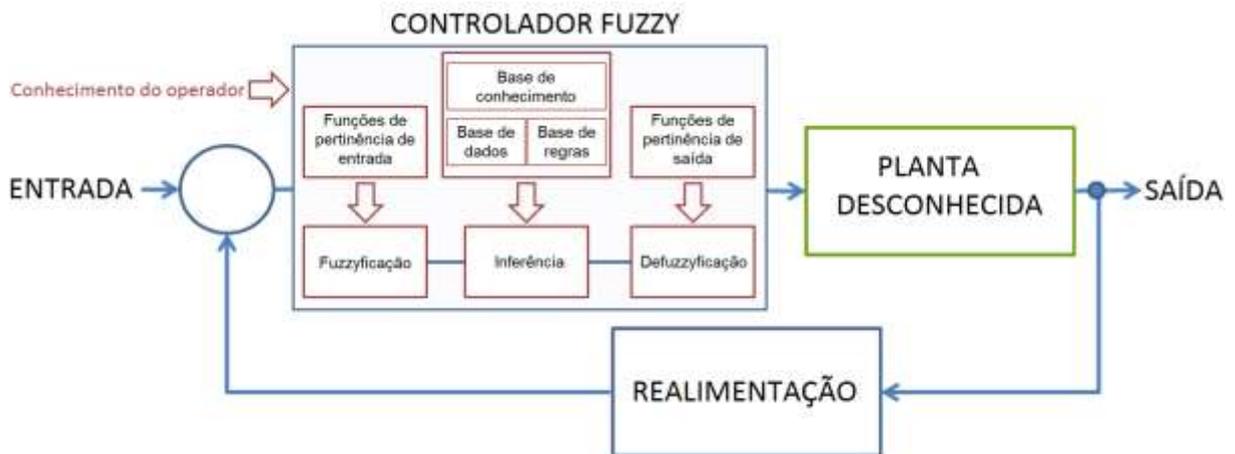


Figura 2.12. Filosofia de controle fuzzy. Fonte: Silva, et al. (2019).

2.2.2. Introdução ao Controle Fuzzy Mamdani

Uma variável linguística x no universo de discurso X é definida em um conjunto de termos, nomes ou rótulos, $A = \{x, \mu_A(x) \mid x \in X\}$, com cada valor sendo um número fuzzy definido em X (SIMÕES e SHAW, 2007; SENA, 2015). Um universo de discurso é um conjunto de valores finitos que a variável linguística pode assumir. A função de pertinência, $\mu_A(x)$, associa a um dado componente do universo de discurso, o seu respectivo grau de pertinência a um determinado conjunto fuzzy.

Em particular, $\mu_A(x) = 0$ e $\mu_A(x) = 1$ indicam, respectivamente, a não pertinência e a pertinência completa de x ao conjunto fuzzy A . A Fig.(2.13) apresenta um exemplo de variáveis linguísticas dentro de uma função de pertinência do tipo triangular, considerando x o nível de líquido de um determinado tanque que apresenta o conjunto de termos $\{\text{muito baixo, baixo, médio, alto, muito alto}\}$, em um universo de discurso $X = [20, 200]$.

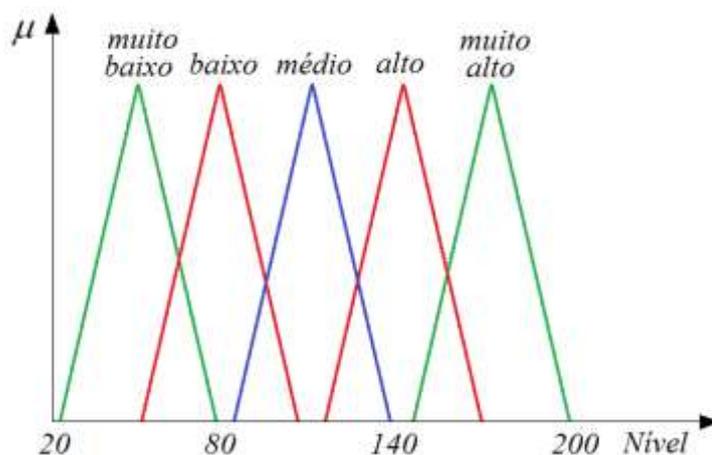


Figura 2.13. Funções de pertinência tipo triangular para variável nível com as devidas variáveis linguísticas. Adaptado de Simões e Shaw (2007).

A função de pertinência é uma função numérica gráfica ou tabulada em que se atribuem valores de pertinência fuzzy para valores discretos de uma variável, em um universo de discurso. Essas funções representam os aspectos fundamentais de todas as ações teóricas e práticas de um sistema fuzzy (SIMÕES e SHAW, 2007; SENA, 2015). Para aplicações de controle e modelagem utilizando a filosofia fuzzy, as mais conhecidas funções que são aplicadas são as de pertinência quadrática, linear por partes (triangular ou trapezoidal), gaussiana e sino. As funções lineares por partes são as mais populares devido à simplicidade dessas funções e ao fato de que o custo computacional adicional exigido pelos outros tipos de função não refletem, em geral, em uma melhoria significativa na qualidade dos valores de saída dos sistemas (YEN e LANGARI, 1999; SENA, 2015). O valor limiar α define a sensibilidade do sistema fuzzy, de modo que um resultado após as operações lógicas abaixo deste valor é considerado como sendo nulo. O valor limiar é escolhido baseado em conhecimentos empíricos (SENA, 2015).

A lógica fuzzy, de maneira análoga à lógica clássica, pode ser descrita em termos de orações que relacionam causa e consequência, normalmente expressas na forma (*if*<antecedente>*then*<consequente>) denominadas regras de inferências. O antecedente é composto por um conjunto de condições que, quando satisfeitas (mesmo parcialmente), determinam o processamento do consequente da regra por um mecanismo de inferência fuzzy, processo este denominado disparo de uma regra. As regras de controle difusas são declarativas, e não sequenciais, de forma que a ordem em que se expressam não é importante. De modo análogo à lógica clássica, na lógica fuzzy existem os operadores de conjunto união, intersecção e complemento; ou na nomenclatura booleana, *ou* (*or*), *e* (*and*) e *negação* (*not*), respectivamente. O análogo ao operador *and* na lógica fuzzy é conhecido como *T-norm*, e o

análogo ao operador *or* como *S-norm* ou *T-conorm*. Existem diversas funções que são utilizadas para estes operadores, tais como: Zadeh, Mean, Yager, Zysno, e Dubois. As mais utilizadas são às funções de *máximo* e de *mínimo* (Zadeh) para os operadores *or* e *and*, respectivamente (FUJIMOTO, 2005; SENA, 2015), onde dados os conjuntos fuzzy *A* e *B* de *X*, a união e a intersecção são conjuntos fuzzy com funções de pertinência definidas pelas seguintes equações, respectivamente:

$$\mu_{A \cup B}(x) = \max\{\mu_A(x), \mu_B(x)\}, \quad \forall x \in X \quad (2.32)$$

$$\mu_{A \cap B}(x) = \min\{\mu_A(x), \mu_B(x)\}, \quad \forall x \in X \quad (2.33)$$

Semelhantemente, para operação complemento existem diversas variações tais como: Zadeh, Yager e threshold, sendo a mais utilizada a definição de Zadek (FUJIMOTO, 2005) da Eq.(2.34).

$$\mu_{\bar{A}}(x) = 1 - \mu_A(x), \quad \forall x \in X \quad (2.34)$$

Na Fig.(2.14) estão representadas graficamente as operações de união, intersecção e complemento.

Existem diversos tipos de inferência para relacionar as entradas e saídas do conjunto de regras. Eles diferem pelo tipo de operador utilizado e pelos tipos de antecedentes e consequentes. As regras podem ser constituídas por uma única entrada e uma única saída (SISO, do inglês *Single-Input/Single-Output*) ou múltiplas saídas (SIMO, do inglês *Single-Input/Multiple-Output*), bem como, por múltiplas entradas e uma única saída (MISO, do inglês *Multiple-Input/Single-Output*) ou múltiplas saídas (MIMO, do inglês *Multiple-Input/Multiple-Output*) (SENA, 2015). Uma regra de inferência tipo MISO pode ser representada pela sintaxe **Se** *x* **é** *A* **e/ou** *y* **não é** *B* **e/ou**... ..*w* **é** *F* **então** *u* **é** *G*.

O operador *then* (*então*) define a implicação de cada regra sobre uma dada variável linguística da saída, podendo ser definido pela função mínimo da Eq.(2.35).

$$\mu_{CFR}[x_j] \leftarrow \min(\mu_{PT}, \mu_{CFS}[x_j]), \quad (2.35)$$

Onde μ_{CFR} é a função de pertinência de saída da regra de inferência após aplicação do operador, μ_{PT} é o grau de veracidade da oração antecedente da regra de inferência e μ_{CFS} é a função de pertinência da oração consequente (FUJIMOTO, 2005; SENA, 2015).

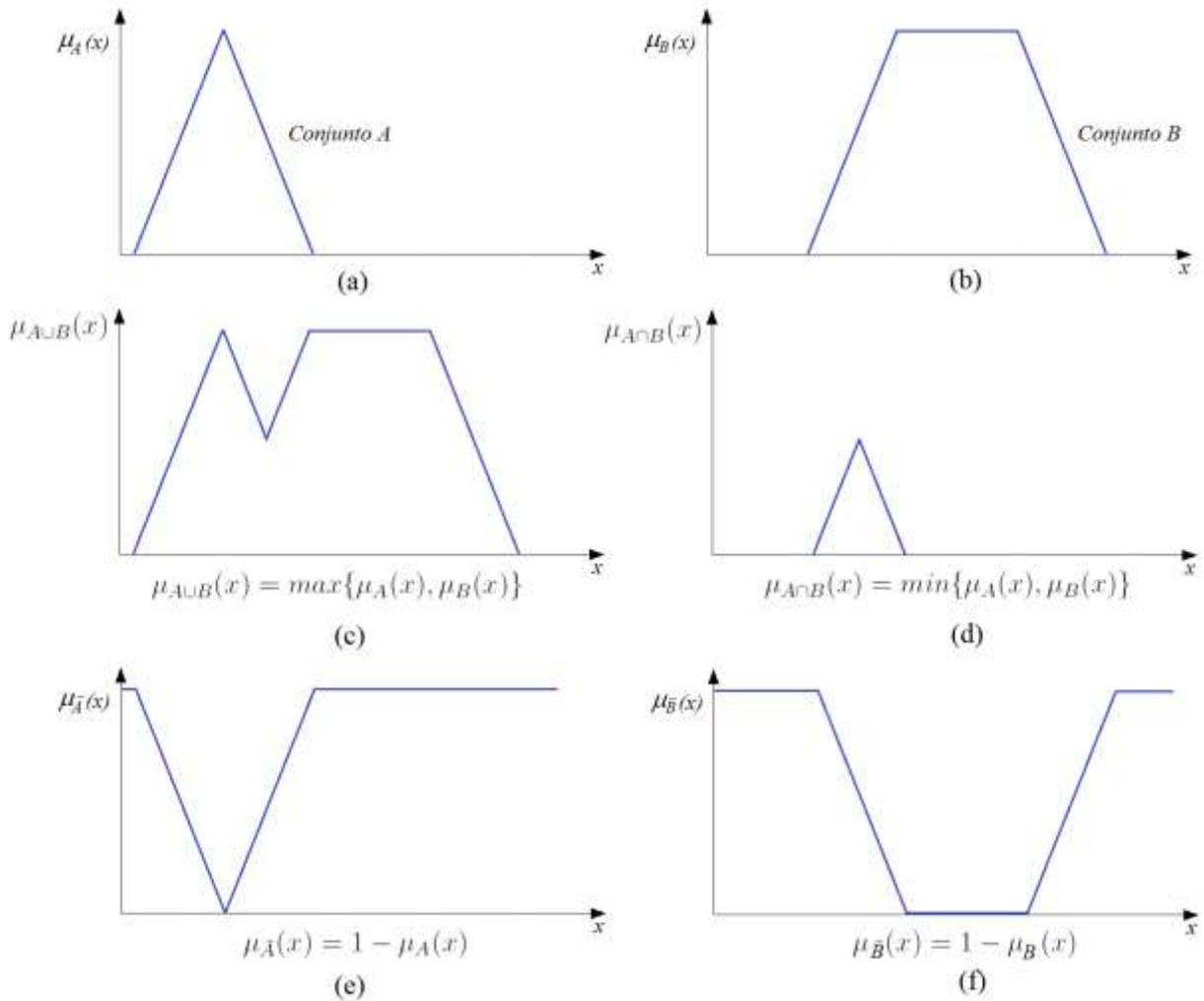


Figura 2.14. Representações gráficas: (a) conjunto A de função triangular, (b) conjunto B de função trapezoidal, (c) operação de união entre A e B, (d) operação de intersecção entre A e B, (e) operação de complemento de A e (f) operação de complemento de B. Fonte: Sena (2015).

O operador de agregação de regras une a influência de todas as regras obtidas após aplicação do operador *then* sobre o resultado final, apresentando a consequência do sistema como um todo devido às entradas do sistema Fuzzy (SENA, 2015). Este operador pode ser definido pela função máximo da Eq.(2.36).

$$\mu_{CFE} [x_i] \leftarrow \max_{j=1}^n (\mu_{CFR_j} [x_i]) \quad (2.36)$$

De acordo com Osiro (2013) os métodos de defuzzyficação podem ser classificados em duas aplicações práticas distintas: melhor compromisso (centro de área e centro do máximo) e resultado mais plausível (média do máximo).

O método de defuzzyficação da Média do Máximo (MoM) calcula a média de todos os valores de saída que tenham os maiores graus de possibilidade. A Eq.(2.37) determina a média do máximo, sendo M a quantidade de pontos com o mesmo valor máximo de $\mu_C(z)$ e z_k

são os valores correspondentes de z onde $\mu_C(z)$ tem valor máximo (YAGER e FILEV, 1993; SENA, 2015).

$$MoM = \frac{1}{M} \sum_{\kappa=1}^M z_{\kappa} \quad (2.37)$$

A principal limitação do método MoM é que ele não considera a forma total do conjunto fuzzy de saída. Sendo assim, duas distribuições de possibilidades que apresentem diferentes formas, porém o mesmo conjunto de valores com grau de pertinência máximo, quando defuzzyficados fornecerá o mesmo valor clássico, o que é contra intuitivo (ORTEGA, 2001; SENA, 2015).

