

UNIVERSIDADE FEDERAL DA PARAÍBA
CENTRO DE ENERGIAS ALTERNATIVAS E RENOVÁVEIS
DEPARTAMENTO DE ENGENHARIA ELÉTRICA

João Salvio da Silva Junior

**Modelagem de Sistemas e Controle de Vazão
em uma Rede de Abastecimento de Água
usando Inteligência Artificial**

João Pessoa
2019

João Salvio da Silva Junior

Modelagem de Sistemas e Controle de Vazão em uma Rede de Abastecimento de Água usando Inteligência Artificial

Trabalho de Conclusão de Curso apresentado à Coordenação do Curso de Engenharia Elétrica da Universidade Federal da Paraíba como pré-requisito para a obtenção do título de Bacharel em Engenharia Elétrica.

Universidade Federal da Paraíba
Centro de Energias Alternativas e Renováveis
Curso de Graduação em Engenharia Elétrica

Orientador: Prof. Dr. Juan Moises M. Villanueva

João Pessoa

2019

© João Sálvio da Silva Júnior

J95m Júnior, João Sálvio da Silva.

Modelagem de Sistemas e Controle de Vazão em uma Rede de Abastecimento de Água usando Inteligência Artificial / João Salvio da Silva Junior. – João Pessoa, 2019.

75 f. : il.

Orientador: Prof. Dr. Juan Moises M. Villanueva
Monografia (Graduação) - UFPB/CEAR

1. Controle de vazão de água. 2. Sistemas Supervisórios. 3. Redes Neurais Artificiais. 4. Lógica Fuzzy. 5. Inteligência Artificial. Villanueva, Juan Moises Maurício. II Título.

UFPB/BC

João Salvio da Silva Junior

Modelagem de Sistemas e Controle de Vazão em uma Rede de Abastecimento de Água usando Inteligência Artificial

Trabalho de Conclusão de Curso apresentado à Coordenação do Curso de Engenharia Elétrica da Universidade Federal da Paraíba como pré-requisito para a obtenção do título de Bacharel em Engenharia Elétrica.

Trabalho aprovado. João Pessoa, 25 de setembro de 2019:

Prof. Dr. Juan Moises M. Villanueva

(Orientador)

Departamento de Engenharia Elétrica

Prof. Dr. Ademar V. da Silva Netto

(Membro)

Departamento de Engenharia Elétrica

Prof. MSc. Ronildo Inácio Soares de Alencar

(Membro)

Departamento de Eng. Civil e Ambiental

João Pessoa

2019

Agradecimentos

Agradeço a divina providência pela vida.

Agradeço a minha avó Teresa por sempre ter me incentivado a estudar, por ter me dado a oportunidade de estudar no Colégio Diocesano Seridoense, durante o ensino médio, e por todo amor e carinho que recebi antes dela partir para a eterna morada.

Agradeço aos meus pais, Jovelina e Luiz Dantas, por terem me dado um lar e irmãos, pelo auxílio financeiro durante a graduação, pelas orações e por sempre acreditarem em mim.

Agradeço ao amigo Bonifácio por ter me apresentado o curso de Engenharia Elétrica, e por meio dele agradeço a todas as amigadas formadas durante a graduação, as quais levarei para a vida.

Agradeço aos professores Nady, Antonio e Juan pela experiência adquirida nas atividades extra curriculares desenvolvidas durante o curso.

Agradeço ao professor Juan por aceitar ser o orientador neste trabalho de conclusão de curso, pelos conhecimentos transmitidos e pela paciência em diversos momentos durante o trajeto.

Agradeço a Universidade Federal da Paraíba pelo ensino gratuito e de qualidade, ao professor Heber, coordenador do LENHS, pela disponibilidade das bancadas para os ensaios e aos colegas Kamila, Thomas e Lucas pela agradável companhia no dia-dia do Laboratório.

Por fim, a todos os que contribuíram direta ou indiretamente para a realização deste trabalho.

Resumo

Sistemas de abastecimento de água são projetados para atender a demanda por água de cidades, complexos industriais, laboratórios de pesquisa, etc. Usualmente estes sistemas operam em malha aberta e com baixo rendimento operacional. Neste sentido, o objetivo deste trabalho é desenvolver um modelo computacional para um sistema específico de abastecimento de água, utilizando técnicas baseadas em Redes Neurais Artificiais e usar este modelo como uma planta para ser controlada por um controlador inteligente, baseado em lógica *fuzzy*. Além disso, fazer uma análise comparativa do desempenho do controlador *fuzzy* com um controlador proporcional clássico. Propõe também, o estudo de técnicas de modelagem de sistemas de abastecimento de água com o intuito de desenvolver sistemas de controle sobre esses modelos, poupando assim, eventuais acidentes em plantas reais, durante os mais diversos testes. Foram realizados experimentos em uma bancada totalmente instrumentalizada, com a capacidade de emular um sistema real de abastecimento de água, a qual possuía seu próprio sistema supervisorio. Os modelos e os controladores foram desenvolvidos por meio do software *Matlab*. Os resultados mostraram que o desempenho do modelo desenvolvido representou com fidelidade o comportamento do sistema de abastecimento de água e que o controlador baseado em lógica nebulosa conseguiu reduzir o tempo de assentamento e tornar o erro do sistema desprezível, se comparado com o controlador proporcional. Sendo assim, pode-se afirmar que a utilização de técnicas baseadas em redes neurais artificiais são eficazes na implementação de modelos de sistemas de abastecimento de água.

Palavras-chave: Sistemas de Abastecimento de Água, Controle de Vazão, Redes Neurais Artificiais, Inversor de Frequência, Lógica *fuzzy* .

Abstract

The purpose of this paper is to develop a computational model for a specific water supply system, using techniques based on Artificial Neural Networks and to use this model as a plant to be controlled by an intelligent controller based on fuzzy logic. Besides that, to do a comparative analysis of the fuzzy controller performance with a proportional classic controller. It also proposes the study of modeling techniques of water supply systems in order to develop control systems on these models, therefore avoiding any accidents on real plants during the several control tests. Experiments were performed on an entirely instrumented workbench, with the capacity to emulate a real water supply system, which had its own supervisory system. The models and the controllers were developed using the Matlab software. The results showed that the performance of the developed model represented with fidelity the behavior of the water supply system and that the fuzzy logic based controller was able to reduce the settling time and make the system error negligible compared to the proportional controller. Thus, it can be affirmed that the use of techniques based on artificial neural networks is effective in the implementation of models of water supply systems.

Keywords: Water Supply Systems, Flow Control, Artificial Neural Networks, Frequency Inverter, Fuzzy Logic.

Lista de ilustrações

Figura 1	–	Esquemático de um sistema de abastecimento de água.	19
Figura 2	–	Tubo condutor de fluido, Campo magnético, Eletrodos.	21
Figura 3	–	Medidor eletromagnético, Campo magnético, Eletrodos.	21
Figura 4	–	Esquema simplificado de um inversor de frequência.	22
Figura 5	–	Comportamento de um controle escalar de velocidade.	23
Figura 6	–	O neurônio de McCulloch e implementações de algumas funções booleanas.	24
Figura 7	–	O neurônio artificial.	25
Figura 8	–	Diagrama de blocos de um controlador proporcional.	28
Figura 9	–	Resposta de um sistema de primeira ordem.	29
Figura 10	–	Perspectivas de Conjuntos clássicos e Conjuntos <i>fuzzy</i>	31
Figura 11	–	Grau de pertinência para diferentes conjuntos.	36
Figura 12	–	Representação da bancada experimental utilizada.	39
Figura 13	–	Transdutor de vazão VMS Pro.	39
Figura 14	–	Conjunto motobomba.	40
Figura 15	–	Inversor de frequência.	41
Figura 16	–	CLP ATOS 4004.	42
Figura 17	–	Curva característica do padrão de treinamento.	43
Figura 18	–	Curva característica do padrão de teste.	44
Figura 19	–	Diagrama de blocos do modelo RNA.	44
Figura 20	–	Filtragem do sinal de vazão de treinamento.	45
Figura 21	–	Comparação entre a saída RNA e o padrão de vazão de teste.	46
Figura 22	–	Diagrama de blocos - Controlador proporcional.	47
Figura 23	–	Método da curva de reação.	47
Figura 24	–	Tabela de <i>Ziegler Nichols</i>	48
Figura 25	–	Interfase <i>Fuzzy Logic Designer</i>	48
Figura 26	–	Diagrama de blocos do controlador Fuzzy.	49
Figura 27	–	Comportamento esperado do sinal de erro.	50
Figura 28	–	Conjunto erro.	51
Figura 29	–	Conjunto derivada do erro.	52
Figura 30	–	Conjunto saída.	52
Figura 31	–	Definição das regras.	53
Figura 32	–	Comportamento do erro e da derivada do erro após um degrau.	53
Figura 33	–	Resposta controlada sobre uma planta genérica do primeiro grau.	54
Figura 34	–	Teste do modelo RNA (Laço Aberto).	56
Figura 35	–	Regressão do erro médio quadrático.	57

Figura 36 – Desempenho do controlador proporcional no experimento I.	58
Figura 37 – Desempenho do controlador <i>fuzzy</i> no experimento I.	58
Figura 38 – Desempenho do controlador proporcional no experimento II.	60
Figura 39 – Desempenho do controlador <i>fuzzy</i> no experimento II.	61
Figura 40 – Desempenho do controlador <i>fuzzy</i> no experimento III.	62

Lista de tabelas

Tabela 1	– Exemplificação de conjuntos <i>fuzzy</i> e graus de pertinência.	33
Tabela 2	– Dados para exemplo de inferência <i>fuzzy</i>	35
Tabela 3	– Especificações do transdutor de vazão Incontrol.	40
Tabela 4	– Parâmetros da centrífuga.	40
Tabela 5	– Parâmetros do motor.	41
Tabela 6	– Parâmetros para a criação da rede neural artificial.	46
Tabela 7	– Definição do conjunto erro.	50
Tabela 8	– Definição do conjunto derivada do erro.	50
Tabela 9	– Definição do conjunto de saída.	51
Tabela 10	– Desempenho do modelo RNA.	56
Tabela 11	– Avaliação do controlador proporcional no experimento I.	59
Tabela 12	– Avaliação do controlador <i>fuzzy</i> no experimento I.	59
Tabela 13	– Avaliação do controlador proporcional no experimento II.	61
Tabela 14	– Avaliação do controlador <i>fuzzy</i> no experimento II.	61
Tabela 15	– Avaliação do controlador <i>fuzzy</i> no experimento III.	62

Lista de abreviaturas e siglas

LENHS	Laboratório de Eficiência Energética e Hidráulica em Saneamento
SAA	Sistema de Abastecimento de Água
RNA	Rede Neural Artificial
MIT	Motor de Indução Trifásico
CC	Corrente Contínua
CA	Corrente Alternada
LM	Levenberg-Marquardt
NA	Neurônio Artificial
CLP	Controlador Lógico Programável
EQM	Erro Médio Quadrático
MSE	Mean squared error

Lista de símbolos

β	Letra grega Beta
η	Letra grega Eta
μ	Letra grega Mi
Σ	Letra grega maiúscula Sigma
φ	Letra grega minúscula Fi
ρ	Letra grega minúscula Rô

Sumário

1	INTRODUÇÃO	15
1.1	Motivação	15
1.2	Objetivos	16
1.2.1	Objetivos Gerais	16
1.2.2	Objetivos Específicos	16
1.3	Metodologia adotada	17
1.4	Organização do trabalho	17
2	FUNDAMENTAÇÃO TEÓRICA	19
2.1	Sistema de abastecimento de água	19
2.1.1	Sensores de vazão	20
2.1.2	Inversores de frequência	22
2.2	Redes neurais artificiais	24
2.2.1	Estrutura de uma rede neural artificial	25
2.2.2	Treinamento	26
2.3	Sistemas de Controle	28
2.3.1	Controle proporcional	28
2.4	Lógica <i>fuzzy</i>	30
2.4.1	Controle <i>fuzzy</i>	30
2.4.2	Conjuntos <i>fuzzy</i>	32
2.4.3	Regras <i>fuzzy</i>	34
2.4.4	Inferência <i>fuzzy</i>	34
3	MATERIAIS E MÉTODOS	38
3.1	Bancada experimental	38
3.2	Modelo RNA	42
3.2.1	Caracterização da bancada	43
3.2.2	Montagem do banco de dados (tabelas)	44
3.2.3	Criação da rede neural artificial	45
3.3	Sintonização do controlador proporcional	46
3.4	Implementação do controlador Fuzzy	48
3.4.1	Definição do domínio	49
3.4.2	Construção dos conjuntos	49
3.4.3	Definição das regras	53
3.4.4	Avaliação	54

4	RESULTADOS	55
4.1	Desempenho do modelo RNA	56
4.2	Experimento I	58
4.3	Experimento II	60
4.4	Experimento III	62
5	CONSIDERAÇÕES FINAIS	63
6	REFERÊNCIAS BIBLIOGRÁFICAS	64
	 APÊNDICES	 68
	APÊNDICE A – CÓDIGOS RNA	69
	APÊNDICE B – CÓDIGOS <i>FUZZY</i>	74

1 Introdução

Atualmente intervenções na tubulação hidráulica de abastecimento de água é feita de maneira bastante invasiva, causando transtornos no trânsito local e comprometendo o abastecimento de água durante esta intervenção, seja para reparos ou para ampliações. Por isso, vemos que a modelagem de sistemas é uma ferramenta que pode contribuir para o planejamento desse tipo de intervenção, prevendo falhas ou ajudando a projetar sistemas mais robustos.

1.1 Motivação

A Secretaria Nacional de Saneamento Ambiental, do Ministério das Cidades, é o órgão federal responsável pelo planejamento, formulação e implementação da política setorial de saneamento básico no Brasil, o que inclui o setor de abastecimento de água. Contudo, outras competências de repasse de recursos para iniciativas de saneamento estão estabelecidas no âmbito federal, como a Fundação Nacional de Saúde, do Ministério da Saúde, que atua em ações de abastecimento de água em municípios com população de até 50.000 habitantes, (PEREIRA, 2014).

A indústria da água no Brasil vem tentando acompanhar o desenvolvimento tecnológico dos demais setores industriais, onde a presença da automação é uma realidade. A automação pode ser definida como qualquer aplicação de técnica computadorizada ou mecânica para diminuir ou suprimir a interferência humana em qualquer processo. Mecanismos automáticos podem ser adotados tanto no auxílio de controle de processos simples como no controle automático de grandes sistemas.(GOMES, 2010)

Há muitos séculos surgiu à necessidade de se medir vazão, devido à canalização de água para consumo doméstico, pois a administração pública necessitava de um sistema para medir a taxa de consumo líquido. Com o passar do tempo e com o surgimento de indústrias, aumentou a demanda por medição de fluidos em geral, tornando esse sistema de fundamental importância no setor. Os medidores de vazão de obstrução, embora antigos, ainda são amplamente utilizados para a medição de vazão, dentre eles podemos citar o tubo de Venturi, sendo este um excelente medidor com característica principal de baixa perda de carga importada ao sistema (SAVIONEK; KARDELI; RATHKE, 2010).

O controle e automação de processos são áreas da engenharia que aplicam técnicas variadas para estabilização de variáveis a níveis operacionais (CASTRUCCI; BITTAR; SALES, 2011). Essas técnicas, quando bem aplicadas, oferecem várias vantagens como a melhora na qualidade dos produtos, redução de perdas por defeitos durante o processo,

prolongamento da vida útil de equipamentos, menor risco de acidentes ocupacionais e atenuação da carga de trabalho do operador (CAMPOS; TEIXEIRA, 2010).

As tecnologias disponíveis para a implementação do controle automático se tornam cada vez mais modernas e seguras. Um exemplo são as vias de comunicação, em que as informações inicialmente eram transmitidas de forma analógica através de correntes numa faixa de 4 a 20mA, e aos poucos vêm sendo substituídas por tecnologias digitais, como o *profibus* e o *fieldbus*, que são mais seguras, a prova de falhas e permitem transmissão de maior número de dados. Outro componente essencial em um processo industrial são os controladores, que recebem os dados de área, os processam e decidem quais elementos e o quanto devem atuar para que seja atingido o objetivo (CAMPOS; TEIXEIRA, 2010).

Neste contexto, a identificação e modelagem de sistemas é muito importante, pois permitem a implementação de controladores mais robustos e eficientes. Sem um modelo, o projeto de um controlador se dá por tentativa e erro, que torna o processo mais complexo e demorado, visto que a sintonia de um controlador sem informações da dinâmica torna a tarefa muito mais difícil (AGUIRRE, 2007).

1.2 Objetivos

A seguir, são descritos os objetivos gerais e os objetivos específicos que nortearam a execução deste trabalho.

1.2.1 Objetivos Gerais

Desenvolvimento de um modelo de sistema de abastecimento de água baseado em redes neurais artificiais (RNA), como também a realização do controle deste modelo por meio de um controlador inteligente baseado em lógica *fuzzy*, ou nebulosa.

1.2.2 Objetivos Específicos

- Caracterização de uma bancada experimental para desenvolvimento de um modelo não-lienar baseado em RNA;
- Desenvolvimento de um controlador *fuzzy* baseado em regras associadas à experiência do operador do sistema;
- Análise do desempenho do controlador *fuzzy* sobre o modelo RNA, por meio da análise do erro médio quadrático, do tempo de acomodação e do erro em regime estacionário;

- Verificação da robustez de um controlador proporcional, em relação ao controlador *fuzzy*.

