

UNIVERSIDADE CATÓLICA DE PELOTAS
ESCOLA DE INFORMÁTICA
PROGRAMA DE PÓS-GRADUAÇÃO EM INFORMÁTICA

**Técnicas Matemático-Computacionais
para o Tratamento de Incertezas
Aplicadas ao Problema do Fluxo de
Potência em Sistemas de Transmissão de
Energia Elétrica**

por
Rogério Rodrigues de Vargas

Dissertação apresentada como
requisito parcial para a obtenção do grau de
Mestre em Ciência da Computação

Orientador: Prof. Dr. Luciano Vitoria Barboza
Co-orientadora: Prof. Dr. Graçaliz Pereira Dimuro

DM-2008/1-002

Pelotas, abril de 2008

Livros Grátis

<http://www.livrosgratis.com.br>

Milhares de livros grátis para download.

Dedico... esse trabalho aos meus pais, Jesus e Élide, pelo incentivo que me deram para chegar até aqui, sempre me apoiando em todos os momentos, e a minha namorada Daiane pelo carinho, apoio e dedicação.

AGRADECIMENTOS

Chegando ao fim de mais uma jornada, gostaria de agradecer a todos aqueles que em contextos diferentes, contribuíram para que eu chegasse até aqui.

Na impossibilidade de citar todos os nomes que fizeram a travessia comigo nesta construção, permito-me a destacar os seguintes nomes, na certeza de que somos gratos a todos que direta ou indiretamente contribuíram para o sucesso deste trabalho.

Assim, apresento-vos a profundidade dos reconhecimentos plenos:

À Deus.

Aos meus pais Jesus Prieto de Vargas e Élide Conceição Rodrigues de Vargas, que depositaram em mim toda a confiança me proporcionando a abertura deste caminho.

À minha namorada e companheira Daiane Conrad Fernandes, por todo amor, compreensão, incentivo, suporte e carinho que me motivaram a prosseguir, mesmo nos momentos mais difíceis.

Ao professor orientador Luciano Vitoria Barboza, por sua tão competente orientação, a minha admiração e agradecimento pelo contínuo apoio em todas as fases de realização até este momento, as discussões críticas que tanto contribuíram para o meu crescimento intelectual e pessoal.

À minha co-orientadora, professora Graçaliz Pereira Dimuro, pelos ensinamentos que com certeza, irei carregar por toda a vida.

A todos os colegas, professores e funcionários do Programa de Pós-Graduação em Informática.

À secretária do curso de Ciência da Computação, Kátia, pelo suporte e apoio e amizade que conquistamos.

Aos colegas do LASE, por terem me aceitado e por diversas discussões em que resultaram conhecimento.

À CAPES pelo apoio financeiro.

*“Não se imagina que tudo
é vago até que se tenta
fazê-lo de maneira precisa”.*

— BERTRAND RUSSEL

SUMÁRIO

LISTA DE FIGURAS	9
LISTA DE TABELAS	12
LISTA DE CÓDIGOS	13
LISTA DE ABREVIATURAS E SIGLAS	14
RESUMO	16
ABSTRACT	17
1 INTRODUÇÃO	18
1.1 Motivação	18
1.2 Contexto Histórico	20
1.3 Objetivos	23
1.4 Organização do Texto	24
2 ESTADO DA ARTE	25
2.1 A Matemática Intervalar	25
2.1.1 Conjunto \mathbb{IR}	26
2.1.2 Operações Básicas	26
2.1.3 Definições em \mathbb{IR}	28
2.2 A Lógica Fuzzy	30
2.2.1 Definição de Conjuntos Fuzzy	31
2.2.2 Função de Pertinência	33
2.2.3 Números Fuzzy	42
2.3 A Biblioteca C-XSC	49
2.3.1 Tipos de dados	50
2.3.2 Uso de Intervalos	51

2.3.3	Avaliação de Expressões com Alta Precisão	52
2.3.4	Arredondamento	52
2.3.5	Aritmética Dinâmica de Múltipla Precisão	53
2.3.6	Operações em C-XSC	53
2.3.7	Entrada e Saída de dados em C-XSC	54
2.3.8	CToolbox	55
2.4	Síntese	55
3	ESTUDO DE CASO: FLUXO DE POTÊNCIA EM REDES DE ENER- GIA ELÉTRICA	57
3.1	O Fluxo de Potência	57
3.1.1	Classificação das Barras	59
3.1.2	Formulação Matemática do Problema	60
3.1.3	Modelagem Matemática do Problema do Fluxo de Potência	61
3.1.4	Método de Newton-Raphson (Abordagem pontual)	63
4	METODOLOGIAS PROPOSTAS	65
4.1	Entrada de dados	66
4.2	Diagrama Unifilar	69
4.3	Método de Krawczyk (Abordagem Intervalar)	69
4.4	Proposta Via Matemática Intervalar	70
4.4.1	Conversão Intervalar das Demandas	70
4.4.2	Erros de Leitura em Instrumentos	71
4.4.3	Variação de Demanda no Tempo	71
4.4.4	Implementação	73
4.5	Proposta Via Números Fuzzy	75
4.5.1	Geração das Demandas como Números Fuzzy	75
4.5.2	Implementação	76
4.5.3	Interpretação da Solução	76
4.6	Descrição dos Recursos Computacionais Utilizados	77
4.7	Síntese	77
5	RESULTADOS NUMÉRICOS	79
5.1	IEEE-6 Barras	80
5.1.1	Resultado Pontual	80
5.1.2	Erro no Equipamento de Medição - Resultado Intervalar pelo MatLab (IntLab)	80
5.1.3	Erro no Equipamento de Medição - Resultado Intervalar pelo C++ (C-XSC)	81
5.1.4	Erro no Equipamento de Medição - Resultado com Números Fuzzy	82

5.1.5	Resultados Numéricos para o Problema da Variação da Demanda no Tempo	85
5.2	SSB-352 Barras	86
5.2.1	Resultado Pontual	86
5.2.2	Erro no Equipamento de Medição - Resultado Intervalar pelo C++ (C-XSC)	87
5.2.3	Erro no Equipamento de Medição - Resultado pela Lógica Fuzzy	87
5.2.4	Resultados Numéricos para o Problema da Variação da Demanda no Tempo	91
6	CONCLUSÕES	93
6.1	Trabalhos Futuros	95
6.2	Publicações Obtidas	95
	REFERÊNCIAS	96
ANEXO A	TUTORIAL DE INSTALAÇÃO	101
A.1	Configuração do equipamento de teste	101
A.2	Instalação da IDE Anjuta	101
A.3	Instalação da Biblioteca C-XSC	105
A.4	Configuração do Anjuta para o C-XSC	106

LISTA DE FIGURAS

Figura 2.1	Representação Geométrica de \mathbb{IR}	26
Figura 2.2	Adição Intervalar	27
Figura 2.3	Inclusão Intervalar	28
Figura 2.4	Representação Geométrica da União de dois Intervalos	29
Figura 2.5	Representação Geométrica da Intersecção de dois Intervalos	29
Figura 2.6	Representação Geométrica do Ponto Médio	29
Figura 2.7	Representação Geométrica da Distância entre dois Intervalos	30
Figura 2.8	Representação Geométrica do Diâmetro de um Intervalo	30
Figura 2.9	Módulo de um Intervalo	30
Figura 2.10	Conceito Clássico de “Quente”	32
Figura 2.11	Conceito Fuzzy de “Quente”	33
Figura 2.12	Conjunto Fuzzy de “Números Pequenos”	34
Figura 2.13	Conjunto Fuzzy de “Alto”	35
Figura 2.14	Função Crescente e Decrescente, respectivamente	36
Figura 2.15	Funções S e Z, respectivamente	36
Figura 2.16	Funções do Tipo Sino	37
Figura 2.17	Função Trapezoidal	38
Figura 2.18	Função Triangular	39
Figura 2.19	Sistemas de Inferência Fuzzy	39
Figura 2.20	Método de Defuzzificação Centro da Área	41
Figura 2.21	α –nível de um Conjunto Fuzzy	42
Figura 2.22	Intervalo α –nível de um Número Fuzzy	43
Figura 2.23	Número Fuzzy Triangular	45
Figura 2.24	Número Fuzzy Triangular	46
Figura 2.25	Representação Gráfica da Operação de Adição entre dois Números Fuzzy	47
Figura 2.26	Representação Gráfica da Operação de Subtração entre dois Números Fuzzy	48

Figura 2.27	Representação Gráfica da Operação de Multiplicação entre Números Fuzzy	49
Figura 3.1	Esquema de Distribuição de Energia	58
Figura 3.2	Usina Geradora, Barra PV	59
Figura 3.3	Subestação, Barra PQ	60
Figura 4.1	Arquivo de Entrada de Dados	67
Figura 4.2	Representação do Diagrama Unifilar da Rede Elétrica	69
Figura 4.3	Exemplo de Variação de Demanda no Sistema	72
Figura 4.4	Diagrama de Resolução do Problema	73
Figura 4.5	Diagrama da Resolução do Problema pelos Números Fuzzy	75
Figura 4.6	Diagrama da Interpretação dos Números Fuzzy	77
Figura 5.1	Dado Fuzzy para a Demanda Ativa da Barra 3	83
Figura 5.2	Dado Fuzzy para a Demanda Reativa da Barra 3	83
Figura 5.3	Dado Fuzzy para a Demanda Ativa da Barra 5	83
Figura 5.4	Dado Fuzzy para a Demanda Reativa da Barra 5	83
Figura 5.5	Dado Fuzzy para a Demanda Ativa da Barra 6	84
Figura 5.6	Dado Fuzzy para a Demanda Reativa da Barra 6	84
Figura 5.7	Resultado Fuzzy para a Magnitude de Tensão da Barra 3	84
Figura 5.8	Resultado Fuzzy para o Ângulo da Fase de Tensão da Barra 3	84
Figura 5.9	Resultado Fuzzy para a Magnitude de Tensão da Barra 5	85
Figura 5.10	Resultado Fuzzy para o Ângulo da Fase de Tensão da Barra 5	85
Figura 5.11	Resultado Fuzzy para a Magnitude de Tensão da Barra 6	85
Figura 5.12	Resultado Fuzzy para o Ângulo da Fase de Tensão da Barra 6	85
Figura 5.13	Dado Fuzzy para a Demanda Ativa da Barra 5	87
Figura 5.14	Dado Fuzzy para a Demanda Reativa da Barra 5	87
Figura 5.15	Dado Fuzzy para a Demanda Ativa da Barra 36	88
Figura 5.16	Dado Fuzzy para a Demanda Reativa da Barra 36	88
Figura 5.17	Dado Fuzzy para a Demanda Ativa da Barra 37	88
Figura 5.18	Dado Fuzzy para a Demanda Reativa da Barra 37	88
Figura 5.19	Dado Fuzzy para a Demanda Ativa da Barra 186	88
Figura 5.20	Dado Fuzzy para a Demanda Reativa da Barra 186	88
Figura 5.21	Dado Fuzzy para a Demanda Ativa da Barra 232	89
Figura 5.22	Dado Fuzzy para a Demanda Reativa da Barra 232	89
Figura 5.23	Resultado Fuzzy para a Magnitude da Tensão da Barra 5	89
Figura 5.24	Resultado Fuzzy para o Ângulo de Fase da Tensão da Barra 5	89
Figura 5.25	Resultado Fuzzy para a Magnitude da Tensão da Barra 36	90

Figura 5.26	Resultado Fuzzy para o Ângulo de Fase da Tensão da Barra 36	90
Figura 5.27	Resultado Fuzzy para a Magnitude da Tensão da Barra 37	90
Figura 5.28	Resultado Fuzzy para o Ângulo de Fase da Tensão da Barra 37	90
Figura 5.29	Resultado Fuzzy para a Magnitude da Tensão da Barra 186	90
Figura 5.30	Resultado Fuzzy para o Ângulo de Fase da Tensão da Barra 186 . . .	90
Figura 5.31	Resultado Fuzzy para a Magnitude da Tensão da Barra 232	91
Figura 5.32	Resultado Fuzzy para o Ângulo de Fase da Tensão da Barra 232 . . .	91

LISTA DE TABELAS

Tabela 2.1	Funções de Arredondamento Direcionados do C-XSC	52
Tabela 2.2	Operadores em C-XSC	54
Tabela 4.2	Análise Intervalar das Demandas	72
Tabela 5.1	Principais Características para os Sistemas Testados	79
Tabela 5.2	Sistema Pontual IEEE-6 Barras	80
Tabela 5.3	MatLab: Sistema Intervalar IEEE-6 Barras	81
Tabela 5.4	C++: Sistema Intervalar IEEE-6 Barras	81
Tabela 5.5	Resultado Intervalar do Sistema IEEE 6 Barras	86
Tabela 5.6	Resultado Pontual para SSB-352 Barras	86
Tabela 5.7	Resultados para o SSB-352 barras	87
Tabela 5.8	Resultado Intervalar do Sistema SSB 352 Barras (Variação da De- manda no Tempo)	91

LISTA DE CÓDIGOS

1.1.1 Codificação em C++: Operação Fracionária	19
2.3.1 Codificação em C-XSC: Operadores Predefinidos	51
2.3.2 Codificação em C-XSC: Uso de Intervalos	52
2.3.3 Codificação em C-XSC: Arredondamento Direcionado	53
2.3.4 Codificação em C-XSC: Múltipla Precisão	53
2.3.5 Codificação em C-XSC: Operador Intersecção	54

LISTA DE ABREVIATURAS E SIGLAS

BIAS	Basic Interval Arithmetic Subroutines
C-XSC	C++ Class Library for eXtended Scientific Computing
CAPES	Coordenação de Aperfeiçoamento de Pessoal de Nível Superior
CoA	Método Centro da Área
E/S	Entrada e Saída
GMFC	Grupo de Matemática e Fundamentos da Computação
IAM-UKA	Institute Für Angewandte Mathematik - Universität Karlsruhe
IEEE	Institute of Electrical and Eletronics Engineers
IntLab	Interval Laboratory
LASE	Laboratório de Sistemas de Energia
MVAr	Mega Volt. Ampere reativo
MW	Mega Watts
PMAPS	Probabilistic Methods Applied to Power Systems
PQ	Subestação
PROFIL/BIAS	Programmer's Runtime Optimized Fast Interval Library
PSCC	Power Systems Computation Conference
pu	phase unbalance
PV	Usina Geradora
SSB	Região Sul e Sudeste do Brasil
TEMA	Tendências em Matemática Aplicada e Computacional
TFN	Número Fuzzy Triangular

UCPel	Universidade Católica de Pelotas
UPF	Universidade de Passo Fundo
UFRN	Universidade Federal do Rio Grande do Norte
UTE	Usina Termelétrica
V	Volt
XSC	eXtensions for Scientific Computation

RESUMO

Em análise de sistemas de potência, uma das principais ferramentas à disposição dos operadores da rede é o cálculo do fluxo de potência. O resultado dessa análise fornece o perfil de tensões do sistema, que é o ponto de partida para o planejamento da operação. Esta análise está baseada no conhecimento a priori do nível de demandas de potências ativa e reativa nas barras da rede. Estes valores de demanda são obtidos por instrumentos de medição ou, em determinadas situações, por uma estimativa de carga. Assim, o presente estudo tem por finalidade propor o desenvolvimento de uma ferramenta computacional para análise das influências dos erros de medição e dos erros gerados na computação numérica no problema de fluxo de potência. Para a representação desses dados imprecisos, buscou-se, na computação científica, duas soluções: (i) a Matemática Intervalar; e (ii) a Lógica Fuzzy. A Matemática Intervalar é altamente eficaz no controle dos diversos tipos de erros e operações computacionais, pois controla de maneira automática todos os arredondamentos criados pelo sistema de ponto-flutuante e propicia que o resultado final seja de máxima exatidão do ponto de vista computacional. Por outro lado, a Lógica Fuzzy é um sistema de apoio à tomada de decisão que mapeia o conhecimento aproximado, impreciso ou incerto, podendo ser definida como uma ferramenta capaz de capturar informações vagas, em geral descritas em uma linguagem natural, e convertê-las para um formato numérico, de fácil manipulação pelos computadores atuais. Com a aplicação destas abordagens matemáticas, desenvolveu-se um software para a solução deste tipo de problema no sistema operacional Linux, na linguagem de programação C++. Para a implementação desta solução, utilizou-se a biblioteca intervalar denominada C-XSC. Na validação da metodologia proposta, são apresentados resultados para um sistema-teste do IEEE e um sistema real representando a região Sul-Sudeste do Brasil.

Palavras-chave: C-XSC, Fluxo de Potência, Lógica Fuzzy, Matemática Intervalar.

TITLE: “COMPUTATIONAL MATHEMATICAL TECHNIQUES FOR TREATING UNCERTAINTIES APPLIED TO POWER FLOW PROBLEM IN ELECTRIC ENERGY TRANSMISSION SYSTEMS”

ABSTRACT

In analysis of power systems, one of the main tools available for helping the transmission operators is the power flow analysis. The result of the power flow problem supplies the voltage profile to the operators, which is the starting point for the planning operations. This analysis is based on the knowledge a priori of the levels of the demands of active and reactive powers in the transmission bars. These values are gotten by measurement instruments or, in some determined situations, by a estimative of the load. Thus, the objective of the present study is to present a computational tool for the analysis of the influences of the measurement errors and the errors generated by numerical computations in the power flow problem. For the representation uncertain data, one searched in the Scientific Computation, two solutions: (i) Interval Mathematics; and (ii) Fuzzy Logic. Interval Mathematics is highly efficient in the control of the diverse types of computational errors and operations, and, therefore it controls in automatic way all the roundings created by the point-floating platform system and propitiates that the final result is of maximum exactness from the computational point of view. On the other hand, Fuzzy Logic is a support system for the decision making with uncertain knowledge, allowing to capture vague information, in general described in natural language, and converting them for a numerical format, of easy manipulation by current computers. Applying these two mathematical approaches, a software for the solution of this problem was developed, using the programming language C++ in the operational system Linux. For the implementation of this solution, the interval library called C-XSC it was used. For the validation of the proposed methodology, results for a IEEE test-system and of a real system representing the South-Southeast region of Brazil were presented.

Keywords: C-XSC, Electric Power Systems, Fuzzy Logic, Interval Mathematics.

1 INTRODUÇÃO

1.1 Motivação

Para a solução matemático-computacional de um problema físico, é possível ocorrer diversas fontes de erros que podem alterar profundamente os resultados obtidos. Dessa forma, torna-se fundamental o conhecimento das causas desses erros para minimizar as suas conseqüências.

Consideram-se basicamente três tipos de erros: os erros de dados, os de modelagem e os erros numéricos.

Quando se utiliza um equipamento de medição, acredita-se que o valor medido está correto. Em princípio este instrumento está calibrado, então a medida realizada está correta. Será mesmo? Supondo que duas pessoas realizem a mesma verificação, quais serão os resultados? Pode-se afirmar que possivelmente os resultados não serão exatamente iguais, apesar de virem do mesmo equipamento. O que poderia estar errado? Os erros de dados podem ocorrer por estimativas de medições, por falha humana ou por instrumentos de medição. Os próprios instrumentos de medição possuem uma tolerância, onde o valor medido possui um determinado grau de certeza, mas conhece-se o valor da imprecisão obtida.

Na etapa da modelagem de um problema real, procura-se obter um modelo matemático que represente o mais fielmente possível o fenômeno real. Na construção de um modelo que represente matematicamente uma situação real são necessárias várias simplificações, ignorando-se algumas variáveis e condicionantes do sistema que está a ser modelado, que se consideram menos importantes para o problema em causa. Nesta fase, portanto, pode ocorrer uma modelagem incorreta, na qual o modelo matemático escolhido não represente a melhor solução do fenômeno físico em estudo. Também pode ocorrer um erro no desenvolvimento e/ou escolha do algoritmo. Para ambos os casos, o resultado final obtido pode apresentar uma informação com baixa exatidão.

Os computadores erram? Imagina-se que eles jamais irão cometer erros. Entretanto, os computadores nem sempre geram uma informação correta, exata. O estudo da

aritmética do ponto de vista computacional apresenta algumas diferenças consideráveis em relação à aritmética real. Algumas propriedades básicas da aritmética real não valem mais quando executadas no computador, pois enquanto na matemática real alguns números são representados por infinitos dígitos, no computador isso não é possível, pois a memória do computador é finita. Também ocorrem erros para a representação em ponto flutuante. Todo número inteiro decimal pode ser representado por um número inteiro binário; porém, isto não é verdadeiro para a maioria dos números fracionários.

O Código 1.1.1 apresenta um programa simples ilustrando o problema de erro computacional e está implementado em C++.

```
1  #include <iostream>
2  using namespace std;
3
4  int main()
5  {
6      int i;
7      float soma = 0;
8      for (i = 1; i <= 10000; i++)
9      {
10         soma = soma + .0001;
11     }
12     cout << soma << endl;          //1.00005
13     return 0;
14 }
```

Código 1.1.1: Codificação em C++: Operação Fracionária

O Código 1.1.1 realiza a soma do número fracionário 0,00001 mil vezes, o resultado fornecido pelo algoritmo é 1.00005 ao invés do número exato 1. Esse pequeno erro computacional da representação do número decimal 0,0001 em binário se propagará pela soma, comprometendo o resultado da soma final.

Para a representação desses dados imprecisos, buscou-se na computação científica duas soluções: (i) a Matemática Intervalar; e (ii) a Lógica Fuzzy. A Matemática Intervalar manipula dados e parâmetros na forma de intervalos, com o objetivo dos computadores suportarem uma aritmética mais poderosa, controlando de maneira automática todos os arredondamentos criados pelo sistema de ponto-flutuante e propicia que o resultado final seja de máxima exatidão do ponto de vista computacional. Por outro lado, a Lógica Fuzzy é um sistema de apoio à tomada de decisão que mapeia o conhecimento aproximado, impreciso ou incerto, podendo ser definida como uma ferramenta capaz de capturar informações vagas, em geral descritas em uma linguagem natural, e convertê-las para um formato numérico, de fácil manipulação pelos computadores atuais.

Como uma aplicação prática, onde os erros discutidos podem aparecer e pode-se aplicar a Matemática Intervalar e a Lógica Fuzzy, tem-se o fluxo de potência em redes

de energia elétrica. A demanda do sistema, ou seja, o consumo de energia é obtido por instrumentos de medição e por estimativas de consumo. A exatidão destes dados é de fundamental importância para que se possa realizar uma estimativa do perfil de tensões do sistema elétrico.

Desta estimação, dependerá toda a operação da rede de uma forma segura e confiável. Sendo assim, realizar-se uma análise de fluxo de carga confiável, rigorosa e automática é essencial para que se tente evitar a propagação dos erros durante o processo computacional bem como realizar-se um controle dos erros de arredondamento e truncamento inerentes ao processo matemático. Uma das formas encontradas para representar esses dados incertos é a sua utilização através de dados na forma de intervalos e de pertinências. Portanto, a Matemática Intervalar e a Lógica Fuzzy podem ser utilizadas para a representação dessas demandas energéticas das quais não se detém uma certeza completa sobre seus valores.

1.2 Contexto Histórico

Esta seção apresenta os principais fatos que conduziram ao surgimento da Matemática Intervalar, da Lógica Fuzzy e o ambiente no qual a biblioteca intervalar C-XSC foi desenvolvida.

- Matemática Intervalar: Para (WIENER, 1914) (WIENER, 1920), no ano de 1914, Norbert Wiener teria sido uma das primeiras pessoas a utilizar intervalos para descrever resultados de medições de distâncias e tempo. Também Burkill e Young (BURKILL, 1924; YOUNG, 1931), descrevem uma forma de aritmética intervalar. No entanto, os primeiros registros de trabalhos independentes devem-se a P. S. Dwyer, M. Warmus, T. Sunaga e R. E. Moore (DWYER, 1951; WARMUS, 1956; SUNAGA, 1958; MOORE, 1959).

Mas foram os artigos de Moore e em particular a sua tese datada de 1962, onde introduz a aritmética intervalar, os principais responsáveis pelo despertar de um trabalho de investigação ativo e contínuo (MOORE, 1966).

Por outro lado, a Matemática Intervalar foi proposta por Leslie Fox em 1974, agrupando diferentes áreas como a Aritmética Intervalar, Análise Intervalar, Topologia Intervalar, Álgebra Intervalar e outras.

Inicialmente, algumas críticas foram feitas à técnica da Matemática Intervalar, a qual segundo os críticos apresentava dois problemas:

- fornecia resultados com intervalos pessimistas, ou seja, demasiadamente grandes, os quais não retornavam uma informação relevante
- necessitava de um grande tempo de processamento computacional

Essas duas críticas foram descartadas (HöLBIG, 2005), uma vez que, com a utilização da aritmética avançada, os resultados são produzidos com a máxima exatidão. E, quanto ao tempo de processamento, este pode ser consideravelmente reduzido dependendo da forma como o algoritmo é implementado, existindo várias possibilidades tanto via software quanto via hardware (para exemplificar essa afirmação ver (RUMP, 1999)).

- **Lógica Fuzzy:** Na antiga Grécia, Aristóteles introduziu as Leis do Conhecimento, que posteriormente seriam o sustento para a Lógica Clássica. Suas três leis fundamentais eram:
 - Princípio de Identidade;
 - Lei da Contradição;
 - Lei do Terceiro Excluído.

A Lei do Terceiro Excluído afirma que para toda proposição P , é p ou $\sim p$, uma destas duas deve ser verdadeira, sem que haja nenhuma proposição entre elas. Em princípio, pode-se pensar que é assim, porém, certamente surgem dúvidas ao pensar em um copo pela metade, o copo estar cheio seria uma proposição p , o copo estando vazio seria $\sim p$. Assim, pode-se concluir que o copo está cheio ou vazio, não havendo uma proposição para este meio termo (o copo está pela metade).