Na Fig.(2.15) apresenta a MoM para a variável de saída C . A linha MoM divide ao meio a região onde $\mu_C(z)$ tem valor máximo.

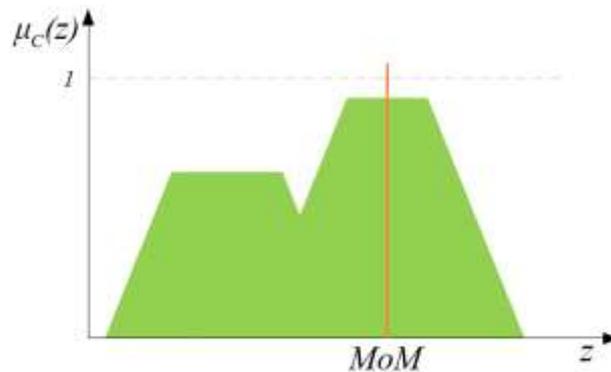


Figura 2.15. Defuzzyficação pela média do máximo (MoM). Adaptado de Osiro (2013).

No método de centro de área (CoA) também denominado método do centro de gravidade (CoG) ou ainda centróide, a partir da área na variável de saída, obtida pela agregação dos consequentes de todas as regras. Diferentemente do MoM, a técnica do Centro de Área para calcular o valor clássico representativo considera toda a distribuição de possibilidade de saída do modelo (SENA, 2015). O procedimento é similar ao usado para calcular o centro de gravidade em física, se consideramos a função de pertinência $\mu_C(z)$ como a densidade de massa de z . Por outro lado, o método do Centro de Área pode ser compreendido como uma média ponderada, onde $\mu_C(z)$ funciona como o peso do valor z . Se z é discreto, então a defuzzyficação da conclusão fuzzy C é dada por:

$$CoA = \frac{\sum_{\kappa=1}^N \mu(z_{\kappa}) \cdot z_{\kappa}}{\sum_{\kappa=1}^N \mu(z_{\kappa})} \quad (2.38)$$

A variável N é a quantidade de funções de pertinência da variável de saída.

Esse tipo de defuzzyficação além de necessitar de um considerável esforço computacional, apresenta limitações quando aplicada em um processo no qual as funções de pertinência não possuem sobreposição ou quando mais de uma regra possuir a mesma saída fuzzy, o que caracteriza uma sobreposição de área (ORTEGA, 2001; SENA, 2015). O método denominado de centro do máximo (CoM) apresenta um cálculo mais simples ao ignorar as áreas das funções de pertinência ao considerar apenas seus respectivos picos (OSIRO, 2013; SENA, 2015). Na Fig.(2.16) a centroide ou centro de gravidade da região em verde está na linha CoA.

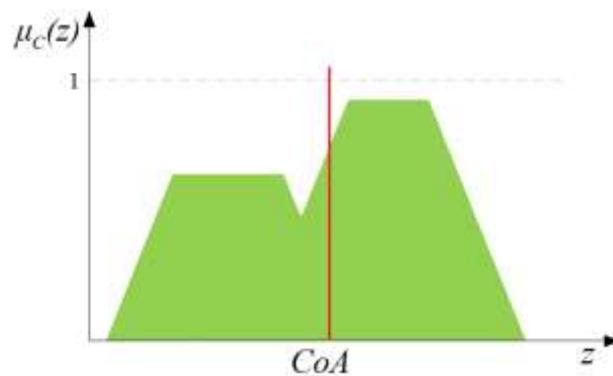


Figura 2.16. Defuzzyficação pelo centro de área (CoA). Adaptado de Osiro (2013).

CAPÍTULO III

INSTRUMENTAÇÃO E ESTRATÉGIA DE CONTROLE

3.1. INSTRUMENTAÇÃO DO PROTÓTIPO ROBÓTICO

Os três atuadores lineares do robô Delta são uma associação de componentes mecânicos tais como fusos, guias, mancais e rolamentos. Os sistemas de posicionamento apresentam um curso útil de 500 mm em guias tipo BR 15, com estrutura confeccionada em alumínio. Os fusos são de esferas padrão SFU1605 com passo de 5 mm e diâmetro de 16 mm, apresentando uma precisão de 0,08mm (SILVA et al., 2019). A Fig.(3.1) apresenta o robô paralelo utilizado.



Figura 3.1. Estrutura do robô delta controlado. Fonte: Silva et. al. (2021).

3.1.1. Descrição do Sistema de Retroalimentação

Sistemas de controle clássico são baseados em um modelo matemático que descrito através de uma ou mais equações diferenciais, definem a resposta do sistema em relação às entradas (SILVA NETO, 2019). Porém, em vários casos, o modelo matemático do processo não existe, não é conhecido ou é muito complexo de ser obtido ou implementado. Nestes casos, admite-se que o sistema é representado por uma função de transferência que é obtida através de um processo chamado de identificação de sistemas (SENA, 2008 apud SILVA NETO, 2019).

Os elementos funcionais de um sistema de aquisição por computador são comumente os sensores e transdutores, os cabos de ligação, o condicionamento de sinal (quando necessário), o equipamento de aquisição, o computador e o software de aquisição (SILVA NETO, 2019). A Fig.(3.2) apresenta o diagrama esquemático da aquisição de sinais de um fenômeno físico para controle por meio de um computador.

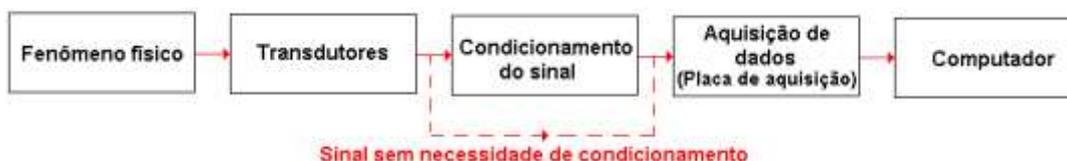


Figura 3.2. Diagrama funcional de um sistema de aquisição por computador. Fonte: Silva Neto (2019).

Nos encoders incrementais, a frequência dos pulsos determina a velocidade do eixo, e a quantidade de pulsos, a partir de um referencial, determina o deslocamento angular, fornecendo a distância linear percorrida (SILVA NETO, 2019). Os encoders que geram o código Gray possuem dois acopladores ópticos distanciados angularmente (canais A e B defasados 90°), cuja codificação precisará ser transformada para código binário. A resolução destes encoders é dependente do número de divisões (furos) arranjadas angularmente (SILVA NETO, 2019). No entanto, o aumento da resolução será limitado ao tempo de aquisição de dados do sistema. A Fig.(3.3) apresenta um encoder incremental com geração de código Gray.

O processo de conversão de um código Gray para binário é realizado repetindo o bit mais significativo, e o valor do bit menos significativo é o resultado da operação lógica “ou exclusivo”, entre os dois bits Gray (SILVA NETO, 2019). O processo de conversão de um número binário em decimal é realizado multiplicando-se cada bit binário pela base do sistema (base dois), elevado à posição que ocupa da direita para esquerda, iniciando-se do índice zero. A contagem decimal primária se limita a representação dos quatro estados possíveis nos encoders. Para varrer todo o conjunto de valores decimais inteiros, geralmente é implementado um algoritmo de incremento ou decremento unitário (SILVA NETO, 2019). Os

encoders incrementais adquiridos (5322-2222-0100) da Hohner, permitem uma resolução de 100 Pulsos/volta, gerando na combinação dos dois canais, um código Gray de 400 estados/volta. O processo de conversão de código Gray para o deslocamento angular é realizado por meio da parametrização das entradas contadoras do módulo funcional, que permitem um resultado direto (biblioteca pronta). Com base no passo dos fusos (5mm) é possível relacionar cada estado gerado nos encoders, ao deslocamento dos sistemas lineares (calculado em 0,0125 mm/estado). Por fim, a retroalimentação foi finalizada com o desenvolvimento de um programa que converte o deslocamento angular dos encoders em movimento linear atuadores (C1, C2 e C3).

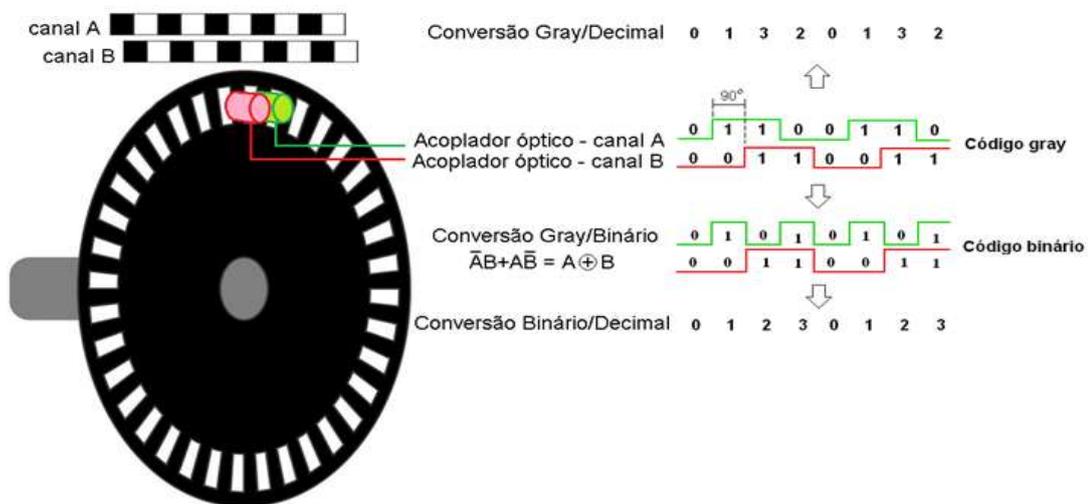


Figura 3.3. Encoder incremental com código Gray. Fonte: Silva Neto (2019).

Para programação foi utilizada a versão Labview Academic Premium Suite, que possibilita gerar códigos executáveis (exe) a partir do código fonte. Após a conversão do código fonte em executável, não é necessário o uso do software de programação.

Para assegurar que o mecanismo conheça sua posição inicial e não ultrapasse os limites de atuação das juntas, foram utilizados sensores de fim de curso que estabelecem a posição inicial ao mesmo tempo em que interrompem o sinal enviado aos motores, sendo necessário um redirecionamento deste sinal em outra porta do microcontrolador para fazer o robô retornar em segurança (SILVA NETO, 2019).

O módulo multifuncional de dados é o hardware responsável pelas entradas e saídas de sinais nos processos de medição e controle de sistemas por computador. O módulo utilizado na montagem foi disponibilizado pelo laboratório de instrumentação do IFPE Caruaru; trata-se do NI DAQ USB-6353 da National Instruments, que é disposto de 48 entradas digitais de 5 VCC TTL/CMOS com 1 MHz (porta P₀ com 32 bits, portas P₁ e P₂ com 8 bits cada) (SILVA

NETO, 2019). A Fig.(3.4) apresenta o hardware de aquisição montado para todo o conjunto robótico.

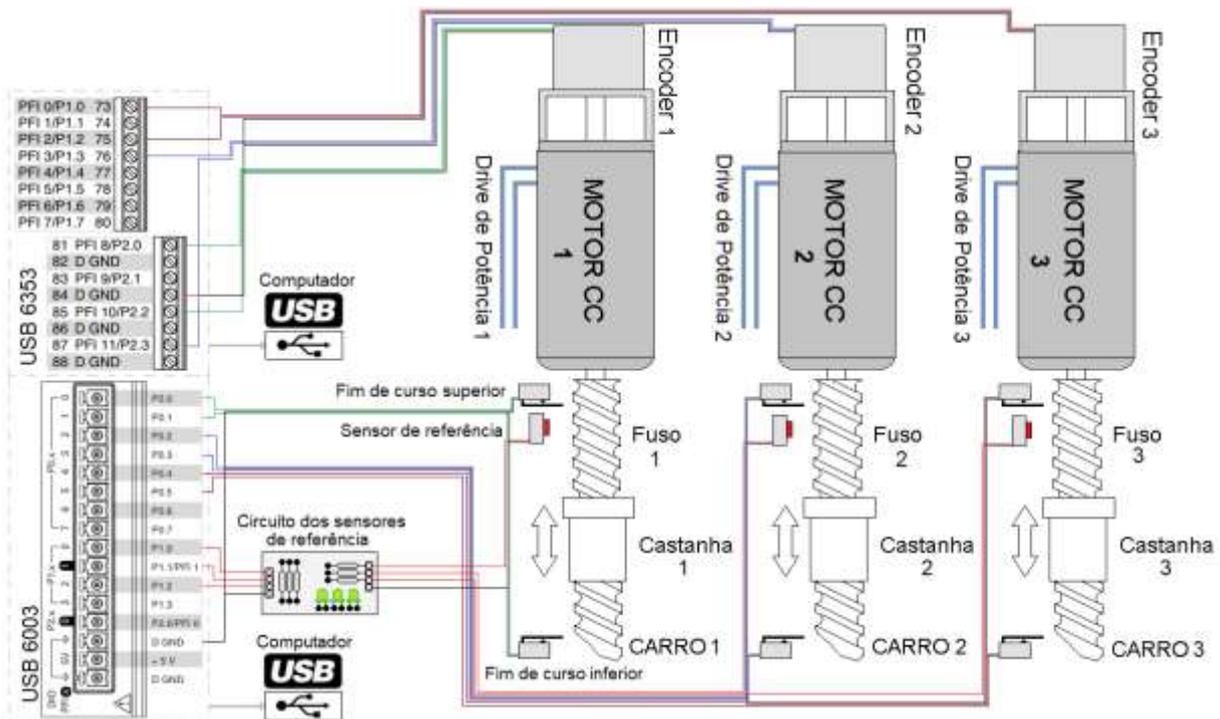


Figura 3.4. Detalhamento do hardware de aquisição montado. Fonte: Silva Neto (2019).

3.1.2. Descrição do Sistema de Acionamento

O sistema de acionamento permite o comando e ajuste da rotação dos motores de Corrente Contínua (motores de CC) por meio de circuitos drivers (circuitos de potência). Os circuitos drivers recebem sinais analógicos dos controladores implementados no ambiente Labview (software instalado no computador) (SILVA NETO, 2019). A programação em Labview utiliza as saídas analógicas do módulo multifuncional. A Fig.(3.5) apresenta o diagrama de blocos do sistema de acionamento.