1.3 Metodologia adotada

Foi realizada uma revisão bibliográfica a partir dos materiais já publicados tais como livros, revistas, artigos científicos, trabalho de conclusões de cursos e dissertações de mestrado.

Foi realizada a calibração dos transdutores de vazão.

Utilizou-se um sistema supervisório próprio do Laboratório de Eficiência Hidráulica e Energética em Saneamento (LENHS) para a aquisição dos dados de interesse.

Sintonização do controlador proporcional por meio do métodos da resposta ao degrau de Ziegler-Nichols.

Foi realizada a caracterização da bancada por meio do registro dos valores de frequência do inversor e dos valores de vazão adquiridos por meio do sistema supervisório.

E por ultimo foi utilizado o *software Matlab* para a implementação do controlador *fuzzy* e do modelo da bancada baseado em RNA.

1.4 Organização do trabalho

Esta seção irá abordar a forma como este relatório de atividade está dividido.

No Capítulo 1 é explanada a motivação do trabalho sendo apurados os objetivos gerais e os objetivos específicos, bem como a organização do trabalho.

No Capítulo 2 está localizada a fundamentação teórica, a qual é base para os experimentos e modelos desenvolvidos neste trabalho.

O Capítulo 3 tem a descrição da bancada experimental que foi utilizada para a realização dos experimentos, tem os procedimentos utilizados na implementação do modelo baseado em RNA, é apresentado o método utilizado na sintonização do controlador proporcional e por fim mostra a construção do controlador *fuzzy*.

Em seguida, o Capítulo 4 mostra o desempenho do modelo desenvolvido baseado em RNA, e são realizados dois experimentos onde se busca investigar, por meio destes, a eficacia do controlador *fuzzy* sobre o modelo criado.

E feita uma análise sobre os resultados adquiridos, no Capítulo 5, como também sugestões de novos trabalhos sobre este tema.

Em seguida, estão listados os referenciais bibliográficos da fundamentação teórica utilizada para a realização deste trabalho.

E por fim, o Apêndice A, onde está o código utilizado na construção do modelo baseado em RNA. E o Apêndice B, o qual possui listadas as regras utilizadas no controlador *fuzzy*.

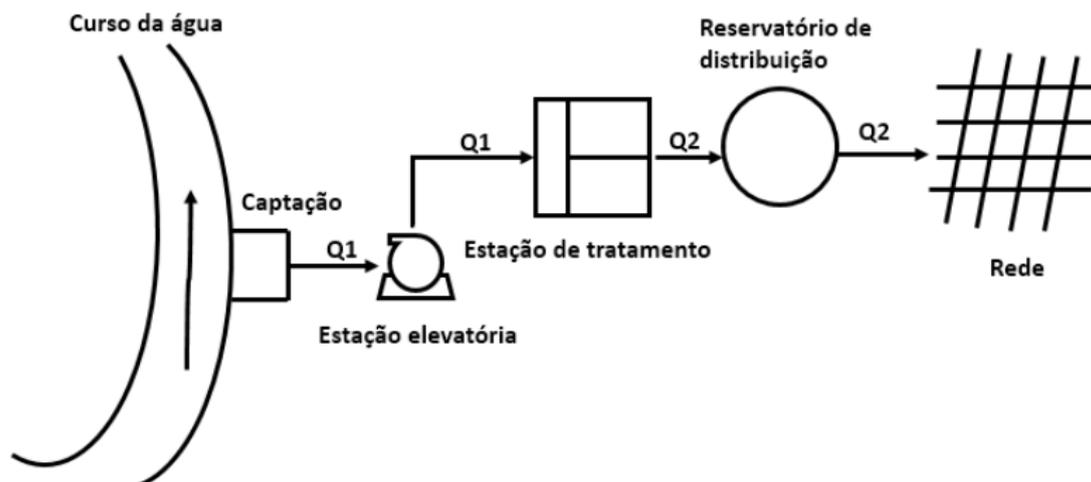
2 Fundamentação teórica

Neste capítulo será exposta a fundamentação teórica associada ao assunto abordado neste trabalho. Na Seção 1 será abordado o conceito de sistemas de abastecimento de água (SAA), e alguns equipamento de que são fundamentais para lidar com SAA's, tais como sensores de vazão e inversores de frequência para o controle de velocidade de motores de indução. Na Seção 2 serão apresentados conceitos sobre o desenvolvimento de redes neurais artificiais. Na Seção 3 é mostrado o princípio de funcionamento de um controlador proporcional. E por ultimo, na Seção 4, é abordada a lógica *fuzzy*.

2.1 Sistema de abastecimento de água

Os sistemas de abastecimento de água são compostos, basicamente por um ponto de captação da água bruta do meio ambiente, em seguida há um tratamento adequado para torná-la potável e, por fim, distribui-se até os consumidores em quantidade satisfatória para atender a demanda de consumo (HELLER, 2006), e têm a finalidade de conduzir e distribuir água em quantidade e qualidade aceitável. Compreende todo conjunto de processos de obtenção de água até sua utilização por parte dos clientes. O diagrama deste tipo de sistema está ilustrado na Figura 1.

Figura 1 – Esquemático de um sistema de abastecimento de água.



Fonte: Adaptado de Tsutiya (2004)

Sendo assim, (HELLER, 2006) define que:

- **Captação:** etapa onde se retira a água da fonte que alimenta o sistema de abastecimento de água, onde esta fonte é o manancial;
- **Estação elevatória:** estas são usadas quando a água precisa atingir pontos mais altos;
- **Estação de tratamento:** consiste em uma série de processos de filtrações e químicos para que a água bruta se torne potável;
- **Rede:** trata-se do conjunto de peças especiais destinadas a conduzir a água até os pontos de torneiras das instalações prediais, ou os pontos de consumo público, sempre de forma contínua e segura;
- **Adução:** estágio de transporte da água de uma unidade do sistema a outro, na Figura 1, é representada pelas flechas que interconectam as unidades do sistema.

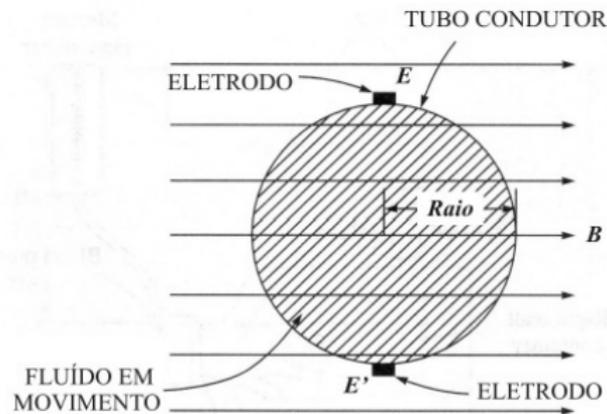
Cada elemento do sistema de abastecimento de água deve estar de acordo com a NBR 12211/1992, a qual determina a metodologia para a elaboração dos estudos de concepção destes sistemas. O atendimento aos pontos desta normal é crucial para o correto funcionamento do sistema, e por conseguinte, evitar superdimensionamento que acarretaria prejuízo financeiro. E ainda, há o risco do subdimensionamento, o qual acarreta dano na prestação de serviço aos consumidores.

2.1.1 Sensores de vazão

A escolha de um medidor de vazão pode ser algo complexo, devido a grande quantidade de técnicas de medição presentes hoje e as características particulares que cada uma apresenta. Por isso, antes de determinar um medidor de vazão, é preciso verificar quais são as condições de funcionamento que o equipamento deve apresentar, tais como: precisão exigida, faixa de medida, custo, facilidade de leitura e tempo de vida em serviço (FOX; MCDONALD, 1998). Entre os métodos mais comuns podemos citar: Molinetes, Magnéticos e Ultrassônicos.

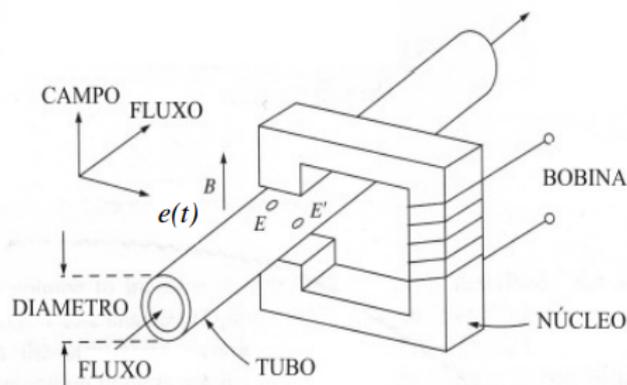
A Figura 2 apresenta um campo magnético (B) atuando no sentido perpendicular a um tubo condutor por onde passa um fluido e uma tensão elétrica, $e(t)$, é induzida no eletrodos (E) e (E'): Lei de Faraday. A Figura 3 apresenta o dispositivo, e com uma bobina pode-se gerar um campo magnético (B) fixo ou variável no tempo (CARR, 1998). Uma tensão elétrica, diretamente proporcional a velocidade do fluxo é gerada nos eletrodos (E) e (E').

Figura 2 – Tubo condutor de fluido, Campo magnético, Eletrodos.



Fonte: Adaptado de (CARR, 1998)

Figura 3 – Medidor eletromagnético, Campo magnético, Eletrodos.



Fonte: Adaptado de (CARR, 1998)

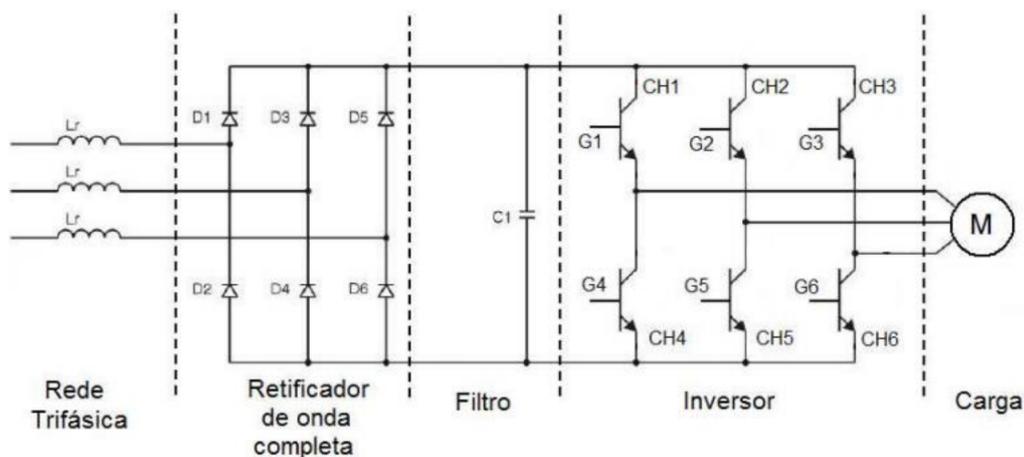
A técnica eletromagnética para medição de vazão é uma das mais flexíveis e mais universais dentre os métodos disponíveis. Sua perda de carga é equivalente a de um trecho reto de tubulação, já que não possui qualquer obstrução. É virtualmente insensível à densidade e a viscosidade do fluido de medição. Medidores magnéticos são ideais para medição de produtos químicos altamente corrosivos, fluidos com sólidos em suspensão, lama, água, polpa de papel. Sua aplicação estende-se desde saneamento até indústrias químicas, papel e celulose, mineração e indústrias alimentícias. A única restrição, em princípio é que o fluido tem que ser eletricamente condutor. Fluidos com propriedades magnéticas adicionam erros de medição.

2.1.2 Inversores de frequência

O inversor de frequência foi utilizado para realizar o controle da velocidade do motor trifásico de indução.

O ajuste de velocidade dos motores de indução trifásicos (MIT) para alcançar melhor eficiência no acionamento depende da estratégia de controle de um conversor CA-CC-CA (FRANCHI, 2011). Este conversor CA-CC-CA, é o elemento que encontra-se entre a rede elétrica de entrada, a qual possui tensão e frequência fixas, e o motor que aciona o equipamento, salvo a existência de sistemas supervisórios. Basicamente a conversão consiste de três estágios: um retificador de entrada para converter a corrente alternada (CA) em corrente contínua (CC), um barramento CC para armazenar e filtrar o nível de tensão de corrente contínua, e um inversor na saída para converter a forma de onda contínua em alternada novamente. O diagrama básico de um inversor é mostrado na Figura 4. Já na saída, os níveis de tensão e frequência são ajustáveis, (GRIPP, 2005).

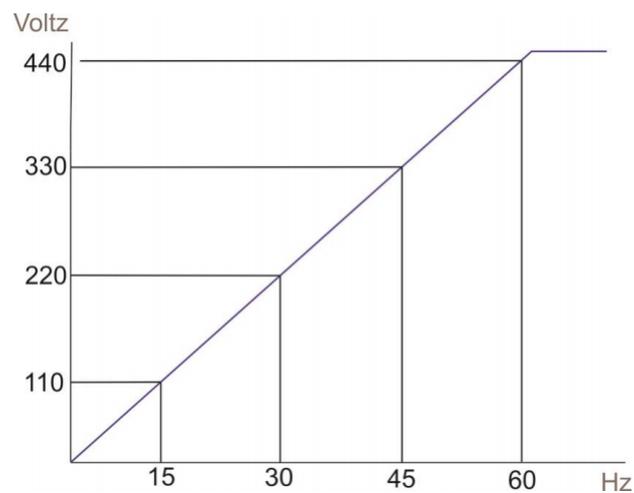
Figura 4 – Esquema simplificado de um inversor de frequência.



Fonte: Adaptado de (SOCIESC, 2015).

Dentre as inúmeras aplicações com inversores de frequência, o controle de partida do motor e a variação de velocidade rotórica são os objetivos mais buscados. Os sistemas de controle de velocidade de menor complexidade que utilizam apenas o método de manter constante a relação tensão/frequência é o controle escalar, porém, algumas limitações são observadas neste tipo de controle. Em situações que exija rápida resposta de aceleração, controle em malha fechada e variação de velocidade em função da variação da carga, o acionamento com controle escalar não é indicado (CAMARGO, 2011). Já que, o inversor com controle escalar tem uma relação de tensão/frequência pré-definidos para quaisquer variações de velocidade, dessa maneira caracteriza um desempenho praticamente linear, conforme a Figura 5.

Figura 5 – Comportamento de um controle escalar de velocidade.



Fonte: Adaptado de (AZEVEDO & MENDES, 2008).

A tecnologia do inversor com controle vetorial, por outro lado, é uma solução mais abrangente para o controle de velocidade e torque de MIT. Suas características são : resposta dinâmica rápida, precisão de velocidade e possibilidade de torque elevado para mínimas rotações, (OLIVEIRA, MORAIS, & PIRES, 2012). Tais características do controle vetorial deixam essa tecnologia muito mais atraente em aplicações específicas. Dessa forma, o controle vetorial apresenta-se mais preciso para uma ampla variação de velocidade, e de cargas no eixo (QUEIROZ, 2008).

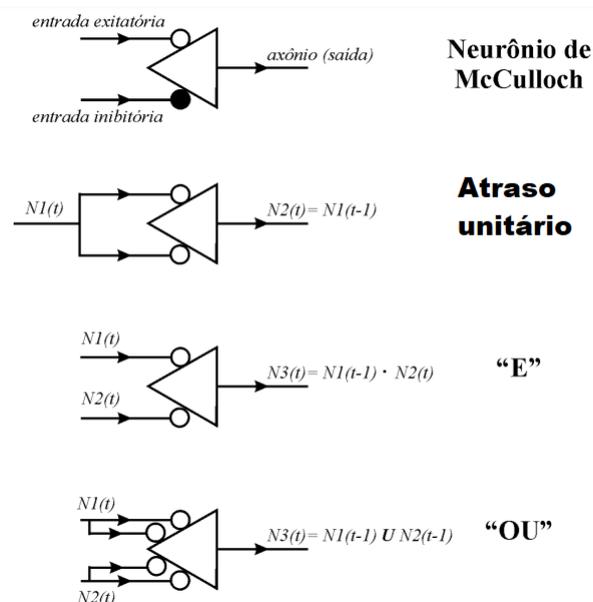
2.2 Redes neurais artificiais

Uma Rede Neural Artificial, RNA, é um Sistema Dinâmico Complexo representado por um grafo arco rotulado em que cada vértice é um Neurônio Artificial (NA). Nesta definição, rótulos são, naturalmente, valores numéricos. Eles correspondem aos valores das conexões entre os diferentes neurônios, (BARRETO 2002).

Conforme afirma, (FINOCCHIO, 2014), de acordo com as diversas estruturas neurais e algoritmos de aprendizagem propostos por vários pesquisadores, as redes neurais possuem certas características exclusivas de sistemas biológicos. Sistemas de computação baseados em redes neurais têm a capacidade de receber ao mesmo tempo várias entradas e distribuí-las de maneira organizada. Geralmente, as informações armazenadas por uma rede neural são compartilhadas por todas as suas unidades de processamento. Essa característica contrasta com os atuais esquemas de memória dos computadores convencionais, onde a informação fica confinada em um determinado endereço.

O primeiro modelo matemático para uma rede neural, proposto por McCulloch e Pitts, era simples diante das informações disponíveis naquela época sobre o funcionamento elétrico de uma célula nervosa, Figura 6. Era um dispositivo binário, sendo que a saída do neurônio poderia ser pulso ou não pulso (ativo ou não), e as várias entradas tinham um ganho arbitrário, podendo ser excitatórias ou inibitórias. Para se determinar a saída do neurônio, calculava-se a soma ponderada das entradas com os respectivos ganhos como fatores de ponderação, excitatórios ou inibitórios. Se o resultado atingisse um certo limiar, a saída do neurônio era pulso (ativo), caso contrário, não pulso (não ativo).