De acordo com (ALTROCK, 1997; RUSS; SIMPSON; DOBBINS, 1996) a Lógica Fuzzy ou Lógica Difusa foi estruturada em 1965, com uma publicação de Lotfi A. Zadeh (ZADEH, 1965), surgindo uma nova teoria de conjuntos. Professor da Universidade da Califórnia, Berkeley, considerado um grande colaborador do controle moderno, Zadeh criou uma teoria de conjuntos em que não há descontinuidades, ou seja, não há uma distinção abrupta entre elementos pertencentes e não pertencentes a um conjunto, são os Conjuntos Nebulosos. Desse ponto, começou a se desenvolver a Lógica Fuzzy, para tratar de variáveis “*imprecisas*”, ou definidas de forma “*vaga*”.

As primeiras aplicações industriais na Europa, ocorreram após 1970. Em Londres, Ebrahim Mamdani usou Lógica Fuzzy para controlar um gerador a vapor. Na Alemanha, Hans Zimmermann usou Lógica Fuzzy para sistemas de apoio à decisão. Ainda nesta época, as poucas aplicações que usavam Lógica Fuzzy, esconderam o fato circunscrevendo Lógica Fuzzy com condições como “lógica de múltiplos valores” ou “lógica contínua”. Após 1980, a Lógica Fuzzy ganhou mais impulso em apoio de decisão e aplicações de análise de dados na Europa. Inspirado pelas primeiras aplicações de Lógica Fuzzy européias, companhias japonesas começaram

a usar Lógica Fuzzy em engenharia de controle. As primeiras aplicações de Lógica Fuzzy foram uma planta de tratamento de água da Fuji Electric em 1983 e um sistema de metrô da Hitachi em 1987. A Lógica Fuzzy é muito usada na área de controle de processos. Máquinas fotográficas e filmadoras usam Lógica Fuzzy. A maioria dos carros japoneses usam Lógica Fuzzy em alguns dos seus componentes. Devido ao grande sucesso, a Lógica Fuzzy é considerada um padrão em sistemas de controle, e tem larga aceitação na comunidade de engenharia.

- C-XSC: A ausência de linguagens que forneçam ferramentas específicas de alta exatidão tornou-se aparente em programação numérica e computação científica. Mesmo com a evolução das linguagens de programação, a programação de algoritmos numéricos não foi amplamente facilitada e nem passou a fornecer resultados mais confiáveis (KOLBERG, 2002).

Desde a década de sessenta são desenvolvidas pesquisas com o objetivo de possibilitar que os computadores suportem uma aritmética mais poderosa que a aritmética de ponto-flutuante ordinária. Estas implementações requerem suporte aritmético que seja satisfatório, e ferramentas de programação poderosas que não estão geralmente disponíveis.

Esta exigência fez com que fosse necessário estender linguagens para a computação científica. Neste trabalho, optou-se por utilizar a extensão XSC, mais precisamente a biblioteca C-XSC.

Além das linguagens com a extensão XSC, alguns softwares relacionados com o uso de intervalos estão apresentados a seguir (conforme (HöLBIG, 2005)):

- Compilador FORTE C++ e FORTE FORTRAN/HPC: este compilador foi desenvolvido pela Sun Microsystems que suporta o uso da aritmética intervalar. Site: <http://developers.sun.com/>
- Compilador GNU FORTRAN: projeto que modificou o compilador GNU para suportar a aritmética intervalar (SCHULTE; SWARTZLANDER, 1996).
- IntLab (*Interval Laboratory*): é um *toolbox* para o MatLab que tem suporte para a aritmética intervalar. Foi desenvolvido com o objetivo de ser rápida e possibilitar o uso de processamento paralelo. A característica de velocidade, é obtida devido ao uso de rotinas da biblioteca BLAS. Um outro aspecto dessa biblioteca, é que possibilita um ambiente de programação interativo, de fácil utilização de rotinas intervalares (RUMP, 1999). Uma comparação qualitativa dessa biblioteca com a do C-XSC é apresentada no capítulo de resultados.
- IntPakX: é um pacote que disponibiliza o desenvolvimento de algoritmos com verificação numérica para o Maple (GRIMMER; PETRAS; REVOL, 2004),

(KRÄMER, 2002).

- Linguagens de programação para a computação científica (linguagens ou bibliotecas XSC): Essas linguagens são formadas pelas bibliotecas Pascal-XSC, C-XSC e ACRITH-XSC (a biblioteca C-XSC será mais detalhada nesta dissertação). Pascal-XSC e C-XSC foram desenvolvidas em 1988 pela Universidade de Karlsruhe (Alemanha), e atualmente é mantida pela Universidade de Wuppertal (Alemanha). O *download* da biblioteca está disponível em: <http://www.xsc.de>.
- PROFIL/BIAS (*Programmer's Runtime Optimized Fast Interval Library*): é uma biblioteca de classes em C++ que suporta operações com números reais e intervalares. Site: <http://www.ti3.tu-harburg.de/english/index.html>.
- Toolbox b4m: é um *toolbox* para o MatLab baseado no BIAS (*Basic Interval Arithmetic Subroutines*) (KNÜPPEL, 1993) tem suporte à aritmética intervalar. Permite que sejam desenvolvidas rotinas no MatLab e as converte para C++. Mais informações podem ser encontradas em (ZEMKE, 1998).

Informações sobre outras bibliotecas/ferramentas para a utilização na computação científica podem ser obtidas no seguinte endereço: <http://www.cs.utep.edu/interval-comp/intsoft.html>.

1.3 Objetivos

O objetivo geral desta dissertação de mestrado é analisar algoritmos para o tratamento das incertezas em fluxo de potência em redes de energia elétrica utilizando como metodologia a Matemática Intervalar e a Lógica Fuzzy. Posteriormente, implementar uma aplicação (software) para o sistema operacional Linux utilizando como recurso de implementação a biblioteca C-XSC.

Por outro lado, os objetivos específicos são:

- estudar a Matemática Intervalar;
- estudar a Lógica Fuzzy;
- analisar algoritmos baseados na Matemática Intervalar para tratamento de incertezas em sistema elétrico de potência;
- estudar a biblioteca C-XSC para cálculos intervalares;
- implementar computacionalmente a aplicação.

1.4 Organização do Texto

Este texto se divide em 6 capítulos. O capítulo 2 trata do estudo do referencial teórico onde são abordados os seguintes aspectos: a teoria da Matemática Intervalar, a conceitualização da Lógica Fuzzy e as principais funcionalidades da biblioteca C-XSC. No capítulo 3, tem-se o estudo de caso, onde discute-se o problema do fluxo de potência em redes de energia elétrica. Para o capítulo 4, são apresentadas as metodologias propostas para a resolução do problema. No capítulo 5, apresenta-se a comparação do algoritmo implementado utilizando-se duas plataformas operacionais distintas: o ambiente de programação do MatLab e a linguagem C++ com o uso da biblioteca C-XSC, e os resultados oriundos da aplicação da Lógica Fuzzy. Finalmente a conclusão e trabalhos futuros são discutidas no capítulo 6.

2 ESTADO DA ARTE

Este capítulo apresenta o estudo de duas técnicas da computação científica, a Matemática Intervalar e a Lógica Fuzzy. Essas duas técnicas foram empregadas para o tratamento das incertezas no fluxo de potência em redes de energia elétrica. Para a aplicação da Matemática Intervalar, utilizou-se uma biblioteca que possibilita o uso de intervalos, a biblioteca C-XSC.

2.1 A Matemática Intervalar

A qualidade do resultado na Computação Científica depende do conhecimento e controle dos erros computacionais nos dados. Para resolver esse problema surgiu a Matemática Intervalar baseada na aritmética de Moore (MOORE, 1966, 1979; ACIÓLY, 1991; BEDREGAL, 1996; LYRA, 1999; SANTIAGO, 1999; DUTRA, 2000; CRUZ, 2000).

A Matemática Intervalar busca resolver problemas que concentram-se basicamente em dois aspectos: (i) na elaboração de um modelo computacional que reflita o controle e análise dos erros que poderão ocorrer durante o processamento computacional, e (ii) na escolha mais adequada das técnicas de programação para o desenvolvimento de softwares na computação científica, em busca da minimização dos erros em seus resultados.

A Matemática Intervalar considera um conjunto de métodos para manipulação de intervalos numéricos que aproximam dados incertos. Segundo (OLIVEIRA; DIVERIO; CLAUDIO, 2001), na computação científica, os intervalos podem ser aplicados para representar valores desconhecidos e, também para representar valores contínuos. Servem para controlar o erro de arredondamento e para representar dados inexatos, aproximações e erros de truncamento de procedimentos. Estes métodos baseiam-se na definição da aritmética intervalar e do produto escalar ótimo (KULISCH, 2002).

Para (DIMURO, 1991), resultados intervalares carregam consigo a segurança de sua qualidade e o grau de sua incerteza, pois o diâmetro do intervalo solução é um indicativo da influência dos erros dos dados de entrada e dos erros de arredondamento e truncamento no erro do resultado final obtido.

2.1.1 Conjunto \mathbb{IR}

Define-se \mathbb{IR} como sendo o conjunto de todos os intervalos reais.

$$\mathbb{IR} = \{[x_1; x_2] \mid x_1, x_2 \in \mathbb{R}, x_1 \leq x_2\} \quad (2.1)$$

Observe na figura 2.1 que associando-se a cada intervalo $[x_1; x_2] \in \mathbb{IR}$, um ponto $(x_1, x_2) \in \mathbb{R}^2$, obtém-se a representação geométrica para \mathbb{IR} .

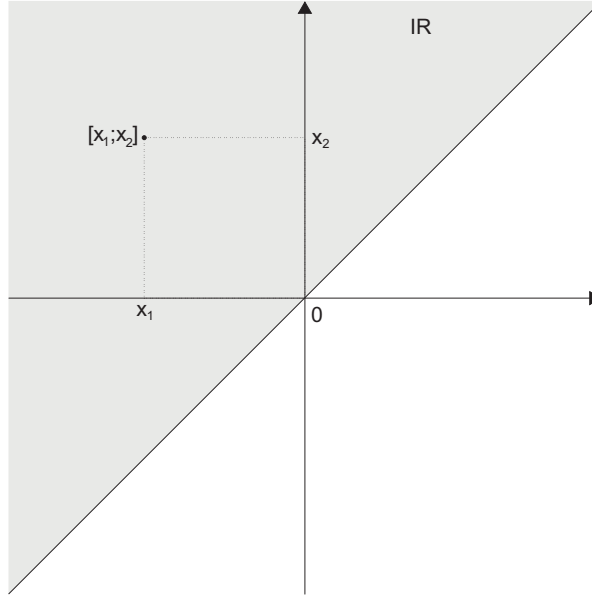


Figura 2.1: Representação Geométrica de \mathbb{IR}

Destacam-se a seguir, as principais operações na Matemática Intervalar.

2.1.2 Operações Básicas

As operações básicas intervalares como adição, subtração, multiplicação e divisão intervalar, estão definidas como segue.

2.1.2.1 Adição Intervalar

Sejam, $X, Y \in \mathbb{IR}$ dois intervalos reais, com $X = [x_1; x_2]$ e $Y = [y_1; y_2]$, a adição intervalar de $X + Y$ é dada na Equação 2.2.

$$X + Y = [(x_1 + y_1); (x_2 + y_2)] \quad (2.2)$$

A Figura 2.2, com base na Equação 2.2, demonstra graficamente a disposição dos intervalos $X = [10; 20]$, $Y = [30; 39]$ e a soma intervalar de $X + Y = [10 + 30; 20 + 39] = [40; 59]$.

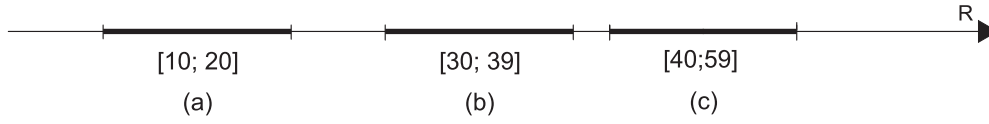


Figura 2.2: Adição Intervalar

Sejam X, Y e $Z \in \mathbb{IR}$. As seguintes propriedades algébricas se aplicam à adição intervalar em \mathbb{IR} :

- **fechamento:** Se $X, Y \in \mathbb{IR}$ então $X + Y \in \mathbb{IR}$;
- **associatividade:** $X + (Y + Z) = (X + Y) + Z$;
- **comutatividade:** $X + Y = Y + X$;
- **elemento neutro:** $\exists 0 = [0; 0] \in \mathbb{IR}$ tal que $X + 0 = 0 + X = X$.

2.1.2.2 Pseudo Inverso Aditivo Intervalar

Seja $X \in \mathbb{IR}$ um intervalo de reais, com $X = [x_1; x_2]$. O pseudo inverso aditivo intervalar é dado conforme a Equação 2.3.

$$-X = [-x_2; -x_1] \quad (2.3)$$

Exemplo: Seja $X = [-4; 9]$, tem-se $-X = [-9; 4]$.

2.1.2.3 Subtração Intervalar

Sejam, $X, Y \in \mathbb{IR}$ dois intervalos reais, com $X = [x_1; x_2]$ e $Y = [y_1; y_2]$, a subtração intervalar de $X - Y$ é dada na Equação 2.4.

$$X - Y = [(x_1 - y_2); (x_2 - y_1)] \quad (2.4)$$

Exemplo: Sejam, $X = [-1; 4]$ e $Y = [3; 8]$. Substituindo X e Y na Equação 2.4 tem-se para $X - Y = [(-1) - 8; 4 - 3] = [-9; 1]$.

2.1.2.4 Multiplicação Intervalar

Sejam, $X, Y \in \mathbb{IR}$ dois intervalos reais, com $X = [x_1; x_2]$ e $Y = [y_1; y_2]$ a multiplicação intervalar de $X \times Y$ é dada na Equação 2.5.

$$X \times Y = [\min \{x_1 \times y_1, x_1 \times y_2, x_2 \times y_1, x_2 \times y_2\}; \max \{x_1 \times y_1, x_1 \times y_2, x_2 \times y_1, x_2 \times y_2\}] \quad (2.5)$$

Exemplo: Sendo $X = [-2; 3]$ e $Y = [4; 5]$, tem-se a multiplicação intervalar de $X \times Y = [\min \{(-2) \times 4, (-2) \times 5, 3 \times 4, 3 \times 5\}; \max \{(-2) \times 4, (-2) \times 5, 3 \times 4, 3 \times 5\}] = [-10; 15]$.

Sejam X, Y e $Z \in \mathbb{IR}$. As seguintes propriedades algébricas se aplicam à multiplicação intervalar em \mathbb{IR} :

- **fechamento:** Se $X, Y \in \mathbb{IR}$ então $X \times Y \in \mathbb{IR}$;
- **associatividade:** $X \times (Y \times Z) = (X \times Y) \times Z$;
- **comutatividade:** $X \times Y = Y \times X$;
- **elemento neutro:** $\exists 1 = [1; 1] \in \mathbb{IR}$ tal que $X \times 1 = 1 \times X = X$;
- **Subdistribuidividade:** $X \times (Y + Z) \subseteq (X \times Y) + (X \times Z)$.

2.1.2.5 Pseudo Inverso Multiplicativo Intervalar

Seja $X \in \mathbb{IR}$ um intervalo de reais, tal que $0 \notin X$ tem-se o pseudo inverso multiplicativo intervalar na Equação 2.6.

$$X^{-1} = \frac{1}{X} = \left[\frac{1}{x_2}; \frac{1}{x_1} \right] \quad (2.6)$$

Exemplo: Seja $X = [3; 4]$, tem-se $X^{-1} = \left[\frac{1}{4}; \frac{1}{3} \right]$.

2.1.2.6 Divisão Intervalar

Sejam, $X, Y \in \mathbb{IR}$ dois intervalos reais, com $X = [x_1; x_2]$, $Y = [y_1; y_2]$ e $0 \notin Y$. Define-se a divisão intervalar de X por Y na Equação 2.7.

$$\frac{X}{Y} = \left[\min \left\{ \frac{x_1}{y_1}, \frac{x_1}{y_2}, \frac{x_2}{y_1}, \frac{x_2}{y_2} \right\}; \max \left\{ \frac{x_1}{y_1}, \frac{x_1}{y_2}, \frac{x_2}{y_1}, \frac{x_2}{y_2} \right\} \right] \quad (2.7)$$

Exemplo: Sejam $X = [2; 4]$ e $Y = [1; 3]$, tem-se $\frac{X}{Y} = \left[\min \left\{ \frac{2}{1}, \frac{2}{3}, \frac{4}{1}, \frac{4}{3} \right\}; \max \left\{ \frac{2}{1}, \frac{2}{3}, \frac{4}{1}, \frac{4}{3} \right\} \right] = [0,6666; 4]$.

2.1.3 Definições em \mathbb{IR}

2.1.3.1 Inclusão Intervalar

Dado $x \in \mathbb{IR}$, diz-se que $X \in \mathbb{IR}$ é uma inclusão intervalar de x se $x \in X$. Um exemplo pode ser visto na Figura 2.3.



Figura 2.3: Inclusão Intervalar

Na Figura 2.3 o intervalo $[3; 4]$ é uma inclusão para o número π .

2.1.3.2 Intervalo Simétrico

Seja $X \in \mathbb{IR}$ um intervalo, X é um intervalo simétrico se $-X = X$.

Exemplo: $[-1; 1]$, $[-\pi; \pi]$ e $[0; 0]$.

2.1.3.3 União entre dois Intervalos

Sejam $X = [x_1; x_2]$ e $Y = [y_1; y_2]$ dois intervalos conforme a Figura 2.4, tais que $X \cap Y \neq \emptyset$. Define-se a união dos intervalos X e Y como o intervalo $X \cup Y = [\min \{x_1; y_1\}; \max \{x_2; y_2\}]$. A representação geométrica da união de dois intervalos é mostrada na Figura 2.4.

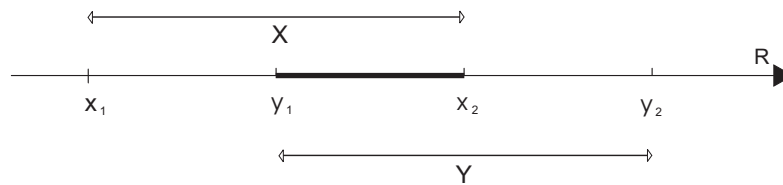


Figura 2.4: Representação Geométrica da União de dois Intervalos

2.1.3.4 Intersecção entre dois Intervalos

Sejam $X = [x_1; x_2]$ e $Y = [y_1; y_2]$ dois intervalos. Define-se a intersecção dos intervalos X e Y como sendo o intervalo $X \cap Y = [\max \{x_1; y_1\}; \min \{x_2; y_2\}]$, se $\max \{x_1; y_1\} \leq \min \{x_2; y_2\}$. Se $\min \{x_2; y_2\} < \max \{x_1; y_1\}$ então $X \cap Y = \emptyset$. A representação geométrica da intersecção é mostrada na Figura 2.5.

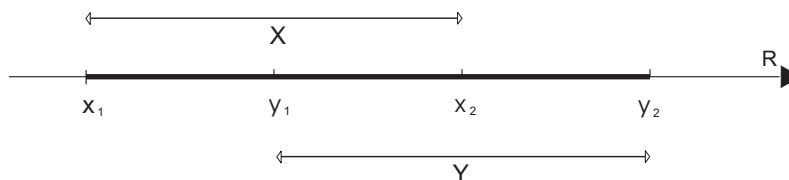


Figura 2.5: Representação Geométrica da Intersecção de dois Intervalos

2.1.3.5 Ponto Médio de um Intervalo

Seja $X = [x_1; x_2] \in \mathbb{IR}$ um intervalo conforme a Figura 2.6. Define-se o ponto médio do intervalo X como sendo o número real $m = \frac{x_1 + x_2}{2}$.

Notação: $med(X) = med([x_1; x_2]) = \frac{x_1 + x_2}{2}$.

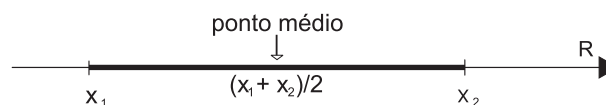


Figura 2.6: Representação Geométrica do Ponto Médio

2.1.3.6 Distância entre dois Intervalos

Sejam $X = [x_1; x_2]$ e $Y = [y_1; y_2]$ dois intervalos conforme Figura 2.7. Define-se a distância de X para Y como sendo o número real não-negativo $\delta = \max \{|x_1 - y_1|; |x_2 - y_2|\}$.

Notação:

$$\text{dist}(X, Y) = \text{dist}([x_1; x_2], [y_1; y_2]) = \max \{|x_1 - y_1|, |x_2 - y_2|\} \geq 0.$$

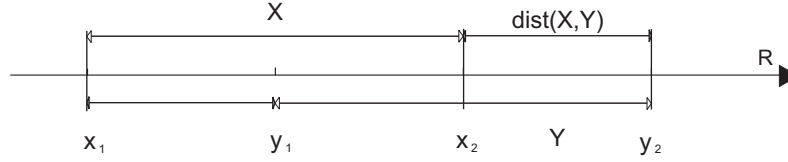


Figura 2.7: Representação Geométrica da Distância entre dois Intervalos

2.1.3.7 Diâmetro Intervalar

Seja $X = [x_1; x_2] \in \mathbb{IR}$ um intervalo conforme a Figura 2.8. Define-se diâmetro do intervalo X como sendo o número real não-negativo $d = x_2 - x_1$.

Notação: $\text{diam}(X) = \text{diam}([x_1; x_2]) = x_2 - x_1 \geq 0$.

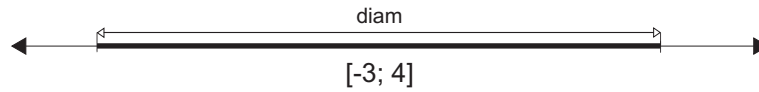


Figura 2.8: Representação Geométrica do Diâmetro de um Intervalo

A Figura 2.8 exibe o diâmetro do intervalo $[-3; 4]$ que corresponde a 7 unidades.

2.1.3.8 Módulo Intervalar

Seja $X = [x_1; x_2] \in \mathbb{IR}$ um intervalo conforme a Figura 2.9. Define-se o módulo do intervalo como X sendo o número real não-negativo $\mu = \text{dist}(X, 0)$, que corresponde à distância de X ao zero.

Notação: $|X| = |[x_1; x_2]| = d(X, 0) = \max \{|x_1|, |x_2|\} \geq 0$.

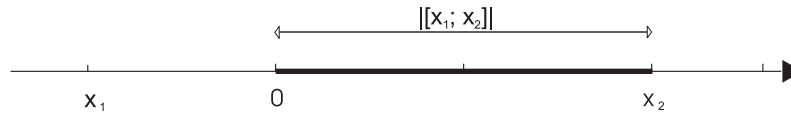


Figura 2.9: Módulo de um Intervalo

2.2 A Lógica Fuzzy

Nas últimas décadas, pesquisadores da inteligência artificial vêm trabalhando com a teoria dos Conjuntos Fuzzy, também conhecida como conjuntos nebulosos, introduzida

por Lofti A. Zadeh em 1965 (COX, 1994), (COX, 1995), (KASABOV, 1996), (TERANO; ASAI; SUGENO, 1992), (ZADEH, 1965). Mas, há poucos anos, esses estudos vêm sendo posto em prática, tendo um grande crescimento no número de pesquisas, possibilitando o desenvolvimento de máquinas com controladores Fuzzy, desde eletrodomésticos como: aspirador de pó, ar condicionado, robôs autônomos e até mesmo chips Fuzzy VLSI (KASABOV, 1996), (KOSKO, 1992), (TSOUKALAS; UHRIG, 1997).

2.2.1 Definição de Conjuntos Fuzzy

Na teoria de conjuntos clássicos, um conjunto pertence ou não pertence a um determinado conjunto. Atribuindo a noção de pertinência, a premissa de pertence ou não pertence é quebrada, um valor de pertinência (de uma função característica) é 0 ou 1 (sendo compreendido dentro do conjunto $\{0, 1\}$). A forma tradicional de representar um elemento x de um determinado conjunto através da função característica é dada pela equação 2.8.

$$\begin{aligned}\chi_A(x) &= 1, & \text{se } x \text{ é um elemento do conjunto } A, \text{ e} \\ \chi_A(x) &= 0, & \text{se } x \text{ não é um elemento do conjunto } A\end{aligned}\quad (2.8)$$

Por exemplo, a seguinte proposição “*a água está quente*”, pode ser modelada através de conjuntos clássicos definindo parâmetros, representando a temperatura da água pela variável T , onde esta representa o conjunto de todos os valores para a temperatura da água. Este conjunto pode ser definido na equação 2.9.

$$A \subseteq T, \text{ onde } A = \{x \in T : 50 \leq x \leq 90\} \quad (2.9)$$

ou também pode ser representado por uma função característica $x_a : T \rightarrow \{0, 1\}$, definida na equação 2.10.

$$x_a(x) = \begin{cases} 1 & \text{para } x \in A \\ 0 & \text{para } x \notin A \end{cases} \quad (2.10)$$

A Figura 2.10 representa o conceito “*quente*” de modo clássico, onde a temperatura entre 50 e 90 graus indica com 100% de certeza de que a água está quente; caso contrário não está.