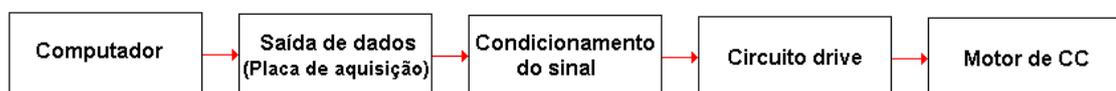


Figura 3.5. Diagrama funcional do sistema de acionamento de acionamento. Fonte: Silva Neto (2019).

O módulo multifuncional possui um conversor Digital Analógico (DAC) que serve para transformar os sinais digitais fornecidos pelo controlador em sinal analógico de controle para os circuitos drivers. O sinal de saída do módulo (sinal de controle) tem valor que varia de -5 a +5 VCC (configurável entre -10 a +10 VCC), com os valores positivos para sentido de giro horário do motor e negativos para determinação de sentido de giro anti-horário. O Labview disponibiliza o bloco DAQ Assistant Express VI, que contém um programa de

configuração (wizard) dos sinais de entrada e saída do módulo multifuncional de dados para realizar quaisquer operações: analog input, analog output, digital I/O (SILVA NETO, 2019). Na configuração das saídas analógicas, definem-se as variações das tensões de saída, escolhendo-se ainda quais canais físicos de saída da placa serão utilizados. O módulo multifuncional utilizado dispõe de quatro saídas analógicas (AO0, AO1, AO2 e AO3) com configuração de tensão em valor máximo de 5 VCC e mínimo de -5 VCC.

De acordo com Silva Neto (2019), para realizar o controle dos motores foi utilizado placas de potência controladas por PWM do tipo EM-176 da ElectromenTM, baseadas em Pontes-H. Estas placas têm as seguintes características: alimentação de 0 a 35VCC, corrente de saída de 10 A (contínuo) e 15 A (pico). A placa de potência dos motores possui as seguintes entradas: DIR, que recebem sinais do computador para determinar o sentido de giro; SPEED (velocidade), que varia a velocidade do motor de acordo com o sinal recebido. Foram utilizados três motores de corrente contínua da Ignis Motors[®] (MR20A-024150-667RPM) de 24VCC e 8A (150 W e 667 RPM) para realizar o acionamento de fusos de esferas recirculantes, que por sua vez converteram o movimento rotacional em linear. Os motores foram selecionados a partir da medição do torque (utilizando um torquímetro mecânico) necessário para girar o fuso de um dos sistemas de posicionamento (atuadores lineares).

O controle de trajetória em malha fechada é associado ao controle de velocidade dos motores, pois enquanto o valor do erro tende a zero, de forma proporcional a velocidade do motor também tenderá a zero por meio da variação do sinal de controle. A Fig.(3.6) apresenta o hardware de acionamento montado para todo o conjunto robótico.

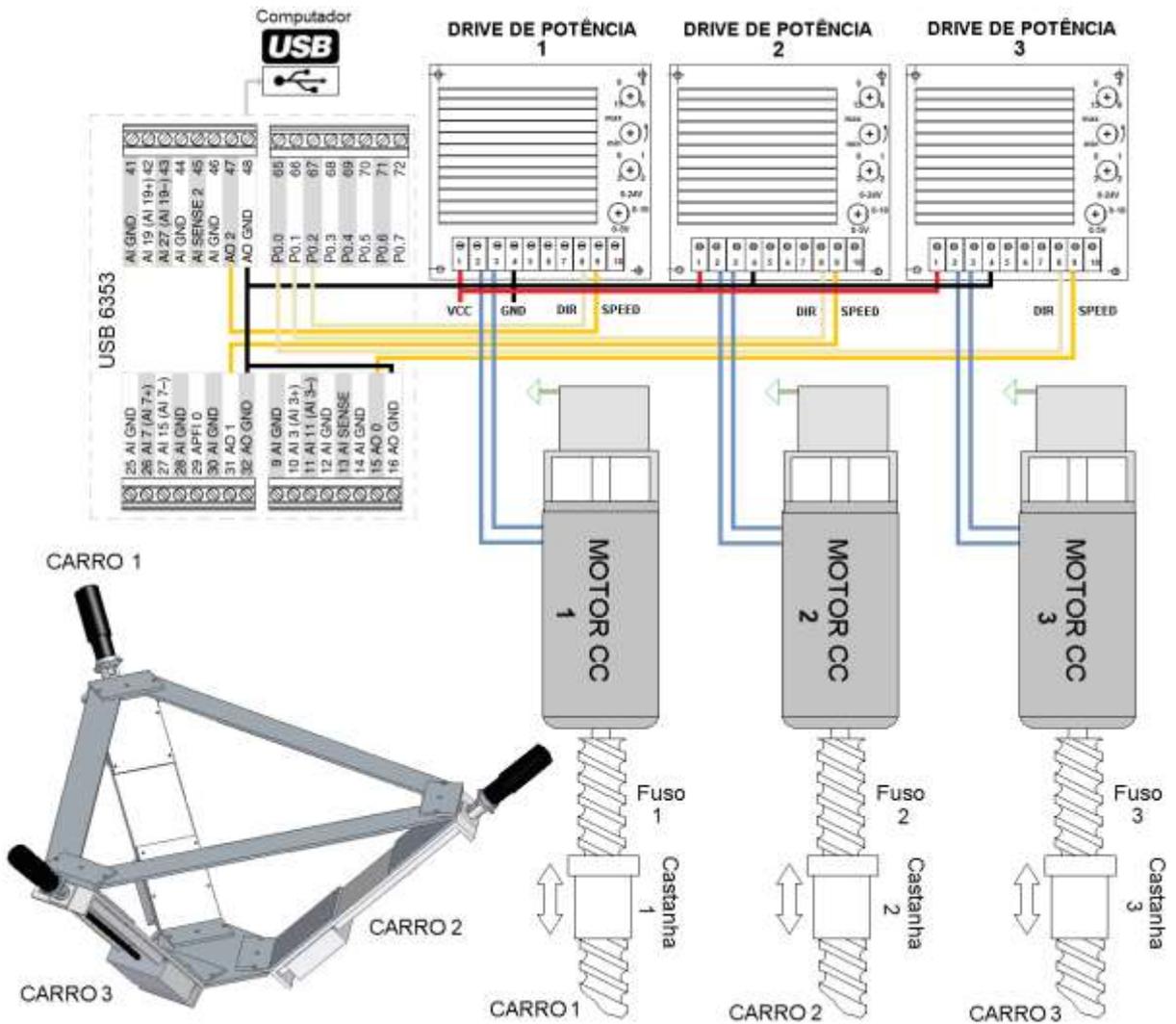


Figura 3.6. Detalhamento do hardware de acionamento montado. Fonte: Silva Neto (2019).

3.2. DIAGRAMA DE CONTROLE EM MALHA FECHADA

Na Fig.(3.7) é apresentado um diagrama funcional, observando os equipamentos utilizados e a comunicação destes.

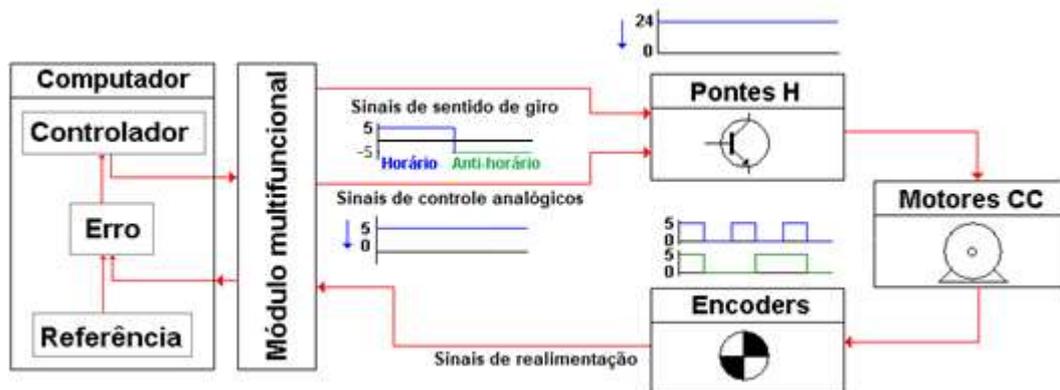


Figura 3.7. Diagrama funcional do sistema de controle. Fonte: Silva Neto (2019).

A estratégia proposta para o sistema de controle de trajetória pode ser observado na Fig.(3.8). Existem três etapas de controle: a etapa de controle de velocidade, a etapa de controle de trajetória e a etapa de geração da trajetória. Na etapa de geração, a velocidade da ponta da ferramenta é definida nas direções X, Y e Z no sentido da trajetória desejada. Na etapa de controle de trajetória, as velocidades são compensadas para manter a ferramenta na trajetória de referência. Na etapa de controle de velocidade, os controladores de velocidade são utilizados para manter os atuadores nas velocidades de referência, de acordo com a saída do modelo de cinemática inversa.

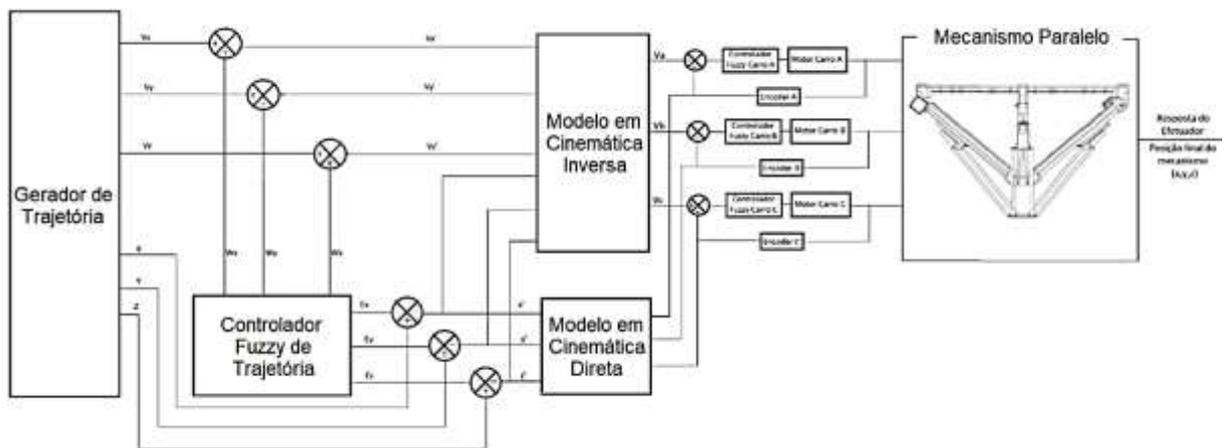


Figura 3.8. Diagrama de blocos proposto. Fonte: Autoria Própria (2021).

Devido à complexidade de descrever este sistema através de equações matemáticas, os controladores de velocidades utilizados são do tipo Fuzzy mamdani. Uma vez que se tratar de um sistema estatístico, não é necessário o conhecimento do modelo matemático. Além disso, outra característica desses controladores é a possibilidade de trabalhar com sistemas MIMO (*Multiplex Input Multiplex Output*).

Uma das grandes dificuldades na implementação de um controlador fuzzy, é o ajuste experimental das funções de pertinência. O método mamdani foi utilizado para a implementação do controlador de velocidade dos motores de CC, e conseqüentemente o controle de trajetória do protótipo final.

CAPÍTULO IV

RESULTADOS EXPERIMENTAIS DO CONTROLE

4.1. CONTROLE DE VELOCIDADE

Os controladores Fuzzy foram implementados pelo método Mamdani através do “VI (Virtual Instrument) Controlador de Lógica Fuzzy” disponível no Fuzzy Logic Toolkit do software Labview[®]. De acordo com os princípios dos sistemas Fuzzy de serem baseados na experiência e nos padrões de raciocínio dos operadores humanos, o modelamento dos controladores foi realizado a partir de um estudo prévio e experimental do comportamento desejado para os regimes transitório e permanente dos atuadores.

Inicialmente, as funções de pertinência e regras fuzzy foram baseadas nas sugestões de Simões e Shaw (2007) e nos trabalhos de Bordon, Andreoti e Menezes (2016) e Silva et al. (2019), mas a posteriori, foram modificadas a partir do conhecimento adquirido do comportamento da máquina nos testes realizados.

No bloco de fuzzificação foi criada uma variável linguística denominada *ERRO*, que representa o desvio entre o sinal de entrada (valor desejado de velocidade) e o sinal de realimentação (velocidade atual do atuador). O universo de discurso da variável *ERRO* abrange o intervalo $[-200, 200]$, representando os valores máximos de erro, em milímetros por segundo, que o sistema pode apresentar. A variável *ERRO* é composta por cinco funções de pertinência com os termos linguísticos denominados: Negativo Grande (NG), Negativo Pequeno (NP), Quase Zero (QZ), Positivo Pequeno (PP) e Positivo Grande (PG). Para as variáveis NG e PG foram utilizadas funções trapezoidais, e para as demais foram utilizadas funções triangulares. A Fig. (4.1) apresenta a variável linguística de entrada dos controladores de velocidade com suas funções de pertinência e termos linguísticos correspondentes. Na Tab.(4.1) são apresentados os parâmetros de cada função de pertinência.