Figura 6 – O neurônio de McCulloch e implementações de algumas funções booleanas.



Fonte: Adaptado de (KOVÁCS, 1996).

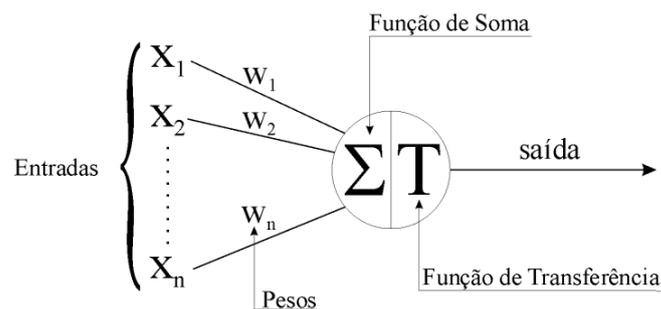
2.2.1 Estrutura de uma rede neural artificial

Um dos atributos de grande importância do neurônio artificial é o peso. Os pesos, também conhecidos por pesos sinápticos, são representados pela letra W (*weight*), vide Figura 7, e representam o grau de importância que determinada entrada possui em relação àquele determinado neurônio.

O valor do peso é alterado em função da intensidade do sinal de entrada, e dessa forma, o peso muda seu valor representativo para a rede, processo de aprendizagem. Deduz-se que quanto mais estimulada for a entrada, mais estimulado será o peso correspondente, e quanto mais for estimulado um peso, mais significativo e influente o mesmo será para o resultado do sinal de saída do respectivo neurônio.

O sinal de excitação do neurônio é resultante do somatório do produto dos sinais de entrada, X_i , representados por um vetor $[x_1, x_2, \dots, x_n]$, pelo peso do neurônio, ou seja, $\sum_{x=0}^n X_i \cdot W_i$. Após esta ponderação, os sinais passam a ser chamados de entradas ponderadas.

Figura 7 – O neurônio artificial.



Fonte: Adaptado de (TAFNER, 1996).

Em seguida o neurônio irá verificar se o valor resultante da soma entre o produto dos sinais de entrada pelos respectivos pesos atingiu ou não um valor predeterminado, denominado de limiar. Se atingiu, o mesmo valor é repassado adiante através da saída. Caso contrário, o sinal será transferido. Esse processo de verificação é chamado de função de transferência, o qual também é conhecido como limiar lógico.

Em alguns modelos simples de redes neurais artificiais, a função de ativação pode ser a própria função de soma das entradas ponderadas do neurônio. Já em modelos mais elaborados, a função de ativação pode possuir um processamento atribuído, o qual pode ser, por exemplo, o uso de um valor prévio de saída como uma entrada para o próprio neurônio, servindo de auto-excitação para o mesmo, (TAFNER, 1996).

O valor de saída do neurônio será produzido após a chamada da função de ativação,

seguido pela função de transferência.

2.2.2 Treinamento

Muitas das leis de aprendizado estão em uso, e a maioria delas são apenas variações da mais difundida lei, que é a lei de Hebb. As pesquisas em torno das funções de aprendizado continuam, e busca-se aproximar cada vez mais estes modelos dos padrões biológicos.

Abaixo, são apresentadas algumas das mais conhecidas e difundidas leis de aprendizado, (DACS, 2000).

- **Princípio de aprendizagem de Hebb** - a primeira e indiscutivelmente a mais conhecida regra de aprendizado, foi apresentada pelo biólogo e psicólogo Donald Hebb. A descrição da mesma foi exposta em seu famoso livro *The Organization of Behavior* em 1949. A regra parte do seguinte pressuposto. Se um neurônio recebe uma entrada, proveniente de outro neurônio, e ambos estão ativos, isto é, possuem o mesmo sinal, os pesos entre os neurônios precisam ser excitados;
- **A Lei de Hopfield** - é praticamente similar ao princípio de aprendizado de Hebb com apenas uma exceção: a Lei de Hopfield especifica a magnitude da excitação ou da inibição. Se a saída desejada e o sinal de entrada estão ambos ativos ou inativos, os pesos são incrementados pela taxa de aprendizado, caso contrário, são decrementados;
- **A Regra Delta de Widrow** - esta regra é uma variação um pouco além do princípio de aprendizado proposto por Hebb. A regra delta apresentada por Widrow é a mais comumente utilizada nos dias atuais. Esta regra se baseia na simples ideia da contínua modificação da intensidade e a importância das conexões de entrada; reduzindo consideravelmente a diferença entre o valor de saída desejado e o atual valor de saída da unidade de processamento, no caso, o neurônio artificial. A regra delta altera os pesos sinápticos de modo que minimize o erro quadrático da rede, trabalhando da seguinte forma: o erro calculado na saída é transformado pela derivação da função de transferência, e conseqüentemente, usado para ajustar os pesos de entrada da camada prévia da rede, ou seja, o erro é retro-propagado às camadas anteriores, sendo uma de cada vez. O processo de retro-propagação dos erros da rede continua até que a primeira camada da rede seja alcançada. Os tipos de redes chamadas de *feedforward* e *backpropagation* derivam seus nomes dos métodos adotados pelo processamento do erro. Quando se usa a regra delta, é importante assegurar que o conjunto de dados de entrada está disposto de forma aleatória ou gerado de forma randômica. Uma vez mal organizado, este conjunto de treinamento pode conduzir

a rede a não convergência da precisão desejada, impossibilitando o aprendizado do problema em questão;

- **A Lei de Aprendizado de Teuvo Kohonen** - desenvolvida por Teuvo Kohonen, a mesma foi inspirada nos sistemas biológicos, onde os elementos competem entre si por uma oportunidade de aprender, ou atualizar/ajustar seus respectivos pesos. A unidade de processamento que possuir o melhor sinal de saída será considerado o mais apto, e conseqüentemente passarão a capacidade e privilégio de inibir os ajustes sinápticos de seus concorrentes e excitar seus vizinhos. Somente a unidade apta e seus respectivos vizinhos terão permissão para ajustar seus pesos. A abrangência e a possível área que uma unidade vizinha pertence está relacionada ao período de treinamento da rede. O paradigma atual usa o seguinte procedimento: é formada uma grande área de vizinhança e a medida que ocorre o processo de treinamento, há uma seleção e conseqüentemente um estreitamento da mesma.

2.3 Sistemas de Controle

Para (NISE, 2002), união de subsistemas e processos (ou plantas) construídos com a finalidade de se obter uma saída com desempenho desejado, dada uma entrada específica, caracteriza um sistema de controle. Em (NISE, 2002), encontramos indicadas quatro razões principais para o desenvolvimento de sistemas de controle, são elas: amplificação de potência, controle remoto, conveniência da forma da entrada e compensação de perturbações.

Assim sendo, nos dois tópicos seguintes, serão abordadas dois modelos de controlador que serão aplicados ao modelo RNA desenvolvido. O primeiro, trata-se de um controlador proporcional clássico. Já o segundo, consiste em um controlador inteligente baseado em lógica *fuzzy*.

2.3.1 Controle proporcional

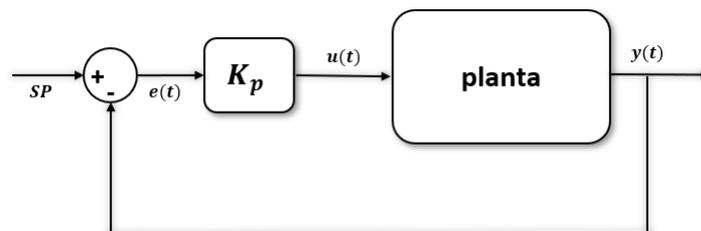
Neste tipo de controlador a relação entre a sua saída, $y(t)$, e o sinal de erro, $e(t)$, é dada pela Equação 2.1, abaixo.

$$u(t) = K_P \cdot e(t) \quad (2.1)$$

em que K_P é denominado ganho proporcional, (OLIVEIRA, J. P. B. M, 1994). E 'SP' representa o valor que se deseja ter na saída.

Um controlador proporcional consiste basicamente em um amplificador com ganho ajustável, conforme pode ser observado na Figura 8.

Figura 8 – Diagrama de blocos de um controlador proporcional.



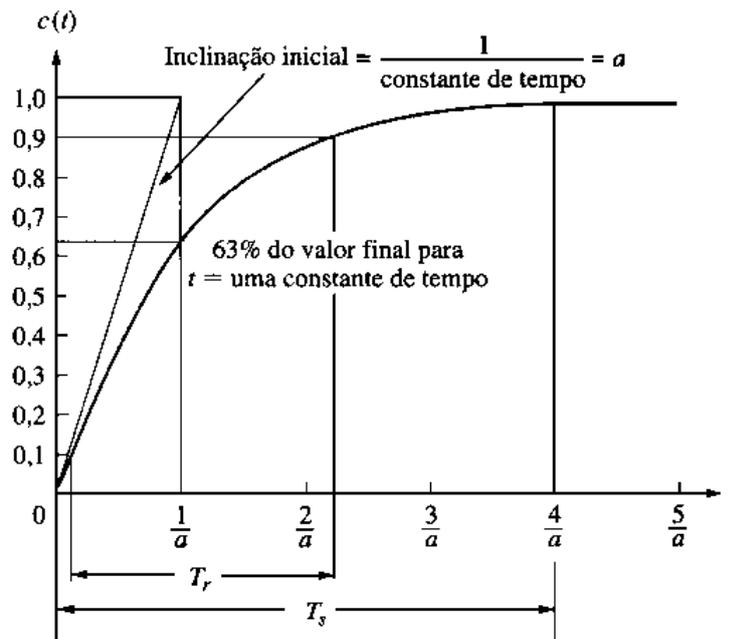
Fonte: Autor

Uma característica importante desta ação de controle é a existência de um erro residual permanente pelo fato do sistema não possuir um integrador. O erro estacionário que é dependente de K_P e da planta, pode ser minimizado por um aumento de K_P . No entanto deve-se notar que o aumento deste parâmetro conduz a um aumento do sobrevalor percentual, e eventualmente pode conduzir a instabilidade, (LOURENÇO, 1997).

A fim de determinar as características dinâmicas do sistema, tanto para o regime transitório, quanto no regime permanente, pode-se aplicar um degrau unitário na entrada e observar a saída. Serão abordadas a seguir as respostas de primeira e segunda ordem.

Uma resposta típica de primeira ordem está ilustrada na Figura 9. Onde (NISE, 2002), define os seguintes parâmetros:

Figura 9 – Resposta de um sistema de primeira ordem.



Fonte: (NISE, 2002)

Onde:

- **Tempo de Subida (T_r):** é o tempo necessário que a forma de onda da resposta vá de 10% a 90% do seu valor final;
- **Tempo de Acomodação (T_s):** é o tempo em que a resposta alcance e fique operando na faixa de 2% do seu valor final;
- **Constante de Tempo (t):** é o tempo necessário para que a resposta alcance 63% do valor do seu valor final.;

2.4 Lógica *fuzzy*

O termo *fuzzy* na língua inglesa pode ter mais de um significado, de acordo com o contexto de interesse, porém o conceito característico desta palavra gira em torno de: vago, indistinto, incerto. Ainda não há uma unanimidade em relação as traduções para o português, sendo que 'nebuloso' ou 'difuso' são, popularmente, os termos mais aceitos quando são realizadas traduções.

Nas próximas seções serão apresentados os principais conceitos concernentes à Lógica *Fuzzy* e aos conjuntos *fuzzy*, que fundamentam essa teoria, bem como as regras de inferência presentes nesse contexto. Posteriormente, são apresentadas as aplicações de maior notabilidade da Lógica *Fuzzy*, mais especificamente relacionadas ao contexto da Inteligência Artificial, designando-se uma atenção especial à utilização da Lógica *Fuzzy* como formalismo de representação do conhecimento para ser representado em sistemas especialistas, construindo, assim, os sistemas especialistas *fuzzy*. Por fim, são elencados aspectos positivos e limitações que essa lógica não-clássica apresenta.

2.4.1 Controle *fuzzy*

Tomando como exemplo a frase a seguir:

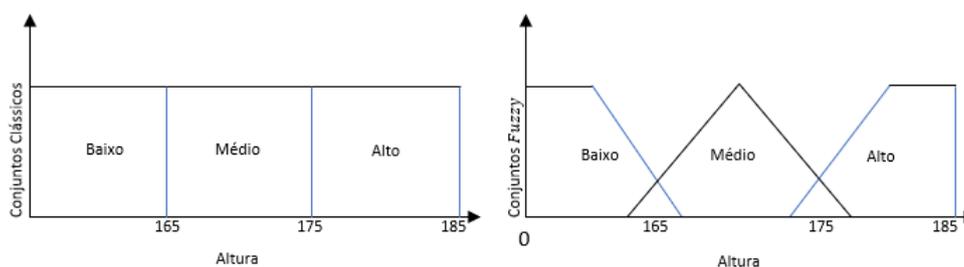
'Embora o transformador esteja um pouco carregado, pode-se utilizá-lo por um tempo.'

Nessa frase existem dois termos (um pouco e um tempo) bastante subjetivos e difíceis de representar. Para um especialista que esteja conversando com outro especialista, o entendimento seria normal; no entanto, durante o processo de aquisição, fica bastante complicado para o engenheiro de conhecimento entender e representar esse tipo de conhecimento. Dessa forma, uma maneira de tentar solucionar o processo de representação de conhecimento impreciso é através da Lógica *Fuzzy*. A Lógica *Fuzzy* tem como principal objetivo a modelagem computacional do raciocínio humano, impreciso, ambíguo e vago.

A Lógica *Fuzzy* foi primeiramente introduzida em 1930 pelo filósofo e lógico polonês Jan Lukasiewicz. Através do estudo de termos do tipo alto, velho e quente, ele propôs a utilização de um intervalo de valores $[0,1]$ que indicaria a possibilidade que uma declaração fosse verdadeira ou falsa. Em 1937, o filósofo Max Black propôs a idéia de que continuidade descrevia graus. Ele definiu o primeiro conjunto *fuzzy* e descreveu algumas idéias básicas de operações com conjuntos *fuzzy*. Em 1965, Lofti Zadeh publicou o artigo *Fuzzy Sets*, que ficou conhecido como a origem da Lógica *Fuzzy*. Na realidade, Zadeh redescobriu a idéia de fuzzyficação, identificou e explorou tal conceito, assim como lutou por ele. Portanto, Zadeh ficou e ainda é conhecido como o "mestre" da Lógica *Fuzzy*.

A teoria clássica de conjuntos permite o tratamento de classes de objetos e suas inter-relações em um universo definido. Nessa teoria, a pertinência de um dado elemento com relação a um conjunto refere-se ao fato de tal elemento pertencer ou não a esse conjunto. De forma ilustrativa, considere o gráfico apresentado no lado esquerdo da Figura 10, que representa um exemplo típico da teoria clássica e descreve a altura de uma pessoa através de três conjuntos: baixo, médio e alto. Nesse exemplo, dado um elemento x qualquer, o mesmo pertencerá a um dos conjuntos do gráfico; por exemplo, se $x = 1,65$, então x pertence ao conjunto médio e não aos demais, ou seja, um elemento pertence ou não a um determinado conjunto e, além disso, tal elemento não pertence a mais de um conjunto.

Figura 10 – Perspectivas de Conjuntos clássicos e Conjuntos *fuzzy*.



Fonte: Autor.

Ao contrário da Lógica convencional, a Lógica *Fuzzy* utiliza a idéia de que todas as coisas admitem (temperatura, altura, velocidade, etc.) graus de pertinências. Com isso, a Lógica *Fuzzy* tenta modelar o senso de palavras, tomada de decisão ou senso comum do ser humano. Ainda tomando como exemplo o gráfico da Figura 10, dados dois elementos $x_1 = 1,69$ e $x_2 = 1,71$, se a Lógica clássica for utilizada, esses dois elementos pertencem a classes diferentes, x_1 pertencendo à classe médio e x_2 à classe alto. No entanto, na realidade fica difícil de dizer que uma pessoa com 1,69 m e outra com 1,71 m pertencem a classes diferentes.

Por outro lado, na Lógica *Fuzzy*, tanto o x_1 quanto o x_2 têm graus de pertinências aos conjuntos *fuzzy* definidos, que podem variar de 0 a 1. Em outras palavras, enquanto que a tomada de decisão na teoria clássica seria como a da Equação 2.2, a da Lógica *Fuzzy* seria como a da Expressão 2.3, considerando um conjunto A e um elemento x com relação a esse conjunto.

$$\mu(x) = \begin{cases} 1 & \text{se, e somente se } x \in A. \\ 0 & \text{se, e somente se } x \notin A. \end{cases} \quad (2.2)$$

$$\mu(x) = \begin{cases} 1 & \text{se, e somente se } x \in A. \\ 0 & \text{se, e somente se } x \notin A. \\ 0 \leq \mu(x) \leq 1 & \text{se } x \text{ pertence parcialmente a } A. \end{cases} \quad (2.3)$$

Dessa forma, a Lógica *Fuzzy* pode ser considerada como um conjunto de princípios matemáticos para a representação do conhecimento baseado no grau de pertinência dos termos (graus de verdade). Como pode ser observado na Equação 2.3, o intervalo de pertinência é $[0,1]$, onde 0 significa que um elemento não pertence a um determinado conjunto, 1 significa completa pertinência ao conjunto, e valores entre 0 e 1 representam graus parciais de pertinências. Assim, na Lógica *Fuzzy*, um elemento pertence a um conjunto com um certo grau de pertinência, fazendo com que uma determinada sentença possa ser parcialmente verdadeira e parcialmente falsa. Além do mais, um mesmo elemento pode ter graus de pertinências diferentes de 0 para mais de um conjunto *fuzzy*.