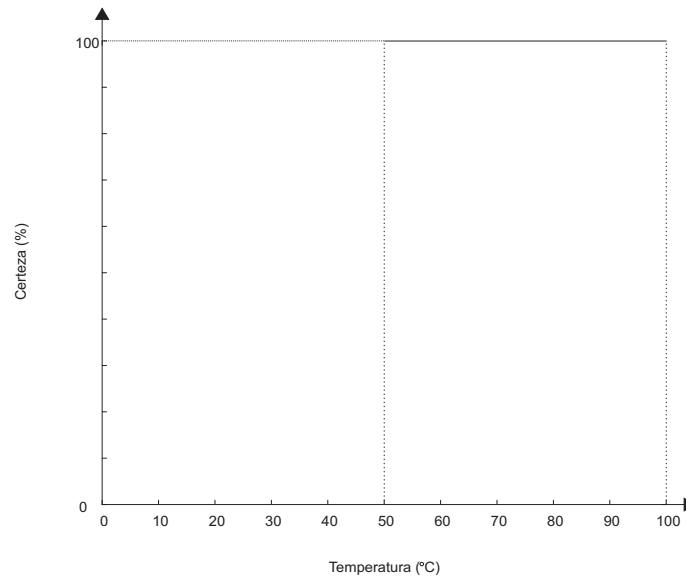


Figura 2.10: Conceito Clássico de “Quente”

Esta modelagem tem muitas aplicações e utilidades práticas, contudo, em muitos casos, sofre perda e distorção da informação. Este exemplo mostra, que a temperatura de 50 graus Celsius é considerada quente e no entanto, uma temperatura de 49°C começa a ser considerada como fria, embora que o fato de não ser quente, não signifique que seja fria.

Analizados esses valores, nota-se que é necessária a existência de uma teoria para representar os valores incertos, aproximados, como *quente*, *pequeno* ou *alto*, pois elementos com essas características exigem graus de pertinência diferentes de simplesmente valores (1) quando pertencer ou (0) quando não pertencer.

Na teoria dos Conjuntos Fuzzy um elemento pode pertencer parcialmente a um dado conjunto. O grau de pertinência é definido através de uma generalização da função característica chamada de *função de pertinência* e é definida pela equação 2.11.

$$\mu_A(x) : U \rightarrow [0, 1] \quad (2.11)$$

onde U é o conjunto universo de discurso e A é um Conjunto Fuzzy.

Os valores da função de pertinência são números reais no intervalo $[0, 1]$, onde 0 significa que o elemento não é membro do conjunto e 1 significa que ele pertence totalmente ao conjunto. Cada valor da função é chamado de *grau de pertinência*.

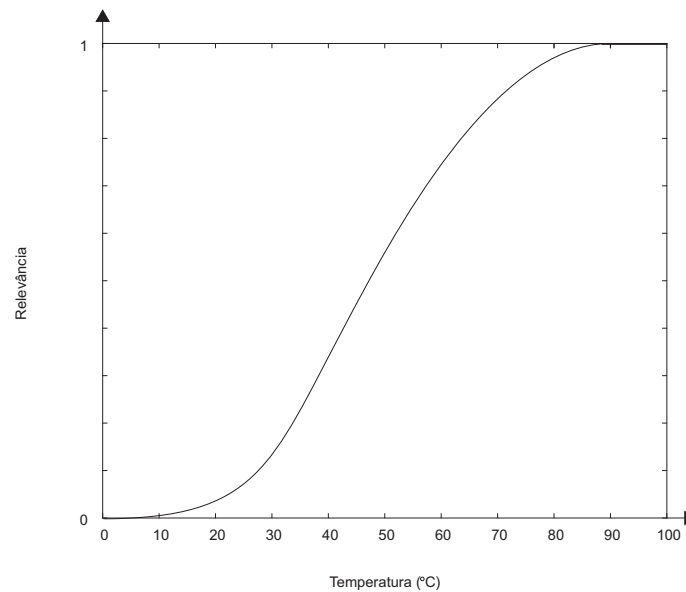


Figura 2.11: Conceito Fuzzy de “Quente”

Na Figura 2.11 tem-se uma possível representação do conceito Fuzzy *quente*. A função de pertinência μ_{QUENTE} associa o quanto a temperatura é considerada *quente*. Pode-se ter, por exemplo, as temperaturas 40, 50 e 60, com graus de pertinências 0.2, 0.3 e 0.6 respectivamente.

2.2.2 Função de Pertinência

2.2.2.1 Definição

A função de pertinência é um conceito fundamental na teoria dos Conjuntos Fuzzy. Ela é uma generalização da função característica da programação funcional (GIARRA-TANO; RILEY, 1994) que estabelece uma correspondência entre um elemento no domínio e um valor verdade que indica o grau de pertinência do elemento no conjunto.

Um Conjunto Fuzzy codifica a imprecisão associada a um fenômeno através de sua superfície; na verdade, o desenho da curva representa a semântica do conceito em questão (GOTTGTROY, 1996).

2.2.2.2 Componentes de um Conjunto Fuzzy

Um Conjunto Fuzzy consiste basicamente de três componentes:

- um eixo horizontal, de valores crescentes monotonicamente, que constituem o conjunto que representa o domínio;
- um eixo vertical, entre 0 e 1, que indica o grau de pertinência ao conjunto;
- uma função, que estabelece a relação entre os dois eixos.

A Figura 2.12, ilustra um exemplo clássico de Conjunto Fuzzy que representa a idéia de *número pequeno*. Para esse exemplo, os componentes do Conjunto Fuzzy *número pequeno* são:

- no eixo horizontal: os números reais positivos, num intervalo considerado representativo do contexto do modelo em questão;
- no eixo vertical: o quanto cada número do eixo horizontal reflete *ser pequeno*;
- uma função, que estabelece a relação entre os dois eixos.

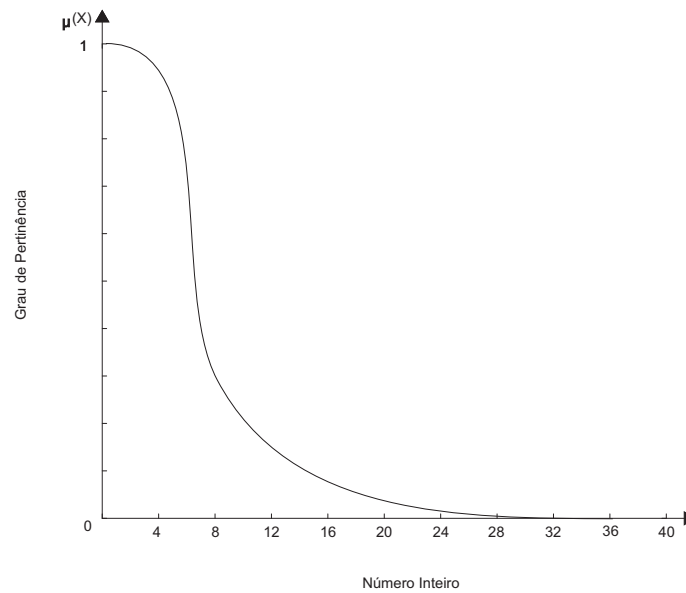


Figura 2.12: Conjunto Fuzzy de “Números Pequenos”

Em um outro exemplo, encontrado em (GOTTGTROY, 1996), pode-se especificar um Conjunto Fuzzy através da explicação da função de pertinência de alguns elementos do domínio; assim, no universo de *altura*, define-se o universo $U = (1.60, 1.70, 1.75, 1.80)$ e pode-se representar o subconjunto Fuzzy *alto* na equação 2.12 por:

$$alto = 0.01/1.60 + 0.3/1.70 + 0.65/1.75 + 1.0/1.80 \quad (2.12)$$

A função de pertinência para o conjunto *alto*, no caso uma curva linear simples, representando que *ser alto* é diretamente proporcional a um valor, que representa a altura em metros.

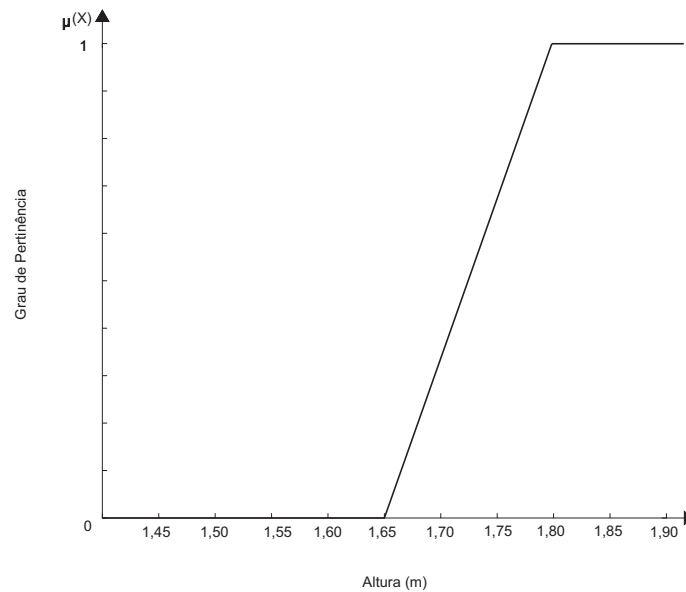


Figura 2.13: Conjunto Fuzzy de “Alto”

O Conjunto Fuzzy “Alto”, Figura 2.13, diz que, abaixo de 1,60m, inclusive, definitivamente não significa ser alto; 1,75m significa moderadamente alto, com grau de pertinência de 0,65; e acima de 1,80m, definitivamente significa ser alto. Observe que esse conjunto está representando o conceito *alto* para a realidade brasileira.

Cabe ao Engenheiro do Conhecimento, pela sua experiência, definir qual o “desenho” que melhor caracteriza o conceito em questão. Entretanto, os sistemas Fuzzy têm demonstrado ser tolerantes a aproximações na representação dos conceitos (COX, 1994), isto é, eles demonstram bom desempenho mesmo quando o desenho do Conjunto Fuzzy não mapeia exatamente o conceito modelado.

2.2.2.3 Tipos de Funções de Pertinência

Existem diversos tipos de curvas para representar um Conjunto Fuzzy, cabe ao Engenheiro do Conhecimento traçar a curva que mais se aproxima do comportamento da variável em questão. Serão mostradas a seguir, as principais curvas para representar um Conjunto Fuzzy.

2.2.2.4 Representação Linear

É o tipo de representação mais simples e uma boa escolha para aproximar conceitos mal compreendidos, que não são Números Fuzzy. A Figura 2.14 mostra uma representação linear crescente e decrescente, respectivamente.

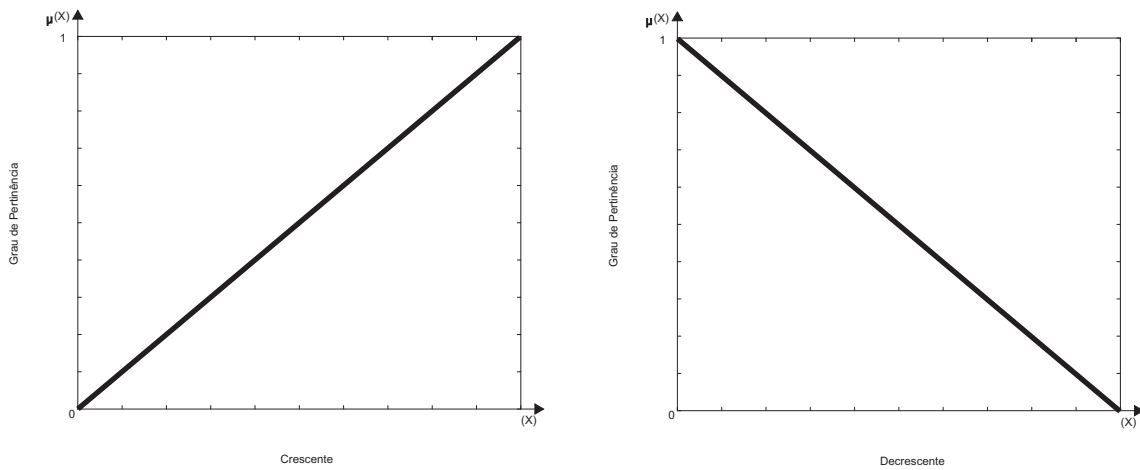


Figura 2.14: Função Crescente e Decrescente, respectivamente

2.2.2.5 Representação em Curva-S

A representação em Curva-S (Sigmóide/Logística) corresponde às superfícies não lineares crescentes e decrescentes, também conhecidas como curva-S e curva-Z, respectivamente. Este tipo de representação é bastante usada para modelar a dinâmica de populações, em que as amostragens de valores individuais aproximam variáveis aleatórias, como por exemplo, a aceleração de um objeto que esteja caindo, o tempo até a falha de um componente mecânico, altura e peso.

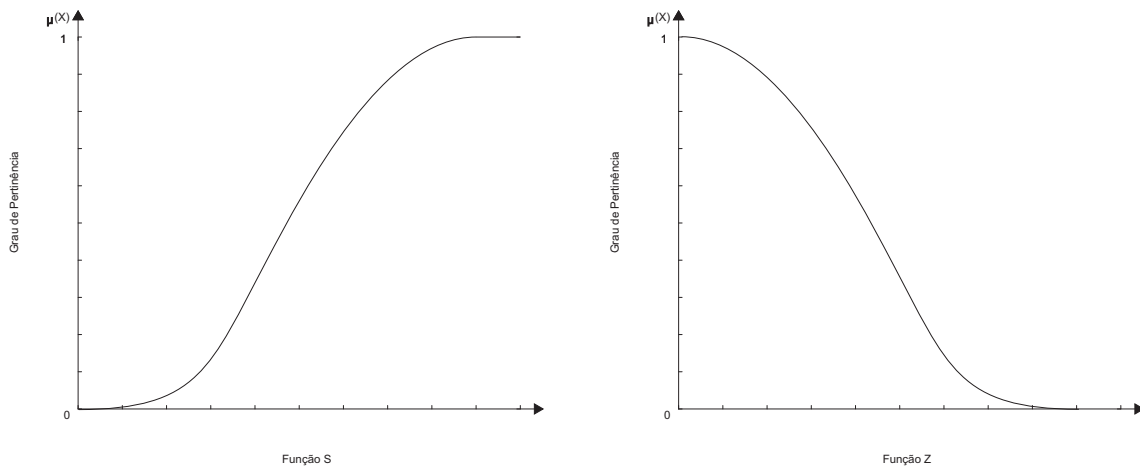


Figura 2.15: Funções S e Z, respectivamente

A curva-s também é bastante utilizada em representações de frequência como “usualmente”, “a maioria de”, “próximo a”, “sempre”, etc., onde o domínio representa um Conjunto Fuzzy de proporções.

2.2.2.6 Representação em Curvas Tipo Sino

É uma classe importante de representações Fuzzy e, é baseada na representação das aproximações de um valor central, sendo graficamente visualizada como uma curva do tipo sino. Esse Conjunto Fuzzy é usado para representar números e espaços Fuzzy.

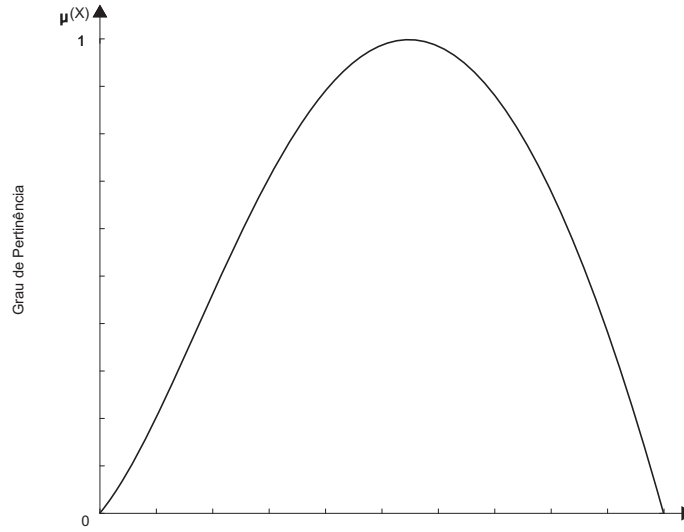


Figura 2.16: Funções do Tipo Sino

2.2.2.7 Representação Trapezoidal

Esta representação é uma variação do tipo sino. Este tipo de representação possui um processamento rápido e possibilita uma descontinuidade no conjunto. Pode ser regido pela seguinte fórmula, formada por uma variável independente x , e quatro parâmetros de formato a , b , c e d :

$$Trap(x, a, b, c, d) = \begin{cases} 0, & \text{para ou se } x \leq a \\ 1 - \frac{(b-x)}{(b-a)} = \frac{x-a}{b-a}, & a < x \leq b \\ 1, & b < x \leq c \\ \frac{(d-x)}{(d-c)}, & c < x \leq d \\ 0, & x > d \end{cases} \quad (2.13)$$

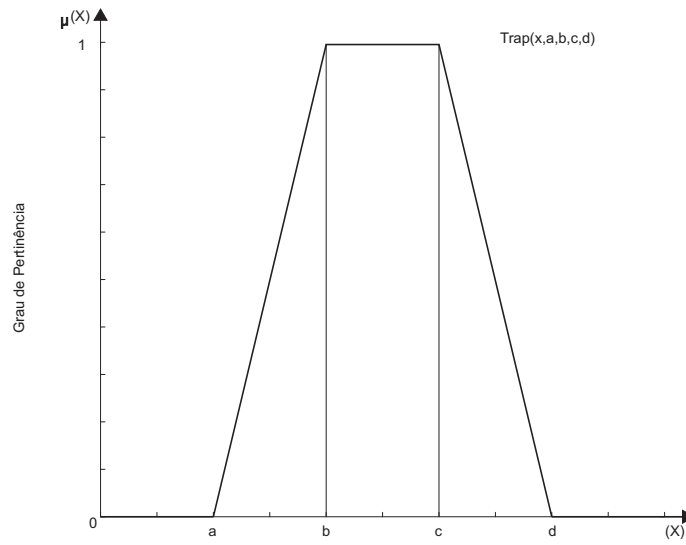


Figura 2.17: Função Trapezoidal

2.2.2.8 Representação Triangular

Também uma extensão da curva tipo sino, porém mais simples que a função trapezoidal.

$$Tri(x, a, b, c) = \begin{cases} 0, x \leq a \\ 1 - \frac{(b-x)}{(b-a)} = \frac{a-x}{b-a}, a < x \leq b \\ \frac{(c-x)}{(c-b)}, b < x \leq c \\ 0, x > c \end{cases} \quad (2.14)$$

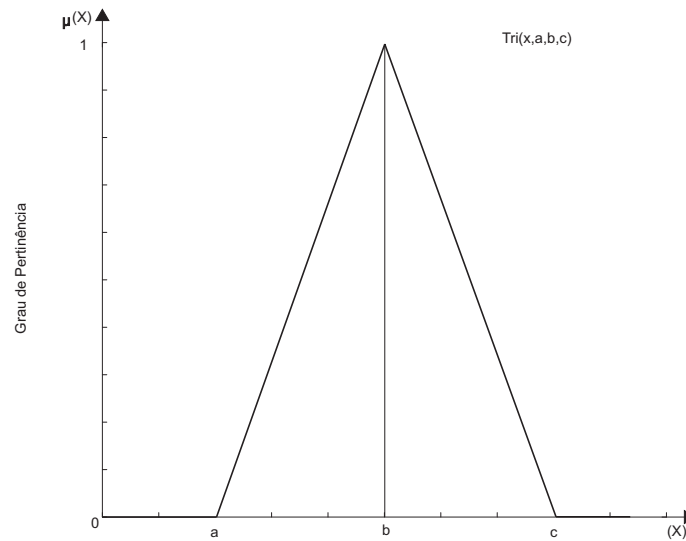


Figura 2.18: Função Triangular

Essas são as principais formas para representar Conjuntos Fuzzy. A seguir serão discutidos os processos de fuzzificação e defuzzificação.

2.2.2.9 Fuzzificação/Defuzzificação

Os sistemas baseados em regras possuem um mecanismo de inferência para gerar conclusões a partir das entradas fornecidas e das regras armazenadas em sua base de conhecimento. É possível definir sistemas de inferência Fuzzy como tendo uma arquitetura conforme a Figura 2.19.

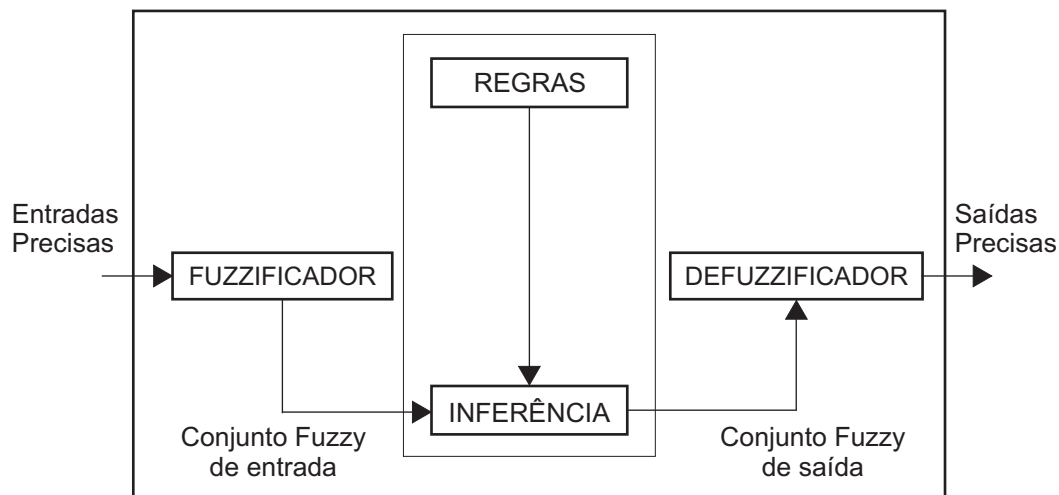


Figura 2.19: Sistemas de Inferência Fuzzy

A Fuzzificação tem como entrada valores concretos nas variáveis de entrada e como saída graus de pertinência, nos Conjuntos Fuzzy (entre 0 e 1). O Fuzzificador

converte as entradas em Conjuntos Fuzzy; a base de regras contém as relações entre as entradas e as saídas desejadas, expressadas como uma coleção de regras de implicação; o módulo de inferência que processa a base de regras, decide quais as regras são ativadas e combina com seus resultados; o Defuzzificador converte o resultado da inferência de volta a um valor preciso.

Como geralmente os dados de entrada são valores obtidos, vindos de resultados de medições, ou observações, é necessário efetuar um mapeamento destes dados precisos para os Conjuntos Fuzzy de entrada relevantes, o que é realizado na Fuzzificação. Na Fuzzificação ocorre também a ativação das regras para uma dada situação. Uma vez obtido o Conjunto Fuzzy de saída através do processamento de inferência, na Defuzzificação é efetuada a interpretação dessa informação (SILVEIRA, 2002).

Fuzzificação é um mapeamento do domínio de números reais (discretos, em geral) para o domínio Fuzzy. Este processo atribui valores lingüísticos, definidos por funções de pertinência, às variáveis de entrada (SHAW; SIMÕES, 1999).

Defuzzificação é um método utilizado para decodificar as variáveis lingüísticas de saída inferidas pelas regras Fuzzy para valores reais (discretos, geralmente). Os principais métodos de defuzzificação são o Centro-da-Área, o Centro-do-Máximo, a Média-do-Máximo e a Média-Ponderada (SHAW; SIMÕES, 1999).

Quando todas as regras foram executadas, o Conjunto Fuzzy solução associado ao problema pode ser Defuzzificado.

- Método de Defuzzificação Centróide

Existem alguns métodos de Defuzzificação que devem ser escolhidos de acordo com a aplicação, dentre os quais, pode-se destacar o Método Centro da Área (CoA), também conhecido como centróide.

O método de Defuzzificação Centro da Área calcula o centróide da área composta, que representa o termo de saída do processo de inferência nebuloso (Conjunto Fuzzy de Saída), esse termo é composto pela união de todas as contribuições de regras. O centróide é um ponto que divide essa área gerada de saída em duas partes iguais, como ilustrado na Figura 2.20 e calculado pela Equação 2.15

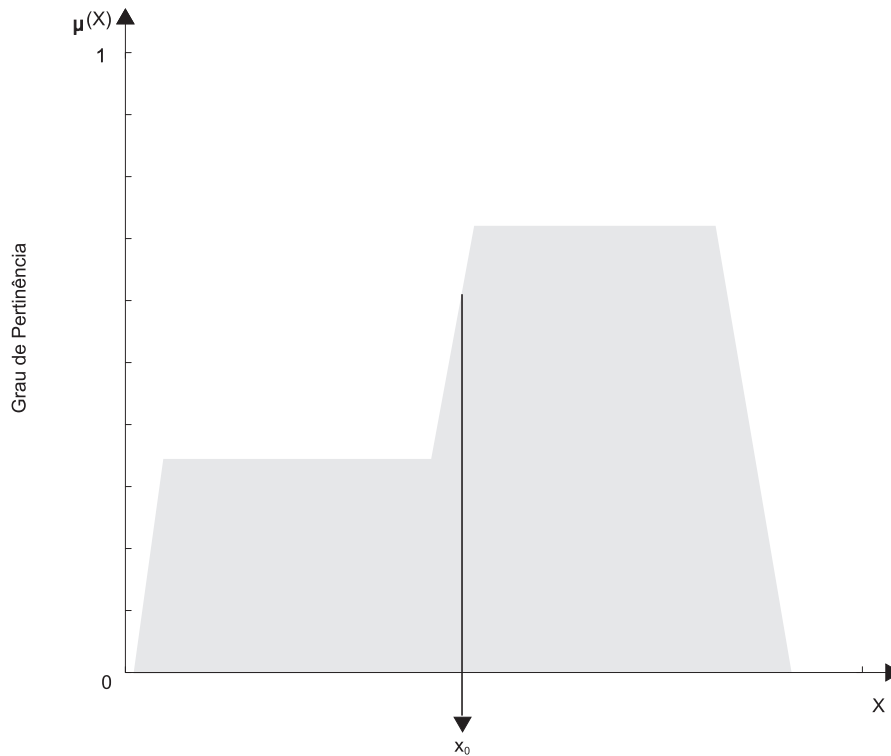


Figura 2.20: Método de Defuzzificação Centro da Área

$$x_0 = \left[\frac{\sum x_i \cdot \mu_{saida}(x_i)}{\sum \mu_{saida}} \right] \quad (2.15)$$

O método de Defuzzificação Centróide é o mais utilizado em aplicações da Lógica Fuzzy na engenharia já que se obtém uma solução única, embora as vezes seja difícil de calcular (PUEYO, 2005).