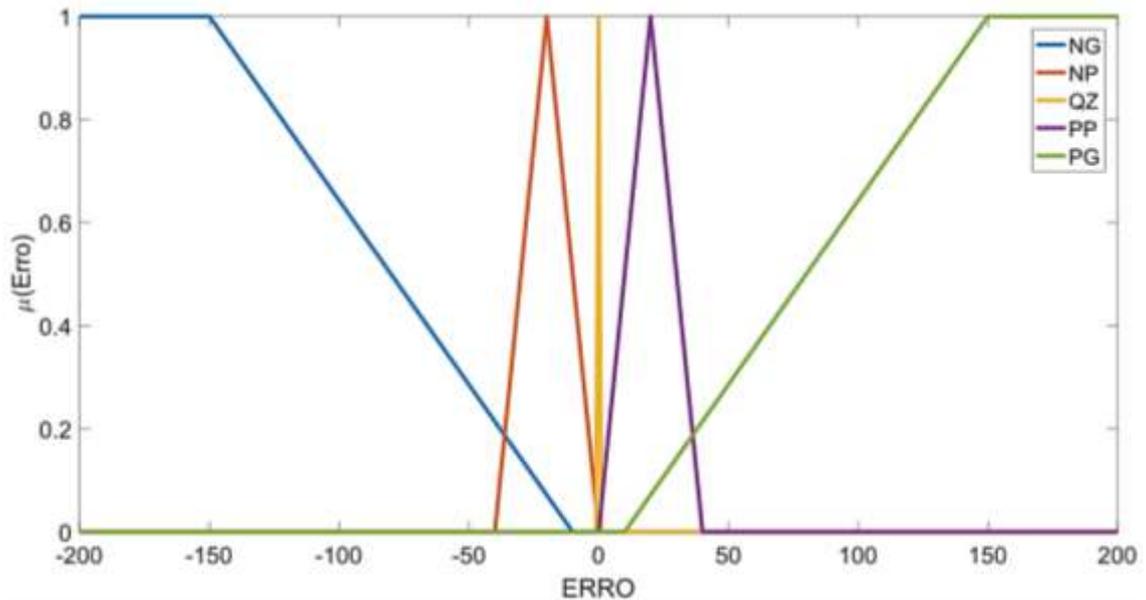


Figura 4.1. Funções de pertinência da variável de entrada dos controladores de velocidade.

Fonte: Autoria Própria (2021).

Na defuzzificação foi criada uma variável linguística denominada *Saída*, que representa o sinal de controle que atuará sobre o sinal de velocidade. O universo de discurso desta variável abrange o intervalo $[-0,3 \ 0,3]$. Para a variável *Saída*, foram associadas cinco funções de pertinência, definidas pelos termos NG, NP, QZ, PP e PG. Para as variáveis NG e PG foram utilizadas funções trapezoidais, e para as demais foram utilizadas funções triangulares. O sinal de controle irá atuar sobre a velocidade de acordo com a eq.(4.1).

$$V_i = V_{sp} + \mu_i \quad (4.1)$$

Onde V_i representa o sinal de acionamento dos atuadores, V_{sp} é o sinal de acionamento do set-point e μ_i o sinal de controle da saída do controlador fuzzy. As variáveis linguísticas de saída dos controladores de velocidade são apresentadas na Fig.(4.2). Na Tab.(4.1) são apresentados os parâmetros de cada função de pertinência.

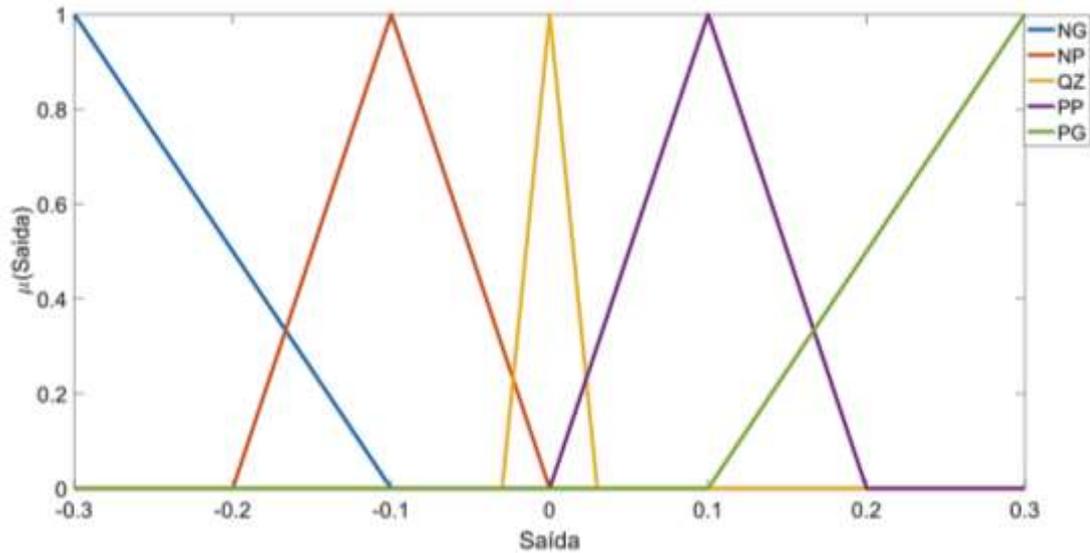


Figura 4.2. Funções de pertinência da variável de Saída dos controladores de velocidade. Fonte: Autoria Própria (2021).

Tabela 4.1. Parâmetros das funções de pertinência dos controladores fuzzy. Fonte: Autoria Própria (2021).

Variável	Controlador de Velocidade	
ERRO	NG (trapezoidal)	[-200, -200, -150, -10]
	NP (triangular)	[-40, -20, 0]
	QZ (triangular)	[-1, 0, 1]
	PP (triangular)	[0, 20, 40]
	PG (trapezoidal)	[10, 150, 200, 200]
SAIDA	NG (trapezoidal)	[-0.3, -0.3, -0.3, -0.1]
	NP (triangular)	[-0.2, -0.1, 0]
	QZ (triangular)	[-0.03, 0, 0.03]
	PP (triangular)	[0, 0.1, 0.2]
	PG (trapezoidal)	[0.1, 0.3, 0.3, 0.3]

As regras dos controladores Fuzzy do tipo SE-ENTÃO, apresentadas na Tab.(4.2), foram elaboradas de acordo com o conhecimento do funcionamento do sistema.

Tabela 4.2. Regras Fuzzy. Fonte: Autoria Própria (2021).

Se ERRO = NG então SAÍDA = NG
Se ERRO = NP então SAÍDA = NP
Se ERRO = QZ então SAÍDA = QZ
Se ERRO = PP então SAÍDA = PP
Se ERRO = PG então SAÍDA = PG

Para validar os controladores de velocidade, os atuadores foram submetidos a entrada do tipo degrau, com valor de 100 mm/s. Nas Fig.(4.3), (4.4), (4.5), (4.6), (4.7) e (4.8) são apresentadas as curvas de resposta dos atuadores (velocidade) e do sinal de referência (*setpoint*), dos atuadores 1, 2 e 3.

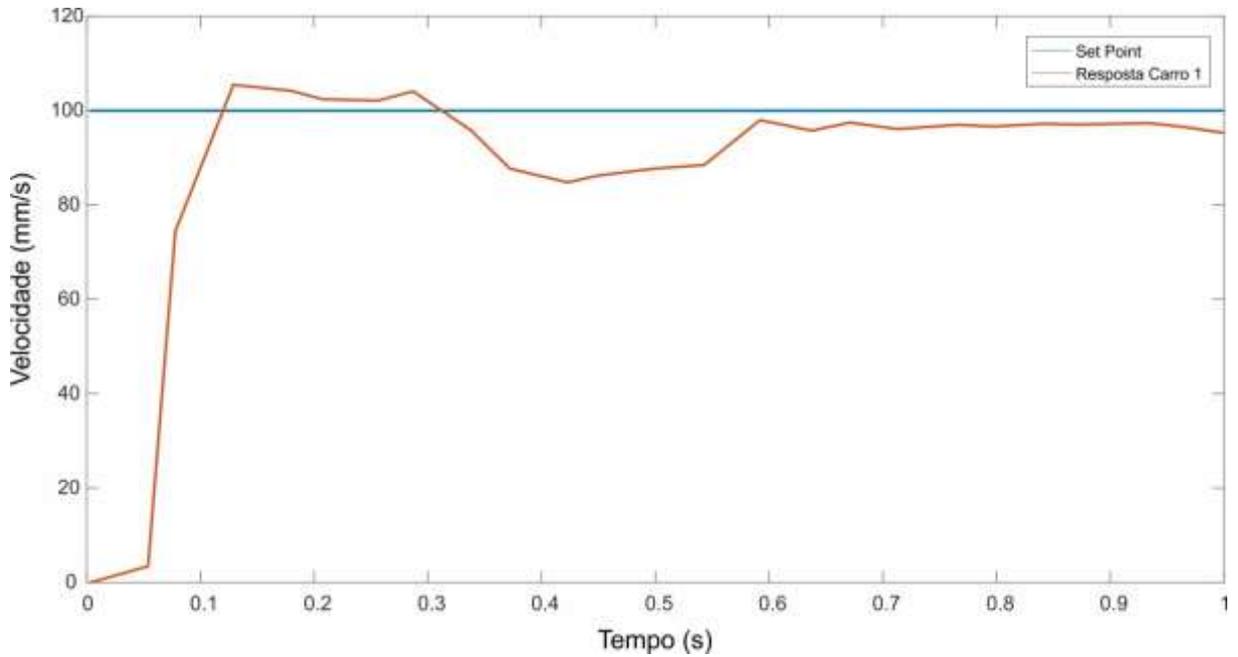


Figura 4.3. Curva de resposta do atuador 1 (100mm/s). Fonte: Autoria Própria (2021).

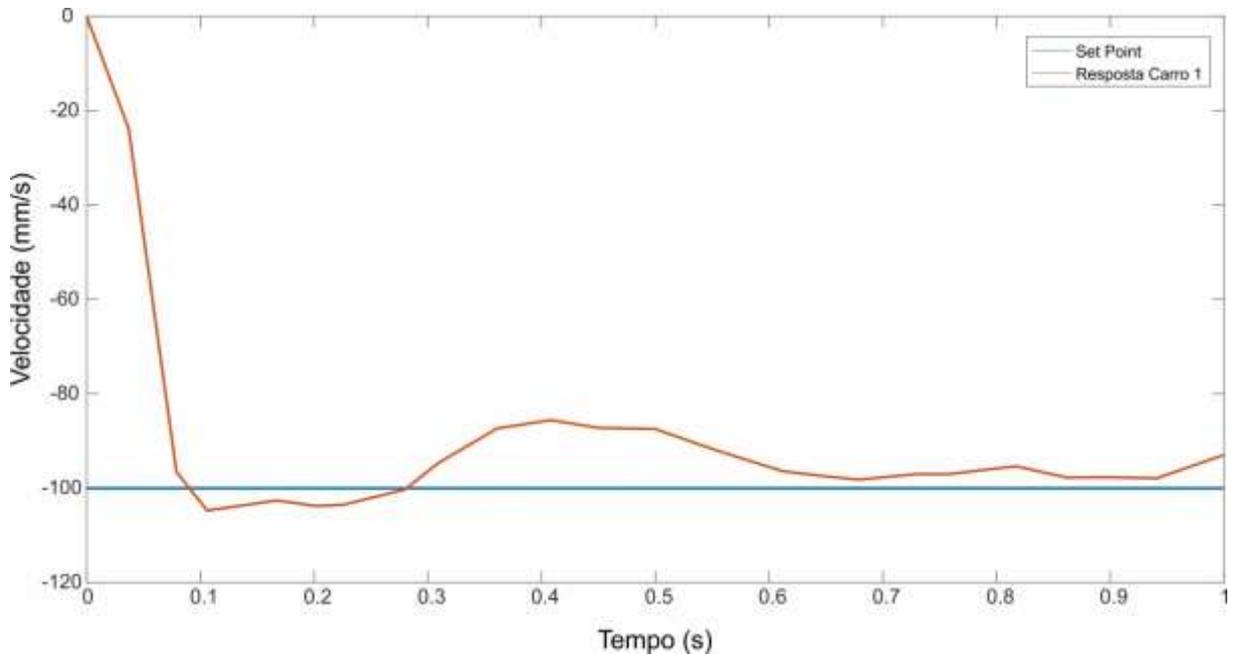


Figura 4.4. Curva de resposta do atuador 1 (-100mm/s). Fonte: Autoria Própria (2021).

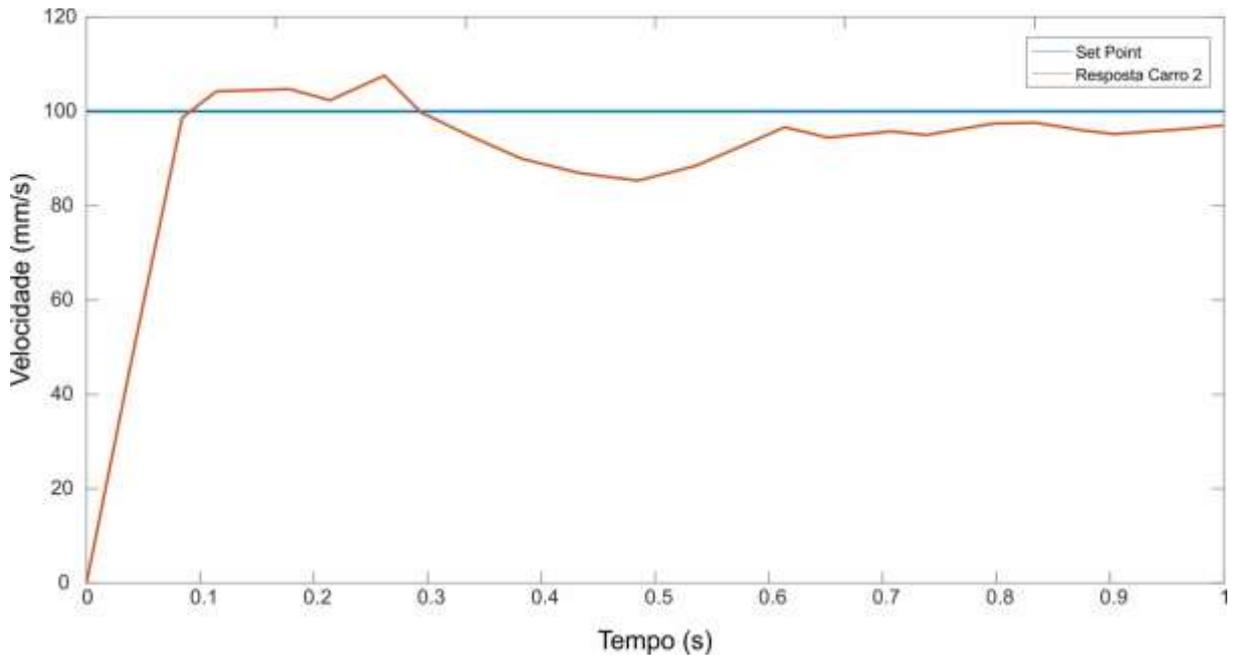


Figura 4.5. Curva de resposta do atuador 2 (100mm/s). Fonte: Autoria Própria (2021).