Para entender melhor o funcionamento da Lógica *Fuzzy*, ainda considerando o exemplo descrito acima, a altura de uma pessoa pode ser descrito através de conjuntos *fuzzy* (explicados com maiores detalhes nas próximas seções). Utilizando os mesmos elementos do exemplo acima, $x_1 = 1,69$ e $x_2 = 1,71$, percebe-se que ambos têm graus de pertinência 0 para o conjunto alto e graus de pertinência parciais para os conjuntos baixo e médio. É possível notar que, com a Lógica *Fuzzy*, não existe um limite abrupto que define os elementos que pertencem ou não a um determinado conjunto, como no caso dos conjuntos alto, médio e baixo. Por outro lado, os graus de pertinência dos elementos possuem variações suaves no intervalo real $[0,1]$, representando, assim, de forma mais realista, o conhecimento humano.

2.4.2 Conjuntos *fuzzy*

Na teoria clássica dos conjuntos, a transição de um elemento entre os conjuntos contidos em um universo é abrupta e bem definida (MARRO, 2010). Isso evita que ambigüidades apareçam e tornam a lógica mais simples. Ainda considerando o exemplo da utilização de conjuntos para separar pessoas pela altura, uma pessoa com 1,69 m seria considerada uma pessoa de altura mediana, se assim fosse definido, estando apenas nesse conjunto e em nenhum outro; já uma pessoa com 1,71 m faria parte do conjunto das pessoas altas, e somente deste. Todavia, na realidade, fica bem difícil ver que pessoas com uma diferença de altura tão mínima pertencem a conjuntos diferentes. Já para um elemento em um universo que contém conjuntos *fuzzy*, essa transição pode ser gradual (ROSS, 2005). Assim, pode-se concluir que os conjuntos *fuzzy* que classificam os elementos de um dado universo são menos rígidos do que aqueles utilizados na teoria clássica visto que eles admitem graus parciais de pertinência.

O primeiro passo na representação de conjuntos *fuzzy* é a escolha da função de pertinência. A escolha dessa função depende do problema a ser modelado e também da capacidade computacional disponível para processar o que se deseja. Funções não-lineares podem ser mais eficientes para problemas mais complicados, porém, elas demandam um poder computacional muito maior do que as funções lineares. Se o universo a ser trabalhado for curto, ou contínuo, torna-se bem mais simples a aplicação de uma função para separar adequadamente os elementos em conjuntos. Um exemplo para a altura seria o apresentado na Tabela 1.

Tabela 1 – Exemplificação de conjuntos *fuzzy* e graus de pertinência.

	alto			médio			baixo		
x	165	175	185	165	175	185	165	175	185
$\mu(x)$	0	0,5	1	0	1	0	1	0,5	0

Fonte: Autor.

Nesse caso, tem-se o grau de pertinência de cada um sendo analisado em relação aos centímetros. Uma pessoa com menos de 165 centímetros não pode ser considerada nem um pouco alta, assim como uma pessoa acima de 180 centímetros não é nada baixa.

Como as opções para a escolha de uma função de pertinência é praticamente infinita, vale a pena sempre gastar algum tempo nessa etapa do processo e com isso escolher a função mais adequada ao problema a ser modelado. A consulta a um especialista, no caso de sistemas especialistas, é sempre uma boa idéia.

Pode-se considerar uma variável linguística (ou *fuzzy*) como uma entidade utilizada para representar de modo impreciso – e, portanto, linguístico – um conceito ou uma variável de um dado problema. Uma variável linguística, diferentemente de uma variável numérica, admite apenas valores definidos na linguagem *fuzzy* que está utilizando-se dela. Por exemplo:

‘João é alto.’

A variável João está recebendo o valor alto, que é um dos conjuntos *fuzzy* definidos para esta variável. Os modificadores são termos ou operações que modificam a forma dos conjuntos *fuzzy* (ou seja, a intensidade dos valores *fuzzy*), podendo-se citar, por exemplo, os advérbios muito, pouco, extremamente, quase, mais ou menos, entre outros. Estes podem ser classificados em aumentadores, quando aumentam a área de pertinência de um conjunto *fuzzy*, ou, analogamente, diminuidores, quando diminuem a área de pertinência de um conjunto *fuzzy*.

2.4.3 Regras *fuzzy*

A expertise dos sistemas de inferência *fuzzy* é descrita de forma qualitativa em forma de regras nebulosas, estas são compostas pelos termos antecedente e o consequente, **SE** <antecedentes> **ENTÃO** <consequentes> e também das operações entre eles (NOGUEIRA, 2013).

As regras *fuzzy* são regras normais utilizadas para operar, da maneira correta, conjuntos *fuzzy*, com o intuito de obter consequentes. Para criar tais regras é preciso de um raciocínio coerente com o que se deseja manusear e obter. Para isso, este raciocínio deve ser dividido em duas etapas: (1) avaliar o antecedente da regra e (2) aplicar o resultado no consequente. Por exemplo, considerando a sentença se x é alto, então x é pesado seguindo os passos 1 e 2 acima, tem-se que para $x = 1,70$ m, deve-se, primeiramente, verificar o grau de pertinência da entrada para o conjunto ao qual se encaixa, alto, que é, para este caso, $\mu(x) = 0.5$. Como o grau de pertinência da entrada x é tal, então se deve passar este valor de pertinência para um $y = 80$ kg (por exemplo), pertencente ao conjunto pesado.

Para casos em que existam vários antecedentes, é preciso encontrar um grau de pertinência resultante de todos os dos antecedentes. Nos casos em que o conectivo entre os antecedentes seja e, deve-se utilizar métodos de combinação, contanto que o resultado não ultrapasse o valor de menor pertinência entre os antecedentes; um exemplo de método seria o mínimo das pertinências. Já nos casos em que o conectivo entre os antecedentes for ou, deve-se utilizar métodos de combinação, contanto que o resultado não seja menor que o maior grau de pertinência. O raciocínio é bem mais simples para casos em que existam vários consequentes, pois o grau de pertinência resultante será o mesmo para todos os consequentes.

2.4.4 Inferência *fuzzy*

A inferência *fuzzy* é um processo de avaliação de entradas com o objetivo de, através das regras previamente definidas e das entradas, obter conclusões utilizando-se a teoria de conjuntos *fuzzy*. Esse processo pode ser feito através de modelos de inferência, cuja escolha deve levar em consideração o tipo de problema a ser resolvido, obtendo-se assim um melhor processamento.

Portanto, Ortega (2001) aponta que o método de inferência mais comum e vastamente utilizado no controle de sistema é o Método de Mamdani, o qual foi escolhido para o presente trabalho.

O estilo de inferência Mandami foi criado pelo professor Ebrahim Mamdani da Universidade de Londres (Reino Unido) em 1975 no contexto do desenvolvimento de sistemas *fuzzy* baseando-se em regras de conjuntos *fuzzy* no intuito de representar experiências da vida real. Para a construção desse sistema, foi definido um processo de raciocínio dividido

em quatro passos: (1) fuzzyficação, (2) avaliação das regras *fuzzy*, (3) agregação das regras *fuzzy* e (4) defuzzyficação.

Para ilustrar cada uma das etapas, considere a análise de riscos num projeto. Nesse domínio, identificam-se três variáveis linguísticas (as duas primeiras de entrada e a última de saída), apresentadas na Tabela 2, bem como seus respectivos valores. Desta feita, quer-se estabelecer, sendo conhecidos um valor x de recurso monetário para o projeto e um número y de funcionários para trabalhar no mesmo, qual o risco z desse projeto.

Tabela 2 – Dados para exemplo de inferência *fuzzy*.

Recurso monetário x	
Valor linguístico	Notação
Inadequado	A1
Razoável	A2
Adequado	A3
Número y	
Valor linguístico	Notação
Pequeno	B1
Grande	B2
Risco z	
Valor linguístico	Notação
Baixo	C1
Normal	C2
Alto	C3

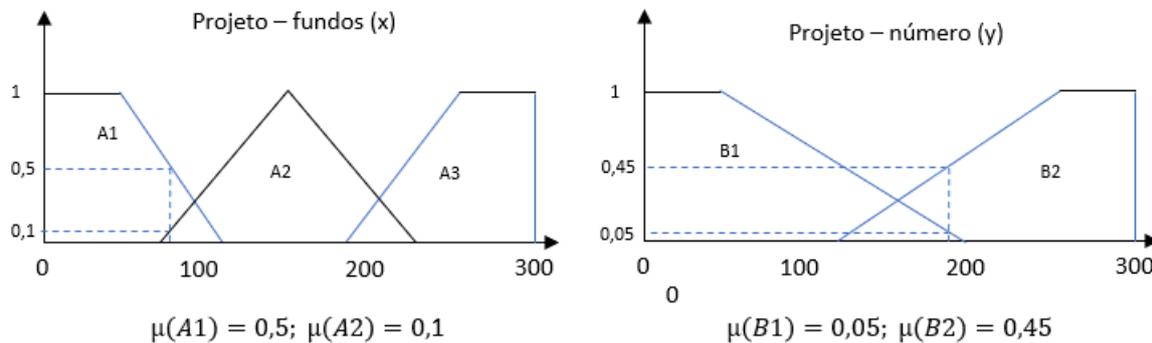
Fonte: Autor.

Fuzzyficação

Essa etapa obtém o grau de pertinência com que cada entrada pertence a cada conjunto *fuzzy*. Cada uma dessas entradas foi previamente limitada no universo de discurso em questão e associada a um grau de pertinência em cada conjunto *fuzzy* através do conhecimento do especialista. Então para obter o grau de pertinência de uma determinada entrada *crisp* basta buscar esse valor na base de conhecimento do sistema *fuzzy*.

Para o exemplo em questão, tem-se os conjuntos fuzzy e graus de pertinência para cada uma das variáveis de entrada, conforme mostrado na Figura 11.

Figura 11 – Grau de pertinência para diferentes conjuntos.



Fonte: Autor.

Avaliação das regras *fuzzy*

Depois de obter as entradas fuzzyficadas é só aplicá-las nos antecedentes obtendo assim o valor do conseqüente para cada regra. Para um antecedente composto, os operadores “e” e “ou” são utilizados para obter um único resultado; no caso do operador “ou” é utilizada a operação de união (pega o maior grau de pertinência), e, no caso do operador “e”, é utilizada a de interseção (pega o menor grau de pertinência). Depois de obter um único valor para o antecedente é necessário obter o valor do conseqüente através de um método de correlação dos mesmos. O método mais comum é conhecido como *clipped*, onde o conseqüente é “cortado” para o nível de valor verdade do antecedente da regra avaliada, ou seja, o valor obtido é simplesmente passado para o conseqüente dessa regra. Por exemplo, com base nos graus de pertinência e nas correlações entre as variáveis linguísticas, têm-se as regras

- 1: IF (x is A3 (0) or y is B1 (0.1)) THEN (z is C1 (0.1));
- 2: IF (x is A2 (0.2) and y is B2 (0.7)) THEN (z is C2 (0.2));
- 3: IF (x is A1 (0.5)) THEN (z is C3 (0.5)).

Agregação das regras *fuzzy*

Como o nome sugere, nessa etapa são agregadas todas as funções membro dos conseqüentes de cada regra em um único conjunto *fuzzy*.

Defuzzyficação

Para se obter uma saída numérica é necessário defuzzyficar a saída obtida na etapa anterior. O método de defuzzyficação mais comum é a técnica do centróide, que obtém o ponto onde uma linha vertical divide ao meio um conjunto agregado. A fórmula matemática que obtém esse ponto é expressa da seguinte forma Expressão 2.4. A precisão do método depende do intervalo escolhido, quanto maior mais impreciso, porém mais rápido de calcular.

$$COG = \frac{\sum_{x=a}^b \mu(x) \cdot x}{\sum_{x=a}^b \mu(x)} \quad (2.4)$$

Onde:

- COG: método de defuzzificação *Center of Gravity*;
- $\mu(x)$: Valor de pertinência;
- x: valor de entrada.

3 Materiais e Métodos

Neste capítulo, propõe-se a realização da modelagem de um sistema de bombeamento de água, representado por uma bancada do LENHS, Figura 12, dos componentes apresentados na figura foram utilizados o sensor de vazão FT_2, um conjunto motobomba, o qual é acionado por meio de um inversor de frequência, e aproximadamente 20 metros de tubulações, além de ser capaz de emular diversos cenários. Para a realização desta modelagem foram utilizadas técnicas de Inteligência Artificial baseadas em Redes Neurais Artificiais.

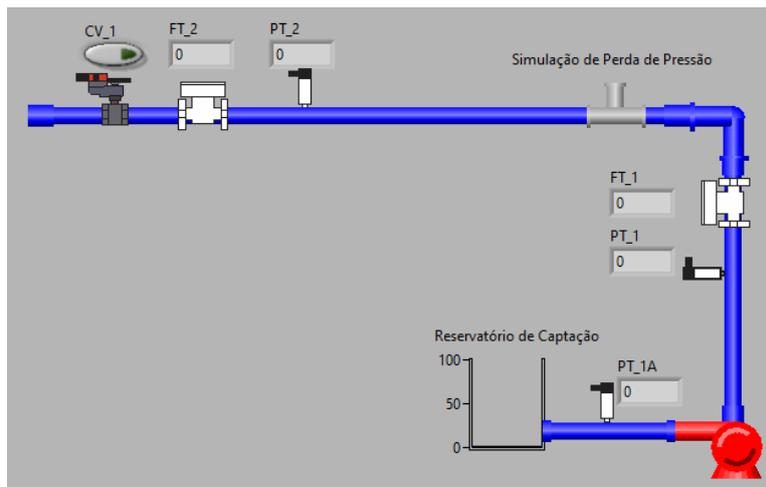
Também se propõe o desenvolvimento de um controle de vazão baseado em Lógica Fuzzy, objetivando-se a obtenção de um controlador robusto às diversas variações dos parâmetros do sistema, como exemplo, a variação de pressão.

Além disso, o monitoramento dos parâmetros do sistema foi realizado por um sistema de aquisição de dados.

3.1 Bancada experimental

O sistema no qual foram realizados os ensaios é representado na Figura 12. Esta bancada é composta por uma rede de tubulações com várias saídas de vazão de água, no entanto, para os propósitos deste trabalho foi criado um ambiente de um sistema simplificado composto por um reservatório de captação, que fornece água para um conjunto motobomba, e dele para um canal de tubulação, no qual tem em sua saída, sensores de vazão, pressão e uma válvula. A água bombeada retorna ao reservatório de captação por meio de canais.

Figura 12 – Representação da bancada experimental utilizada.



Fonte: Autor.

O monitoramento da vazão foi realizado por um transdutor de pressão eletromagnético, FT_2 , presente na saída do sistema, o qual é mostrado na Figura 13. As suas principais especificações podem ser observadas na Tabela 3. Para se ter uma medição livre de interferências, neste caso, é importante respeitar as recomendações exigidas pelo fabricante, ou seja, obedecer o comprimento mínimo de trecho reto, para se evitar as singularidades, a montante e a jusante do medidor.

Figura 13 – Transdutor de vazão VMS Pro.

Fonte: www.rucken.com.br

Tabela 3 – Especificações do transdutor de vazão Incontrol.

Fabricante	Incontrol
Modelo	VMS 038
Faixa de medição	1,24 a 40,8 m^3/h
Sinal de Saída	2 fios (4 a 20 mA)
Tempo de resposta	1 ms
Precisão	0,2%
Tensão de alimentação	24 V

Fonte: Catálogo do fabricante.

Figura 14 – Conjunto motobomba.



Fonte: Autor

O conjunto motobomba presente no sistema utilizado é composto pelo acoplamento de uma bomba centrífuga a um motor elétrico trifásico, vide Figura 14. As especificações para a centrífuga e para o motor utilizado estão presentes nas Tabelas 4 e 5, respectivamente.

Tabela 4 – Parâmetros da centrífuga.

Fabricante	KSB Bombas Hidráulicas.
Modelo	MEGANORM 65-200
Vazão máxima	50 m^3/h
Velocidade de rotação	1750 rpm
Altura manométrica	17 mca

Fonte: Catálogo do fabricante.

Tabela 5 – Parâmetros do motor.

Fabricante	WEG
Potencia	3,7 kW
Tensão de alimentação	380/440 V
Frequência de alimentação	60 Hz
Índice de proteção	IP 55
Velocidade nominal	1720 prn

Fonte: Catálogo do fabricante.

O inversor de frequência é um dispositivo que permite o ajuste da frequência da tensão de alimentação do motor trifásico que opera na bomba, este ajuste permite o controle da velocidade de rotação do motor, e conseqüentemente o controle da vazão na bomba hidráulica. O modelo utilizado neste trabalho foi o CFW 08, fabricado pela WEG SA. Possui tensão de alimentação trifásica 240/480 Vac (+10%/−15%), frequência de alimentação de 50/60Hz, frequência de chaveamento entre 10 e 15kHz, frequência de saída podendo ser ajustada entre 0 a 300 Hz, potência de até 12,2 kVA, potência dissipada de 240 W e corrente nominal de 16 A. Este inversor está ilustrado na Figura 15.

Figura 15 – Inversor de frequência.