2.2.2.10 α –nível de um Conjunto Fuzzy

O α –nível de um Conjunto Fuzzy A é o conjunto A_α que contém todos os elementos do universo de discurso U que possuem graus de pertinência $\mu_A(x)$ acima de um certo valor de α .

A equação 2.16 mostra a definição deste conjunto

$$A_\alpha = \{x \in U | \mu_A(x) \geq \alpha\}, \alpha = 0.2 \quad (2.16)$$

A Figura 2.21 ilustra o α –nível de um Conjunto Fuzzy com $\alpha = 0.2$.

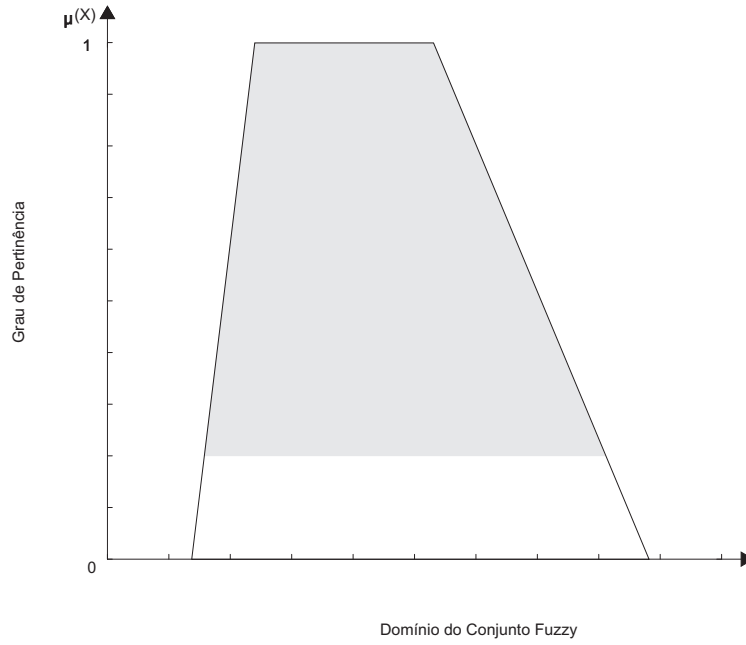


Figura 2.21: α -nível de um Conjunto Fuzzy

2.2.3 Números Fuzzy

Assim como no caso clássico, os Números Fuzzy têm também como objetivo realizar operações aritméticas. A diferença é que nos Números Fuzzy pretende-se calcular quantidades imprecisas. Por exemplo, quando se mede a altura de um indivíduo, o que se obtém é um valor numérico carregado de imprecisões. Tais imprecisões podem ter sido causadas pelos instrumentos de medição, pelos indivíduos que estão medindo, pelo indivíduo que está sendo medido, enfim, diversos outros fatores podem influenciar. Finalmente, opta-se por um valor preciso (um número real) a para indicar a altura. No entanto, seria mais prudente dizer que a altura está *em torno de* a . Neste caso, matematicamente indica-se a expressão *em torno de* a por um Conjunto Fuzzy A cujo domínio é o conjunto dos números reais. Também é razoável esperar que $A(a) = 1$.

O Número Fuzzy (KLIR; YUAN, 1995; ZIMMERMANN, 1996; FANG; CHEN, 1990; GIACHETTI; YOUNG, 1997; PEDRYCZ; GOMIDE, 1998) é um Conjunto Fuzzy definido em um universo de discurso sobre os números reais, $U \in R$.

A escolha dos números reais, para os possíveis valores para a *altura*, criou-se porque estes objetos (números), generalizam os números reais. Tais objetos serão chamados de Números Fuzzy.

A operação de α -nível também é aplicada para os Números Fuzzy. Denota-se com A_α o intervalo α -nível para um Número Fuzzy A . A_α -nível é um intervalo *crisp* definido na Equação 2.17 como:

$$A_\alpha = [a_1^{(\alpha)}, a_3^{(\alpha)}], \forall \alpha \in [0, 1], a_1, a_3, a_1^{(\alpha)}, a_3^{(\alpha)} \in R \quad (2.17)$$

Com esta definição é possível conhecer qualquer intervalo *crisp* dentro do Número Fuzzy, correspondente a um α -nível qualquer, Figura 2.22.

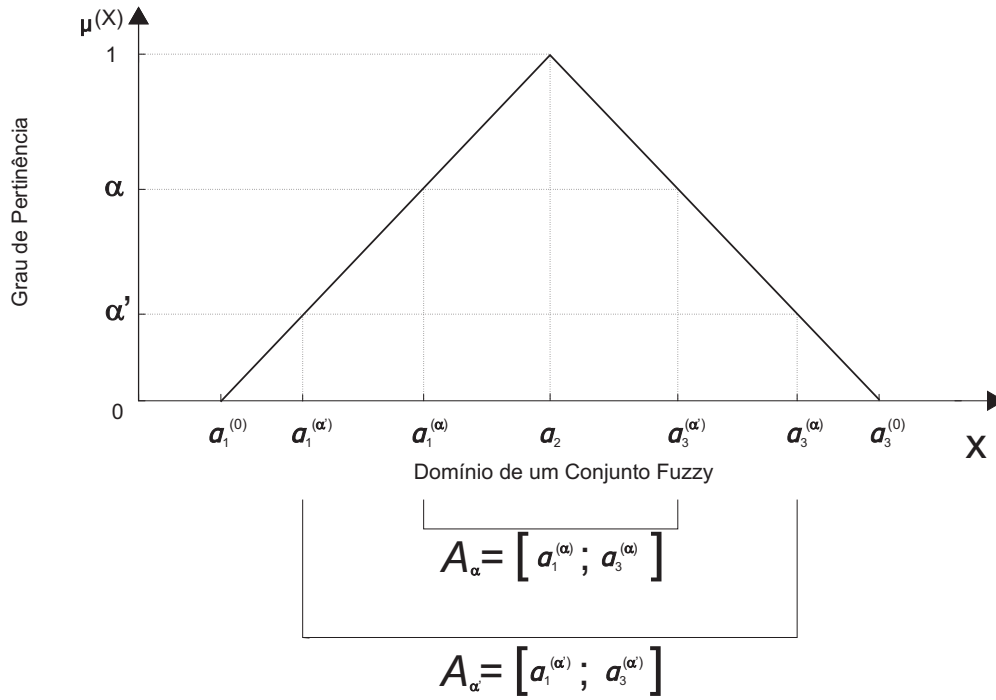


Figura 2.22: Intervalo α -nível de um Número Fuzzy

Para que um Conjunto Fuzzy seja definido como um Número Fuzzy, este deve cumprir as seguintes condições:

- estar definido nos números reais (todos os α -níveis de A são não vazios, com $0 \leq \alpha \leq 1$);
- a função de pertinência deve ser contínua;
- o Conjunto Fuzzy deve ser normalizado;
- o Conjunto Fuzzy deve ser convexo.

Logo, um Número Fuzzy deverá ser normalizado e convexo, onde a condição de normalização implica que o máximo valor de pertinência é 1; ou ainda, não satisfazia a condição da Equação 2.18.

$$\exists x \in R, \mu_A(x) = 1 \quad (2.18)$$

A condição de convexidade significa que a linha traçada por todo α -nível é contínua e o intervalo α -nível satisfaz a relação a seguir. Seja o intervalo A_{α} definido pela Equação 2.19:

$$A_\alpha = [a_1^{(\alpha)}, a_3^{(\alpha)}] \quad (2.19)$$

Logo:

$$(\alpha' \leq \alpha) \Rightarrow (a_1^{(\alpha')} \leq a_1^{(\alpha)}, a_3^{(\alpha')} \geq a_3^{(\alpha)}) \quad (2.20)$$

2.2.3.1 Operações com Números Fuzzy

A seguir define-se brevemente a forma de realizar as operações com Números Fuzzy.

- Operações com intervalo α -nível:

Os intervalos α -nível de um Número Fuzzy $A = [a_1, a_3]$ podem ser referenciados como um conjunto *crisp*. A_α é um intervalo *crisp* definido pela equação 2.17.

Logo as operações vistas no estudo da Matemática Intervalar podem ser aplicadas para o intervalo α -nível A_α .

Se o intervalo α -nível B_α de um Número Fuzzy B é definido por:

$$B = [b_1, b_3], b_1, b_3 \in R, B_\alpha = [b_1^{(\alpha)}, b_3^{(\alpha)}], \forall \alpha \in [0, 1], b_1, b_3, a_1^{(\alpha)}, b_3^{(\alpha)} \in R$$

então, as operações entre A_α e B_α podem ser descrita da seguinte forma:

$$\begin{aligned} A_\alpha + B_\alpha &= [a_1^{(\alpha)}, a_3^{(\alpha)}] + [b_1^{(\alpha)}, b_3^{(\alpha)}] = [a_1^{(\alpha)} + b_1^{(\alpha)}, a_3^{(\alpha)} + b_3^{(\alpha)}] \\ A_\alpha - B_\alpha &= [a_1^{(\alpha)}, a_3^{(\alpha)}] - [b_1^{(\alpha)}, b_3^{(\alpha)}] = [a_1^{(\alpha)} - b_3^{(\alpha)}, a_3^{(\alpha)} - b_1^{(\alpha)}] \end{aligned}$$

Isto é, para cada grau de pertinência do Número Fuzzy estabelece-se um α -nível, criando-se intervalos α -nível. Logo, as operações com o Número Fuzzy serão realizadas para cada nível (cada α -nível) segundo a teoria intervalar. Esta metodologia é também aplicável às operações de multiplicação, divisão e outras.

- Solução Numérica:

Na implementação computacional, o grau de pertinência $\alpha = [0, 1]$ é discretizado em n níveis, por exemplo 10, e as operações de intervalo de confiança (aritmética intervalar) são realizadas para cada nível. A Figura 2.23 apresenta o Número Fuzzy como um conjunto de Números Fuzzy como um conjunto de números intervalares, $[a_1^\alpha, a_3^\alpha]$, que são definidos para cada nível de pertinência α . Observe que, quando α cresce, isto é, o grau de confiança aumenta, o diâmetro do intervalo diminui.

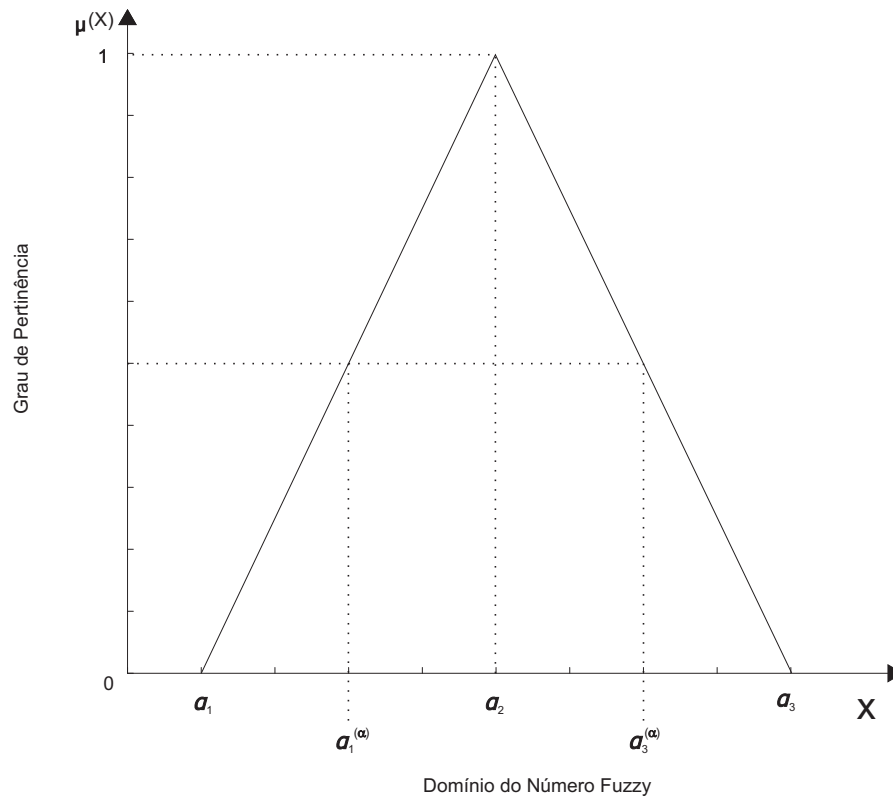


Figura 2.23: Número Fuzzy Triangular

2.2.3.2 Operações Aritméticas com Números Fuzzy Triangulares

A Figura 2.24 representa esquematicamente um Número Fuzzy Triangular (TFN), onde ao valor a_2 se atribui pertinência 1. Valores menores que a_1 e maiores que a_3 são atribuídos pertinência zero. Valores intermediários entre $[a_1, a_2]$ e $[a_2, a_3]$ são atribuídos pertinências entre 0 e 1.

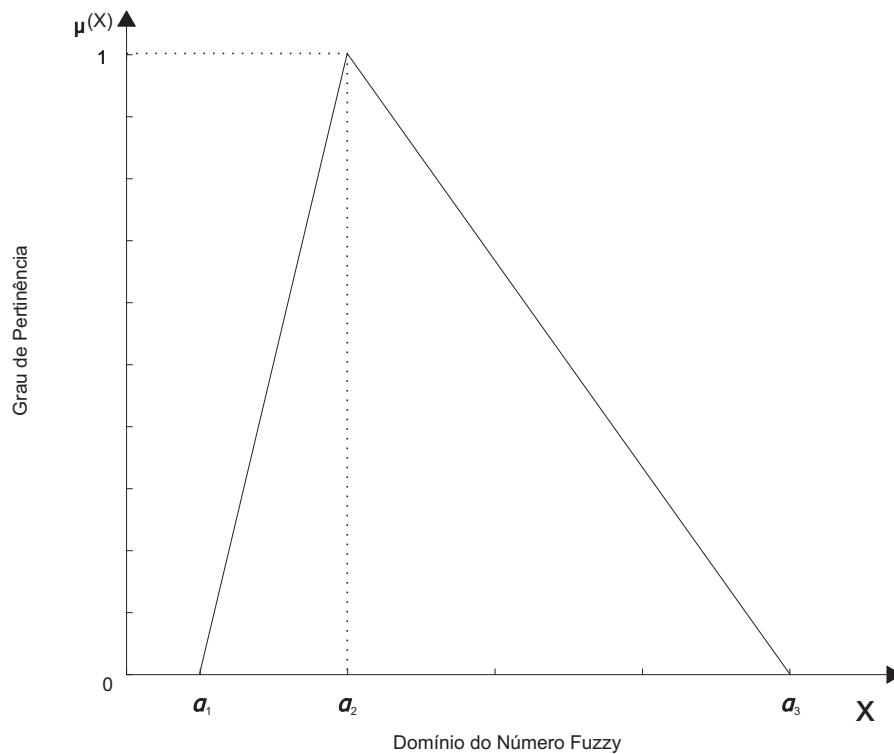


Figura 2.24: Número Fuzzy Triangular

As operações de adição, subtração e multiplicação por um escalar são operações fechadas no conjunto dos Números Fuzzy Triangulares.

A seguir, as operações de adição, subtração e a multiplicação por um escalar são detalhadas. As operações algébricas utilizadas neste estudo são definidas e representadas graficamente a seguir.

2.2.3.3 Operação Algébrica de Adição

Sejam dois Números Fuzzy Triangular A e B , com $A = (a_1, a_2, a_3)$ e $B = (b_1, b_2, b_3)$, então $A + B = (a_1, a_2, a_3) + (b_1, b_2, b_3) = (a_1 + b_1, a_2 + b_2, a_3 + b_3)$.

Exemplo: Com $A = (2, 3, 6)$ e $B = (1, 4, 5)$, então temos que $A + B = (3, 7, 11)$.

A representação gráfica desta operação é mostrada na Figura 2.25.

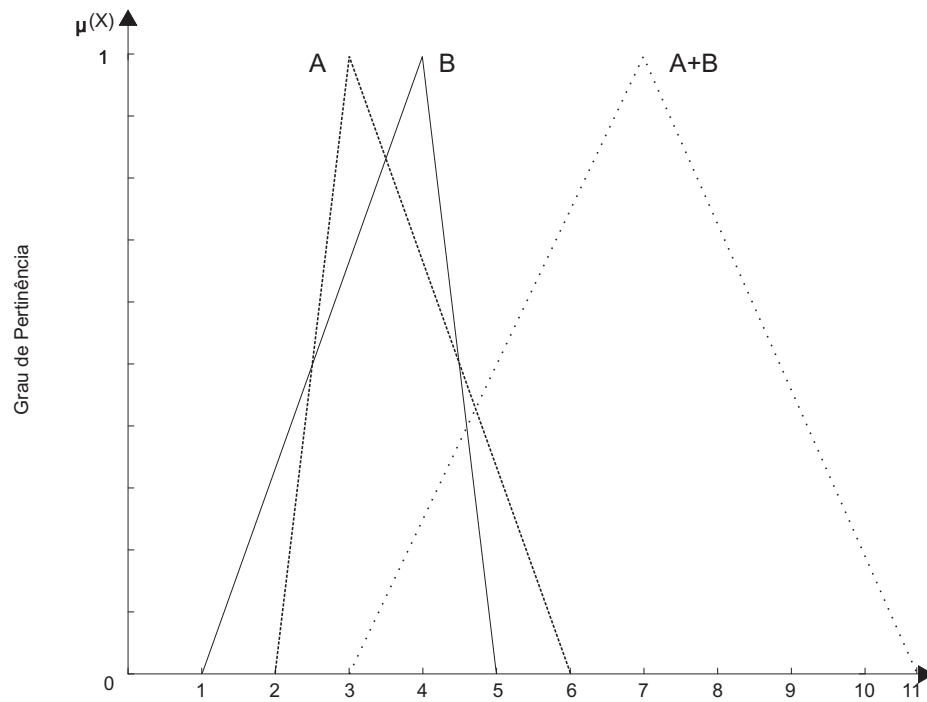


Figura 2.25: Representação Gráfica da Operação de Adição entre dois Números Fuzzy

2.2.3.4 Operação Algébrica de Subtração

Sejam dois Números Fuzzy Triangular A e B , com $A = (a_1, a_2, a_3)$ e $B = (b_1, b_2, b_3)$, então $A - B = (a_1, a_2, a_3) - (b_1, b_2, b_3) = (a_1 - b_3, a_2 - b_2, a_3 - b_1)$.

Exemplo: Com $A = (2, 3, 6)$ e $B = (1, 4, 5)$, então temos que $A - B = (2 - 5, 3 - 4, 6 - 1) = (-3, -1, 5)$. Essa representação pode ser visto na Figura 2.26.

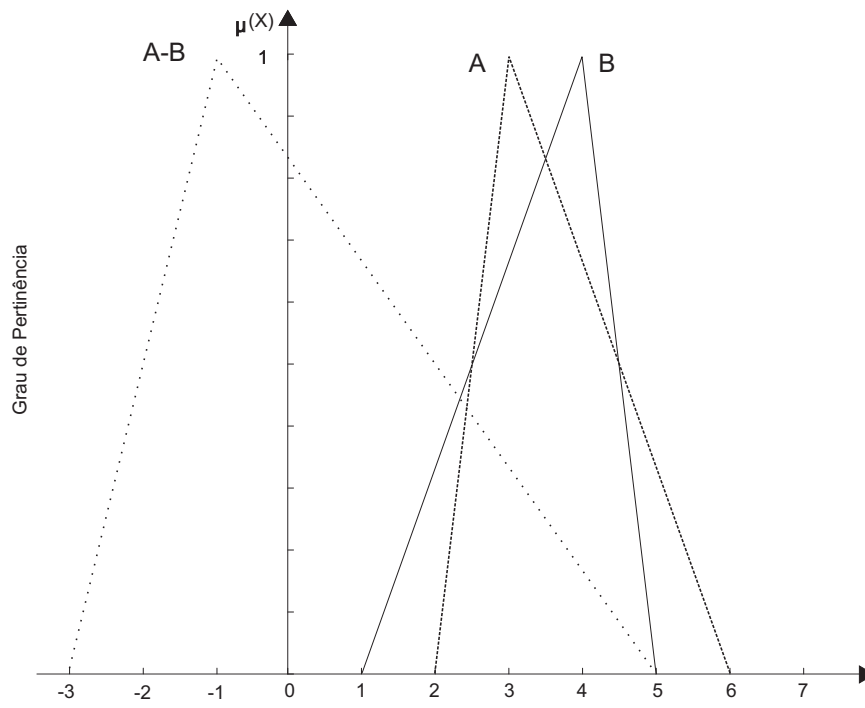


Figura 2.26: Representação Gráfica da Operação de Subtração entre dois Números Fuzzy

2.2.3.5 Operação Algébrica de Multiplicação por um Escalar

Seja um TFN A , com $A = (a_1, a_2, a_3)$ então $2 \times (a_1, a_2, a_3) = (2 \times a_1, 3 \times a_2, 3 \times a_3)$.

Exemplo: Com $A = (2, 3, 6)$, então temos que $2 \times A = (4, 6, 12)$. Essa representação pode ser visto na Figura 2.27.

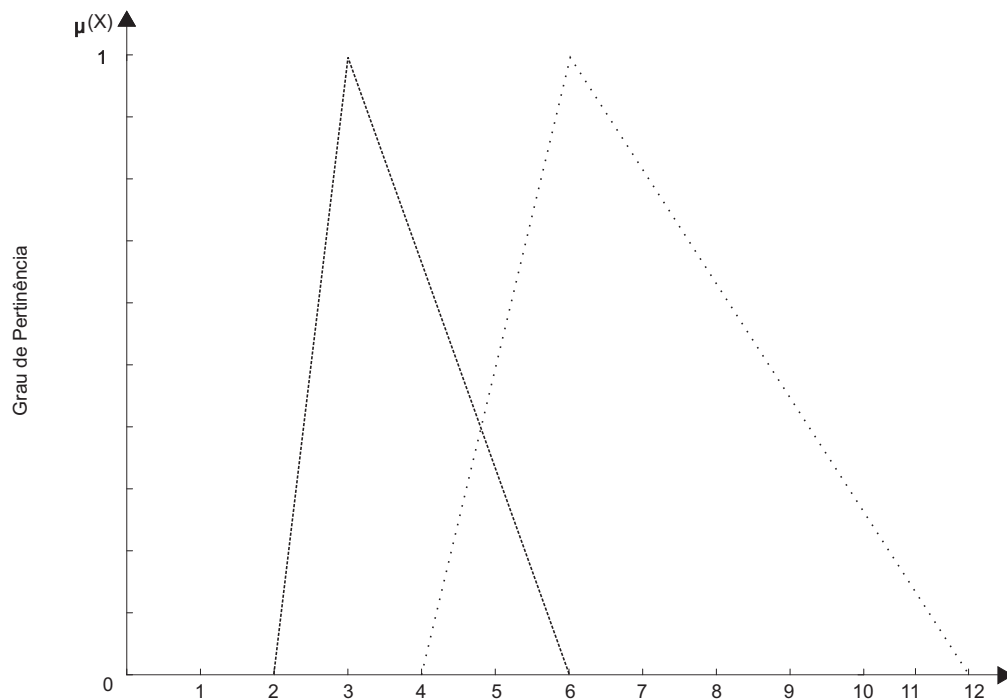


Figura 2.27: Representação Gráfica da Operação de Multiplicação entre Números Fuzzy

2.3 A Biblioteca C-XSC

Os computadores possuem uma aritmética de ponto-flutuante que serve para aproximar os números reais e suas operações.

A linguagem de programação C++ é uma linguagem orientada a objeto, e é uma extensão a linguagem de programação C. Esta extensão C++, tem o novo conceito de estrutura de dados abstratos (classes) e o conceito de operadores overloaded (sobrecarregados).

As características da linguagem C++, por ser orientada a objeto, fornecem recursos adicionais poderosos que reduzem o esforço de programação e aumentam a confiabilidade e a clareza dos programas.

Bibliotecas XSC (eXtensions for Scientific Computation) estão disponíveis nas linguagens de programação FORTRAN, PASCAL e C (ACRITH-XSC, PASCAL-XSC e C-XSC, respectivamente).

De acordo com (HOFSCHUSTER; KRÄMER; WIETHOFF, 2001), o C-XSC é uma ferramenta para o desenvolvimento de algoritmos numéricos propiciando exatidão e resultados verificados automaticamente. A biblioteca C-XSC é muito aplicada na computação científica como uma biblioteca numérica, para o desenvolvimento dos algoritmos numéricos, utilizando a linguagem de programação C++. Assim, o C-XSC permite a programação de alto nível de aplicações numéricas em C++ (HöLBIG, 2005).

Com suas estruturas de dados abstratos, operadores e funções predefinidos, o C-

XSC fornece uma interface entre a computação científica e as linguagens de programação C e C++. Além disso, C-XSC suporta a programação de algoritmos que incluem automaticamente a solução de problemas matemáticos em certos limites confiáveis. Tais algoritmos fornecem uma declaração matemática precisa sobre a verdadeira solução.

A biblioteca C-XSC fornece um grande número de tipo de dados, dispondo de extensões para a computação científica que propicia vários benefícios, dentre os quais se destacam:

- aritmética real, complexa, intervalar e intervalar complexa, com propriedades matematicamente definidas;
- vetores e matrizes dinâmicos;
- subvetores de vetores e matrizes;
- tipo de dados **dotprecision** (tipo de dado em alta precisão);
- operadores aritméticos predefinidos de alta exatidão;
- função padrão de alta exatidão;
- controle de arredondamento para entrada/saída de dados;
- resultados numéricos com rigor matemático.

Nas próximas subseções serão apresentadas e exemplificadas algumas características dessa biblioteca.