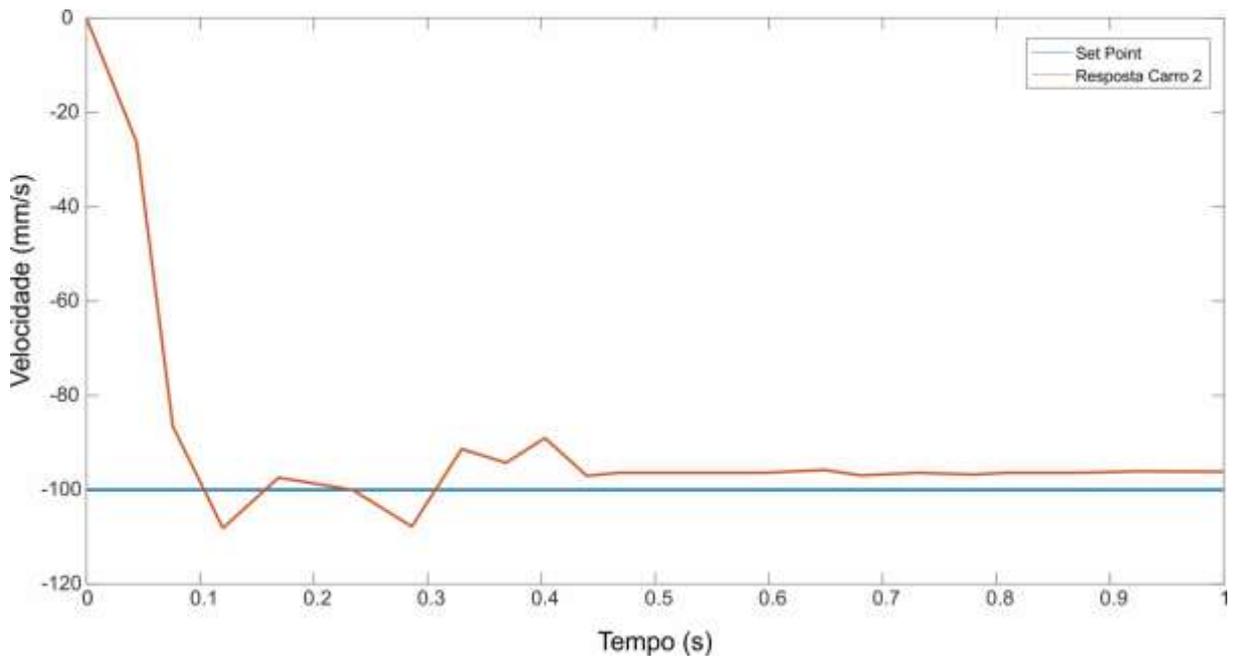


Figura 4.6. Curva de resposta do atuador 2 (-100mm/s). Fonte: Autoria Própria (2021).

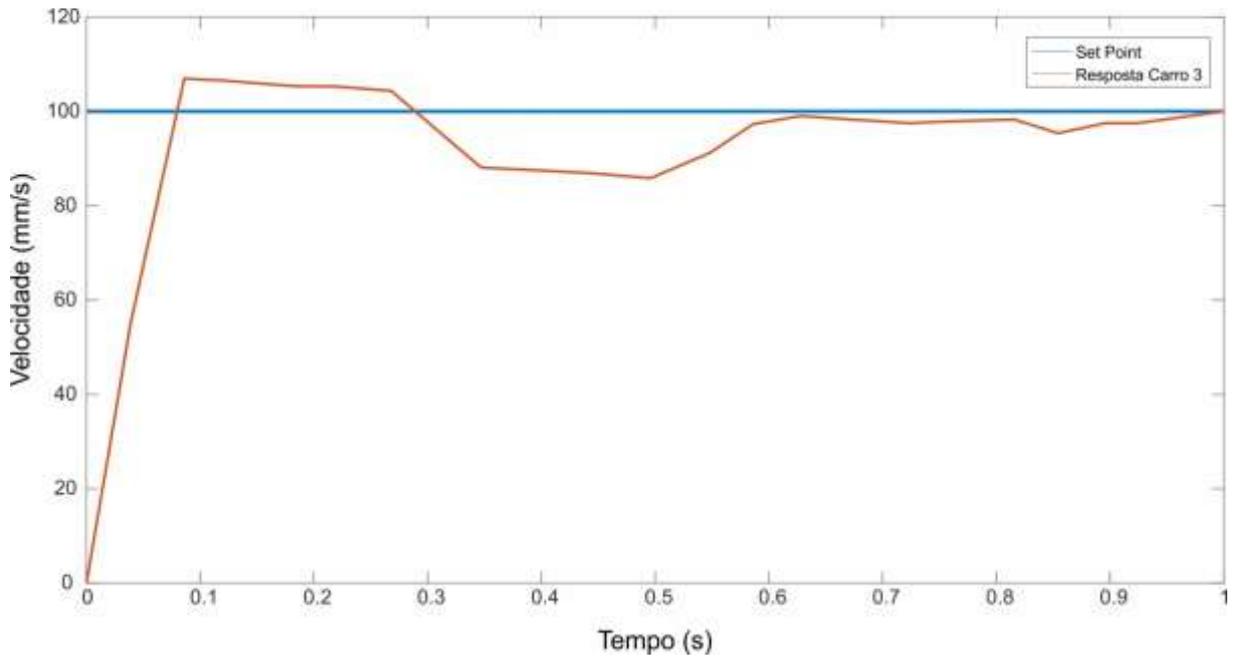


Figura 4.7. Curva de resposta do atuador 3 (100mm/s). Fonte: Autoria Própria (2021).

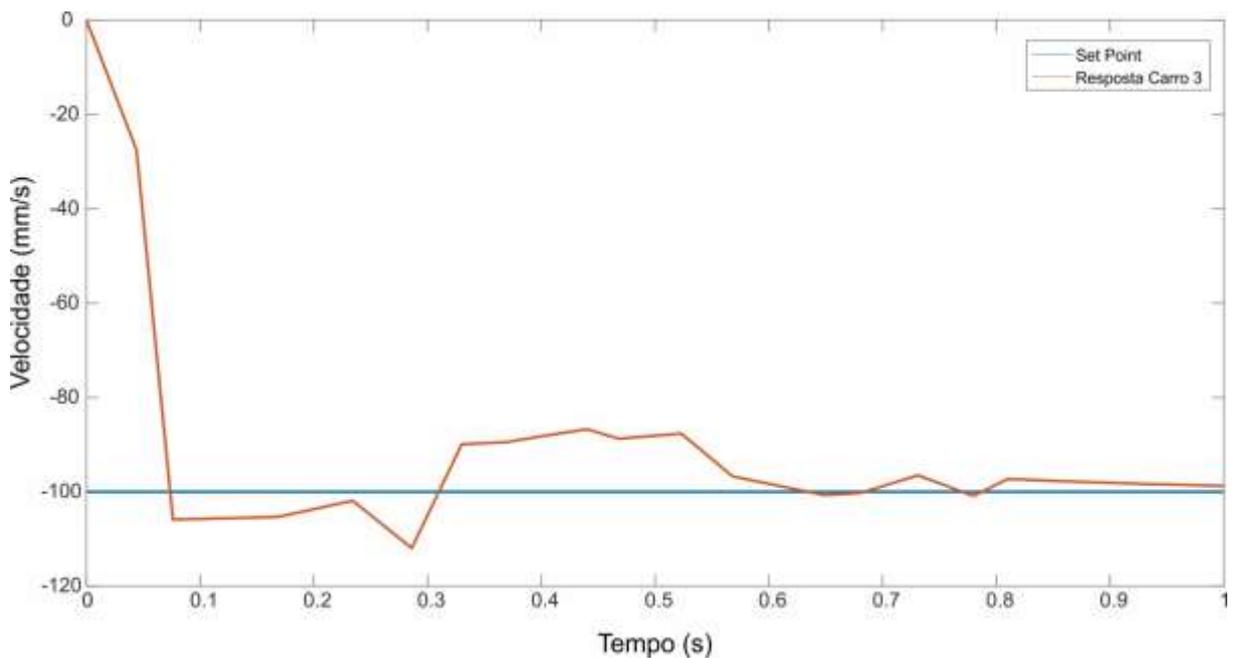


Figura 4.8. Curva de resposta do atuador 3 (-100mm/s). Fonte: Autoria Própria (2021).

Por meio das curvas de resposta, foi possível obter alguns parâmetros das respostas nos regimes transitório e permanentes, que são importantes para a análise do desempenho do controlador. Na Tab.(4.3) são apresentados o tempo de acomodação T_s (tempo necessário para o sistema alcançar e permanecer dentro de uma faixa de 2% em torno de seu valor final), o tempo de subida T_r ou “rise time” que é o tempo para o sinal sair de 10 à 90% do valor final, o sobressinal M_p (valor percentual máximo da resposta em relação ao sinal de referência) e erro de regime permanente e_{ss} (erro que o sistema apresenta após o tempo de acomodação).

Tabela 4.3. Índices de desempenho dos controladores. Fonte: Autoria Própria (2021).

Parâmetros analisado	Controlador atuador 1		Controlador atuador 2		Controlador atuador 3	
	100mm/s	-100mm/s	100mm/s	-100mm/s	100mm/s	-100mm/s
T_s (s)	0,63	0,679	0,628	0,44	0,628	0,681
T_r (s)	0,085	0,05	0,076	0,087	0,065	0,053
M_p (%)	5,5	4,7	7,6	8,1	6,9	11,9
e_{ss} (mm/s)	2,52	-2,33	2,41	-3,61	1,73	-1,7

A partir da Tab. (4.3) é possível verificar que o sistema apresentou erros de regime permanente dentro do previsto para as características do sistema. Além de um tempo de acomodação aceitável, visto que será adicionado ao sistema um controlador de trajetória para corrigir os erros nos eixos X, Y e Z.

4.2. CONTROLE DE TRAJETÓRIA

Assim como os controladores de velocidade, o controlador de trajetória Fuzzy foi implementado pelo método Mamdani através do ‘VI (Virtual Instrument) Controlador de Lógica Fuzzy’ disponível no Fuzzy Logic Toolkit do software Labview®. De acordo com os princípios dos sistemas Fuzzy de serem baseados na experiência e nos padrões de raciocínio dos operadores humanos, o modelamento do controlador foi realizado a partir de um estudo prévio e experimental do comportamento desejado para os regimes transitório e permanente do efetuador.

As funções de pertinência e regras fuzzy foram baseadas no trabalho de Bordon, Andreoti e Menezes (2016), e posteriormente foram adaptadas para o sistema proposto. Afim de coletar informações para definir o universo de discurso das variáveis de entrada, foi realizado um teste sem o controlador de trajetória. Os resultados deste teste são apresentados na Fig.(4.9).

No bloco de fuzzificação foram criadas três variáveis linguísticas denominadas *ErroX*, *ErroY* e *ErroZ*, que representa o desvio de trajetória nas direções X, Y e Z respectivamente. Com base nos dados apresentados na Fig.(4.9) o universo de discurso das variáveis *ErroX*, *ErroY* e *ErroZ* abrangem o intervalo de [-30, 30], representando os valores máximos de erro em milímetros. As variáveis linguísticas de entrada foram compostas por cinco funções de pertinência com os termos linguísticos denominados: Negativo Grande (NG), Negativo Pequeno (NP), Quase Zero (QZ), Positivo Pequeno (PP) e Positivo Grande (PG). Para as

variáveis NG e PG foram utilizadas funções trapezoidais, e para as demais foram utilizadas funções triangulares.

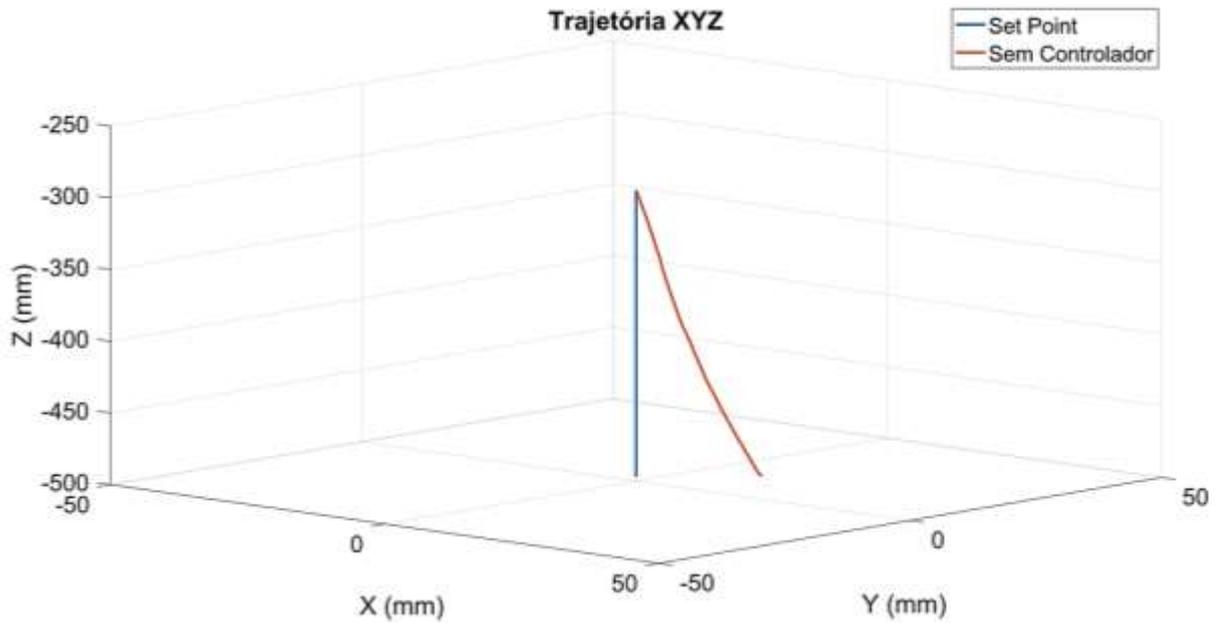


Figura 4.9. Trajetória sem o controlador. Fonte: Autoria Própria (2021).

As Fig.(4.10), (4.11) e (4.12) apresentam as variáveis linguísticas das entradas do controlador de trajetória com suas funções de pertinência e termos linguísticos correspondentes. Na Tab.(4.4) são apresentados os parâmetros de cada função de pertinência.