Fonte: Autor

O sistema de aquisição de dados utilizado, que realiza a comunicação entre os sinais elétricos dos sensores e o sistema supervisor, é composto pelo Controlador Lógico Programável (CLP) ATOS 4004. O qual opera com tensão de 24 V, possui velocidade de comutação de 1 ms, corrente máxima de carga de 2 A, níveis dos sinais de entrada de 0 a 10 Vcc e 0 a 20 mA. Este CLP se comunica com o supervisor por meio do

padrão industrial RS 232 a uma taxa de até 20 *kbit/s*, e o supervisor deste sistema foi implementado na plataforma do LabView em trabalhos anteriores no LENHS.

Figura 16 – CLP ATOS 4004.



Fonte: Autor

3.2 Modelo RNA

A criação do modelo baseado em RNA, que representou a bancada, se deu em três etapas: inicialmente realizou-se a caracterização da bancada por meio de um experimento onde foram adquiridos os valores de vazão e frequência do sistema; em seguida esses dados foram manipulados com o intuito de se obter os padrões de treinamento e de teste da rede; e por ultimo a implementação de um código no software *MatLab* com a topologia pretendida.

3.2.1 Caracterização da bancada

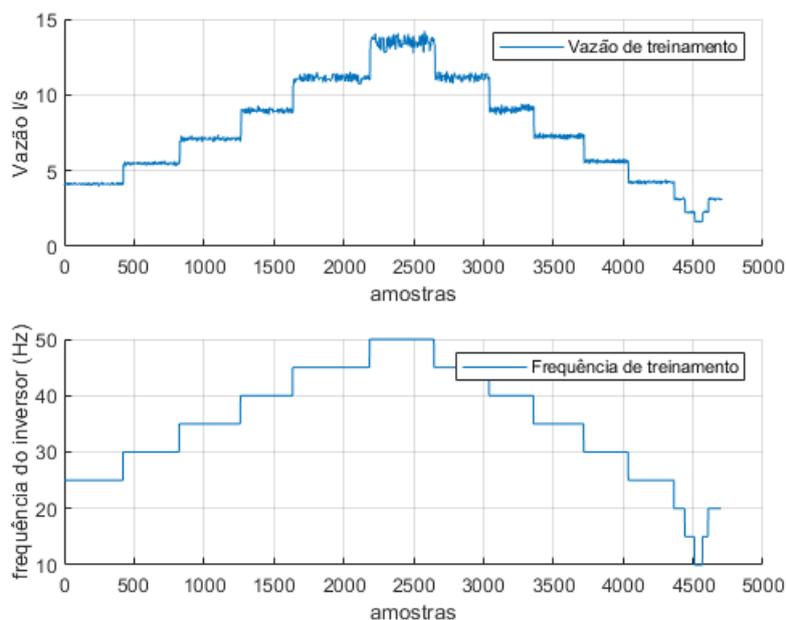
Nesta etapa do trabalho foram feitos dois experimentos, um primeiro para criação do padrão de treinamento da rede neural artificial, e outro onde foram realizadas mudanças aleatórias no valor da frequência do inversor, que foi utilizado para testar o modelo RNA.

Para a criação do padrão de treinamento foi realizado um experimento onde a frequência do inversor foi alterada a cada dois minutos. Inicialmente incrementando 5 Hz , a partir de 25 Hz , até o valor de 55 Hz , em seguida decrementando este valor de 5 Hz até o valor de 20 Hz . O limite superior de 55 Hz foi estipulado levando em consideração as altas pressões de operações existentes no sistema a partir deste valor, as quais poderiam comprometer a bancada. Já o limite inferior foi fixado em 20 Hz , pois abaixo deste valor de frequência a vazão do sistema era desprezível.

Para a obtenção do padrão de teste, este experimento foi refeito sendo que as mudanças nos valores de frequência do inversor foram realizadas de maneira aleatória, dentro da faixa de variação das variáveis do processo.

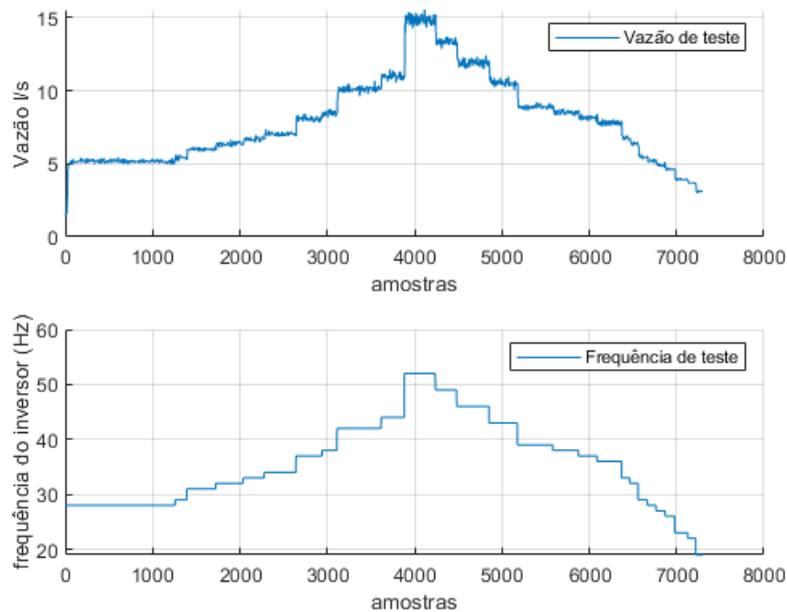
A seguir, podemos observar as curvas características para os padrões de treinamento e de teste, nas Figuras 17 e 18, respectivamente. O gráfico da Figura 17, apresenta um padrão composto por degraus com o intuito de fazer uma varredura entre os mais distintos valores de vazão que o sistema comporta, pois esse intervalo de valores é o que tem que ser aprendido pela RNA. Já na Figura 18, buscou-se construir um padrão que emulasse o manejo de um operador do sistema.

Figura 17 – Curva característica do padrão de treinamento.



Fonte: Autor

Figura 18 – Curva característica do padrão de teste.

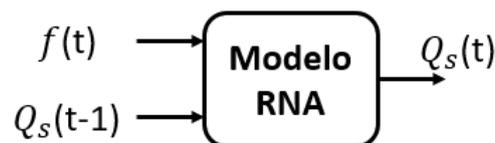


Fonte: Autor

3.2.2 Montagem do banco de dados (tabelas)

Em posse dos dados dos experimentos descritos anteriormente, foi feita uma tratativa de modo a se obter os padrões de dados de treinamento (P_{tr}) e de teste (P_{tt}). Estes padrões consistem em três vetores, o primeiro é o target $Q_s(t)$ que representa o padrão de comportamento dinâmico que o modelo deve aprender, representado pelos dados de vazão adquiridos no experimento para treinamento do modelo RNA, o segundo é composto pelos valores de frequência $f(t)$ que foram adquiridos neste mesmo experimento, e por último, o vetor de realimentação definido como sendo o vetor $Q_s(t-1)$, ou seja, a $Q_s(t)$ com um atraso unitário; foram realizados experimentos com atrasos não unitários e verificou-se que o ganho se manteve estável.

Figura 19 – Diagrama de blocos do modelo RNA.

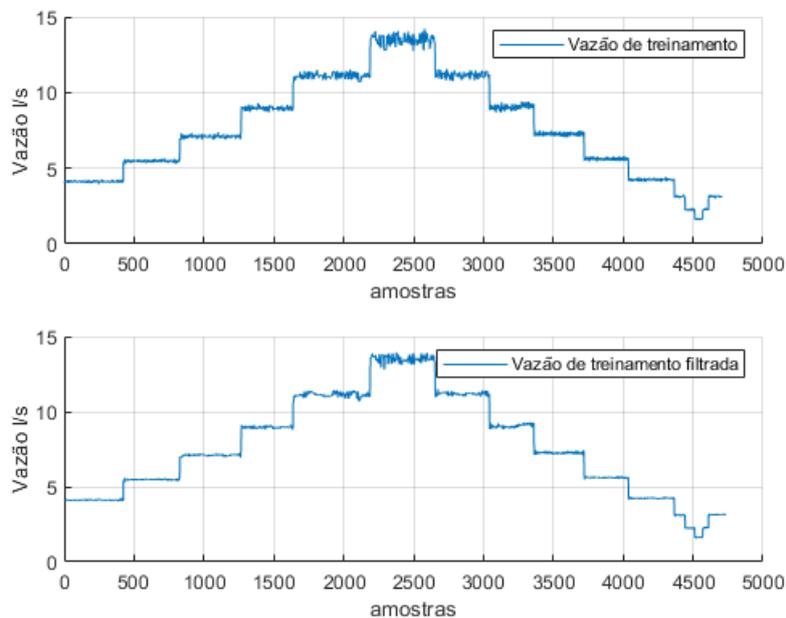


Fonte: Autor

Após, foi feita uma filtragem no ruído presente na vazão do experimento de treinamento utilizando-se o filtro Hampel, este é um filtro linear de limpeza de dados

espúrios, (RINCON, 2012). onde para cada amostra de vazão foi calculada uma mediana em uma janela de 30 amostras circunvizinhas, ou 15 segundos, e a estimação do desvio padrão de cada amostra em relação a mediana de janela, sendo o tempo dessa janela de amostra de três segundos, utilizando o desvio absoluto mediano. Sendo assim se uma amostra diferiu da mediana em mais de uma vez o desvio padrão, ela foi substituída pela mediana. Na figura 20, observamos uma comparação entre os dados antes e depois de ser submetido à filtragem.

Figura 20 – Filtragem do sinal de vazão de treinamento.



Fonte: Autor

3.2.3 Criação da rede neural artificial

Dando prosseguimento aos trabalhos, foi utilizado o software Matlab para a implementação de uma rede neural artificial que otimiza sua resposta com base em valores anteriores ou *Feed-Forward Backpropagation Network*.

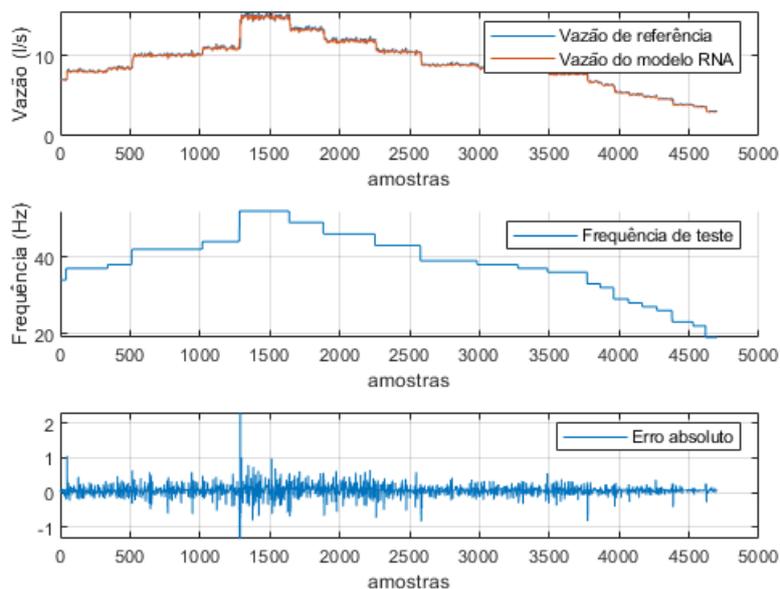
Os parâmetros utilizados na topologia estão descritos na Tabela 6. Estes valores foram obtidos por meio da tentativa e observação, de forma a se diminuir o erro absoluto da resposta da rede. Notou-se que o erro absoluto negativo, era em média, menor que o erro absoluto positivo, por isso foi definido um erro absoluto negativo de -1,5 e um erro absoluto positivo 2,5 para a rede ideal. Por meio de um algoritmo, realizando uma série de interações, esta topologia foi considerada satisfatória.

Tabela 6 – Parâmetros para a criação da rede neural artificial.

Função de ativação	Purelin/logsig
Função de treinamento	trainlm
Número de camadas	5
Número de neurônios por camada	10
Número máximo de épocas	100
Erro mínimo desejado	0,001

Fonte: Autor.

Figura 21 – Comparação entre a saída RNA e o padrão de vazão de teste.



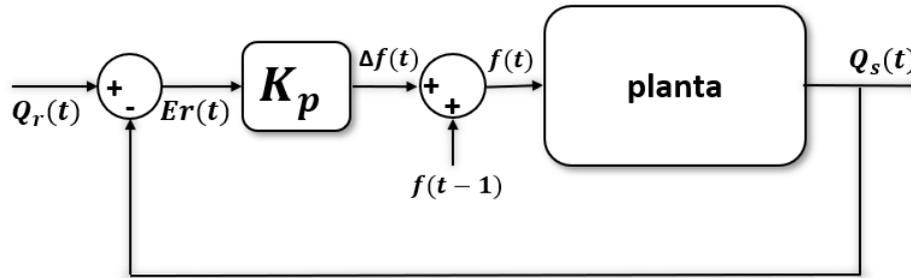
Fonte: Autor

Na Figura 21 podemos observar a resposta da rede neural criada em relação ao valor de vazão adquirido experimentalmente.

3.3 Sintonização do controlador proporcional

Para realizar a sintonização do controlador proporcional, representado pelo diagrama de blocos mostrado na Figura 22, foi utilizado o método da curva de reação de *Ziegler Nichols*, sendo assim, foi aplicado um degrau ao sistema e adquirida a forma de onda da resposta para a posterior extração do ganho proporcional.

Figura 22 – Diagrama de blocos - Controlador proporcional.

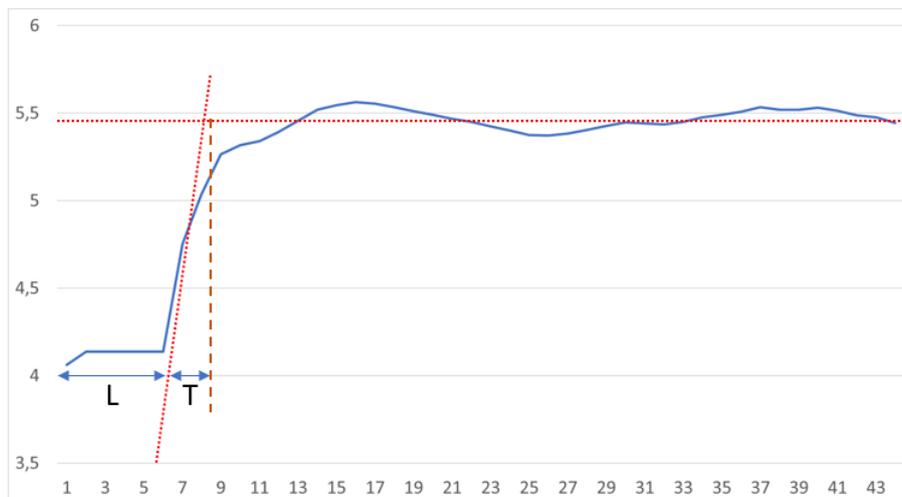


Fonte: Autor

- $Q_r(t)$: Vazão de referência;
- E_r): Erro;
- K_p : Ganho proporcional;
- $\delta f(t)$: frequência diferencial;
- $f(t)$: Frequência;
- $Q_s(t)$: Vazão de saída.

Na Figura 23, podemos observar a reação do sistema ao degrau e a análise da curva. Como o experimento foi feito a um tempo de amostragem de $0,1s$ temos que $L = 0,1 * 6 = 0,6s$ e $T = 0,1 * 2 = 0,2s$.

Figura 23 – Método da curva de reação.



Fonte: Autor

Logo, de acordo com a tabela do método da curva de reação, exibida na Figura 24, temos que o valor para o ganho proporcional é de: $K_p = T/L = 0,2/0,6 = 0,33$.

Figura 24 – Tabela de *Ziegler Nichols*.

	Kp	Ki	Kd
P	T/L	INFINITO	0
PI	0,9*(T/L)	T/(0,3)	0
PID	1,2*(T/L)	2L	0,5L

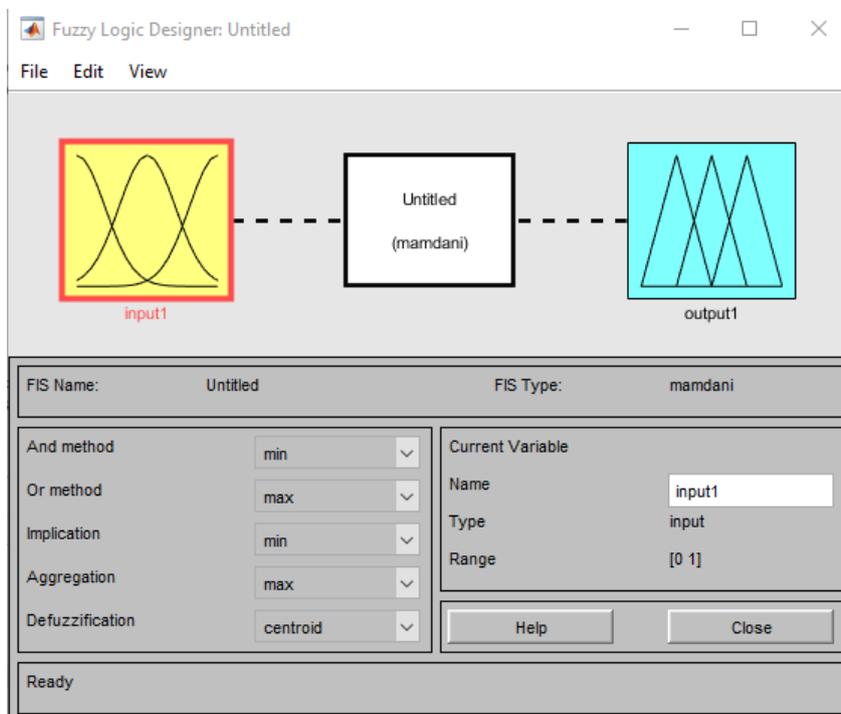
Fonte: Autor

3.4 Implementação do controlador Fuzzy

Nesta etapa do trabalho foi implementado um algoritmo no *software Matlab*, através do método direto de *Mamdani*, o qual foi utilizado para realizar o controle de vazão de um sistema do tipo MISO (*Multiple Input and Single Output*).