2.3.1 Tipos de dados

Todos os operadores aritméticos predefinidos fornecem resultados com exatidão de, no mínimo, 1 **ulp** (erro de arredondamento na última posição da mantissa). Desta forma, eles são de máxima exatidão do ponto de vista da computação científica.

Além de todos os tipos de dados matemáticos do C/C++, o C-XSC oferece outros tipos de dados numéricos simples:

- *real*;
- *interval* (intervalo de reais);
- *complex* (número complexo);
- *cinterval* (intervalo complexo).

Para os tipos de dados escalares apresentados acima, a biblioteca C-XSC oferece uma extensão para a sua utilização na forma de matrizes e vetores, são elas:

- *rvector* (vetor de reais);
- *ivector* (vetor de intervalos reais);
- *cvector* (vetor de complexos);
- *civector* (vetor de intervalos complexos);
- *rmatrix* (matriz de reais);
- *imatrix* (matriz de intervalos reais);
- *cmatrix* (matriz de complexos);
- *cimatrix* (matriz de intervalos complexos).

O C-XSC também possui uma grande variedade de funções matemáticas (com nomes genéricos) com suporte aos tipos de dados real, complex, interval e cinterval. Algumas das funções predefinidas são mostradas no Código 2.3.1.

```

1  #include <iostream>
2  #include <gop.hpp>
3  using namespace cxsc;
4  using namespace std;
5
6  int main()
7  {
8      cout << power(3,3) << endl; // 27.000000
9      cout << sqrt(81) << endl;   // 9
10     return 0;
11 }
```

Código 2.3.1: Codificação em C-XSC: Operadores Predefinidos

Esse algoritmo (Código 2.3.1) mostra algumas operações aritméticas como potência e raiz quadrada utilizando algumas funções predefinidas do C-XSC. A linha 8 mostra a potência de 3^3 , e a linha 9 tem a operação da raiz quadrada de 81 ($\sqrt{81}$) obtendo como resultado o valor 9.

2.3.2 Uso de Intervalos

O C-XSC permite trabalhar com intervalos reais e também com intervalos complexos (interval e cinterval). Existem funções predefinidas para a manipulação de intervalos, nos quais destacam-se *Inf()*, *Sup()*, *diam()* e *mid()* (extremo inferior de um intervalo, extremo superior de um intervalo, diâmetro de um intervalo e o ponto médio de um intervalo, respectivamente).

No Código 2.3.2 é apresentado algumas operações aritméticas utilizando o tipo de dado intervalar (interval).

```

1  #include <iostream>
2  #include <interval.hpp>
3  using namespace cxsc;
4  using namespace std;
5
6  int main()
7  {
8      interval oper1, oper2;
9      real a = 1.0;
10     real b = 3.0;
11     oper1 = interval(a, b);
12     oper2 = interval(b, b);
13     cout << oper1 + oper2 << endl; // [ 4.000000, 6.000000]
14     cout << oper1 * 2 << endl;    // [ 2.000000, 6.000000]
15     cout << mid(oper1) << endl;    // 2.000000
16     cout << Inf(oper1) << endl;    // 1.000000
17     return 0;
18 }

```

Código 2.3.2: Codificação em C-XSC: Uso de Intervalos

Para o Código 2.3.2, tem-se: na linha 13 é realizado a operação de soma de dois intervalos (**oper1** e **oper2**), a linha 14 faz a multiplicação do intervalo **oper1** por 2, o ponto médio do intervalo **oper1** é realizado na linha 15 e o extremo inferior de um intervalo é obtido na linha 16.

2.3.3 Avaliação de Expressões com Alta Precisão

Nas avaliações de expressões aritméticas, a precisão tem um fator decisivo em muitos algoritmos numéricos. Mesmo se todos os operadores de aritmética e funções padrão forem de máxima precisão, expressões compostas de vários operadores e funções, podem não oferecer resultados com máxima precisão. No C-XSC foram desenvolvidos métodos para avaliar expressões com máxima precisão. Para obter uma avaliação com precisão de 1 **ulp**, a extensão C-XSC provê os tipos de dados de alta precisão, são eles: **dotprecision**, **cdotprecision**, **idotprecision** e **cidotprecision**.

2.3.4 Arredondamento

No C-XSC é possível realizar arredondamentos direcionados, ou seja, dada determinada operação é possível arredondar o resultado para cima ou para baixo. A Tabela 2.1 mostra as funções de arredondamento direcionados.

Tabela 2.1: Funções de Arredondamento Direcionados do C-XSC

Descrição	Função
adição com arredondamento para cima	<i>addu()</i>
adição com arredondamento para baixo	<i>addd()</i>
subtração com arredondamento para cima	<i>subu()</i>
subtração com arredondamento para baixo	<i>subd()</i>
multiplicação com arredondamento para cima	<i>mulu()</i>
multiplicação com arredondamento para baixo	<i>muld()</i>
divisão com arredondamento para cima	<i>divu()</i>
divisão com arredondamento para baixo	<i>divd()</i>

Um exemplo de arredondamento direcionado pode ser visto no Código 2.3.3.

```

1  #include <iostream>
2  #include <gop.hpp>
3  using namespace cxsc;
4  using namespace std;
5
6  int main()
7  {
8      double a = 4;
9      double b = 3;
10     cout << SetPrecision(2,16) << a/b << endl;           //1.33333
11     cout << SetPrecision(2,16) << divu(a,b) << endl;       //1.3333333333333335
12
13     return 0;
14 }
```

Código 2.3.3: Codificação em C-XSC: Arredondamento Direcionado

A linha 11 do Código 2.3.3 mostra a divisão de 4 por 3, com a realização de um arredondamento direcionado para cima na décima sexta posição decimal usando a função *divu()*.

2.3.5 Aritmética Dinâmica de Múltipla Precisão

Além das classes *real* e *interval*, no C-XSC existem as classes dinâmicas de maior precisão, vejam o exemplo a seguir, o uso das classes reais longos (*l_real*) e intervalo longo (*l_interval*).

```

1  #include <iostream>
2  #include <interval.hpp>
3  using namespace cxsc;
4  using namespace std;
5
6  int main()
7  {
8      l_interval a, b;
9      a = 1.0;           // a = [ 1.0, 1.0 ]
10     b = 3.0;           // b = [ 3.0, 3.0 ]
11     cout << a/b << endl;
12 }
```

Código 2.3.4: Codificação em C-XSC: Múltipla Precisão

Como resultado da divisão de a por b ($a \div b$), obtido na linha 11, tem-se o intervalo: $[0, 33333333333333333333; 0, 33333333333333333334]$.

No C-XSC também são implementados os vetores e matrizes dinâmicos correspondentes, inclusive com toda a aritmética, operadores relacionais e funções **standard** de múltipla precisão também.

2.3.6 Operações em C-XSC

A tabela 2.2 (HOFSCHUSTER; KRÄMER; WIETHOFF, 2001) mostra as operações aritméticas possíveis entre os diferentes tipos de dados.

Tabela 2.2: Operadores em C-XSC

TIPOS DADOS	integer real complex	interval interval	rvector cvector	ivector ci-vector	rmatrix cmatrix	imatrix ci-matrix
integer real complex	+, -, *, /,	+, -, *, /,	*	*	*	*
interval interval	+, -, *, /,	+, -, *, /, , &	*	*	*	*
rvector cvector	*, /	*, /	*, -, * ¹ , /,	*, -, * ¹ , /		
ivector ci-vector	*, /	*, /	*, -, * ¹ , /,	*, -, * ¹ , /, , &		
rmatrix cmatrix	*, /	*, /	* ¹	* ¹	*, -, * ¹ , /,	*, -, * ¹ , /,
rmatrix cmatrix	*, /	*, /	* ¹	* ¹	*, -, * ¹ , /,	*, -, * ¹ , /, , &

O operador | é o polígono convexo (menor polígono possível com todos os pontos da tela), o operador *¹ indica o produto com alta precisão e destaca-se o operador &, o qual realiza a intersecção entre os intervalos, o qual é exemplificado no Código 2.3.5.

```

1  #include <iostream>
2  #include <interval.hpp>
3  using namespace cxsc;
4  using namespace std;
5
6  int main()
7  {
8      interval A, B, interseccao;
9      A = interval(3, 5);
10     B = interval(4, 6);
11     interseccao = A&B;
12     cout << A << endl;           // [ 3.000000, 5.000000]
13     cout << B << endl;           // [ 4.000000, 6.000000]
14     cout << interseccao << endl; // [ 4.000000, 5.000000]
15     return 0;
16 }
```

Código 2.3.5: Codificação em C-XSC: Operador Intersecção

No Código 2.3.5 a linha 14 realiza a intersecção entre os intervalos [3, 5] e [4, 6], obtendo como resultado o intervalo [4, 5].

Ao invés do programador ter que desenvolver diversas rotinas, muitas destas já estão previamente codificadas.

2.3.7 Entrada e Saída de dados em C-XSC

Usando o conceito de *stream* e sobrecarga de operadores << e >> do C++, o C-XSC permite arredondamento e controle do formato durante E/S de dados. Parâmetros de

E/S, como direção de arredondamento, tamanho dos campos, entre outros, também usam sobrecarga de operadores para manipular dados de E/S.

2.3.8 CToolbox

O *C++ Toolbox for Verified Computing* (HAMMER; HOCKS; RATZ, 1995) é um conjunto de ferramentas para resolução de diversos problemas numéricos com a biblioteca C-XSC. Esse *toolbox* dispõe de funções para computação de raízes de equações, resolução de sistemas lineares, otimização de sistemas, entre outras aplicações da área científica. Além das funções que são disponibilizadas ao instalar o C-XSC, que pode ser obtido no seguinte endereço: www.xsc.de, existem outras funções (softwares adicionais) que podem ser agregadas ao C-XSC, disponível em: http://www.math.uni-wuppertal.de/~xsc/xsc/cxsc_software.html.

2.4 Síntese

Neste capítulo apresentou-se uma revisão bibliográfica do que foi utilizado de técnicas matemáticas para resolver o problema do fluxo de potência em redes de energia elétrica.

Com o uso da Matemática Intervalar e/ou Lógica Fuzzy pode-se tirar um grande proveito para a exatidão de resultados. Pode-se empregar essas técnicas onde se tenha incertezas, como de erros de modelagem, de equipamentos de medição e até mesmo erros computacionais.

Uma correta utilização desses conhecimentos possibilita conhecer ou delimitar os graus de erros, ou seja, conhecer os seus limites.

Com o uso da Matemática Intervalar pode-se perceber que obter uma resposta intervalar não garante que ela contenha algo de interesse. Os algoritmos a serem desenvolvidos devem ser algoritmos intervalares e não versões intervalares dos algoritmos pontuais. Os sistemas computacionais atuais são incapazes de representar todos os números reais.

A utilização da Lógica Fuzzy não possui uma complexidade grande, no sentido de que pode ser facilmente implementada.

O uso de Números Fuzzy possibilita que as avaliações das variáveis sejam expressas com incertezas baseadas em uma avaliação quantitativa. Isto evita realizar outras avaliações quantitativas objetivando conhecer as incertezas.

As operações com números Fuzzy seguem as mesmas propriedades das operações intervalares. A diferença é que, com os números Fuzzy, as operações são realizadas para cada nível de pertinência, como se o Número Fuzzy fosse “fatiado” em diversos intervalos (GIACHETTI; YOUNG, 1997; ANILE; DEODATO; PRIVITERA, 1995; PEREIRA, 2002).

A escolha da biblioteca C-XSC deve-se por dois motivos: (i) disponibilidade; e

(ii) usabilidade. Disponibilidade por ser uma biblioteca *freeware* e por vir ao encontro das necessidades do grupo de pesquisa. E usabilidade, pois o C-XSC é uma biblioteca para trabalhar com a linguagem de programação C/C++ e permite escrever algoritmos numéricos produzindo resultados confiáveis num ambiente de programação confortável. Esta linguagem está praticamente disponível em todos os sistemas operacionais, inclusive o sistema operacional *livre* Ubuntu (Linux).

3 ESTUDO DE CASO: FLUXO DE POTÊNCIA EM REDES DE ENERGIA ELÉTRICA

Este capítulo apresenta as principais definições matemático-computacionais relativas ao problema do fluxo de potência em redes de energia elétrica. Este problema foi escolhido por se tratar de uma aplicação prática da teoria da Matemática Intervalar e da Lógica Fuzzy e ainda possibilitar a exploração dos recursos e funcionalidades da biblioteca intervalar C-XSC.

O problema do fluxo de potência ou fluxo de carga tem sido exaustivamente estudado no âmbito da Engenharia Elétrica por sua importância para o planejamento e a operação de sistemas de transmissão de energia elétrica (BARÁN; KASZCUREWICZ; FALCÃO, 1995; MONTICELLI, 1983).

Segundo (CISNEROS, 2006), até os anos 40, a quantidade de interconexões nos sistemas elétricos eram muito pequenas, dessa forma os sistemas elétricos eram predominantemente radiais. Os estudos (cálculos) desses sistemas eram relativamente simples, ao menos, era possível realizar sem recorrer a grandes recursos de cálculo, que na época não existiam. Com o crescimento das cidades e as interconexões das redes, a complexidade dos sistemas elétricos foi crescendo, e os estudos nessa área foi mais requerido. Casualmente, houve a coincidência de evolução dos sistemas elétricos com os computadores digitais. O primeiro artigo mencionando do uso do computador com os estudos de fluxo de potência deu-se por (DUSTAN, 1947). Entretanto, o crédito pela formulação do problema com uma orientação adequada para sua programação nos computadores, se concede, geralmente, a (DIGITAL COMPUTER SOLUTION OF POWER-FLOW PROBLEMS, 1956).

3.1 O Fluxo de Potência

O objetivo de estudar o fluxo de potência é obter o perfil de tensões do sistema elétrico. Conhecendo o perfil de tensões, pode-se determinar os fluxos nas linhas de transmissão e nos transformadores. É importante salientar que as aplicações do fluxo de

potência são tão amplas como importantes. É essencial o estudo do fluxo de potência para a análise, planejamento e o projeto de sistemas elétricos, como para a sua operação e controle.

Para exemplificar os conceitos de fluxo de potência, utiliza-se da Figura 3.1 que mostra um esquema da distribuição de energia. A energia elétrica é gerada numa usina (por exemplo: esta pode ser eólica, hidroelétrica, nuclear, solar, termoelétrica, etc). A energia é transportada por fios, ou seja, por linhas de transmissão até uma subestação. Dessa subestação, a energia encaminha-se até as residências. A partir das usinas é possível interligar diversas subestações, das subestações, é possível conectar diversas cidades ou bairros.

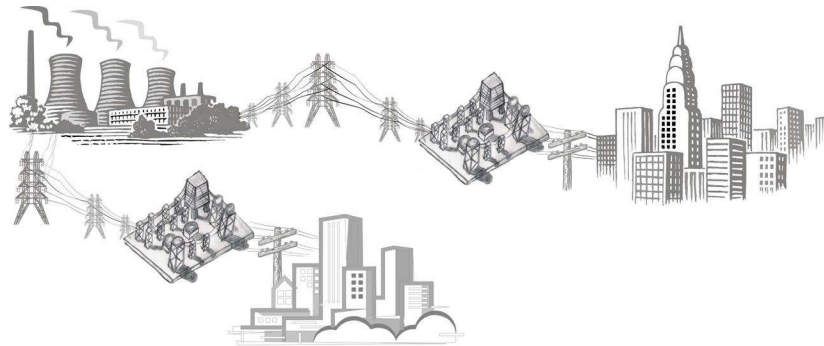


Figura 3.1: Esquema de Distribuição de Energia

Para este modelo de fluxo de potência, é necessário conhecer a distribuição de energia, isto é, necessita-se obter respostas às seguintes perguntas:

- Será que a rede elétrica vai suprir a demanda da cidade ou bairro?
- As linhas de transmissão serão capazes de transportar a energia elétrica solicitada pelas demandas?

Determinando o perfil de tensões da rede elétrica, é possível verificar se as magnitudes de tensão nas barras de carga estão dentro de limites aceitáveis, determinar o carregamento das linhas de transmissão e transformadores e calcular as perdas na transmissão do sistema como um todo.

A demanda do sistema, ou seja, o consumo de energia é obtido por instrumentos de medição, nos quais existe certa limitação, isto é, os dados informados podem apresentar distorções, sendo assim, imprecisos ou inexatos. Também, em determinados casos, o consumo de energia é obtido por estimativa, sendo mais impreciso do que os providos pelos instrumentos de medição.

O cálculo do fluxo de potência é, em geral, realizado utilizando-se métodos computacionais desenvolvidos especificamente para a resolução de sistemas de equações algébricas não-lineares que constituem o modelo estático da rede.

Os componentes de um sistema de energia elétrica podem ser classificados em dois grupos:

- os que estão ligados entre uma barra e a terra (exemplo: geradores, cargas, reatores e capacitores);
- os que estão ligados entre duas barras quaisquer da rede, ou seja, os circuitos. Por exemplo: linhas de transmissão e transformadores.

Os geradores e cargas são considerados a parte externa do sistema e são modelados através de injeções de potências nas barras. Os demais componentes formam a parte interna do sistema. As equações do fluxo de potência são obtidas impondo-se a conservação das potências ativa e reativa em cada barra da rede, isto é, a potência líquida injetada tem que ser igual à soma das potências que fluem pelos componentes internos que têm esta barra como um de seus terminais (BARBOZA, 2001).

3.1.1 Classificação das Barras

A modelagem do fluxo de potência é feita, inicialmente, pela definição de barras: barra PV (Figura 3.2), são as barras de geração; barra PQ (Figura 3.3) para as subestações, pode ser, por exemplo, uma subestação rebaixadora que atende uma determinada região ou município; e uma única barra de folga ($V\theta$) para o sistema. PV significa a especificação de potência ativa e magnitude de tensão, PQ é a especificação de potência ativa e reativa e por fim $V\theta$ significa a especificação da magnitude e ângulo de fase da tensão complexa.



Figura 3.2: Usina Geradora, Barra PV



Figura 3.3: Subestação, Barra PQ

3.1.2 Formulação Matemática do Problema

Um fluxo de potência em uma rede de transmissão, pode ser modelado em coordenadas polares ou retangulares. Neste trabalho optou-se pelas coordenadas retangulares, visto que estas apresentam uma forma quadrática.

Para cada barra do sistema são associadas quatro variáveis, sendo que duas são incógnitas e duas são dados.

- e_k - parte real da tensão complexa na barra k ;
- f_k - parte imaginária da tensão complexa na barra k ;
- P_k - injeção líquida de potência ativa na barra k ;
- Q_k - injeção líquida de potência reativa na barra k .

O cálculo do fluxo de potência corresponde à aplicação da primeira lei de Kirchhoff a cada barra do sistema elétrico e são expressas por:

$$P_k = \sum_{m \in \Omega_k} P_{km}(e_k, e_m, f_k, f_m) \quad (3.1)$$

$$Q_k + Q_k^{sh} = \sum_{m \in \Omega_k} Q_{km}(e_k, e_m, f_k, f_m) \quad (3.2)$$

onde $k = 1, 2, \dots, nb$, sendo nb o número de barras da rede em análise;

- Ω_k é o conjunto de barras vizinhas à barra k ;
- e_k e e_m são as partes reais das tensões complexas nas barras k e m , respectivamente;
- f_k e f_m são as partes imaginárias das tensões complexas nas barras k e m , respectivamente;

- P_{km} é o fluxo de potência ativa no circuito $k - m$;
- Q_{km} é o fluxo de potência reativa no circuito $k - m$;
- Q_{km}^{sh} é o componente da injeção de potência reativa devido ao elemento shunt conectado na barra k .

Shunt correspondem aos equipamentos de um sistema elétrico que são colocados entre uma barra e a terra. Por exemplo, compensadores síncronos, bancos capacitivos, bancos indutivos.

3.1.3 Modelagem Matemática do Problema do Fluxo de Potência

Em coordenadas retangulares, as injeções de potências ativa e reativa na barra i são dadas por:

$$\begin{aligned}
 P_i^{calc} &= \sum_{k=1}^n [e_i(e_k G_{ik} - f_k B_{ik}) + f_i(f_k G_{ik} - e_k B_{ik})] \\
 &= (e_i^2 + f_i^2) G_{ii} + \sum_{\substack{k=1 \\ k \neq i}}^n [e_i(e_k G_{ik} f_k B_{ik}) + f_i(f_k G_{ik} + e_k B_{ik})] \\
 Q_i^{calc} &= \sum_{k=1}^n [f_i(e_k G_{ik} - f_k B_{ik}) - e_i(f_k G_{ik} - e_k B_{ik})] \\
 &= -(e_i^2 + f_i^2) B_{ii} + \sum_{\substack{k=1 \\ k \neq i}}^n [f_i(e_k G_{ik} f_k B_{ik}) - e_i(f_k G_{ik} + e_k B_{ik})]
 \end{aligned}$$

Dessa forma, os resíduos de potência ativa para as barras PV e PQ são:

$$\Delta P_i = (e_i^2 + f_i^2) G_{ii} - \sum_{\substack{k=1 \\ k \neq i}}^n [e_i(e_k G_{ik} - f_k B_{ik}) + f_i(f_k G_{ik} - e_k B_{ik})] - P_{gi} + P_{di} \quad (3.3)$$

e os resíduos de potência reativa para as barras PQ são:

$$\Delta Q_i = -(e_i^2 + f_i^2) B_{ii} - \sum_{\substack{k=1 \\ k \neq i}}^n [f_i(e_k G_{ik} - f_k B_{ik}) - e_i(f_k G_{ik} + e_k B_{ik})] - Q_{gi} + Q_{di} \quad (3.4)$$

Considerando ainda que a barra i seja uma barra PV , então os valores de e_i e f_i

devem satisfazer a relação:

$$e_i^2 + f_i^2 = (V_i^{esp})^2 \quad (3.5)$$

e, portanto, os resíduos dos quadrados das magnitudes das tensões nas barras PV são:

$$\Delta|V_i|^2 = e_i^2 + f_i^2 - |V_i^{esp}|^2 \quad (3.6)$$

As Equações 3.3 e 3.4 representam as equações estáticas do fluxo de potência. Por outro lado, a Equação 3.6 é uma garantia da não violação do módulo da tensão nas barras PV do sistema. No entanto, é necessário estabelecer uma análise em função das variáveis do sistema, conforme o tipo de barra em análise. Assim, sabendo que o problema do fluxo de potência em coordenadas retangulares envolve $2nb - 2$ equações não-lineares com o mesmo número de incógnitas, o sistema de equações não-lineares a ser resolvido é:

$$\begin{cases} \Delta P_i = P_k^{calc} - P_{G_k} + P_{D_{ki}} = 0 \text{ Barras } PV \text{ e } PQ \\ \Delta Q_i = Q_k^{calc} - Q_{G_k} + Q_{D_{ki}} = 0 \text{ Barras } PQ \\ \Delta|V_i|^2 = e_k^2 + f_k^2 - |V_i^{esp}|^2 = 0 \text{ Barras } PV \end{cases} \quad (3.7)$$

onde

$$\begin{aligned} +P_{G_k} - Q_{D_{ki}} &= P_k^{esp} \\ +Q_{G_k} - Q_{D_{ki}} &= Q_k^{esp} \end{aligned}$$

As variáveis do sistema podem ser agrupadas da seguinte maneira:

$$x = \begin{bmatrix} e \\ f \end{bmatrix} \quad (3.8)$$

onde e e f são os vetores com as partes reais e imaginárias dos fasores tensão complexa em todas as barras do sistema elétrico, exceto na barra de folga.

A Equação 3.7 pode ser colocada na forma vetorial como sendo:

$$\begin{aligned} \Delta P &= P^{calc} - P^{esp} = 0 \text{ Barras } PQ \text{ e } PV \\ \Delta Q &= Q^{calc} - Q^{esp} = 0 \text{ Barras } PQ \text{ e } PV \\ \Delta|V|^2 &= |V_i^{esp}|^2 + e_i^2 + f_i^2 \text{ Barras } PV \end{aligned} \quad (3.9)$$

onde

- ΔP é o vetor dos desbalanço de potência ativa nas barras PQ e PV ;
- ΔQ é o vetor dos desbalanço de potência reativa nas barras PQ ;
- $\Delta|V|^2$ é o vetor dos ajustes no módulo do fasor tensão complexa nas barras PV .

Considerando $f(x)$ o vetor das funções

$$f(x) = \begin{bmatrix} \Delta P \\ \Delta Q \\ \Delta |V|^2 \end{bmatrix} \quad (3.10)$$

O sistema de equações não-lineares a ser resolvido pode ser colocado na seguinte forma:

$$f(x) = 0 \quad (3.11)$$

3.1.4 Método de Newton-Raphson (Abordagem pontual)

Expandindo a função $f(x)$ em série de Taylor e truncado-a no termo de primeira ordem, obtém-se:

$$f(x + \Delta x) = f(x) + \frac{\partial f(x)}{\partial x} \Delta x \quad (3.12)$$

onde $\frac{\partial f(x)}{\partial x}$ é a matriz Jacobiana de $f(x)$.