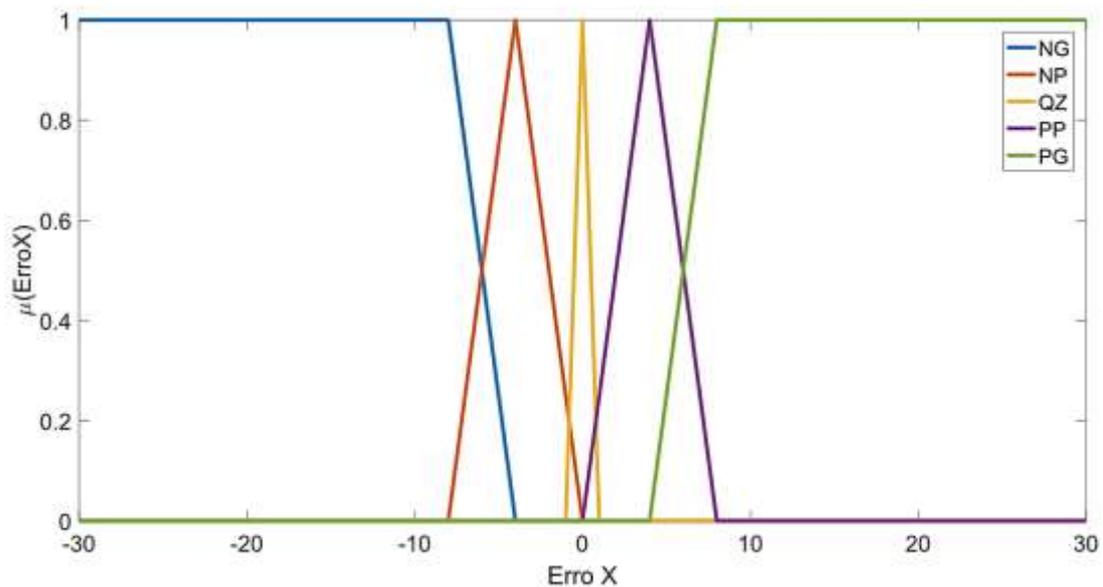


Figura 4.10. Funções de Pertinência da variável de entrada ErroX. Fonte: Autoria Própria (2021).

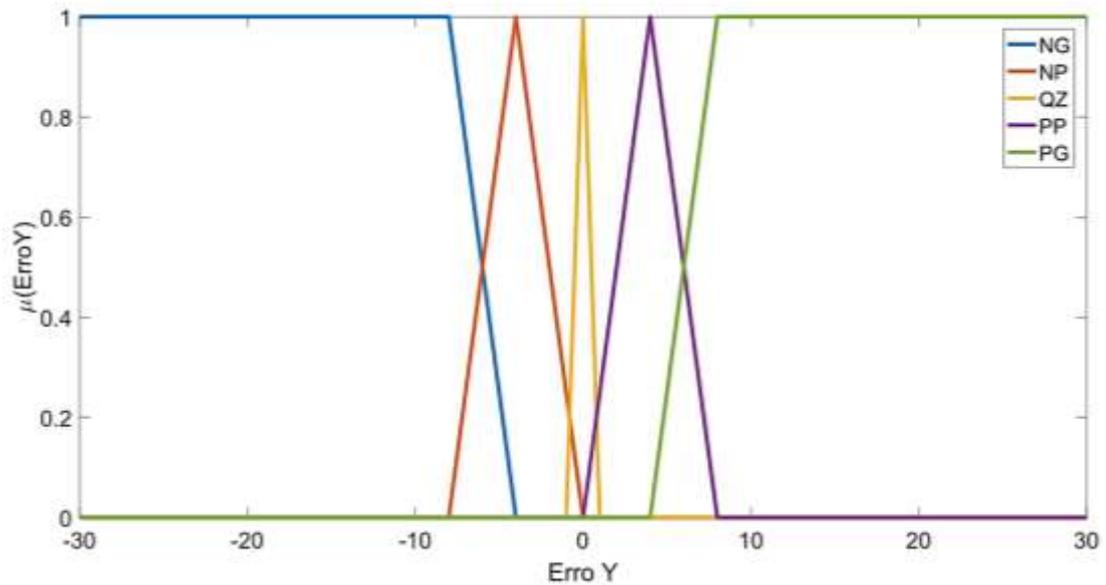


Figura 4.11. Funções de Pertinência da variável de entrada ErroY. Fonte: Autoria Própria (2021).

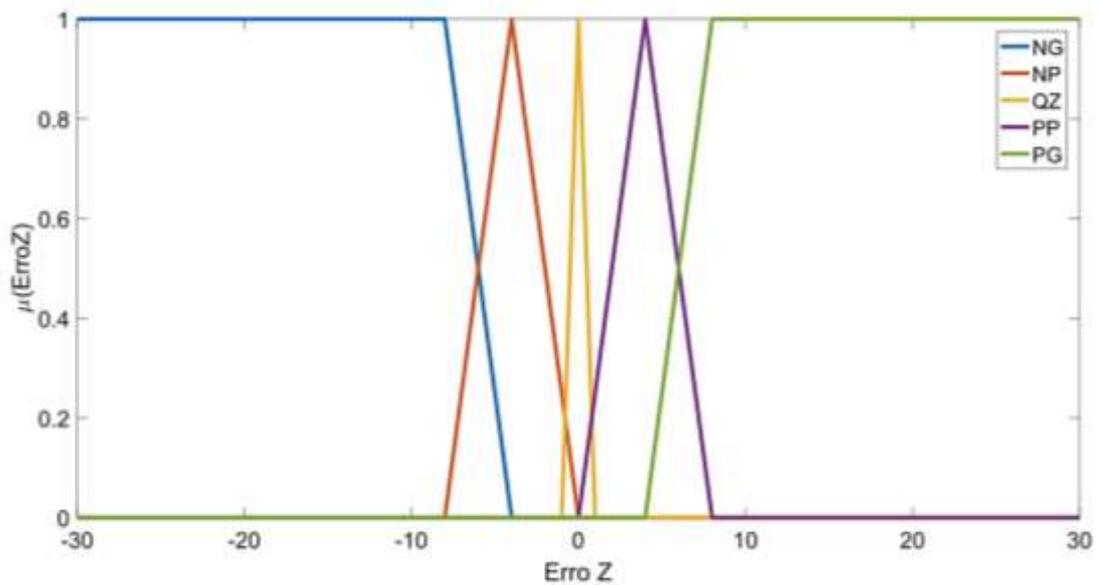


Figura 4.12. Funções de Pertinência da variável de entrada ErroZ. Fonte: Autoria Própria (2021).

Na defuzzificação foram criadas três variáveis linguísticas denominadas *SaídaX*, *SaídaY* e *SaídaZ* que representam o sinal de controle que atuará sobre o sinal de velocidade. O universo de discurso desta variável abrange o intervalo $[-15 \ 15]$. Para as variáveis *SaídaX*, *SaídaY* e *SaídaZ* foram associadas cinco funções de pertinência, definidas pelos termos NG, NP, QZ, PP e PG. Para as variáveis NG e PG foram utilizadas funções trapezoidais, e para as demais foram utilizadas funções triangulares. O sinal de controle irá atuar sobre a velocidade de acordo com a eq.(4.2).

$$V_i = V_{sp} + \mu_i \quad (4.2)$$

Onde V_i representa o sinal das velocidades desejadas em cada direção, V_{sp} é o sinal de acionamento do set-point e μ_i o sinal de controle da saída do controlador fuzzy. As variáveis linguísticas de saída do controlador de trajetória são apresentadas na Fig.(4.13), (4.14) e (4.15). Na Tab.(4.4) são apresentados os parâmetros de cada função de pertinência.

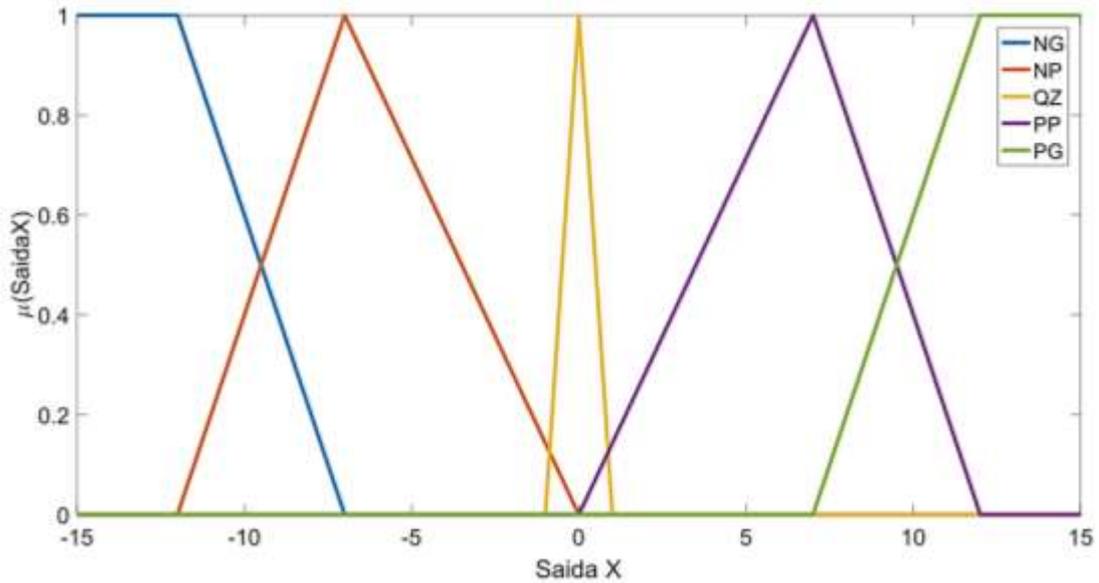


Figura 4.13. Funções de Pertinência da variável de saída SaídaX. Fonte: Autoria Própria (2021).

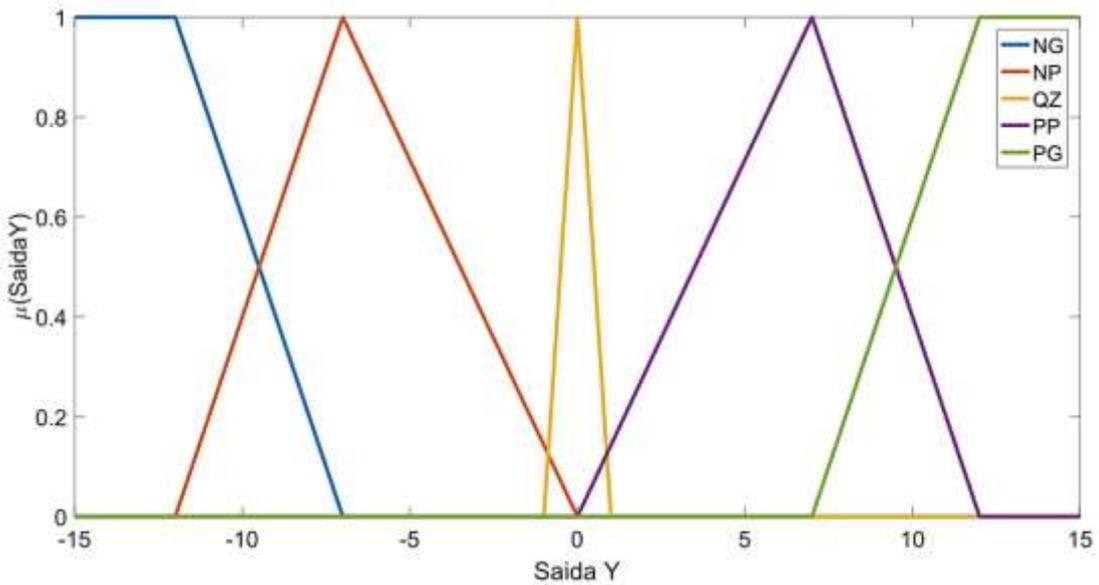


Figura 4.14. Funções de Pertinência da variável de saída SaídaY. Fonte: Autoria Própria (2021).

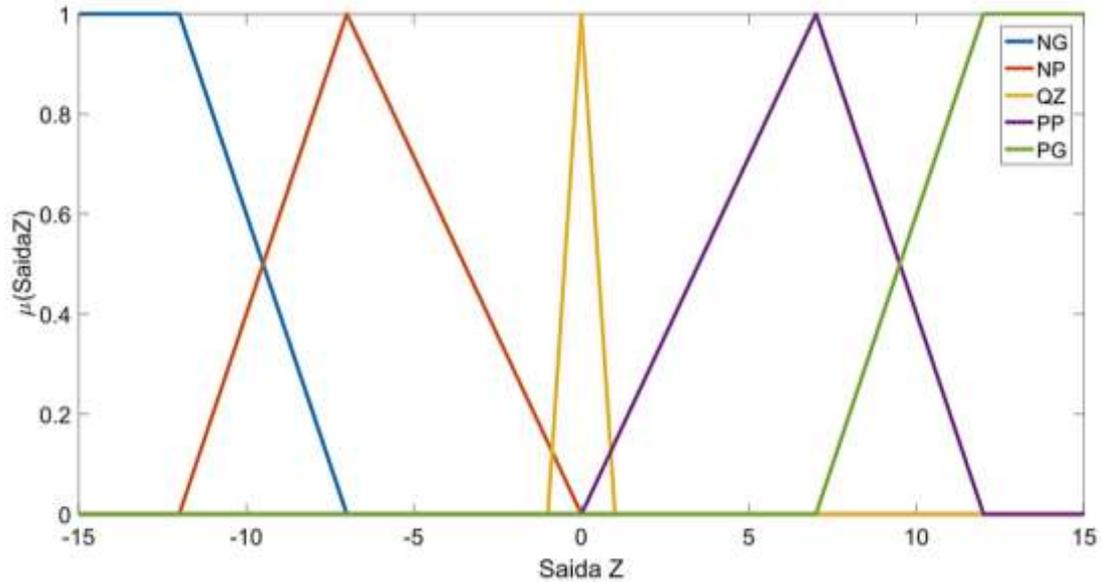


Figura 4.15. Funções de Pertinência da variável de saída SaídaZ. Fonte: Autoria Própria (2021).

Tabela 4.4. Parâmetros das funções do controlador de trajetória. Fonte: Autoria Própria (2021).