Para implementar o controlador foi utilizada a ferramenta *Fuzzy Logic Designer*, na Figura 25 podemos observar a interfase desta ferramenta do *Matlab*.

Figura 25 – Interfase *Fuzzy Logic Designer*.

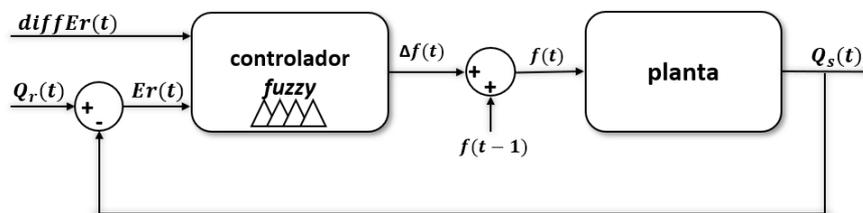


Fonte: Autor

3.4.1 Definição do domínio

Este controlador foi composto por dois sinais de entrada e um sinal de saída. Foram definidos como entradas, o erro do sistema de realimentação e a derivada deste erro. Já a saída foi a frequência atuante sobre o inversor de frequência. Na Figura 26, é mostrado o diagrama de blocos onde estão ilustradas as entradas e saídas do controlador *fuzzy*. O erro é calculado via software realizando a subtração entre o valor de referência e o valor de saída do sistema, já a derivada do erro é igual a: $diffEr = E_r(t) - E_r(t - 1)$. O valor de saída do controlador *fuzzy* foi incrementado, ou decrementado, do valor atual da frequência do inversor de frequência.

Figura 26 – Diagrama de blocos do controlador Fuzzy.



Fonte: Autor

Onde,

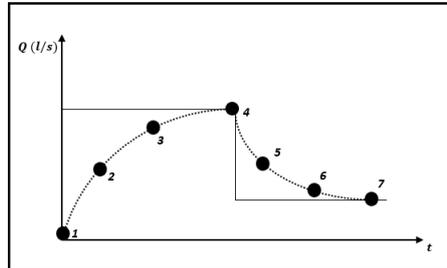
- $diffEr$ → derivada do erro;
- $Q_r(t)$ → vazão de referência (1/s);
- Er → erro do sistema;
- $f(t)$ → frequência do inversor;
- $Q_s(t)$ → vazão de saída (1/s);

3.4.2 Construção dos conjuntos

Analisando-se o comportamento do erro e da derivada do erro, foram definidos sete pontos para o auxílio da definição das regras e dos conjuntos *fuzzy*, conforme pode ser observado na ilustração da Figura 27. Sendo assim, foi feita a seguinte analogia: no ponto um, o erro é máximo e positivo; no ponto dois, o erro possui um valor mediano e positivo; no ponto três, o erro é pequeno e positivo; no ponto quatro, temo erro igual a zero; no ponto cinco o erro é grande e negativo; no ponto seis, o erro é pequeno e negativo; e no ponto sete volta a ser igual a zero. Com base nesse raciocínio foram definidos os conjuntos de erro, derivada do erro e o conjunto de variação de frequência, os quais estão descritos

nas Tabelas 7, 8 e 9, respectivamente, onde cada grupo representa um ponto na curva mostrada na Figura 27.

Figura 27 – Comportamento esperado do sinal de erro.



Fonte: Autor

Uma observação importante de citar é que o domínio destes conjuntos foi normalizado de -1 até 1. O intuito de trabalhar com os parâmetros normalizados foi de poder trabalhar com ganhos constantes, e assim poder regular o sistema.

Tabela 7 – Definição do conjunto erro.

Símbolo	Nome do grupo	Função	Suporte
BN	Erro grande e negativo	trapmf	[-1 -1 -0,82 -0,8]
MN	Erro médio e negativo	trimf	[-0,9 -0,6 -0,5]
SN	Erro pequeno e negativo	trimf	[-0,6 -0,25 -0,05]
Z	zero	trimf	[-0,1 0 0,1]
SP	Erro pequeno e positivo	trimf	[0,05 0,25 0,6]
MP	Erro médio e positivo	trimf	[0,5 0,6 0,9]
BP	Erro grande e positivo	trapmf	[0,8 0,82 1 1]

Fonte: Autor.

Tabela 8 – Definição do conjunto derivada do erro.

Símbolo	Nome do grupo	Função	Suporte
BN	diffErro grande e negativo	trapmf	[-1 -1 -0,82 -0,8]
MN	diffErro médio e negativo	trimf	[-0,9 -0,6 -0,5]
SN	diffErro pequeno e negativo	trimf	[-0,6 -0,25 -0,05]
Z	zero	trimf	[-0,1 0 0,1]
SP	diffErro pequeno e positivo	trimf	[0,05 0,25 0,6]
MP	diffErro médio e positivo	trimf	[0,5 0,6 0,9]
BP	diffErro grande e positivo	trapmf	[0,8 0,82 1 1]

Fonte: Autor.

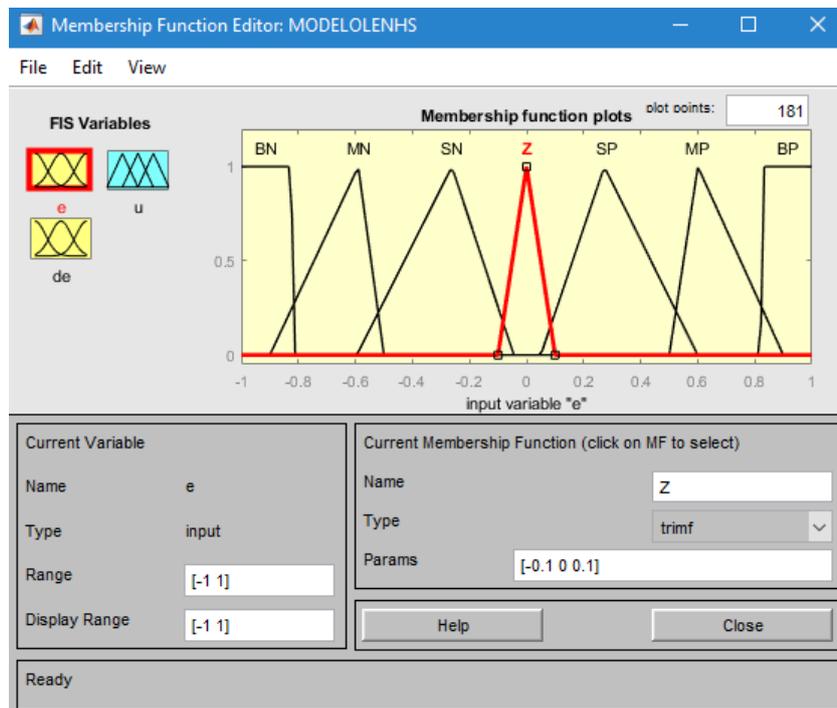
Tabela 9 – Definição do conjunto de saída.

Símbolo	Nome do grupo	Função	Suporte
BD	Decremento grande	trapmf	[-1 -1 -0,8 -0,75]
MD	Decremento médio	trimf	[-0,8 -0,45 -0,4]
SD	Decremento pequeno	trimf	[-0,5 -0,25 -0,1]
Z	zero	trimf	[-0,1 0 0,1]
SI	Incremento pequeno	trimf	[0,1 0,25 0,5]
MI	Incremento médio	trimf	[0,4 0,45 0,8]
BI	Incremento grande	trapmf	[0,75 0,8 1 1]

Fonte: Autor.

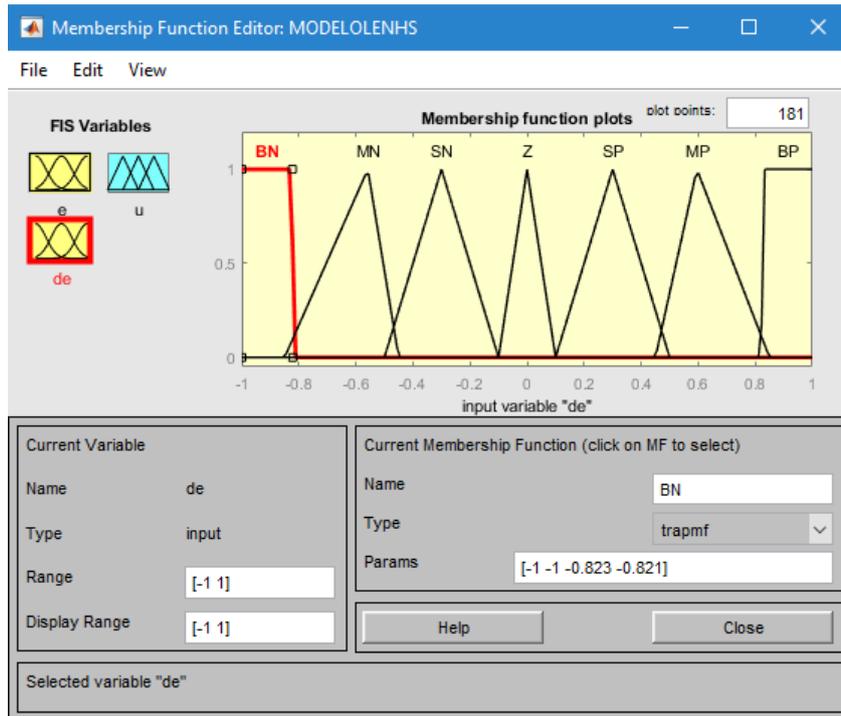
Nas Figuras 28, 29 e 30, está ilustrado a forma dos conjuntos de erro, derivada de erro e saída, respectivamente.

Figura 28 – Conjunto erro.



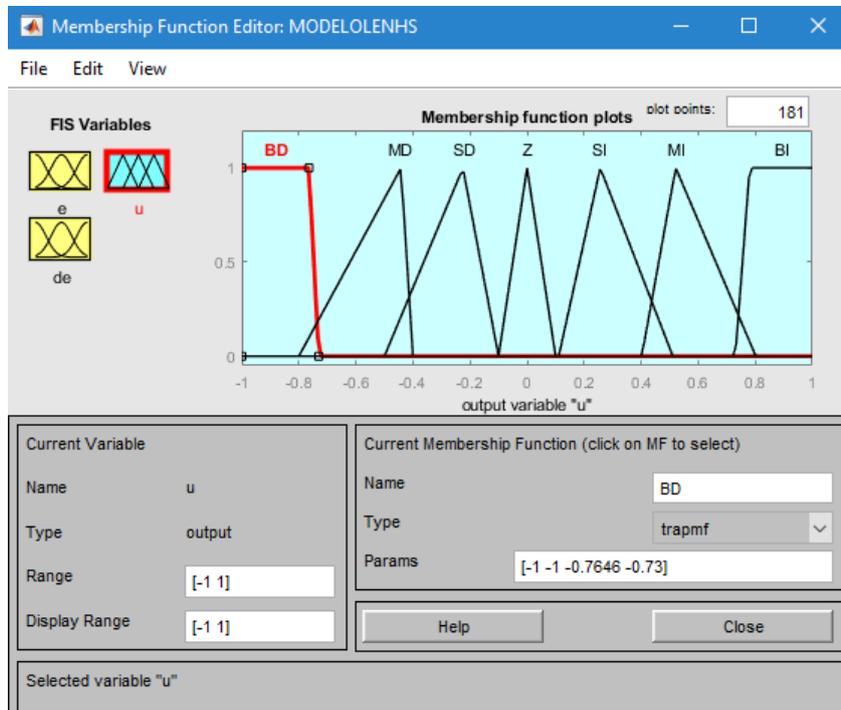
Fonte: Autor

Figura 29 – Conjunto derivada do erro.



Fonte: Autor

Figura 30 – Conjunto saída.



Fonte: Autor

3.4.3 Definição das regras

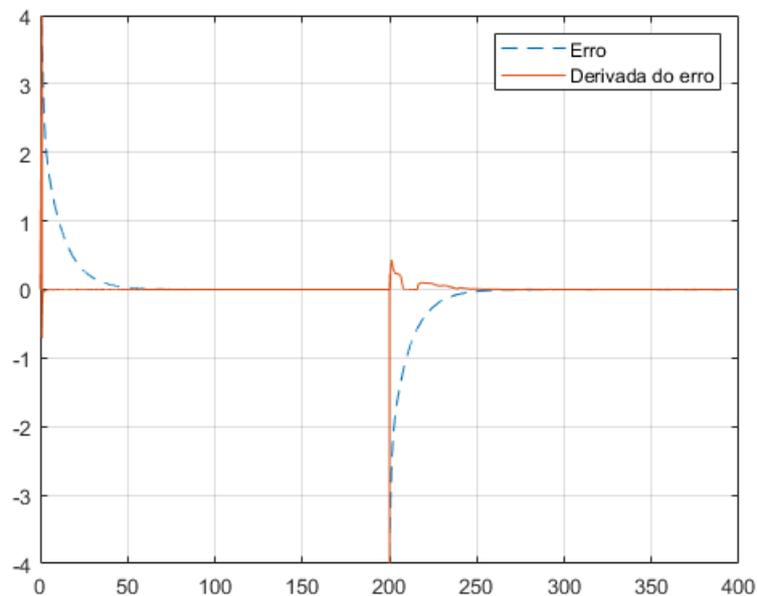
Dando continuidade, foram definidas as regras de acordo com a combinação entre os grupos, dos conjuntos de entrada. Tomando como base o comportamento do erro e da derivada do erro, ilustrados na Figura 32, foi implementado quadro exibido na Figura 31, o qual resume a relação das entradas e saídas para a definição das regras, que foram do tipo *if/then*.

Figura 31 – Definição das regras.

		DERIVADA DO ERRO						
		BN	MN	SN	Z	SP	MP	BP
ERRO	BN	<i>SD</i>	<i>SD</i>	<i>SD</i>	<i>SD</i>	<i>SD</i>	<i>BD</i>	<i>BD</i>
	MN	<i>SD</i>	<i>SD</i>	<i>SD</i>	<i>SD</i>	<i>SD</i>	<i>MD</i>	<i>BD</i>
	SN	<i>SD</i>	<i>SD</i>	<i>SD</i>	<i>SD</i>	<i>SD</i>	<i>SD</i>	<i>SD</i>
	Z	<i>SI</i>	<i>SI</i>	<i>SI</i>	<i>Z</i>	<i>SD</i>	<i>SD</i>	<i>SD</i>
	SP	<i>SI</i>	<i>SI</i>	<i>SI</i>	<i>SI</i>	<i>SI</i>	<i>SI</i>	<i>SI</i>
	MP	<i>BI</i>	<i>MI</i>	<i>SI</i>	<i>SI</i>	<i>SI</i>	<i>SI</i>	<i>SI</i>
	BP	<i>BI</i>	<i>BI</i>	<i>SI</i>	<i>SI</i>	<i>SI</i>	<i>SI</i>	<i>SI</i>

Fonte: Autor

Figura 32 – Comportamento do erro e da derivada do erro após um degrau.

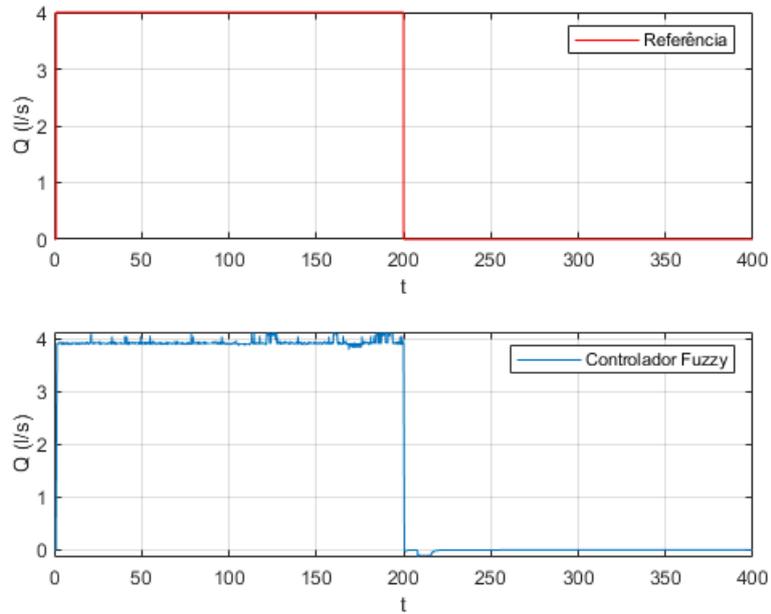


Fonte: Autor

3.4.4 Avaliação

Para realizar um teste no controlador implementado foi realizado o controle de uma planta genérica do primeiro grau e sua resposta pode ser observada na Figura 33.

Figura 33 – Resposta controlada sobre uma planta genérica do primeiro grau.