Assim, a cada iteração k , sucessivas aproximações são obtidas como:

$$\Delta x^k = - \left[\frac{\partial f(x^k)}{\partial x} \right]^{-1} \cdot f(x^k) \quad (3.13)$$

$$x^{k+1} = x^k + \Delta x^k \quad (3.14)$$

As Equações 3.8 e 3.10 ficam da seguinte forma:

$$f(x^k) = \begin{bmatrix} \Delta P^k \\ \Delta Q^k \\ \Delta (|V|^2)^k \end{bmatrix} \quad (3.15)$$

$$x^k = \begin{bmatrix} e^k \\ f^k \end{bmatrix} \quad (3.16)$$

Então, a matriz Jacobiana $J(x)$ é

$$J(x) = \begin{bmatrix} \frac{\partial \Delta P}{\partial e} & \frac{\partial \Delta P}{\partial f} \\ \frac{\partial \Delta Q}{\partial e} & \frac{\partial \Delta Q}{\partial f} \\ \frac{\partial \Delta |V|^2}{\partial e} & \frac{\partial \Delta |V|^2}{\partial f} \end{bmatrix} \quad (3.17)$$

Assim, substituindo as Equações 3.15, 3.16 e 3.14 em 3.13, resulta em

$$\begin{bmatrix} \Delta e^k \\ \Delta f^k \end{bmatrix} = - \begin{bmatrix} \frac{\partial \Delta P}{\partial e} & \frac{\partial \Delta P}{\partial f} \\ \frac{\partial \Delta Q}{\partial e} & \frac{\partial \Delta Q}{\partial f} \\ -\frac{\partial \Delta |V|^2}{\partial e} & \frac{\partial \Delta |V|^2}{\partial f} \end{bmatrix}^{-1} \cdot \begin{bmatrix} \Delta P^k \\ \Delta Q^k \\ (\Delta |V|^2)^k \end{bmatrix} \quad (3.18)$$

Obtidas as tensões complexas em todas as barras do sistema elétrico, os fluxos de potências ativa e reativa nas linhas de transmissão podem ser calculados.

4 METODOLOGIAS PROPOSTAS

Confiabilidade é a capacidade de um dispositivo desempenhar seu propósito durante o período de tempo desejado, sob as condições operativas encontradas. Para um sistema de energia elétrica ser confiável, a aferição das variações nas demandas ativa e reativa podem oscilar no máximo de dois por cento, isto é, a confiabilidade para os resultados obtidos deve ser, no mínimo, 98%. Um erro nos equipamentos de medição maior que 2% é inaceitável para a determinação do perfil de tensões quando se utiliza uma ferramenta para o cálculo do fluxo de potência em redes de energia elétrica.

A obtenção de modelos de manutenção para sistemas sujeitos a deterioração é um problema clássico na literatura (MARCHI; CARVALHO; MORAIS, 2001). Em algumas situações reais, o nível de deterioração do sistema não pode ser observado diretamente.

A bibliografia que trata de engenharia de avaliações e depreciação normalmente cita os seguintes fatores de condições físicas como influentes na vida útil de bens (MELEK, 2007):

- danos por acidente;
- danos por catástrofe;
- deterioração pelo tempo;
- dano e desgaste pelo uso.

Uma descrição melhor das condições físicas que causam retiradas de equipamentos pode ser detalhada como:

- **catástrofes:** incêndios, ventos, chuvas, enchentes, terremotos;
- **deterioração pelo tempo:** é aquela que se desenvolve e aumenta com a idade e a extensão da exposição a elementos destrutivos tais como: ferrugem, elementos químicos, variação de temperatura, efeitos mecânicos de congelamento e aquecimento, mesmo com reparos e manutenção e sem uso;

- **dano e desgaste pelo uso:** fricção, impacto, vibração, tração, compressão e temperatura.

Considere que a maior parte dos equipamentos de medição utilizados em sistemas de potência possui um erro de 1%, dado informado pelo fabricante, e que este equipamento possa sofrer uma depreciação de 5% a cada ano, no erro associado a ele. Dessa forma, este equipamento de medição poderá ser utilizado até dez anos sem sofrer nenhum tipo de manutenção.

Este capítulo apresenta as metodologias empregadas na análise do problema de fluxo de potência onde esta característica dos medidores é considerada.

Primeiramente é abordado o uso da Matemática Intervalar e posteriormente a utilização dos Números Fuzzy. A estrutura do software também será caracterizada e detalhada.

Na Matemática Intervalar, com o objetivo de obter um perfil de tensão intervalar para o sistema de transmissão, são discutidas duas hipóteses: (i) uma análise dos erros de leitura nos instrumentos que medem as demandas ativa e reativa nas barras PQ (percentual informado pelo fabricante) e (ii) uma análise da variação das demandas ativa e reativa do sistema em um especificado período de tempo.

Para os Números Fuzzy, analisa-se os erros dos equipamentos de medição nas demandas ativa e reativa das barras PQ (incluindo os erros informados pelo fabricante e pela depreciação do instrumento).

A seguir, é apresentado a entrada de dados do sistema desenvolvido.

4.1 Entrada de dados

A entrada de dados no sistema é feita através de um arquivo texto (*.txt*), onde todas as informações necessárias para o cálculo do fluxo de potência em um sistema elétrico estão presentes. Neste arquivo são encontrados valores para as demandas ativa e reativa nas barras de carga do sistema, níveis de tensão das máquinas síncronas do sistema, resistências e reatâncias das linhas de transmissão, entre outros. Reatâncias são parâmetros relativos às linhas de transmissão e são ocasionados pela indutância e capacitância das mesmas. A Figura 4.1 apresenta o arquivo de entrada dos dados, para um sistema de seis barras, o qual possui duas barras de geração e sete linhas de transmissão.

```

1
**** Sistema de 6 barras (R.N. Dhar) ****
4
  1      4      8.00   37.00   3.00
  1      6      12.3   51.80   4.20
  2      3      72.3  105.00   0.00
  2      5      28.2   64.00   0.00
  3      4       0.00   13.30   0.00.909
  4      6       9.70   40.70   3.00
  5      6       0.00   30.00   0.00.975
9999
5
  1  2 barra1      1.10      -9999+9999
  2  1 barra2      1.10      50.    -15.0  20.0      0.    0.
  3  0 barra3                      55.   13.
  4  0 barra4                      0.    0.
  5  0 barra5                      30.   18.
  6  0 barra6                      50.    5.
9999

```

Figura 4.1: Arquivo de Entrada de Dados

O arquivo texto tem uma formatação padronizada e tem posições fixas de leitura, conforme a informação que se necessite para o cálculo do fluxo de potência. Isso facilita o armazenamento de suas informações em diversas variáveis. Suas informações são dispostas na seguinte estrutura:

- Leitura do arquivo até o código 1. Todo o conteúdo da linha seguinte é considerado o título do arquivo com o caso a ser simulado
- Leitura do arquivo até o código 4. As informações a seguir são os dados de circuitos e devem prosseguir com a leitura até o código 9999 ser encontrado. Este código informa o final desta informação
- Leitura do arquivo até o código 5. As informações a seguir são os dados de barras e devem prosseguir com a leitura até o código 9999 ser encontrado. Este código informa o final desta informação

Dados de Linha (Código 4)

- campos 1 - 4: número da barra de origem;

- campos 9 - 12: número da barra de destino;
- campo 14: número do circuito;
- campos 18 - 23: resistência do circuito em porcentagem;
- campos 24 - 29: reatância do circuito em porcentagem;
- campos 30 - 35: capacitância total do circuito em MVA;
- campos 36 - 40: tap dos transformadores.

Dados de Linha (Código 5)

- campos 1 - 4: número da barra;
- campo 8: tipo da barra;
 - 0: barra PQ;
 - 1: barra PV;
 - 2: barra de referência;
- Campos 10 - 21: nome da barra;
- Campos 23 - 26: magnitude de tensão complexa em pu;
- Campos 27 - 30: ângulo de fase da tensão complexa em graus;
- Campos 31 - 35: potência ativa gerada em MW;
- Campos 36 - 40: potência reativa gerada em MVar;
- Campos 56 - 60: demanda ativa na carga em MW;
- Campos 61 - 65: demanda reativa na carga em MVar;
- Campos 66 - 70: shunt reativo em MVar.

Dessa forma, a leitura de cada campo do arquivo de entrada de dados é realizada e os dados lidos são armazenados em variáveis. Com isso, é possível realizar os cálculos do fluxo de potência.

Por exemplo, os valores 55 e 13 na linha 15 correspondentes à barra 3 no arquivo informam os valores das demandas ativa (55.0 MW) e reativa (13.0 MVar), respectivamente. Esses dois dados citados representam dados imprecisos, ou seja, foram obtidos por instrumentos de medições ou por estimação.

4.2 Diagrama Unifilar

Um sistema de energia elétrica pode ser visualizado graficamente através de um diagrama unifilar. Este consiste na representação das barras no sistema elétrico e dos elementos de interconexão entre elas (linhas de transmissão e transformadores).

A Figura 4.2 apresenta o diagrama unifilar de um sistema elétrico hipotético com 6 barras.

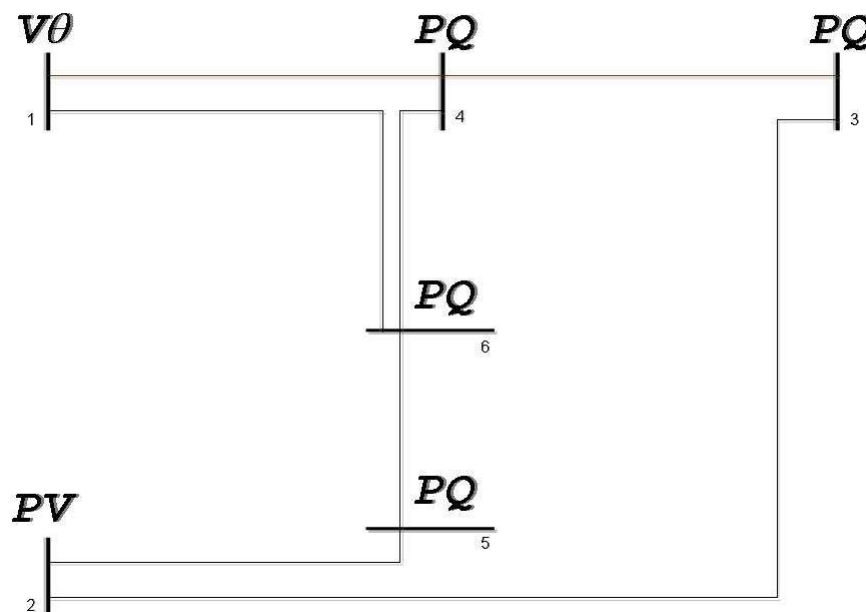


Figura 4.2: Representação do Diagrama Unifilar da Rede Elétrica

Observe que, na figura acima, a rede elétrica em questão, possui uma barra PV (barra 2) e quatro barras PQ (barras 3, 4, 5 e 6). A barra de folga é a barra 1. Observa-se também a existência de sete linhas de transmissão 1-4, 1-6, 2-5, 2-3, 3-4, 4-6 e 5-6.

4.3 Método de Krawczyk (Abordagem Intervalar)

Para resolver o problema do fluxo de potência, foram empregados dois métodos. Para os cálculos pontuais foi utilizado o método de Newton-Raphson (visto no Capítulo 3); para o fluxo de potência intervalar utilizou-se o método de Krawczyk. A solução do fluxo de potência via Newton-Raphson é utilizada como inicialização para o fluxo de potência intervalar. Este procedimento foi adotado visto que no método de Newton-Raphson, a cada iteração, é resolvido um sistema linear, enquanto no método de Krawczyk o sistema não-linear é resolvido somente pelo produto de matrizes.

De acordo com (BARBOZA; DIMURO; REISER, 2004), o qual implementou uma aplicação utilizando o método de Krawczyk em sistemas elétricos de potência, a vantagem deste operador é que nenhum sistema de equações lineares necessita ser resolvido para determinar a atualização na iteração corrente, embora se calcule a matriz inversa pontual. Assim, é possível, através do operador de Krawczyk, resolver um sistema de equações não-lineares somente pelo produto de matrizes.

O operador de Krawczyk $K : R^n \times IR^n \rightarrow IR^n$ é definido por (KRAWCZYK, 1969):

$$K(\tilde{x}, X) = \tilde{x} - Cf(\tilde{x}) - (I - CJ(X))(\tilde{x} - X) \quad (4.1)$$

onde

- \tilde{x} é o intervalo corrente;
- I é uma matriz identidade;
- X é o ponto médio do intervalo corrente;
- C é a inversa do ponto médio de $J(\tilde{x})$.

A convergência do método de Krawczyk é obtida realizando diversas intersecções intervalares. Inicialmente, é definido um intervalo de grande diâmetro, o qual necessariamente deve conter o intervalo-solução do problema. Assim, as atualizações nas variáveis intervalares podem ser realizadas como:

$$X^{k+1} = X^k \cap K(\tilde{x}^k, X^k) \text{ com } k = 0, 1, \dots \quad (4.2)$$

A convergência desse método é definida pelo critério da distância, conforme uma tolerância pré-estabelecida. É realizado uma análise no sistema de equações não-lineares intervalares buscando verificar qual a maior distância entre o ínfimo e o supremo de cada intervalo. Conhecendo esses valores, compara-se com o valor de tolerância especificada pelo usuário, permitindo determinar a convergência ou não do método.

4.4 Proposta Via Matemática Intervalar

4.4.1 Conversão Intervalar das Demandas

Desejando conhecer o perfil de tensão na forma intervalar no sistema, torna-se necessário determinar as demandas ativa e reativa intervalares nas barras PQ do sistema. Para isso, a seguinte metodologia é empregada:

$$\begin{aligned} P_d &= [\textit{infimum}; \textit{supremum}] \\ Q_d &= [\textit{infimum}; \textit{supremum}] \end{aligned} \quad (4.3)$$

O *infimum* representa o limite inferior do intervalo, enquanto que o *supremum* representa o limite superior do intervalo. No entanto, para a definição destes limites é necessário definir qual o tipo de problema em análise. É possível analisar o sistema sob o ponto de vista dos erros em instrumentos de medição ou sob o ponto de vista das variações nas demandas das barras PQ do sistema em um determinado período de tempo.

4.4.2 Erros de Leitura em Instrumentos

Para este tipo de problema, a transformação das demandas ativa e reativa em dados intervalares é feita utilizando os erros associados com cada instrumento de medição.

Dado uma demanda ativa P , deseja-se converter em um intervalo, no qual o erro relativo ao instrumento de medição seja considerado. A Equação 4.4 mostra este processo.

$$P_{int} = [P_i - \varepsilon_i; P_i + \varepsilon_i] \quad (4.4)$$

onde

- P_{int} é o intervalo resultante da conversão da demanda ativa pontual para um dado intervalar;
- ε_i é o erro associado ao instrumento de leitura de potência ativa na barra i do sistema de potência;
- P_i é o valor pontual da potência ativa medida pelo instrumento de medição na barra i do sistema de potência.

Aplica-se a Equação 4.4 em todas as barras PQ do sistema.

Dessa forma, todas as demandas ativa e reativa podem ser expressas em uma forma intervalar, abrangendo o erro de medição do equipamento.

4.4.3 Variação de Demanda no Tempo

As demandas de um sistema de potência sofrem variações conforme o nível de carregamento do sistema e podem apresentar valores bem distintos durante um período de tempo.

Esse tipo de problema consiste em analisar o sistema de potência durante um período de tempo predefinido. Assim, é possível, utilizando a Matemática Intervalar determinar um perfil de tensão na forma de intervalos durante o período de tempo desejado.

A Figura 4.3 mostra como é possível fazer uma análise da variação da demanda em uma barra do sistema de potência elétrico.

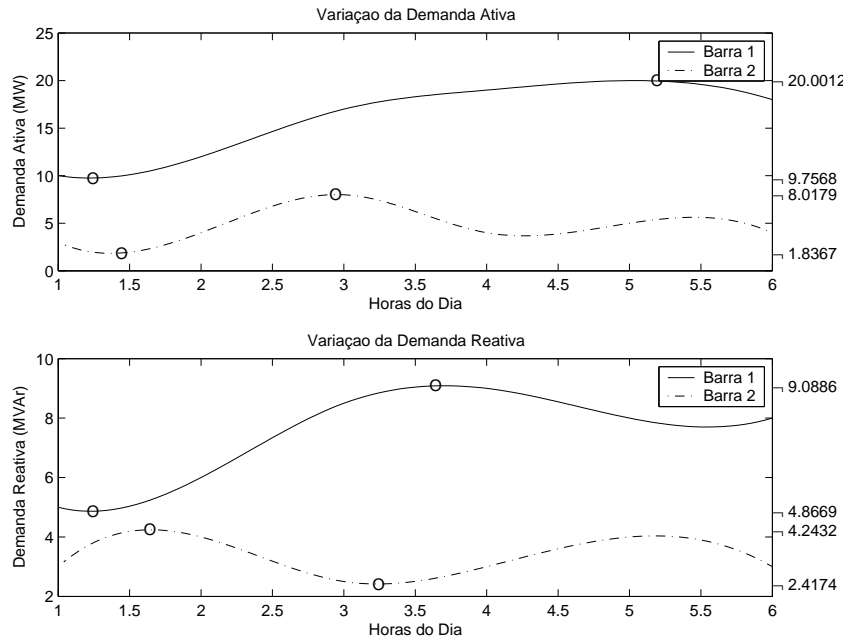


Figura 4.3: Exemplo de Variação de Demanda no Sistema

A Figura 4.3 é a representação da variação de demanda ativa e reativa de duas barras do sistema.

Para a demanda ativa, a barra 1 apresenta valores máximos e mínimos de 20,0012 e 9,7568 MW, respectivamente. A barra 2 apresenta valores máximos e mínimos de 8,0179 e 1,8367 MW, respectivamente. Por outro lado, a demanda reativa na barra 1, tem valores máximo e mínimos de demanda reativa de 9,0886 e 4,8669 MVar; e na barra 2, os valores máximos e mínimos de demanda reativa de 4,2432 e 2,4174 MVar. A Tabela 4.2 apresenta os intervalos para as demandas geradas a partir da análise da Figura 4.3.

Tabela 4.2: Análise Intervalar das Demandas			
Barra	Demanda Ativa	Demanda Reativa	
Barra 1	[9, 7568; 20, 0012]	[4, 8669; 9, 0886]	
Barra 2	[1, 8367; 8, 0179]	[2, 4174; 4, 2432]	

Esta análise permite conhecer o máximo e mínimo das demandas ativa e reativa. Estes pontos correspondem, respectivamente, ao *supremum* e *infimum* do intervalo demanda. E esta análise deve ser feita em cada barra *PQ* do sistema. Assim, é possível determinar o comportamento intervalar da demanda do sistema, pois todos os prováveis pontos de operação dos sistema estão compreendidos nos intervalos de demanda.

4.4.4 Implementação

O objetivo dessa implementação é desenvolver um software para determinar o perfil das tensões em forma intervalar. A análise de ambos os problemas apresentados anteriormente, podem ser realizados de maneira semelhante. A diferença entre ambos é a inicialização dos intervalos de tensão complexas nas barras PQ .

Para o problema de erros de leitura em instrumentos de medição, o diâmetro dos intervalos das demandas ativa e reativa são pequenos, visto que os erros considerados neste tipo de problema variam em torno de 1% do valor pontual (leitura). Por outro lado, no problema da variação de demanda no tempo, os intervalos iniciais possuem grande diâmetro, pois, devido a uma demanda extremamente flutuante durante um intervalo de tempo, os valores iniciais são relativamente grandes quando comparados com os erros no problema dos erros em instrumentos.

A Figura 4.4 mostra um diagrama do processo computacional, partindo dos cálculos pontuais e finalizando no processo intervalar obtendo a magnitude de tensão e o ângulo de fase intervalar.

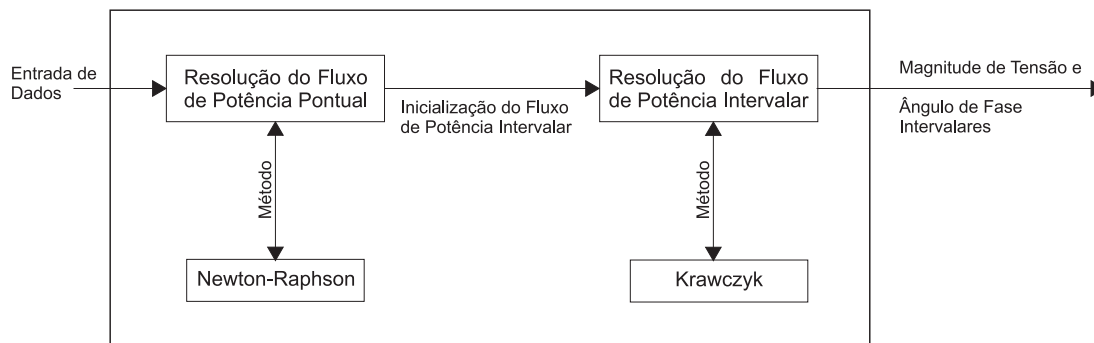


Figura 4.4: Diagrama de Resolução do Problema

O processo intervalar só é executado com a convergência do processo pontual. Assim, se o sistema não possuir convergência no método pontual (Newton-Raphson) o processo intervalar não será executado. Isso significa que o sistema em análise não possui solução e caracteriza a não convergência do método.

A inicialização do processo pontual se dá atribuindo 1,0 pu (por unidade) para a magnitude e 0° para o ângulo de fase das tensões complexas em todas as barras PQ do sistema. Nas barras PV os valores de magnitude de tensão complexa são lidos no arquivo texto.

As etapas da aplicação desenvolvida na metodologia pontual são:

1. montar a matriz admitância nodal;
2. inicializar as tensões complexas;

3. inicializar o contador de iterações;
4. calcular os desbalanços de potências ativa e reativa;
5. testar a convergência;
6. testar o número de iterações;
7. incrementar o contador de iterações;
8. montar a matriz Jacobiana;
9. resolver o sistema linear;
10. atualizar as variáveis pontuais;
11. retornar ao passo 4.

Por outro lado, a inicialização do processo intervalar se dá através da consideração do perfil de tensão pontual determinado anteriormente. Este valor é utilizado como o ponto médio da magnitude de tensão e ângulo de fase no processo intervalar. Dessa forma, seleciona-se qual tipo de erro será considerado (erros de medição ou variação da demanda no tempo) e inicializa-se o problema do fluxo de potência intervalar.

As etapas de aplicação para o desenvolvimento do sistema intervalar são:

1. inicializar as tensões complexas;
2. inicializar o contador de iterações;
3. calcular os desbalanços de potências ativa e reativa;
4. testar a convergência;
5. testar o número de iterações;
6. incrementar o contador de iterações;
7. aplicar o método de Krawczyk;
8. atualizar as variáveis intervalares;
9. retornar ao passo 4.

Realizando todas as tarefas o perfil de tensões intervalares do sistema pode ser encontrado.

4.5 Proposta Via Números Fuzzy

4.5.1 Geração das Demandas como Números Fuzzy

Pela Lógica Fuzzy, quem define a imprecisão nas medidas é o especialista do conhecimento. Neste caso, considerou-se o erro informado pelo fabricante (de 1%) e o erro máximo admissível para os instrumentos de medição na área em estudo de 2%.

Baseado nessas premissas, considera-se as demandas indicadas pelos instrumentos na forma de Números Fuzzy. A Figura 4.5 mostra o diagrama de como é realizada a conversão das demandas para Números Fuzzy.

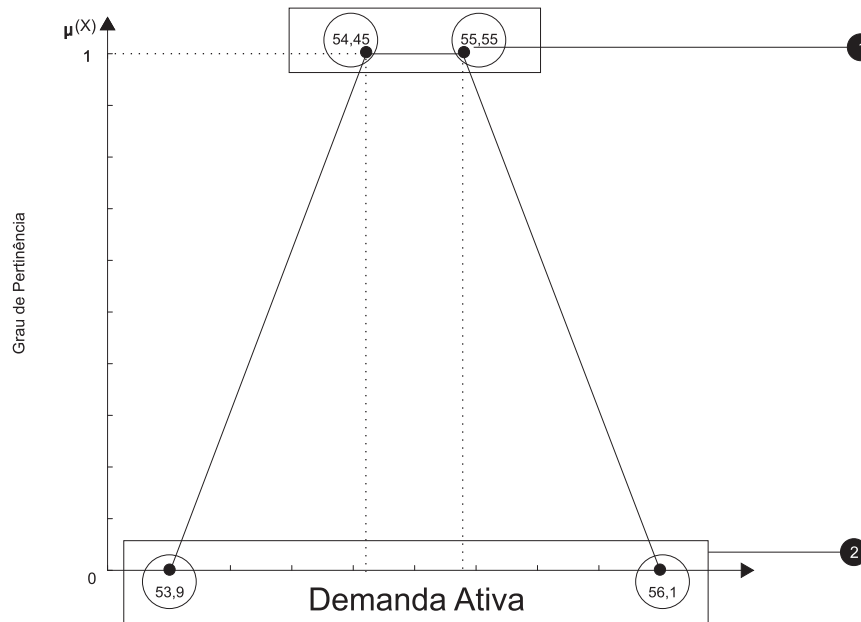


Figura 4.5: Diagrama da Resolução do Problema pelos Números Fuzzy

Pode-se explicar o processo de geração dos Números Fuzzy correspondentes às demandas de potências ativa e reativa do sistema analisando a Figura 4.5. Tem-se:

1. as porcentagens de imprecisão são definidas em 1% para os erros de medição e 2% para o limite máximo de erro admissível;
2. é atribuído o grau de pertinência 1 ao intervalo gerado com a variação de 1% nas demandas ativa e reativa do arquivo de entrada de dados (ponto 1 na Figura 4.5);
3. é atribuído o grau de pertinência zero ao intervalo gerado pela variação de 2% nas demandas ativa e reativa do arquivo de entrada de dados (ponto 2 na Figura 4.5);
4. uma reta é traçada entre os dois pontos conhecidos, para as pertinências zero e um.

Por exemplo, considere o valor da demanda de potência ativa em uma barra do arquivo de entrada de dados de um sistema elétrico hipotético de 55 MW. Aplicando as

regras do especialista, esse valor sofre uma variação de 1% e 2%, respectivamente. Dessa forma, os seguintes intervalos são criados: $[54, 45; 55, 55]$ e $[53, 90; 56, 10]$, respectivamente. Atribui-se o grau de pertinência 1 para o primeiro intervalo gerado e o grau de pertinência zero para o segundo intervalo gerado. Uma reta é traçada nos pontos conhecidos, formando uma representação trapezoidal.