ErroX		ErroY		ErroZ	
NG (trapezoidal)	[-30,-30,-8,-4]	NG (trapezoidal)	[-30,-30,-8,-4]	NG (trapezoidal)	[-30,-30,-8,-4]
NP (triangular)	[-8,-4,0]	NP (triangular)	[-8,-4,0]	NP (triangular)	[-8,-4,0]
QZ	[-1,0,1]	QZ	[-1,0,1]	QZ	[-1,0,1]
PP (triangular)	[0,4,8]	PP (triangular)	[0,4,8]	PP (triangular)	[0,4,8]
PG (trapezoidal)	[4,8,30,30]	PG (trapezoidal)	[4,8,30,30]	PG (trapezoidal)	[4,8,30,30]
SaídaX		SaídaY		SaídaZ	
NG (trapezoidal)	[-15,-15,-12,-7]	NG (trapezoidal)	[-15,-15,-12,-7]	NG (trapezoidal)	[-15,-15,-12,-7]
NP (triangular)	[-12,-7,0]	NP (triangular)	[-12,-7,0]	NP (triangular)	[-12,-7,0]
QZ	[-1,0,1]	QZ	[-1,0,1]	QZ	[-1,0,1]
PP (triangular)	[0,7,12]	PP (triangular)	[0,7,12]	PP (triangular)	[0,7,12]
PG (trapezoidal)	[7,12,15,15]	PG (trapezoidal)	[7,12,15,15]	PG (trapezoidal)	[7,12,15,15]

As regras do controlador Fuzzy do tipo SE-ENTÃO, apresentadas na Tab.(4.5), foram elaboradas de acordo com o conhecimento do funcionamento do sistema.

Tabela 4.5. Regras Fuzzy do controlador de trajetória. Fonte: Autoria Própria (2021).

Se ErroX = NG então SaídaX = NG	Se ErroY = NG então SaídaY = NG	Se ErroZ = NG então SaídaZ = NG
Se ErroX = NP então SaídaX = NP	Se ErroY = NP então SaídaY = NP	Se ErroZ = NP então SaídaZ = NP
Se ErroX = QZ então SaídaX = QZ	Se ErroY = QZ então SaídaY = QZ	Se ErroZ = QZ então SaídaZ = QZ
Se ErroX = PP então SaídaX = PP	Se ErroY = PP então SaídaY = PP	Se ErroZ = PP então SaídaZ = PP
Se ErroX = PG então SaídaX = PG	Se ErroY = PG então SaídaY = PG	Se ErroZ = PG então SaídaZ = PG

A Fig.(4.16) apresenta o resultado da trajetória após a implementação do controlador de trajetória.

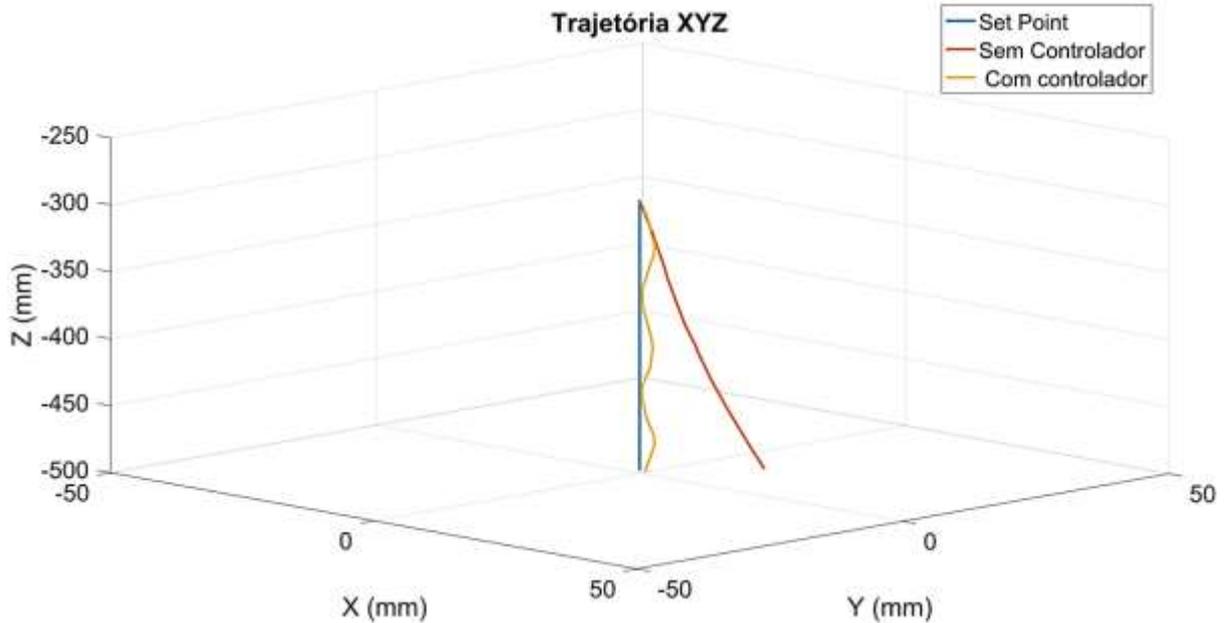


Figura 4.16. Trajetória XYZ com e sem controlador de trajetória. Fonte: Autoria Própria (2021).

Na Tab.(4.6) é apresentada a comparação do desempenho da trajetória com e sem controlador.

Tabela 4.6. Comparação de Trajetória Com e Sem controlador. Fonte: Autoria Própria (2021).

	Sem Controlador (mm)	Com Controlador (mm)	Redução (%)
Erro em X	-12,27	-1,5	87,77
Erro em Y	-11,72	-1,76	84,98
Erro em Z	-1	1,02	2%

Com base na Fig.(4.16) e na Tab.(4.6) é possível afirmar que o controlador se apresentou efetivo em sua utilização, reduzindo 87,77% do erro na direção X, em 84,98% na direção Y e em Z o erro se manteve praticamente constante. Assim, comparando o vetor erro, a trajetória realizada o erro de trajetória foi de 1,26%.

CAPÍTULO V

CONCLUSÕES E TRABALHOS FUTUROS

5.1. CONSIDERAÇÕES FINAIS

Este trabalho apresentou o controle de trajetória baseado em lógica fuzzy para um protótipo com estrutura cinemática paralela tipo delta linear. O conjunto eletromecânico foi acionado de forma exitosa por meio de motores de corrente contínua (CC) com técnica PWM, interfaceados por circuitos drivers de potência (ponte H), que receberam sinais de controle de um sistema computadorizado composto por módulo multifuncional de sinais e software Labview[®]. Os gráficos apresentados e os critérios de desempenho calculados foram obtidos por meio do software Matlab[®]. Foi planejada uma Interface Homem-Máquina (IHM) tipo supervisório em LabView[®] que permitiu a introdução dos dados de entrada tais como as posições XYZ (setpoint) e a velocidade de movimentação. O sistema supervisório também permitiu a visualização das saídas (trajetória e velocidades), entradas (setpoint) e erros, por meio de gráficos plotados em tempo real. Um modelo proposto foi desenvolvido por meio do método de soma vetorial, onde é realizada uma análise geométrica da estrutura para definir os vetores atuantes. O modelo matemático associa as coordenadas dos atuadores (1, 2, 3) e o efetuador final (X, Y, Z), com base em uma análise de cinemática inversa. A análise do desempenho dos controladores de velocidade apresentou um erro máximo de -3,61 mm/s no acionamento do atuador 2 com o setpoint de -100mm/s. Este erro é referente a 3,61%, o que é considerado satisfatório pelos autores. A análise do controle de trajetória apresentou um desempenho muito satisfatório com um erro de regime permanente de -1,50mm para a trajetória em X, de -1,76mm para a trajetória em Y e de 1,02mm para a trajetória em Z, resultando em um erro de trajetória de 1,26%.

5.2. TRABALHOS FUTUROS

Como trabalhos futuros sugere-se a utilização de um sistema de retroalimentação da posição espacial da plataforma móvel por meio de visão computacional, o que pode promover uma redução dos erros presentes pela aproximação dos modelos matemáticos. Também é sugerida a geração de trajetórias circulares.

REFERÊNCIAS BIBLIOGRÁFICAS

- ABREU, J.L.M.C. et al. **Embedded System for the Position Control of a Low-Cost Linear Delta Linear Robot**. In: Congresso Internacional de Gestão e Tecnologia do Programa Despertando Vocações, 2020, Virtual. Anais do IV COINTER PDVGT. Recife: Editora IIDV, 2020. v. 1.
- ALMEIDA JÚNIOR, A.G. et al. **Desenvolvimento e Controle de Robô Delta para Manipulação Industrial**. In: II Congresso Internacional de Gestão e Tecnologias, João Pessoa. Anais do II COINTER PDVGT, 2018a.
- ALMEIDA JÚNIOR, A.G. et al. **Modelamento Matemático e Otimização do Espaço de Trabalho para Robô Paralelo Delta**. In: II Congresso Internacional de Gestão e Tecnologias, João Pessoa. Anais do II COINTER PDVGT, 2018b.
- ALMEIDA JÚNIOR, A.G. et al. **Projeto Eletromecânico e Controle de Robô Paralelo Tipo Linapod**. In: II Congresso Internacional de Gestão e Tecnologias, João Pessoa. Anais do II COINTER PDVGT, 2018c.
- ARAÚJO-GÓMEZ et al. 2019. **Kinematic Analysis and Dimensional Optimization of a 2R2T Parallel Manipulator**. Journal of the Brazilian Society of Mechanical Sciences and Engineering. 41. 10.1007/s40430-019-1934-1.
- BARRIENTOS et al. **Fundamentos de Robótica**. Madrid: McGraw-Hill, 1997.
- BORDON, M.E.; ANDREOTTI S.E.; MENEZES T.A. **Controle de Trajetória de um Robô Móvel com Tração Diferencial Usando Lógica Nebulosa**. In: VI Mostra Nacional de Robótica, Recife, PE, 2016.
- CAMPOS, M.M.; SAITO, K. 2004. **Sistemas Inteligentes em Controle e Automação de Processos**. Rio de Janeiro, Editora Ciência Moderna.
- CARBONE, G. 2011. **Validation of Stiffness Analysis and Experimental Robotic Systems**. Frontiers Mechanical Engineering, vol. 6, issue 2, pp. 182-196.
- CASTRO, A.L.A. **Projeto e Construção de um Robô Manipulador SCARA**. Trabalho de Conclusão de Curso. (Graduação em Engenharia Mecânica), Universidade de Brasília, 2019.
- CLAVEL, R. 1988. **Delta: a Fast Robot, with Parallel Geometry**. Proceedings of the XVIII Int. Symposium on Industrial Robot, Lausanne.

CRUZ, J.A.G. **Monitoração e Análises da Penetração do Cordão de Solda Através da Observação da Oscilação da Poça de Fusão no Processo GMAW-S**. 122p. Tese de Doutorado (Engenharia Mecânica), Universidade de Brasília, Faculdade de Tecnologia, 2018.

DORF, R.C.; BISHOP, R.H. 2018. **Sistemas de Controle Modernos**. 13 ed. Rio de Janeiro: Editora LTC.

FILHO, S.C.T. 2006. **Modelagem e Otimização de um Robô de Arquitetura Paralela para Aplicações Industriais**. Dissertação de Mestrado em Engenharia Mecânica, Escola Politécnica da Universidade de São Paulo.

FUJIMOTO, R.Y. **Diagnóstico Automático de Defeitos em Rolamentos Baseado em Lógica Fuzzy**. Dissertação de mestrado em Engenharia Mecânica. Escola Politécnica da Universidade de São Paulo, 2005.

GASCA, E.A.R. **Desenvolvimento de um Robô com Cinemática Paralela Delta Linear para Manufatura Aditiva Aderente a STEP-NC**. 289p. Dissertação de Mestrado (Engenharia Mecânica), Universidade de Brasília, Faculdade de Tecnologia, 2018.

GONÇALVES, J.A.L.J. **Modeling and Trajectory Control Simulation for Robot Manipulators With Three Degrees of Freedom of Rigid Elements**. 2016. 84 f. Trabalho de Conclusão de Curso (Bacharelado em Engenharia Eletrônica). Universidade Tecnológica Federal do Paraná. Ponta Grossa, 2016.

GONÇALVES, R.S. 2009. **Estudo de rigidez de Cadeias Cinemáticas Fechadas**. Dissertação de Mestrado em Engenharia Mecânica, Universidade Federal de Uberlândia.

KNAPP, W.; COBET, M. 2000. **The IWF Hexaglide: A New Concept for High Speed Machining**. IWF, Zurich.

MOSER, A.D.L; MALUCELLI, M.F.; BUENO, S.N. **Cadeia Cinética Aberta e Fechada: uma Reflexão Crítica**. *Fisioter. mov.* 23 (4), 2010.

NORTON, R.L. **Projeto de Máquinas: Uma abordagem integrada**. 4ª ed. Porto Alegre: Bookman, 2013. 1028 p.

OLIVEIRA NETO, I.A. **Desenvolvimento de uma Cadeira de Rodas Robótica para Transporte de Portador de Necessidades Especiais**. Dissertação de Mestrado (Engenharia Elétrica), Universidade Federal do Rio Grande do Norte, Natal-RN, Brasil, 2013.

ORTEGA, N.R.S. **Aplicação da Teoria de Conjuntos Fuzzy a Problemas da Biomedicina**. Tese de doutorado em Ciências. Universidade de São Paulo, 2001.

OSIRO, L. **Uso da Lógica Fuzzy para Avaliação e Desenvolvimento de Fornecedores Baseado em Modelos de Portfólio**. Tese de doutorado em Engenharia de Produção. Universidade de São Paulo, 2013.

SENA, A.P.C. **Diagnóstico de Avarias em Transmissão por Engrenagens Baseado na Corrente do Motor de Indução e Entropia Wavelet**. 114f. Tese de Doutorado (Engenharia Mecânica), Universidade Federal da Paraíba, João Pessoa – PB, Brasil, 2015.

SENA, A.P.C. et al. **Sistema Embarcado para o Controle de Posição de um Robô Paralelo Tipo Delta Linear de Baixo Custo**. Perspectivas das Engenharias na Sociedade 5.0: Educação, Ciência, Tecnologia e Amor. 1ed. Recife: Instituto Internacional Despertando Vocações, 2020, v. 1, p. 82-98.