Fonte: Autor

4 Resultados

Neste capítulo estão expostos as informações referentes ao desempenho do modelo RNA e foi feita uma análise sobre o desempenho dos controladores em relação ao modelo proposto, esta análise foi composta de três experimentos. No primeiro experimento foi observado o comportamento das respostas em relação a um padrão de vazão que representa uma demanda de um SAA, no segundo experimento foi observado o comportamento das respostas em relação a vazão do padrão de treinamento P_{tr} e no terceiro experimento foi observado o comportamento das respostas em relação a vazão do padrão de teste P_{tt} .

A avaliação do desempenho foi feita por meio da análise do erro médio quadrático, Equação 4.1, erro máximo positivo e erro mínimo negativo.

Também foi analisado o tempo de acomodação da resposta do controlador *fuzzy* em relação ao tempo de acomodação em laço aberto do modelo RNA.

$$EQM = \frac{1}{N} \sum_1^N (e_i)^2 = \frac{1}{N} \sum_1^N (Q_{FUZZY} - Q_{ref})^2 \quad (4.1)$$

Onde:

- EQM → erro quadrático médio;
- e → erro;
- Q_{FUZZY} → vazão controlada;
- Q_{ref} → vazão de referência.

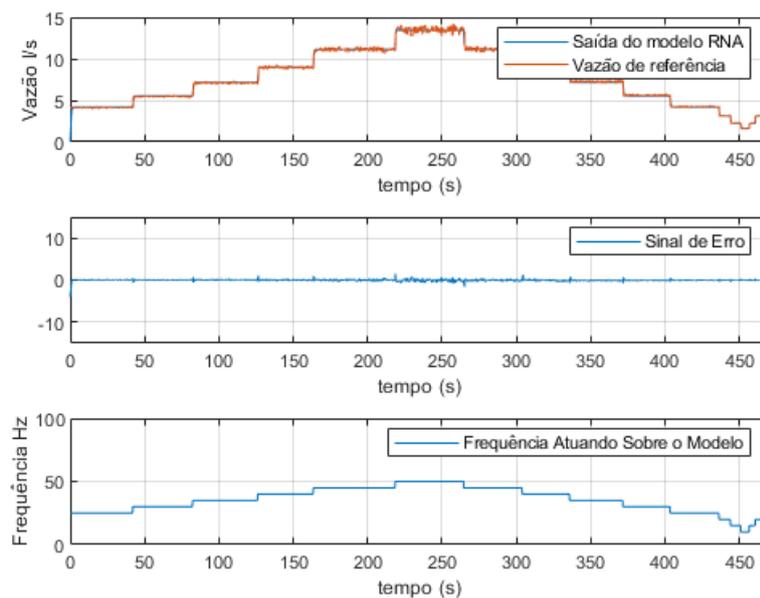
A configuração utilizada no controle da planta pode ser vista na Figura 26, para o controlador *fuzzy*, e na Figura 22, para o controlador proporcional.

4.1 Desempenho do modelo RNA

Após a rede neural artificial ser criada, foi realizado um experimento em laço aberto onde o modelo foi submetido aos parâmetros do padrão de teste (P_{tt}), obtidos experimentalmente.

Na Figura 34, podemos observar o comportamento da saída do modelo RNA em relação ao dados adquiridos experimentalmente, está exibido também os valores de frequência que foram utilizados na entrada da RNA e o comportamento do sinal de erro associado ao modelo RNA em relação aos dados de vazão obtidos experimentalmente.

Figura 34 – Teste do modelo RNA (Laço Aberto).



Fonte: Autor

Como parâmetros para realizar a avaliação desta resposta foram utilizados o erro quadrático médio, erro máximo positivo, erro mínimo negativo e o tempo de acomodação. Na Tabela 10 estão listados estes valores.

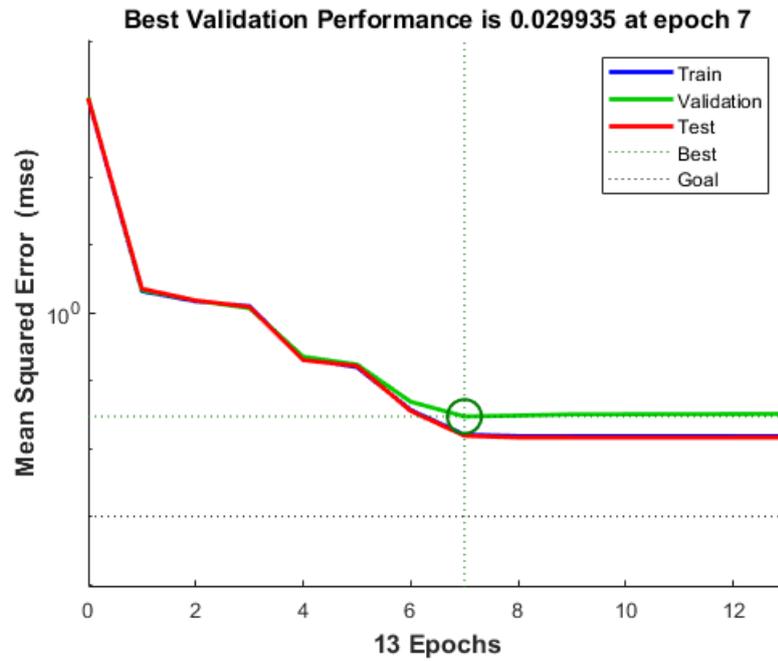
Tabela 10 – Desempenho do modelo RNA.

Erro médio quadrático	0,029935
Erro máximo positivo	2.2549
Erro mínimo negativo	-1.3701
Tempo de acomodação	1,9 s
Erro regime estacionário	0

Fonte: Autor.

Na Figura 35, é mostrada a evolução do *Mean Squared Error* (SME) em relação ao número de épocas, durante a criação da RNA. Foi requerido um SME de 10^{-3} , e como pode ser observado na figura foi obtido um de 3×10^{-2} , com uma simulação que levou sete épocas para obter esse resultado.

Figura 35 – Regressão do erro médio quadrático.



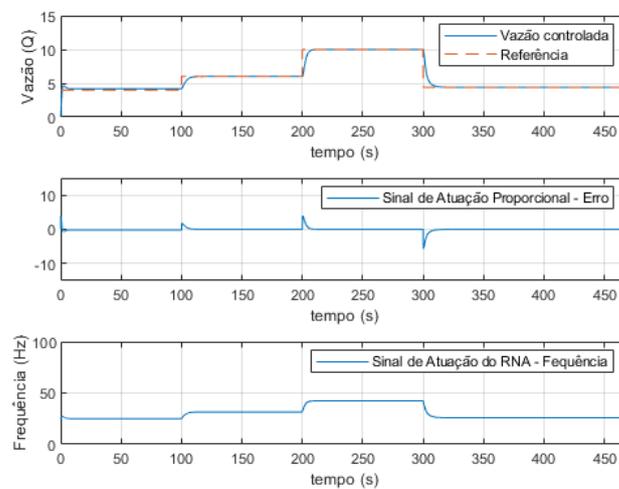
Fonte: Autor

4.2 Experimento I

Neste experimento foram utilizados como vazão de referência, Q_r , um padrão que representa uma demanda de variação de vazão de um SAA genérico, com o intuito de verificar o comportamento dos controladores e do modelo RNA a um referencial constante.

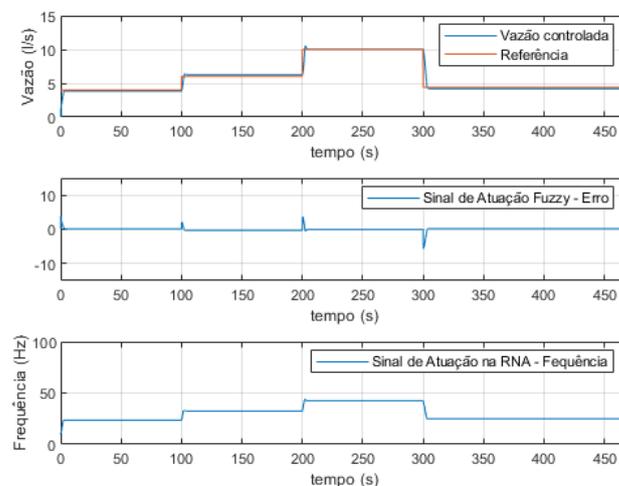
Na Figura 36, pode ser visto a resposta da planta controlada pelo controlador proporcional em relação a sua referência, o sinal de atuação erro proporcional e o sinal de frequência que atua no modelo RNA. Da mesma forma na Figura 37, sendo neste caso para a resposta da planta controlada pelo controlador *fuzzy*.

Figura 36 – Desempenho do controlador proporcional no experimento I.



Fonte: Autor

Figura 37 – Desempenho do controlador *fuzzy* no experimento I.



Fonte: Autor

Na Tabela 11, estão listados os valores de erro médio quadrático, o erro máximo positivo, o erro mínimo negativo e o tempo de acomodação obtidos na resposta da planta controlada pelo controlador proporcional em relação a vazão de referência. E na Tabela 12, estão listados os valores obtidos na resposta da planta controlada pelo controlador *fuzzy*.

Tabela 11 – Avaliação do controlador proporcional no experimento I.

Erro médio quadrático	0,2030
Erro máximo positivo	4,0000
Erro mínimo negativo	-5,6000
Tempo de acomodação	12 s
Erro regime estacionário	-1.4984e-13

Fonte: Autor.

Tabela 12 – Avaliação do controlador *fuzzy* no experimento I.

Erro médio quadrático	0,1837
Erro máximo positivo	4,0000
Erro mínimo negativo	-5,6180
Tempo de acomodação	4,2 s
Erro regime estacionário	0.0523

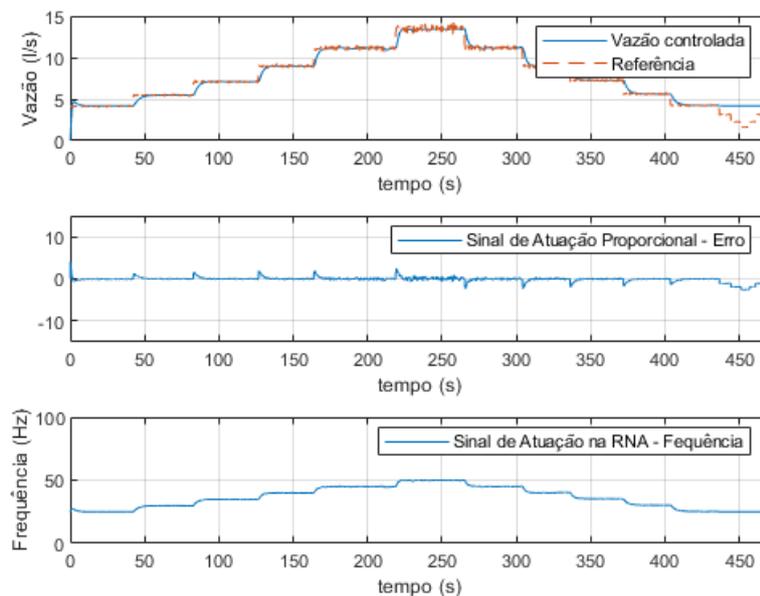
Fonte: Autor.

4.3 Experimento II

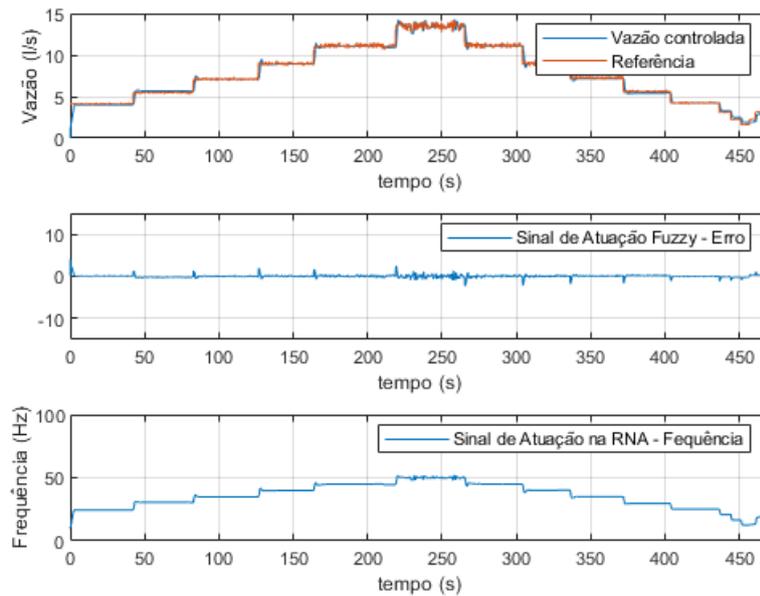
Neste experimento foram utilizados como vazão de referência, Q_r , os valores adquiridos experimentalmente na bancada e que foram utilizados no treinamento do modelo RNA. Visto que em uma aplicação prática o referencial de vazão será constante, o intuito de usar um referencial que possui um ruído (que são os valores adquiridos experimentalmente) é verificar a eficácia dos controladores sobre o modelo RNA.

Na Figura 38, pode ser visto a resposta da planta controlada pelo controlador proporcional em relação a vazão de referência, o sinal de atuação erro proporcional e o sinal de frequência que está atuando no modelo RNA. Da mesma forma na 39, sendo que neste caso para a resposta da planta controlada pelo controlador *fuzzy*.

Figura 38 – Desempenho do controlador proporcional no experimento II.



Fonte: Autor

Figura 39 – Desempenho do controlador *fuzzy* no experimento II.

Fonte: Autor

Na Tabela 13, estão listados os valores de erro médio quadrático, o erro máximo positivo, o erro mínimo negativo e o tempo de acomodação obtidos na resposta da planta controlada pelo controlador proporcional em relação a vazão de referência. E na Tabela 14, para a resposta da planta controlada pelo controlador *fuzzy*.

Tabela 13 – Avaliação do controlador proporcional no experimento II.

Erro médio quadrático	0,3581
Erro máximo positivo	4,0375
Erro mínimo negativo	-2,344
Tempo de acomodação	10,6 s
Erro regime estacionário	0.0046

Fonte: Autor.

Tabela 14 – Avaliação do controlador *fuzzy* no experimento II.

Erro médio quadrático	0,1116
Erro máximo positivo	4,0375
Erro mínimo negativo	-2,6236
Tempo de acomodação	3 s
Erro regime estacionário	-0.0043

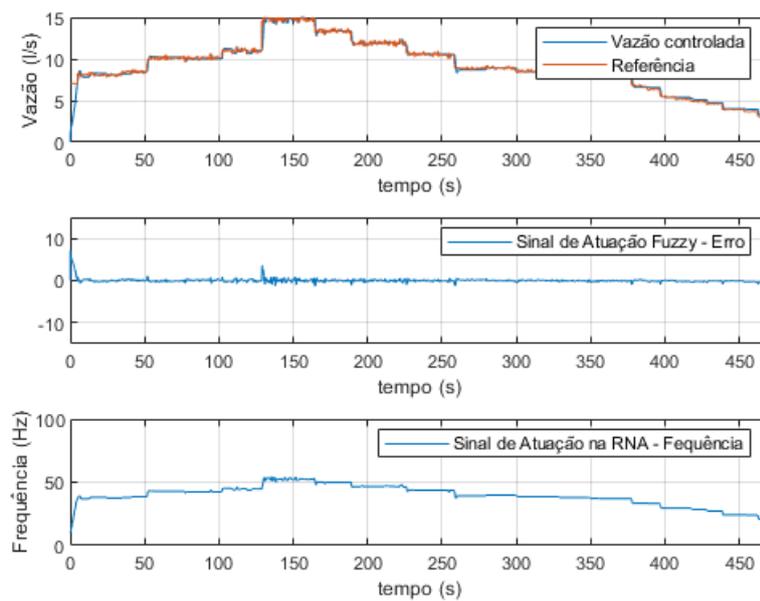
Fonte: Autor.

4.4 Experimento III

Neste experimento foram utilizados como vazão de referência, Q_r , os valores adquiridos experimentalmente na bancada e que foram utilizados no teste do modelo RNA. Da mesma forma que o experimento II, este experimento buscou testar a eficácia do controlador *fuzzy*.

Na Figura 40, está exposto a resposta da planta controlada pelo controlador *fuzzy* em relação a vazão de referência, o sinal de atuação erro *fuzzy* e o sinal de frequência que está atuando no modelo RNA.

Figura 40 – Desempenho do controlador *fuzzy* no experimento III.



Fonte: Autor

Na Tabela 15, estão listados os valores de erro médio quadrático, o erro máximo positivo, o erro mínimo negativo e tempo de acomodação obtidos na resposta da planta controlada pelo controlador *fuzzy* em relação a vazão de referência.

Tabela 15 – Avaliação do controlador *fuzzy* no experimento III.

Erro médio quadrático	0,2207
Erro positivo máximo	7,1000
Erro negativo mínimo	-1,3802
Tempo de assentamento	1,4 s
Erro regime estacionário	0.1249

Fonte: Autor.

5 Considerações finais

Neste capítulo foram realizadas considerações sobre os resultados obtidos e também foi feita uma conclusão sobre os trabalhos descritos neste relatório.