4.5.2 Implementação

O fluxo de potência intervalar é realizado duas vezes, um fluxo com os intervalos dos valores com pertinência 1 das demandas ativa e reativa (ponto 1 na Figura 4.5) e outro com os intervalos dos valores de pertinência zero das demandas ativa e reativa (ponto 2 na Figura 4.5).

Baseado na Figura 4.5, tem-se a seguinte ordem de eventos para o fluxo de potência Fuzzy:

1. executa-se um fluxo de potência intervalar com os valores cujas pertinências são um. À solução obtida é atribuído grau de pertinência um;
2. executa-se um fluxo de potência intervalar com os valores cujas pertinências são zero. À solução obtida é atribuído grau de pertinência zero;
3. traçam-se duas retas, uma com os ínfimos dos intervalos solução, e outra, com os supremos. Assim, gera-se um Número Fuzzy Trapezoidal para a solução.

4.5.3 Interpretação da Solução

Os resultados obtidos caracterizam os graus de pertinências das magnitudes de tensão e ângulos de fase em uma determinada barra da rede elétrica.

A Figura 4.6 ilustra a representação das magnitudes de tensão dos Números Fuzzy.

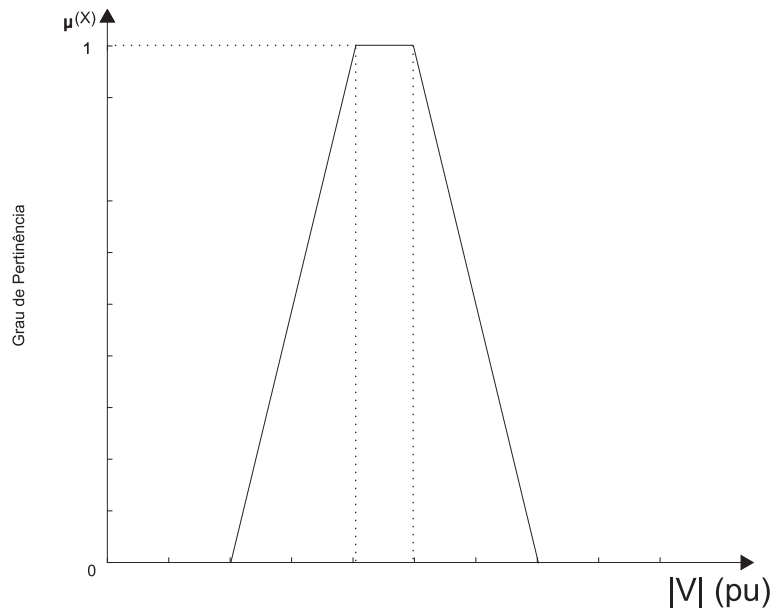


Figura 4.6: Diagrama da Interpretação dos Números Fuzzy

Pode-se depreender que a partir dos intervalos vistos na Figura 4.6 (pertinência 1 e pertinência zero), que, qualquer valor para outras pertinências são obtidas através das equações das retas.

4.6 Descrição dos Recursos Computacionais Utilizados

A implementação computacional foi realizada no sistema operacional Linux (Ubuntu 6.06 LTS - Dapper Drake), utilizando o compilador g++ 4.0, a biblioteca C-XSC versão 2.0. A geração dos gráficos é feita utilizando o Gnuplot versão 4.0. O ambiente de desenvolvimento e gerenciamento de projetos foi a IDE Anjuta 2.2.2.

A configuração do equipamento de teste é a seguinte:

- **processador:** Celeron (R) 2,8 GHz;
- **memória RAM:** 1 GB;
- **memória de vídeo:** 128 MB;
- **disco rígido:** 80 GB.

4.7 Síntese

A Matemática Intervalar foi escolhida para modelar o problema do fluxo de potência devido a sua característica de tratar os números não como valores pontuais, mas sim através de um conjunto de valores (intervalo).

Adotou-se a Lógica Fuzzy por apresentar as seguintes características em relação a outras técnicas de controle:

- robusta por não requerer entradas precisas;
- modificada facilmente, pois é baseada em regras;
- solução mais rápida e barata em alguns casos;
- implementável em microprocessadores.

Com um especialista, que define algumas regras, têm-se os valores de pertinência, no qual é possível representar os valores graficamente, gerando uma visualização mais agradável e mais precisa dos resultados.

Em relação à escolha do método de Krawczyk para resolver o sistema de equações não-lineares intervalares, este foi implementado em (BARBOZA; DIMURO; REISER, 2004) no software computacional MatLab utilizando a biblioteca IntLab. Então, baseou-se na codificação já implementada e esta foi adaptada ao software proposto. De acordo com (BARBOZA; DIMURO; REISER, 2004), a vantagem deste operador é que nenhum sistema de equações lineares necessita ser resolvido para determinar a atualização da próxima iteração.

5 RESULTADOS NUMÉRICOS

Este capítulo apresenta os resultados obtidos pelas metodologias descritas no capítulo anterior para dois sistemas elétricos: um do IEEE com 6 barras e outro, sistema real de 352 barras representando a região Sul-Sudeste do Brasil. As metodologias foram desenvolvidas na linguagem de programação C++ (compilador g++ 4.0), utilizando a biblioteca C-XSC (versão 2.0), foram simuladas em um PC INTEL CELERON 2,8 GHz com Sistema Operacional Linux (Ubuntu 6.06) com 1 Gb de memória RAM.

Para os sistemas testados, são apresentados os seguintes resultados:

- influência dos erros dos equipamentos de medição, via Matemática Intervalar, no perfil de tensões do sistema elétrico;
- influência da variação das demandas em um período de tempo especificado, via Matemática Intervalar, sobre perfil de tensões do sistema elétrico;
- influência dos erros de equipamentos de medição e da depreciação dos mesmos, via Números Fuzzy sobre o perfil de tensões da rede.

No sistema IEEE-6 barras os resultados obtidos pelo software proposto são comparados com os obtidos em (BARBOZA; DIMURO; REISER, 2004, 2005) que utilizou a biblioteca IntLab no ambiente MatLab.

No sistema real de 352 barras, os resultados apresentados são para algumas barras cujas as magnitudes de tensão são as menores do sistema e possuem demanda de potência.

A Tabela 5.1 apresenta as principais características dos sistemas simulados.

Tabela 5.1: Principais Características para os Sistemas Testados

Sistema	Barras	Circuitos	Geradores
IEEE-6	6	7	2
SSB-352	352	476	29

As colunas 2, 3 e 4 apresentam, respectivamente, o número total de barras, de circuitos e de geradores existente em cada sistema elétrico.

5.1 IEEE-6 Barras

Neste sistema de seis barras, as demandas totais de potências ativa e reativa estão expressas com o valor pontual de 135 MW e 36 MVar, respectivamente. Considerou-se uma variação em torno do ponto médio das demandas de potências ativa e reativa em cada barra de carga de 2%, tanto nas simulações do MatLab, como do C++. Esse valor percentual é devido à consideração de erros de leitura nas demandas de potências ativa e reativa nas barras de carga.

Utilizando Números Fuzzy considerou-se um erro de 1% e 2%. A variação de 1% corresponde ao erro do instrumento informado pelo fabricante, e a variação de 2% considera o erro máximo de depreciação admitida para o equipamento de medição.

Para a obtenção dos resultados intervalares no problema da variação da demanda no tempo foi considerada uma variação de 15% em todas as demandas pontuais ativa e reativa das barras PQ do sistema.

5.1.1 Resultado Pontual

A Tabela 5.2 refere-se aos resultados obtidos para o fluxo de potência pontual.

Tabela 5.2: Sistema Pontual IEEE-6 Barras

Barra	Magnitude(pu)	Ângulo(°)
1	1,1000	0
2	1,1000	-3,1890
3	0,9051	-11,7277
4	1,0089	-8,8368
5	0,9384	-11,5112
6	0,9880	-11,0893

Os resultados da Tabela 5.2 são utilizados como ponto médio para a inicialização das variáveis do processo intervalar e são extremamente importantes para a convergência do método.

5.1.2 Erro no Equipamento de Medição - Resultado Intervalar pelo MatLab (IntLab)

A Tabela 5.3 apresenta os resultados obtidos no MatLab. Esses resultados correspondem às magnitudes e ângulos de fase das tensões complexas em todas as barras da rede elétrica.

Tabela 5.3: MatLab: Sistema Intervalar IEEE-6 Barras

Barra	Magnitude(pu)	Erro (%)	Ângulo(°)	Erro (%)
1	[1, 0999; 1, 1001]		[0, 0000; 0, 0000]	
2	[1, 0988; 1, 1012]	0, 0011	[−3, 1954; −3, 1826]	0, 0020
3	[0, 8930; 0, 9174]	0, 0135	[−12, 0675; −11, 3934]	0, 0287
4	[0, 9983; 1, 0195]	0, 0105	[−9, 0777; −8, 5992]	0, 0271
5	[0, 9226; 0, 9542]	0, 0168	[−11, 8858; −11, 1450]	0, 0322
6	[0, 9749; 1, 0011]	0, 0133	[−11, 4116; −10, 7724]	0, 0288

Comparando os resultados das Tabelas 5.2 e 5.3, observa-se que os resultados de magnitude de tensão pontual estão contidos nos resultados de magnitude de tensão intervalar.

5.1.3 Erro no Equipamento de Medição - Resultado Intervalar pelo C++ (C-XSC)

A Tabela 5.4 apresenta os resultados obtidos pelo C++. Esses resultados correspondem às magnitudes e ângulos de fase das tensões complexas em todas as barras da rede elétrica.

Pode-se verificar que os resultados intervalares apresentados na Tabela 5.4 contém os resultados obtidos no fluxo de potência pontual (Tabela 5.2).

Tabela 5.4: C++: Sistema Intervalar IEEE-6 Barras

Barra	Magnitude(pu)	Erro (%)	Ângulo(°)	Erro (%)
1	[1, 0999; 1, 1000]		[0, 0000; 0, 0000]	
2	[1, 0993; 1, 1006]	0, 0006	[−3, 1907; −3, 1872]	0, 0005
3	[0, 8998; 0, 9104]	0, 0059	[−11, 7810; −11, 6750]	0, 0045
4	[1, 0043; 1, 0134]	0, 0045	[−8, 8703; −8, 8037]	0, 0038
5	[0, 9319; 0, 9448]	0, 0069	[−11, 5720; −11, 4511]	0, 0053
6	[0, 9820; 0, 9939]	0, 0060	[−11, 1434; −11, 0359]	0, 0048

Analisando a Tabela 5.3, para a barra dois, a magnitude de tensão está compreendida entre [1, 0988; 1, 1012] e no C++, este valor é [1, 0993; 1, 1006]. Observe que o diâmetro da magnitude de tensão obtido no C++ é menor do que o obtido no MatLab. Além disso, se a comparação for realizada em termos do ângulo de fase da tensão complexa na barra dois, o diâmetro do intervalo obtido no C++ também é menor do que o do MatLab ([−3, 1954; −3, 1826] no MatLab, [−3, 1907; −3, 1872]) no C++).

Para a barra cinco, a magnitude de tensão no MatLab está compreendida entre $[0,9226; 0,9542]$ e no C++ entre $[0,9319; 0,9448]$. O intervalo para o ângulo de fase da tensão complexa nesta barra, no MatLab está entre $[-11,8858; -11,1450]$ e no C++ entre $[-11,5720; -11,4511]$. Note que o intervalo da magnitude de tensão no C++ é menor do que no MatLab e que o intervalo do ângulo de fase da tensão complexa no MatLab é maior do que no C++.

Finalmente, analisando a barra 6, os intervalos $[0,9749; 1,0011]$ e $[0,9820; 0,9939]$ correspondem aos valores das magnitudes de tensão, respectivamente, no MatLab e C++. Para os intervalos do ângulo de fase da tensão complexa, obteve-se no MatLab $[-11,4116; -10,7724]$ e no C++ $[-11,1434; -11,0359]$. Assim, as mesmas conclusões anteriores são novamente encontradas.

Observa-se nas Tabelas 5.3 e 5.4 que os diâmetros das magnitudes de tensão são bem pequenos. Isto se deve ao fato de que, para esta rede elétrica, as barras 1 e 2 são, respectivamente, a barra de folga e uma barra *PV*. Nestes casos, as magnitudes de tensão destas barras não fazem parte das variáveis do processo iterativo. Nesta tabela, também observa-se que o ângulo de fase da tensão complexa na barra 1 possui diâmetro nulo. Recorde que a barra 1 é a barra de folga do sistema elétrico, portanto o ângulo de fase de sua tensão complexa não é variável do processo iterativo. As grandezas discutidas anteriormente são especificadas a priori para o problema do fluxo de potência.

5.1.4 Erro no Equipamento de Medição - Resultado com Números Fuzzy

As demandas tanto ativa como reativa do sistema de seis barras são modeladas com Números Fuzzy na forma trapezoidal, seguindo os princípios da seção 4.5.1.

A barra 2 é uma barra de geração (*PV*). Portanto, a magnitude de sua tensão é definida a priori pelo especialista do conhecimento. Dessa forma, esta variável não participa do processo iterativo.

A seguir, são mostrados nas figuras os valores das demandas ativa e reativa das barras 3, 5 e 6, respectivamente. Essas demandas estão representadas na forma trapezoidal.

As Figuras 5.1 e 5.2 mostram, respectivamente, as demandas ativa e reativa da barra 3. Os valores pontuais das demandas de potências ativa e reativa, nesta barra, são, respectivamente, 55 MW e 13 MVar.

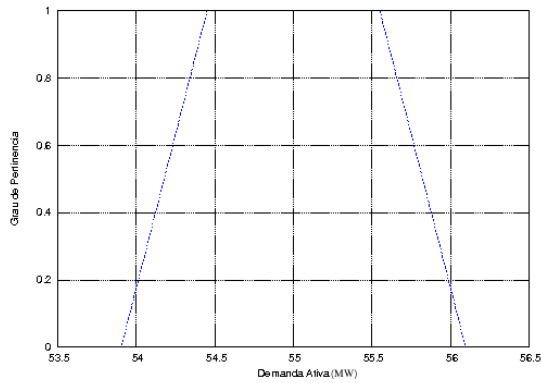


Figura 5.1: Dado Fuzzy para a Demanda Ativa da Barra 3

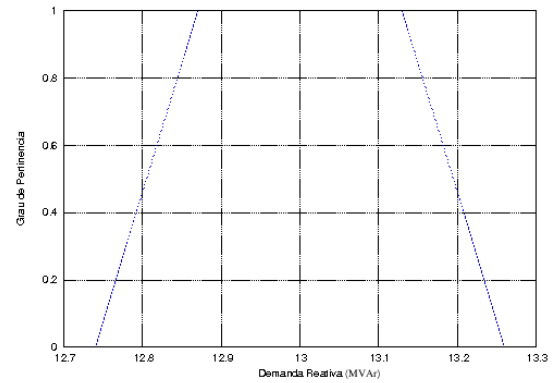


Figura 5.2: Dado Fuzzy para a Demanda Reativa da Barra 3

As Figuras 5.3 e 5.4 caracterizam as demandas de potências ativa e reativa na barra 5. O grau de pertinência um é atribuído aos intervalos $[54, 45; 55, 55]$ MW e $[12, 87; 13, 13]$ MVar.

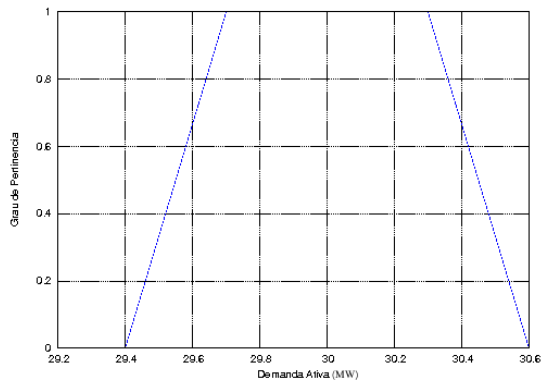


Figura 5.3: Dado Fuzzy para a Demanda Ativa da Barra 5

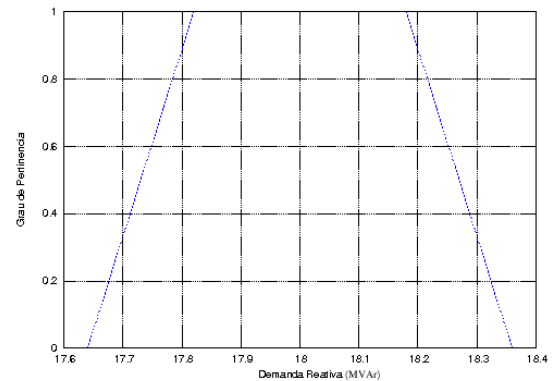


Figura 5.4: Dado Fuzzy para a Demanda Reativa da Barra 5

A demanda de potência ativa da barra 6, a Figura 5.5, apresenta os valores com pertinência 1 para o intervalo $[49, 5; 50, 5]$ MW, e com pertinência zero, o intervalo $[49; 51]$ MW. A demanda de potência reativa, Figura 5.6, possui com pertinência zero o intervalo $[4, 9; 5, 1]$ MVar. Valores de demanda reativa inferiores a 4,9 MVar e superiores a 5,1 MVar não pertencem ao conjunto dos Números Fuzzy que caracteriza a demanda pontual de 5 MVar medida na barra 6.

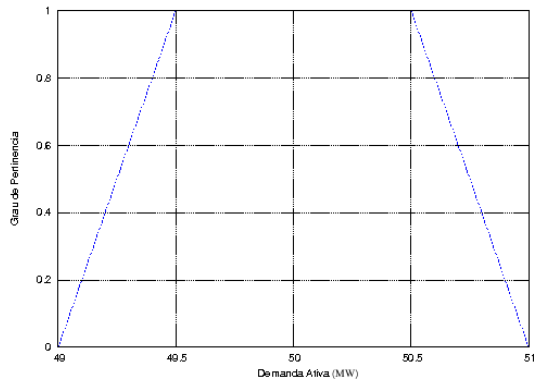


Figura 5.5: Dado Fuzzy para a Demanda Ativa da Barra 6

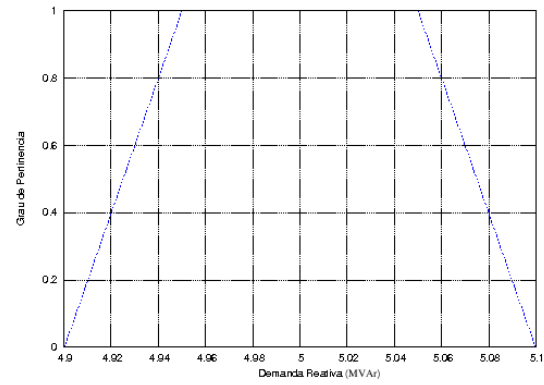


Figura 5.6: Dado Fuzzy para a Demanda Reativa da Barra 6

Então, para obter a magnitude de tensão e o ângulo de fase das tensões complexas nas barras da rede elétrica usando os Números Fuzzy, executa-se dois fluxo de potência com as demandas ativa e reativa com pertinência 1 e pertinência zero, respectivamente.

O resultado para a barra 3 da magnitude de tensão e ângulo de fase são mostrados nas Figuras 5.7 e 5.8, respectivamente.

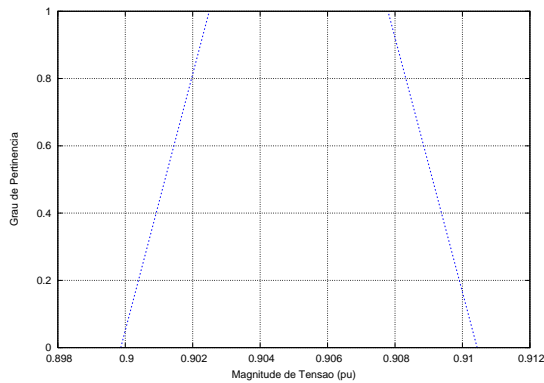


Figura 5.7: Resultado Fuzzy para a Magnitude de Tensão da Barra 3

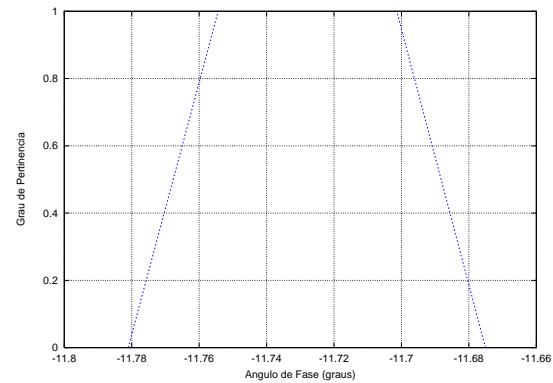


Figura 5.8: Resultado Fuzzy para o Ângulo da Fase de Tensão da Barra 3

Os Números Fuzzy para a magnitude de tensão e para o ângulo de fase da tensão complexa da barra 5 estão mostrados nas Figuras 5.9 e 5.10.

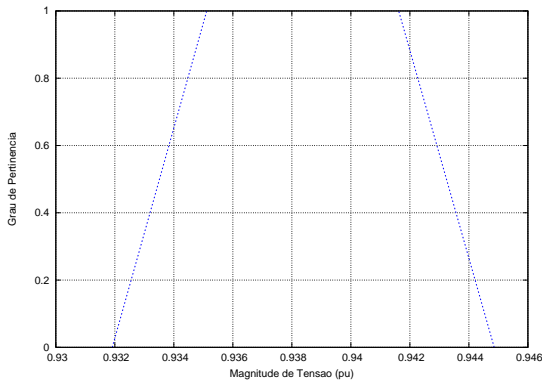


Figura 5.9: Resultado Fuzzy para a Magnitude de Tensão da Barra 5

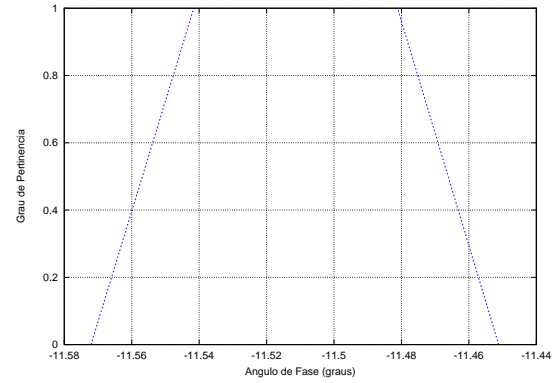


Figura 5.10: Resultado Fuzzy para o Ângulo da Fase de Tensão da Barra 5

Analizando a Figura 5.11, a magnitude da tensão na barra 6 está compreendida no intervalo $[0,9820; 0,9939]$. Por outro lado, tensões entre 0,9849 e 0,9909 pu fazem parte do conjunto dos Números Fuzzy com grau de pertinência 1. Ainda pode-se determinar, a partir da Figura 5.11, que a tensão de 0,984 pu apresentam um grau de pertinência 0,68.

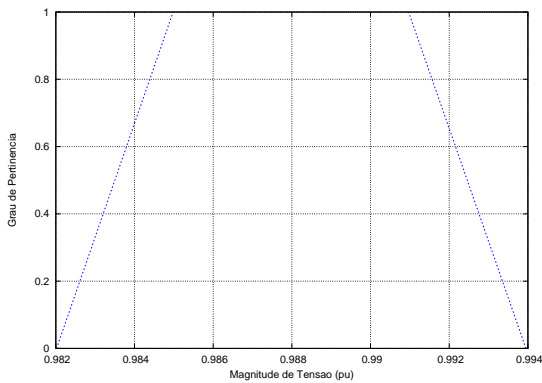


Figura 5.11: Resultado Fuzzy para a Magnitude de Tensão da Barra 6

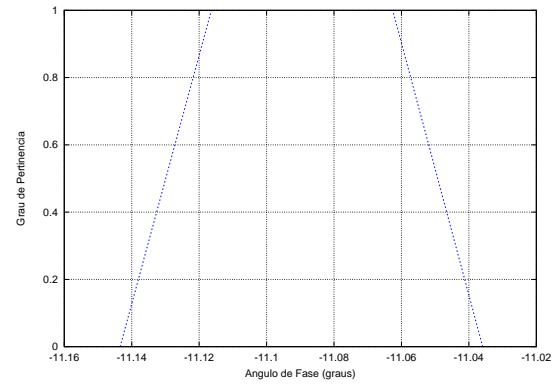


Figura 5.12: Resultado Fuzzy para o Ângulo da Fase de Tensão da Barra 6

Ainda na barra 6, a Figura 5.12 mostra o ângulo de fase da tensão complexa. Observa-se que o ângulo de fase $-11,1^\circ$ tem como pertinência 1 e que o ângulo de fase de $-11,02^\circ$ possui pertinência zero, isto é, não faz parte do Conjunto Fuzzy.

5.1.5 Resultados Numéricos para o Problema da Variação da Demanda no Tempo

A Tabela 5.5 apresenta os resultados intervalares obtidos, considerando o problema da variação da demanda no tempo.

Tabela 5.5: Resultado Intervalar do Sistema IEEE 6 Barras

Barra	Magnitude(pu)	Ângulo(°)
1	[1, 0999; 1, 1000]	[0, 0000; 0, 0000]
2	[1, 0988; 1, 1011]	[−3, 1921; −3, 1857]
3	[0, 8766; 0, 9349]	[−12, 0106; −11, 4578]
4	[0, 9851; 1, 0335]	[−9, 0075; −8, 6725]
5	[0, 9034; 0, 9760]	[−11, 8293; −11, 2096]
6	[0, 9558; 1, 0214]	[−11, 3777; −10, 8152]

O resultado intervalar da magnitude de tensão na barra quatro $[0, 985; 1, 033]$. A mesma análise pode ser feita para o ângulo de fase na barra seis $[−11, 38; −10, 82]$.