SENA, A.P.C. **Implementação Experimental de Sistema de Controle de Posição de Mesa de Coordenadas**. 2008. 49 f. Trabalho de Conclusão de Curso (Automação Industrial), Instituto Federal de Educação, Ciência e Tecnologia da Paraíba, Paraíba, 2008.

SENA, A.P.C.; SILVA, A.V.S.; SILVA, A.S. **Construction and Control PID of Delta Parallel Robot**. In: 24th ABCM International Congress of Mechanical Engineering, 2017, Curitiba. v. 1, 2017.

SILVA NETO, M.B. **Construção Eletromecânica e Controle de Posição de um Protótipo com Estrutura Cinemática Paralela Tipo Robô Delta**. Trabalho de Conclusão de Curso. (Graduação em Engenharia Mecânica), Instituto Federal de Pernambuco, 2019.

SILVA, A.S. et al. **Avaliação de Controladores PID em Protótipo Robótico com Cinemática Paralela Aplicado a Manipulação Industrial**. In: Congresso Internacional do Programa Despertando Vocações Tecnológicas, Recife, 2019.

SILVA, A.S. et al. **Construção e Controle de Posição para um Robô Paralelo com Aplicação na Indústria Confeccionista**. In: Henrique Ajuz Holzmann; João Dallamuta; Marcelo Henrique Granza. (Org.). A Produção do Conhecimento na Engenharia Elétrica 2. 1ed. Ponta Grossa: Atena Editora, 2020, v. 2, p. 82-94.

SILVA, A.S. et al. **Lógica Fuzzy Aplicada ao Controle de Posição de um Robô Delta**. In: Congresso Internacional do Programa Despertando Vocações Tecnológicas, Recife, 2021.

SIMÕES, M.G.; SHAW, I.S. **Controle e Modelagem Fuzzy**. Editora Blucher, 2ª ed. São Paulo, 2007.

SIMPLÍCIO, P.V.G.; LIMA, B.R.; JUNKES, J.A. **Manipuladores Robóticos Industriais**. Cadernos de Graduação: Ciências exatas e tecnológicas, Aracaju, v. 3, n. 3, p. 85-94, Outubro 2016.

STAN, S-D. et al. 2011. **Performance Analysis of 3 DOF Delta Parallel Robot**. HSI2011.

TARTARI FILHO, S.C. **Modelagem e Otimização de um Robô com Arquitetura Paralela para Aplicações Industriais**. 2006. 227 f. Dissertação de Mestrado (Engenharia Mecatrônica), Escola Politécnica da Universidade de São Paulo, São Paulo, 2006.

THOMAS, M.; JOY, M.L.; SUDHEER. AP. 2020. **Kinematic and Dynamic Analysis of a 3-PRUS Spatial Parallel Manipulator**. Chinese Journal of Mechanical Engineering. 33. 10.1186/s10033-020-0433-8.

XAVIER FILHO, A.F. **Controlador Fuzzy Aplicado a um Sistema de Controle de Posição**. Dissertação de Mestrado (Engenharia Mecânica), UFPB, João Pessoa – PB, Brasil. 2008.

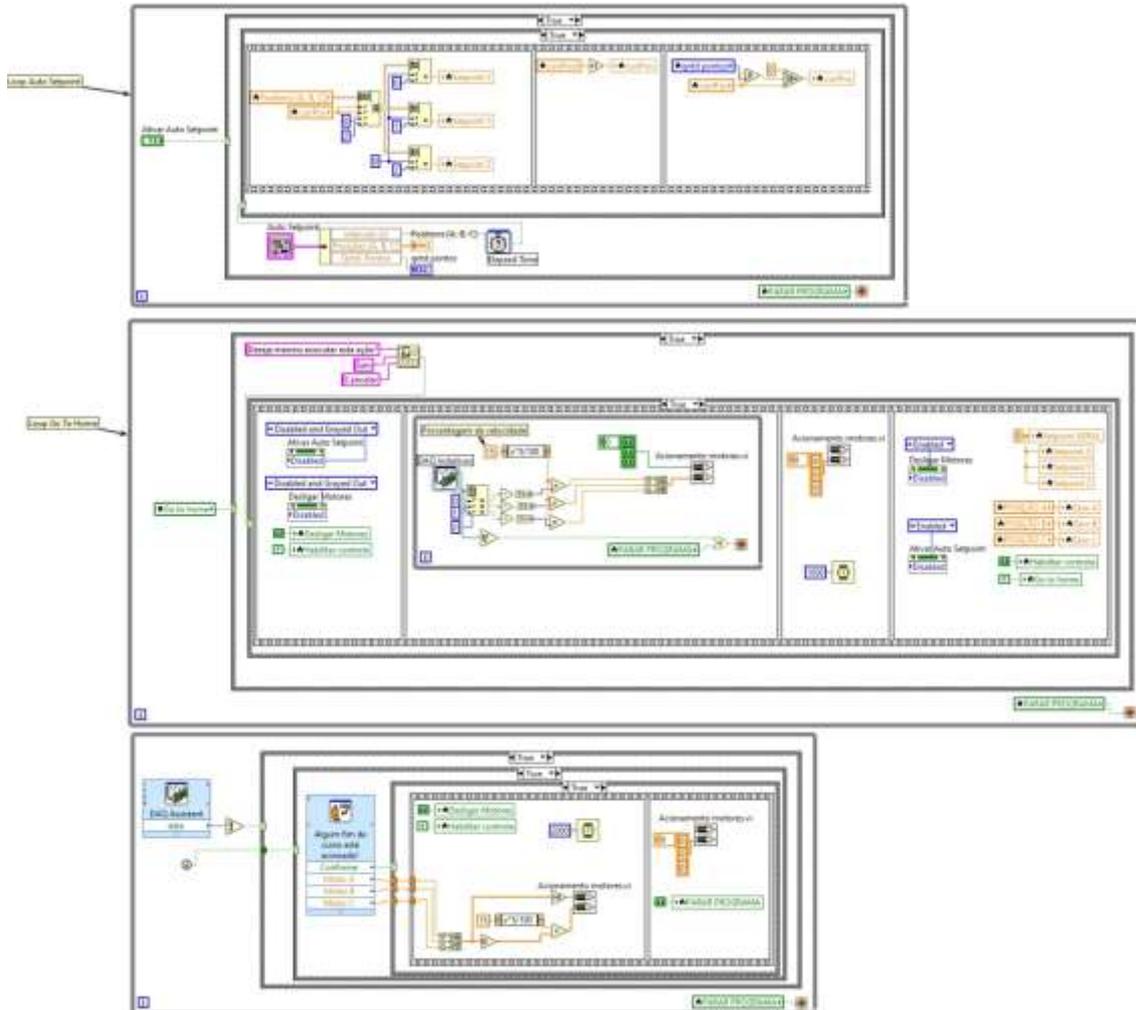
YAGER R.; FILEV, D. **A Simple Adaptive Defuzzification Method**. Ieee Transaction On Fuzzy Systems, v. 1, n. 1, pp.69-78, 1993.

YEN, J.; LANGARI, R. **Fuzzy Logic: Intelligence, Control, and Information**. Prentice Hall, EUA, 1999.

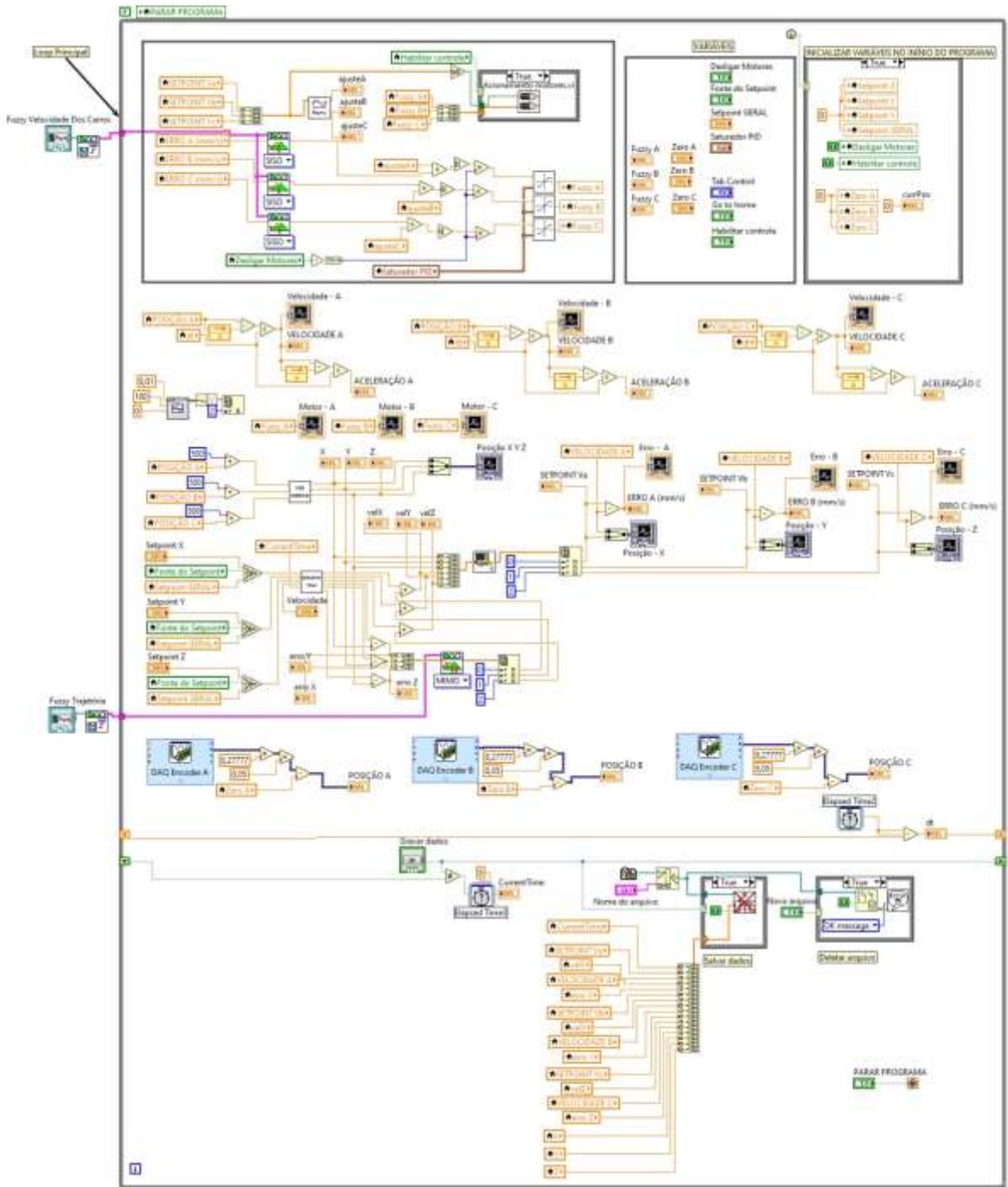
PEREIRA JÚNIOR, M. **Aplicação de Controlador Fuzzy Supervisório em um Secador na Indústria de Fertilizantes**. 82p. Dissertação de Mestrado (Controle e Automação de Processos de Mineração), Universidade Federal de Ouro Preto, Ouro Preto-MG, 2018.

LI, Y.; XU, Q. **Kinematic Analysis of a 3-PRS Parallel Manipulator**. Robotics and Computer Integrated Manufacturing, Vol. 23, pp. 395–408, 2007.

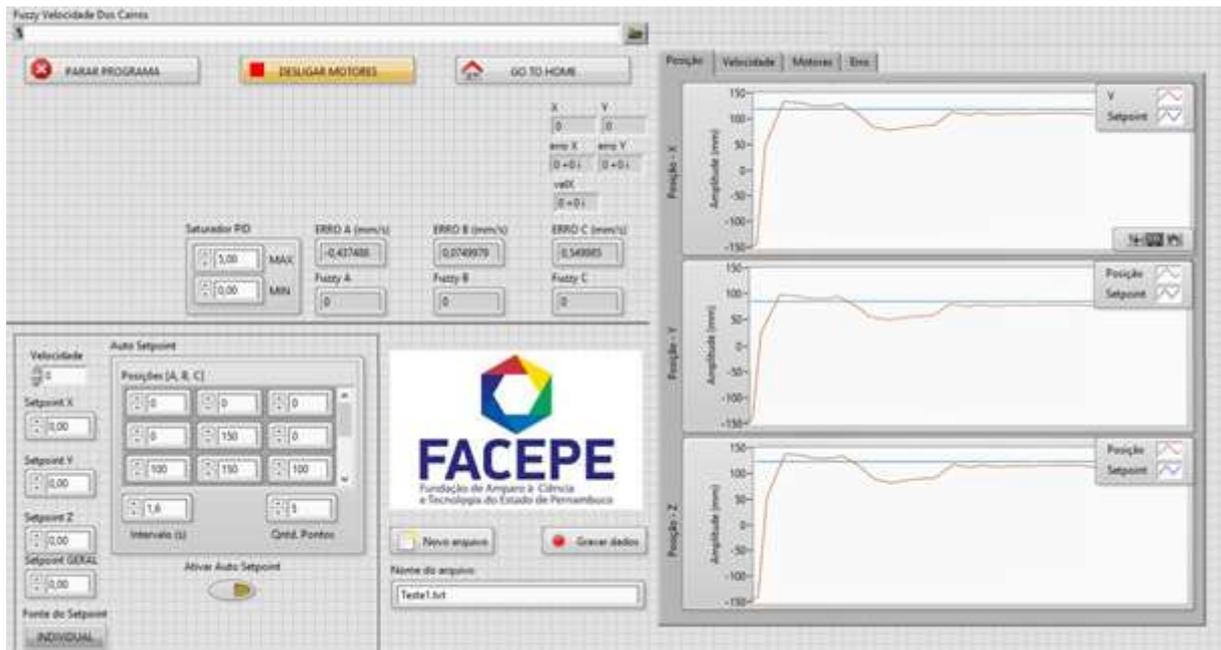
ANEXOS



Programação auxiliar para segurança nas operações. Fonte: Autoria Própria (2021).



Programação dos controladores da máquina. Fonte: Autoria Própria (2021).



Interface Homem Máquina (IHM) do protótipo. Fonte: Autoria Própria (2021).