- A caracterização da bancada foi realizada satisfatoriamente com o auxílio do software *LabView*. Quando foi abordados valores de vazão abaixo de 2 l/s notou-se que o sensor de vazão teve dificuldades para realizar a leitura dos dados, porém isso não comprometeu a caracterização.
- O desenvolvimento de regras associadas as experiências de um operador foi realizado com eficácia por meio da análise do erro e da derivada do erro da resposta ao degrau de um sistema do primeiro grau, a qual representa o comportamento de decisão de um operador do sistema.
- O desempenho do controlador *fuzzy* se mostrou mais eficiente que o controlador proporcional, como já era esperado.
- Foi constatado que a resposta em malha aberta possuiu uma resposta mais rápida, em relação a qualquer um dos cenários abordados, como também um menor erro estacionário. Isso demonstra que o modelo RNA aprendeu com fidelidade o comportamento da bancada experimental.
- No experimento III foram dispensados os resultados do controlador proporcional pois foi constatado uma falha no controle do experimento II.
- Por mais que o sistema a ser controlado seja do primeiro grau, foi constatado que o controle pode ser inviável em determinadas técnicas tradicionais de controle. Pois, o comportamento do fluxo de água mostrou um padrão não linear.

Para trabalhos futuros, que busquem desenvolver o modelo proposto, pode-se incluir a supervisão de sensores de pressão para monitorar o aumento de pressão, funcionando assim como cão de guarda para anunciar limites de pressão que comprometam a integridade da tubulação.

Outra proposta é a inserção de perturbações no modelo de SAA, para analisar a robustez dos controladores desenvolvidos. Também pode ser pertinente a investigação de um sistema MIMO.

Por tudo isso, chegou-se a conclusão que é viável o desenvolvimento de modelos de sistema baseados em RNA e que o controlador *fuzzy* tem uma maior robustez em relação a técnicas clássicas de controle.

6 Referências bibliográficas

AGUIRRE, L. A. **Introdução à Identificação de Sistemas: Técnicas Lineares e Não-Lineares Aplicadas a Sistemas Reais**. 3. ed. Belo Horizonte: Editora UFMG, 2007.

ANDRADE, L. A. de et al. **Estudo comparativo dos métodos de medição de vazão - uma aplicação em comissionamento de turbinas hidráulicas**. Centro de Pesquisas em Hidráulica e Recursos Hídricos, 2014. Disponível em: <http://www.cph.eng.ufmg.br/do_cscph/matevento15.pdf>.

ASSOCIAÇÃO BRASILEIRA DE NORMAS TÉCNICAS - ABNT. NBR 12211 **Estudos de concepção de sistemas públicos de abastecimento de água**. Rio de Janeiro, 1992.

BARRETO, Jorge M. **Introdução às Redes Neurais Artificiais**, Laboratório de Conexão e Ciências Cognitivas UFSC - Departamento de Informática e de Estatística 88040-900 - Florianópolis - SC, 2002.

CAMARGO, C. B. **Acionamentos de Motores de Indução Através das Técnicas de Controle Escalar e Vetorial**. Dissertação, 116. Sorocaba, SP: Programa de Graduação da Faculdade de Engenharia de Sorocaba, (2011).

CAMPOS, M. C. M. M.; TEIXEIRA, H. C. G. **Controles Típicos De Equipamentos e Processos Industriais**. 2. ed. Editora Blucher, São Paulo, 2010.

CARVALHO, T. M. de. **Técnicas de medição de vazão por meios convencionais e não convencionais**. Revista Brasileira de Geografia Física, v. 01, n. 01, p. 73–85, Mai/Aug 2008.

CARR, J.J.;BROWN, J.M. **Introduction to biomedical equipment technology** 3^a ed, New Jersey: Prentice-Hall, 1998. 703p.)

CASTRUCCI, P. L.; BITTAR, A.; SALES, R. M. **Controle Automático**. 1. ed. Rio de Janeiro: LTC, 2011.

DATA & ANALYSIS CENTER FOR SOFTWARE. **Artificial Neural Networks Technology**, Disponível em www.dacs.dtic.mil, 2000.

FINOCCHIO, Marco Antonio Ferreira . **NOÇÕES DE REDES NEURAIIS ARTIFICIAIS**, Apostila, Laboratório de Segurança Iluminação e eficiência energética, Universidade Federal do Paraná, Cornélio Procópio - PR, 2014.

FOX, R. W.; MCDONALD, A. T. **Introdução à Mecânica dos Fluidos**. 5. ed. [S.l.]: LTC, 1998.

FRANCHI, C. M. **Acionamentos Elétricos**. São Paulo, Editora Érica, 2011.

GOMES, Heber Pimentel, **Sistemas de Saneamento – Eficiência Energética** , 1ª edição, João Pessoa, Editora Universitária – UFPB, 2010.

GRIPP, P. E. **Estudo das Tecnologias e Aplicações dos Inversores de Frequência de Média Tensão**. Dissertação, 65. Vitória, ES: Programa de Pós Graduação da Universidade Federal do Espírito Santo, (2005).

HELLER, Léo; DE PÁDUA, Valter Lúcio. **Abastecimento de água para consumo humano**. Editora UFMG, 2006.

KOVÁCS, Zsolt L. **Redes Neurais Artificiais: Fundamentos e Aplicações.**, Segunda edição, Collegium Cognitio, 1996.

LOURENÇO, João. **Apostila de Sintonia de Controladores PID.**, Apostila de controladores PID, Escola Superior de Tecnologia, Setúbal, 1997.

MARRO, Alessandro Assi et al. **Lógica fuzzy: conceitos e aplicações**. Natal: Universidade Federal do Rio Grande do Norte (UFRN), APOSTILA, 2010.

NISE, Norman S.; DA SILVA, Fernando Ribeiro. **Engenharia de sistemas de controle**. LTC, 2002.

NOGUEIRA, Maycon Mariano. **Aplicando lógica fuzzy no controle de robôs móveis**

usando dispositivos lógicos programáveis e a linguagem VHDL., 2013.

OLIVEIRA, A. L., MORAIS, M. S., & PIRES, I. A. **Estudo de Interferências na Alimentação Elétrica de Motores de Indução por Inversores de Frequência.** E-xata, UniBH, 159-173, (2012).

OLIVEIRA, J. P. B. M. **Auto-tuning Techniques for Industrial PI Controlers.**, Dissertação de Mestrado. University of Salford, 1994. LTC, 2002.

ORTEGA, Neli Regina Siqueira. **Aplicação da Teoria de Conjuntos Fuzzy a problemas da Biomedicina.** São Paulo (SP): Instituto de Física/USP, 2ªED, 2001.

PEREIRA, José Almir Rodrigues, **Abastecimento de Água: Informação para Eficiência Hidroenergética**, 1ª edição, João Pessoa, Editora Universitária – UFPB, 2014.

QUEIROZ, F. C. **Controle vetorial de velocidade de um motor de indução trifásico com estimação neural de fluxo.** Dissertação, 87. Natal: Programa de Pós Graduação da Universidade Federal do Rio Grande do Norte, (2008).

RAMOS, R. **Desenvolvimento de um medidor de vazão de gás por Ultra-som: Modelagem, protótipo e testes.** Dissertação (Mestrado), Universidade Federal do Espírito Santo, 2006.

RINCÓN, Antonio Quintero. **Preprocesamiento de EEG con Filtros Hampel**, Instituto Tecnológico de Buenos Aires - IEEE Latin American Transactions - Cordoba, Argentina - volumen 2012.

SANTOS, G. R. dos. **Desenvolvimento de Técnicas de Ultra-som para Medida de Espessura de Filme de Líquido em Escoamentos Gás-Líquido e Líquido-Líquido.** Dissertação (Mestrado), UNIVERSIDADE ESTADUAL DE CAMPINAS FACULDADE DE ENGENHARIA MECÂNICA, 2006.

SAVIONEK, D.; KARDELI, F.; RATHKE, T. **Construção e operação de um medidor de vazão volumétrica tipo Venturi.** [S.l.], 2010.

SOCIESC. **Acionamento de Motores com o Inversor de Frequência**. Curitiba, PR, Brasil: SOCIESC, (2015).

TAFNER, Malcon A. **Redes Neurais artificiais: Introdução e Princípios de Neuro-computação**, Santa Catarina - SC, EKO, 2014.

Apêndices

APÊNDICE A – CÓDIGOS RNA

```
%PRIMEIRO LEITURA DE DADOS%
```

```
M = []; % Cria um vetor vazio
% Cria um vetor M1 que recebe os dados do arquivo xls em questão
M1=xlsread('medNwRNA');
M = M1; % Atribui o valor de M1 a M
size(M1)
```

```
%SEGUNDO BANCO DE DADOS RNA%
```

```
%Saida para treinamento
```

```
Q6 = M(:,2)';
mediaQ6 = hampel(Q6,30,1); %Filtragem do sinal de vazão
%Entradas para treinamento
freq = M(1:4700,1)';
Q6atrasada = [0 Q6(1:4700-1)];
```

```
%Saida para teste
```

```
Q6_tt = M(1:7300,4)';
%Entradas para teste
freq_tt = M(1:7300,3)';
Q6atr_tt = [0 Q6_tt(1:7300-1)];
```

```
% Treinamento
```

```
%Padroes de entrada
Ptr = [freq;
       Q6atrasada];
%Target (objetivo)
Itr = [Q6(1:4700)];
```

```
% Teste
```

```
%Padroes de entrada
Ptt = [freq_tt;
       Q6atr_tt];
%Target (objetivo)
Itt = [Q6_tt(1:7300)];
```

```

%TERCEIRO TREINAMENTO RNA%

%Criando a rede com os dados de entrada e saída
%Função de ativação
TransFcn = {'purelin','purelin','logsig','purelin','purelin'};
%Função de treinamento
TrainFcn = 'trainlm';
numNeuronios = [10 10 10 10 10]; %Número de neurônios utilizados
rede = newff(Ptr,Ttr,numNeuronios,TransFcn,TrainFcn);
%-----
%Parametros de treinamento da Rede Madaline
rede.trainParam.epochs = 100; %Número Máximo de Épocas
rede.trainParam.goal = 1e-3; %Erro Mínimo Desejado
%rede.trainParam.ValRatio = 0.50;
%-----
%Treinamento e teste da Rede Neural
m = 2;
n = -2;
Emq = 0.1;
while (-1.5 > n) || (m > 2.5)
    redet = train(rede,Ptr,Ttr); %criação da redeneural artificial

    output = sim(redet,Ptt); %teste da rede neural
    Q6RNA = output;
    Q6 = Ttt;
    Eab = Q6 - Q6RNA;

    for i = 1:4700
        Ea(i)=Eab(i)*Eab(i);
    end

    Emq = sum(Ea)/length(Q6RNA); %cálculo do erro médio quadrático
    m = max(Eab)
    n = min(Eab)
end

%QUARTO TESTE%

% Laço Aberto
fo = Freq; %Frequencia de Entrada
Q = 0; %Vazão Inicial
Q1 = [Q]; %Guardando a vazao

for i = 1:L-1
    Qrna = sim(redet,[fo(i) Q]');
    Q = Qrna;
    Q1 = [Q1 Q];
end

```

```
%QUARTO TESTE%
% Laço Fechado - Controlador Proporcional
Frequencia = 25; %Sinal de Entrada Frequencia
Q = 0; %Vazão Inicial
Q1 = [Q]; %Guardando a vazao
%Qref = 4*ones(1,L);
%Sinal construido com diferentes degraus
%Qref = [4*ones(1,1000) 6*ones(1,1000) 10*ones(1,1000) 4.4*ones(1,L-3000)];
%Qref = M(1:L,2);
Qref = Q6_tt(2601:L+2600);
Kp = 0.1; %Ganho do controlador
erro = 0;
for i = 1:L-1
    erro(i+1) = Qref(i) - Q;
    fo(i) = Kp*erro(i+1);
    %saturador
    if Frequencia(i) >=50
        Frequencia(i) = 50;
    elseif Frequencia(i)<=25
        Frequencia(i) =25;
    end
    %=====
    Qrna = sim(redet,[Frequencia(i) Q]');
    Q = Qrna;
    Q1 = [Q1 Q];
    Frequencia(i+1) = Frequencia(i) + fo(i);
end
```

```

%QUARTO TESTE%
% -----
% Laço Fechado - Controlador Fuzy
GanhoErro = 1/7.5;
GanhodErro = 1/7;
GanhoFreq = 2.1;
p=4
Frequencia = 10; %Sinal de Entrada Frequencia
Q = 0; %Vazão Inicial
Q1 = [Q]; %Guardando a vazao
%Qref = 4*ones(1,L);
%Sinal construido com diferentes degraus
%Qref = [4*ones(1,1000) 6*ones(1,1000) 10*ones(1,1000) 4.4*ones(1,L-3000)];
%Qref = M(1:L,2);
Qref = Q6_tt(2601:L+2600);

Kp = 0.12; %Ganho do controlador
erro = 0;
derro = 0;
for i = 1:L-1
    erro(i+1) = Qref(i) - Q;
    derro (i+1) = erro(i+1) - erro(i);
    e1 = GanhoErro*erro(i+1);
    del = GanhodErro*derro (i+1);
    saidaFuzzy = evalfis([e1 del],MODELOLENHS);
    fo(i) = GanhoFreq*saidaFuzzy;
    ttfzzy = saidaFuzzy
    ttfreq = fo(i)
    tmp = i
    %saturador
    if Frequencia(i) >=60
        Frequencia(i) = 55;
    elseif Frequencia(i)<=0
        Frequencia(i) =0;
    end
    %=====
    Qrna = sim(redet,[Frequencia(i) Q]');
    Q = Qrna;
    Q1 = [Q1 Q];
    Frequencia(i+1) = Frequencia(i) + fo(i);
end
end

```


APÊNDICE B – CÓDIGOS *FUZZY*

```
1  [System]
2  Name='MODELOLENHS'
3  Type='mamdani'
4  Version=2.0
5  NumInputs=2
6  NumOutputs=1
7  NumRules=49
8  AndMethod='min'
9  OrMethod='max'
10 ImpMethod='min'
11 AggMethod='max'
12 DefuzzMethod='centroid'
13
14 [Input1]
15 Name='e'
16 Range=[-1 1]
17 NumMFs=7
18 MF1='BN': 'trapmf', [-1 -1 -0.85 -0.80]
19 MF2='MN': 'trimf', [-0.9 -0.6 -0.5]
20 MF3='MP': 'trimf', [0.5 0.6 0.9]
21 MF4='BP': 'trapmf', [0.8 0.82 1 1]
22 MF5='SN': 'trimf', [-0.6 -0.25 -0.05]
23 MF6='SP': 'trimf', [0.05 0.25 0.6]
24 MF7='Z': 'trimf', [-0.1 0 0.1]
25
26 [Input2]
27 Name='de'
28 Range=[-1 1]
29 NumMFs=7
30 MF1='BN': 'trapmf', [-1 -1 -0.85 -0.80]
31 MF2='MN': 'trimf', [-0.9 -0.6 -0.5]
32 MF3='MP': 'trimf', [0.5 0.6 0.9]
33 MF4='BP': 'trapmf', [0.8 0.82 1 1]
34 MF5='SN': 'trimf', [-0.6 -0.25 -0.05]
35 MF6='SP': 'trimf', [0.05 0.25 0.6]
36 MF7='Z': 'trimf', [-0.1 0 0.1]
37
38 [Output1]
39 Name='u'
40 Range=[-1 1]
41 NumMFs=7
42 MF1='BD': 'trapmf', [-1 -1 -0.75 -0.7]
43 MF2='MD': 'trimf', [-0.8 -0.45 -0.4]
44 MF3='SD': 'trimf', [-0.5 -0.25 -0.1]
45 MF4='Z': 'trimf', [-0.1 0 0.1]
46 MF5='MI': 'trimf', [0.4 0.45 0.8]
47 MF6='SI': 'trimf', [0.1 0.25 0.5]
48 MF7='BI': 'trapmf', [0.7 0.75 1 1]
```

```
50 [Rules]
51 1 1, 3 (1) : 1
52 1 2, 3 (1) : 1
53 1 3, 3 (1) : 1
54 2 1, 3 (1) : 1
55 2 2, 3 (1) : 1
56 2 3, 3 (1) : 1
57 5 1, 3 (1) : 1
58 5 2, 3 (1) : 1
59 5 3, 3 (1) : 1
60 6 1, 6 (1) : 1
61 6 2, 6 (1) : 1
62 6 3, 6 (1) : 1
63 3 1, 7 (1) : 1
64 3 2, 5 (1) : 1
65 3 3, 6 (1) : 1
66 4 1, 7 (1) : 1
67 4 2, 7 (1) : 1
68 4 3, 7 (1) : 1
69 1 5, 1 (1) : 1
70 1 6, 1 (1) : 1
71 1 7, 1 (1) : 1
72 2 5, 3 (1) : 1
73 2 6, 2 (1) : 1
74 2 7, 1 (1) : 1
75 5 5, 3 (1) : 1
76 5 6, 3 (1) : 1
77 5 7, 3 (1) : 1
78 6 5, 6 (1) : 1
79 6 6, 6 (1) : 1
80 6 7, 6 (1) : 1
81 3 5, 6 (1) : 1
82 3 6, 6 (1) : 1
83 3 7, 6 (1) : 1
84 4 5, 6 (1) : 1
85 4 6, 6 (1) : 1
86 4 7, 6 (1) : 1
87 7 1, 6 (1) : 1
88 7 2, 6 (1) : 1
89 7 3, 6 (1) : 1
90 7 4, 4 (1) : 1
91 7 5, 3 (1) : 1
92 7 6, 3 (1) : 1
93 7 7, 3 (1) : 1
94 1 4, 3 (1) : 1
95 2 4, 3 (1) : 1
96 5 4, 3 (1) : 1
97 6 4, 6 (1) : 1
98 3 4, 6 (1) : 1
99 4 4, 6 (1) : 1
```