Neste caso, notam-se intervalos de diâmetro relativamente grandes, pois inicialmente as demandas ativa e reativa nas barras PQ foram supostas com variações de 15% em torno do dado pontual.

Assim, esta variação se reflete no perfil de tensão e acaba tornando o diâmetro dos intervalos maiores.

5.2 SSB-352 Barras

Para o sistema equivalente real abrangendo as regiões Sul e Sudeste do Brasil com 352 barras foram selecionadas as 5 barras mais problemáticas deste sistema, ou seja, são mostrados resultados para as barras com menor magnitude de tensão e que possuem demanda de potência. As demandas totais de potências ativa e reativa são, respectivamente, 4.404,90 MW e 1.558,70 MVar.

5.2.1 Resultado Pontual

A Tabela 5.6 refere-se aos resultados obtidos para o fluxo de potência pontual.

Tabela 5.6: Resultado Pontual para SSB-352 Barras

Barra	Magnitude(pu)	Ângulo(°)
5	0, 9252	−56, 1196
36	0, 9168	−48, 0402
37	0, 9298	−46, 6055
186	0, 9273	−39, 6909
232	0, 8715	−49, 4225

Observa-se pelos resultados da Tabela 5.6 que os valores das magnitudes de ten-

são nas barras selecionadas estão relativamente baixos. Em uma operação normal da rede elétrica, pode-se aceitar, por breves períodos de tempo, magnitudes de tensão de, no mínimo, 0,9 pu.

5.2.2 Erro no Equipamento de Medição - Resultado Intervalar pelo C++ (C-XSC)

A Tabela 5.7 refere-se aos resultados obtidos para o fluxo de potência intervalar.

Tabela 5.7: Resultados para o SSB-352 barras

Barra	Magnitude(pu)	Ângulo(°)
5	[0,9006; 0,9501]	[−57,5903; −54,6967]
36	[0,8914; 0,9425]	[−49,3780; −46,7563]
37	[0,9063; 0,9534]	[−47,7528; −45,5001]
186	[0,9036; 0,9511]	[−40,6379; −38,7802]
232	[0.8408; 0.9025]	[−51.1102; −47.8159]

Observa-se uma pequena elevação no diâmetro dos intervalos. Isto se deve a topologia do sistema de 352 barras e ao ponto de operação considerado, pois o sistema apresenta maiores valores de demandas ativa e reativa e maior número de interconexões entre os circuitos, o que aumenta um pouco o diâmetro dos intervalos-solução.

5.2.3 Erro no Equipamento de Medição - Resultado pela Lógica Fuzzy

Inicialmente, são apresentadas as demandas de potências ativa e reativa das barras selecionadas na sua forma de Números Fuzzy.

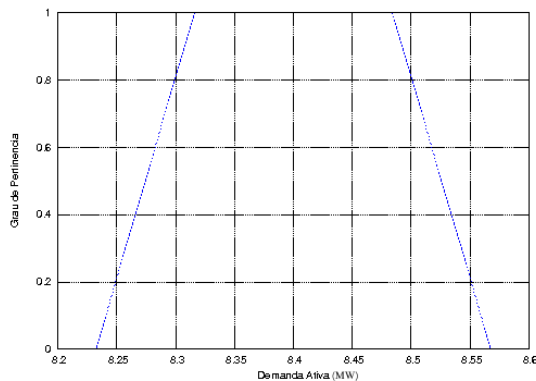


Figura 5.13: Dado Fuzzy para a Demanda Ativa da Barra 5

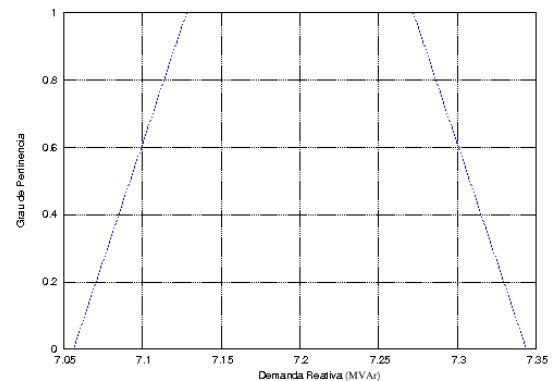


Figura 5.14: Dado Fuzzy para a Demanda Reativa da Barra 5

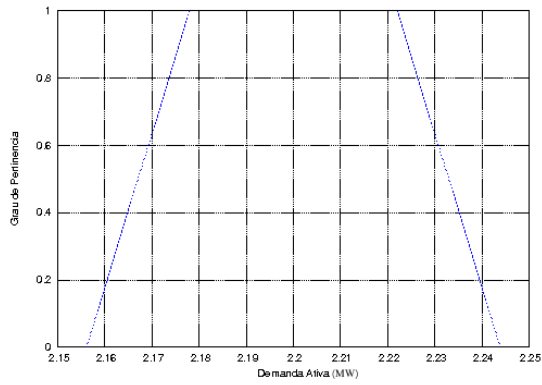


Figura 5.15: Dado Fuzzy para a Demanda Ativa da Barra 36

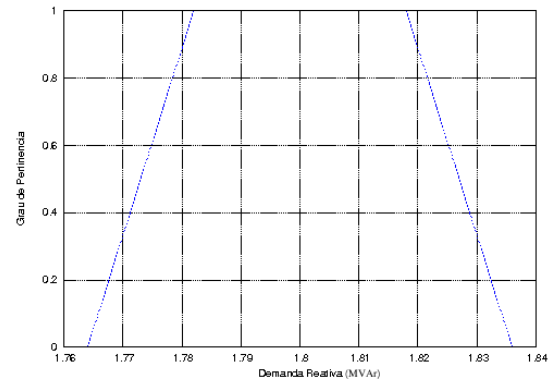


Figura 5.16: Dado Fuzzy para a Demanda Reativa da Barra 36

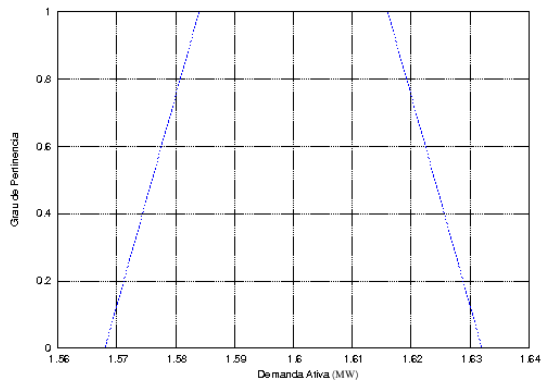


Figura 5.17: Dado Fuzzy para a Demanda Ativa da Barra 37

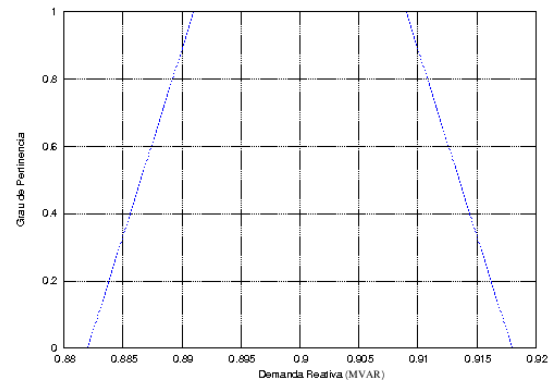


Figura 5.18: Dado Fuzzy para a Demanda Reativa da Barra 37

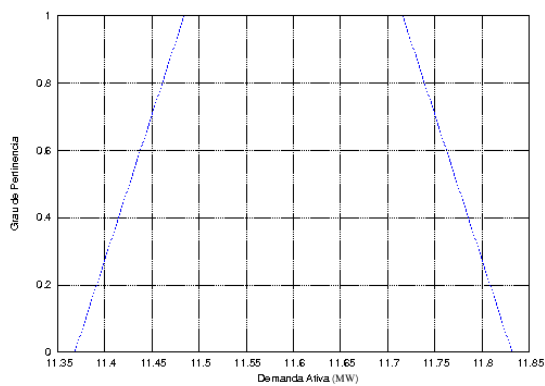


Figura 5.19: Dado Fuzzy para a Demanda Ativa da Barra 186

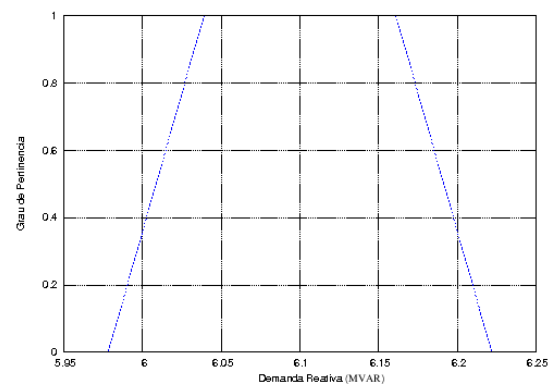


Figura 5.20: Dado Fuzzy para a Demanda Reativa da Barra 186

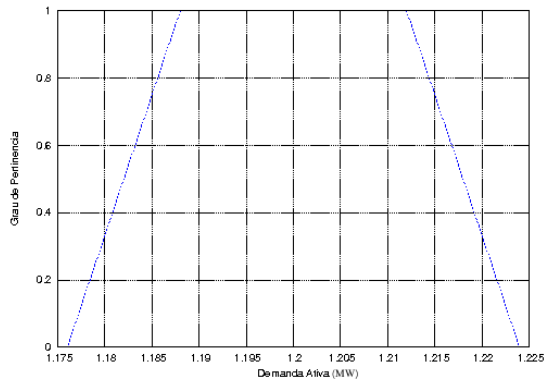


Figura 5.21: Dado Fuzzy para a Demanda Ativa da Barra 232

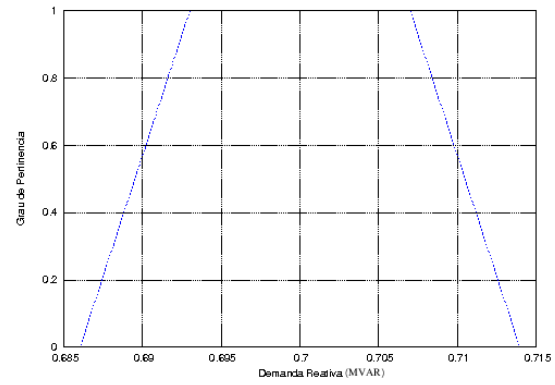


Figura 5.22: Dado Fuzzy para a Demanda Reativa da Barra 232

As magnitudes de tensão e os ângulos de fase das tensões complexas nas barras selecionadas são mostradas a seguir.

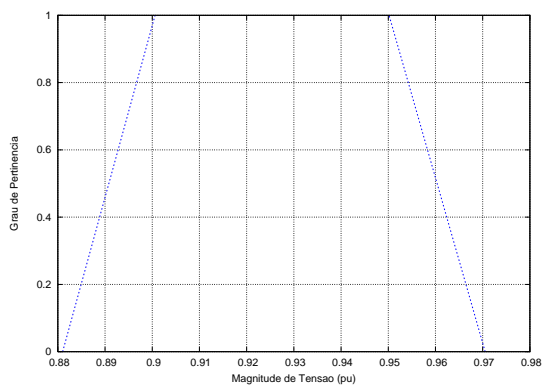


Figura 5.23: Resultado Fuzzy para a Magnitude da Tensão da Barra 5

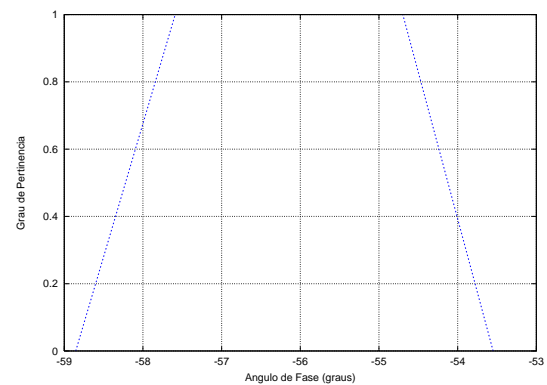


Figura 5.24: Resultado Fuzzy para o Ângulo de Fase da Tensão da Barra 5

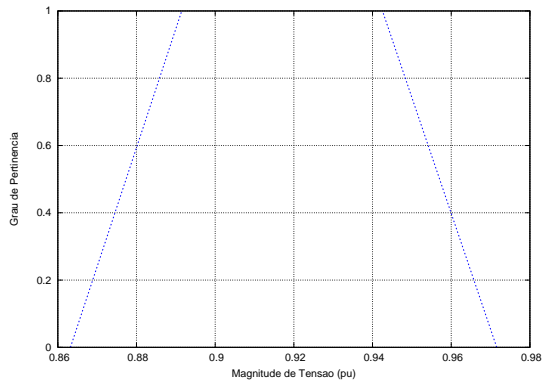


Figura 5.25: Resultado Fuzzy para a Magnitude da Tensão da Barra 36

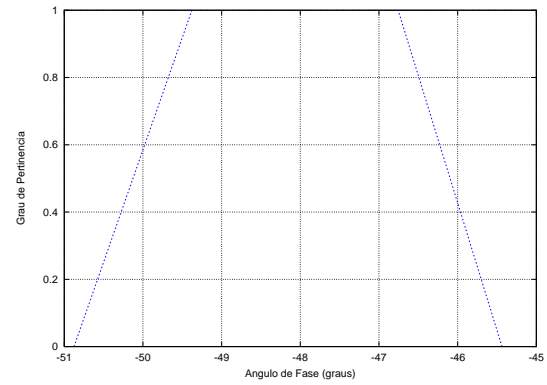


Figura 5.26: Resultado Fuzzy para o Ângulo de Fase da Tensão da Barra 36

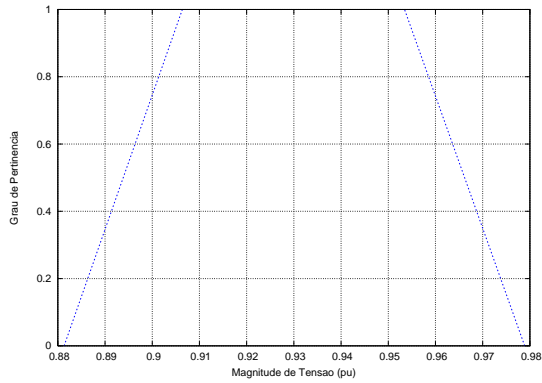


Figura 5.27: Resultado Fuzzy para a Magnitude da Tensão da Barra 37

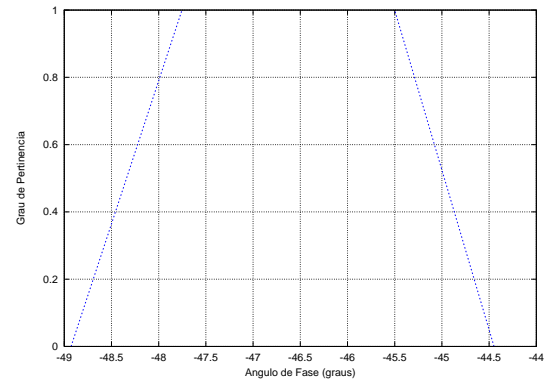


Figura 5.28: Resultado Fuzzy para o Ângulo de Fase da Tensão da Barra 37

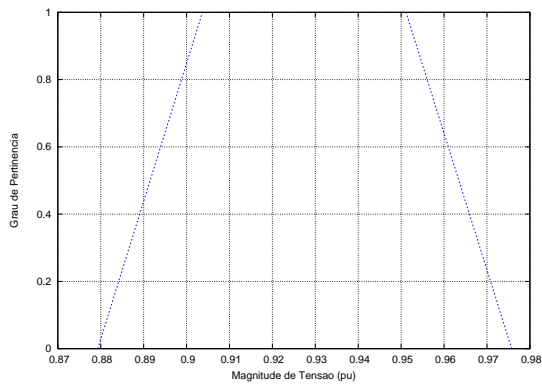


Figura 5.29: Resultado Fuzzy para a Magnitude da Tensão da Barra 186

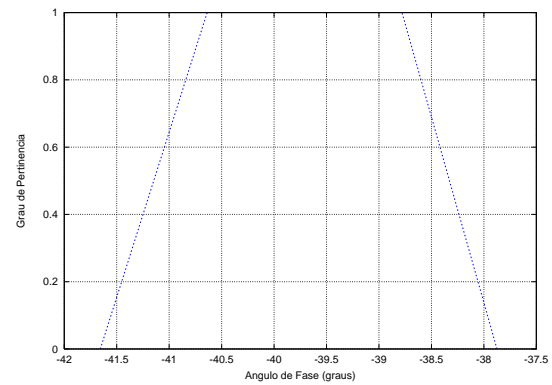


Figura 5.30: Resultado Fuzzy para o Ângulo de Fase da Tensão da Barra 186

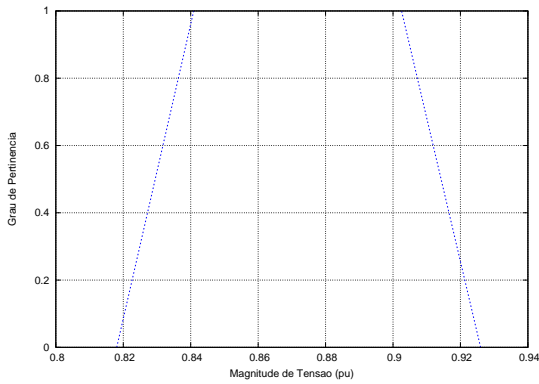


Figura 5.31: Resultado Fuzzy para a Magnitude da Tensão da Barra 232

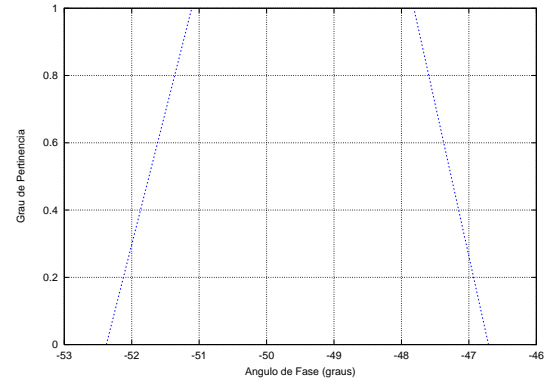


Figura 5.32: Resultado Fuzzy para o Ângulo de Fase da Tensão da Barra 232

Os Números Fuzzy Trapezoidais obtidos como solução para o perfil de tensões da rede elétrica sintetizam os possíveis valores a serem encontrados para as magnitudes de tensão e ângulos de fase das tensões complexas na operação do sistema elétrico e o seu respectivo grau de pertinência no conjunto Fuzzy.

5.2.4 Resultados Numéricos para o Problema da Variação da Demanda no Tempo

Considerando uma variação de 15% nas demandas de potências ativa e reativa em todas as barras de carga do sistema elétrico em um determinado período de tempo, a Tabela 5.8 mostra as magnitudes de tensão e os ângulos de fase nas barras selecionadas obtidos com a aplicação da metodologia proposta.

Tabela 5.8: Resultado Intervalar do Sistema SSB 352 Barras (Variação da Demanda no Tempo)

Barra	Magnitude(pu)	Ângulo(°)
5	[0, 6337; 1, 2308]	[−71, 4411; −44, 7941]
36	[0, 5626; 1, 2809]	[−65, 7929; −36, 5553]
37	[0, 5746; 1, 2966]	[−64, 6982; −35, 1883]
186	[0, 5547; 1, 3126]	[−58, 9332; −28, 9557]
232	[0, 5156; 1, 2332]	[−66, 8188; −37, 8983]

Observa-se uma grande elevação no diâmetro dos intervalos. Isto se deve a topologia e principalmente ao carregamento deste sistema. Sistemas de potência reais operam muito próximos de seus limites de carregamento e suas demandas de potência são extremamente grandes. Assim, quando uma variação de 15% em torno do ponto médio é

realizada para a geração dos intervalos iniciais de potências ativa e reativa, sua influência no perfil de tensão intervalar é extremamente grande, o que acaba por aumentar o diâmetro do intervalo-solução.

No entanto, os resultados apresentados informam que, independentemente do valor de demanda, desde que dentro do intervalo analisado, tem-se a garantia de que o perfil de tensão não excederá os limites inferior e superior das magnitudes e ângulos de fase das tensões complexas. Isto significa que, os valores contidos dentro do intervalo-solução do perfil de tensão representam todas as possíveis combinações de demandas ativa e reativa nas barras PQ do sistema.

6 CONCLUSÕES

Nesta dissertação estudou-se uma metodologia baseada na Matemática Intervalar e nos Números Fuzzy com importante aplicação ao problema do fluxo de potência em redes de energia elétrica. O objeto de estudo foi a influência da inexatidão de dados e de erros de modelagem e numéricos sobre o perfil de tensões do sistema elétrico. Com o desenvolvimento da aplicação, o engenheiro eletricitista pode conhecer a influência dos erros sobre as magnitudes de tensão e os ângulos de fase das tensões complexas nas barras da rede elétrica.

As conclusões referentes a Matemática Intervalar estão descritas a seguir:

- a obtenção dos resultados obtidos pela Matemática Intervalar permite conhecer a dimensão do erro ocasionado pelos equipamentos de medição;
- apenas com o percentual de erro informado pelo fabricante do equipamento de medição, pode-se converter as demandas ativa e reativa na forma de intervalos;
- o uso de intervalos como resposta não garante que ela contenha algo de interesse. Os algoritmos a serem desenvolvidos devem ser algoritmos intervalares e não versões intervalares dos algoritmos pontuais.

Referente aos Números Fuzzy, pode-se dizer que:

- a representação por ser de forma gráfica é intuitiva para a interpretação;
- para a geração dos resultados Fuzzy não necessitou-se de grande processamento computacional, utilizou-se apenas uma biblioteca **free** para a geração dos gráficos;
- através dos Números Fuzzy foi possível tratar o erro ocasionado pela depreciação dos equipamentos de medição, além de tratar o erro inerente de medição do equipamento.

Para obter resultados do fluxo de potência, tanto intervalares como resultados Fuzzy, utilizou-se a biblioteca C-XSC. A seguir, algumas considerações referentes a essa biblioteca são mostradas:

- no C-XSC, o programador pode desenvolver os seus sistemas usando programação orientada a objeto que é característica do C++. Pode-se combinar qualquer linha de código do C-XSC com linhas de código do C++. Os programas utilizando o C-XSC sempre fornecem os mesmos resultados numéricos em computadores com diferentes compiladores C++ (HOFSCHUSTER; KRÄMER; WIETHOFF, 2001);
- com a biblioteca C-XSC, além de tratar os erros dos equipamentos de medição e variação do fluxo de potência no tempo, foi possível também tratar eventuais erros que possam acontecer por arredondamentos e representações da mantissa do computador;
- a biblioteca intervalar C-XSC é de fácil operação e diversas soluções já estão previamente codificadas. Com o auxílio da biblioteca, através de seus operadores predefinidos, foi possível implementar os algoritmos de Newton-Raphson e de Krawczyk, utilizando alocação dinâmica de memória;
- o uso da biblioteca C-XSC é uma boa alternativa para aplicações da Matemática Intervalar para o tratamento das incertezas do fluxo de potência em redes de energia elétrica, pois permite conhecer a influência da inexatidão de dados e de erros numéricos sobre o perfil de tensões do sistema elétrico;
- em contraste com o C e C++, no C-XSC todos os operadores aritméticos predefinidos apresentam um resultado de pelo menos 1 **ulp** de precisão. Para os programadores que utilizam a linguagem de programação C, não há a necessidade de aprender as características da linguagem de programação C++ (que é o ambiente de programação do C-XSC). Na maioria dos casos, o conhecimento da linguagem C é suficiente para utilizar o C-XSC.

Uma comparação quantitativa dos algoritmos propostos implementados no software MatLab com IntLab e na linguagem de programação C++ com a biblioteca C-XSC foi realizada. Observou-se que o diâmetro dos intervalos, tanto na magnitude de tensão como no ângulo de fase da tensão, é menor no C++ do que os resultados obtidos no MatLab.

Simulações com sistemas reais como o que foi apresentado no capítulo anterior, não foram possíveis com o uso do software computacional MatLab. Por ser uma linguagem interpretado, torna-se muito lento para solução de problemas de grande dimensão. Entretanto, combinando a biblioteca C-XSC com a linguagem C++, é possível resolver um fluxo de potência em sistemas reais de grande porte.

6.1 Trabalhos Futuros

Após a apresentação dos principais resultados e das contribuições desta dissertação é necessário identificar os principais pontos de pesquisa que poderão ser trabalhados com base nos resultados que foram obtidos até o momento. Entre esses possíveis pontos de pesquisa, destacam-se:

- desenvolvimento de uma interface gráfica para o ambiente, possibilitando especificar diversas variáveis que atualmente estão definidas no código, como por exemplo, tolerâncias para o processo iterativo, número máximo de iterações, fator de carga e etc;
- aperfeiçoamento dos algoritmos, visando diminuir o tempo de resposta;
- simulação em outros sistemas reais de grande porte;
- desenvolver o algoritmo para ser executado em *clusters* de computadores.

Além disso, é importante ressaltar a continuidade das pesquisas que estão sendo realizadas no momento. Grupos de pesquisa da UPF, UFRN e da Universidade de Wuppertal continuam realizando estudos com a biblioteca C-XSC. Também grupos da UCPel, UFRN e outras instituições brasileiras e estrangeiras vêm utilizando aplicado a Lógica Fuzzy e Matemática Intervalar a problemas reais.

6.2 Publicações Obtidas

Durante o desenvolvimento da pesquisa relacionada a esta dissertação foram elaborados alguns artigos e resumos para congressos e revistas científicas.

- publicou-se um artigo completo, no ano de 2007, no XXX CNMAC (Congresso Nacional de Matemática Aplicada e Computacional), realizado em Florianópolis - SC. Este artigo apresentou o uso da Matemática Intervalar no fluxo de potência;
- submeteu-se um artigo completo para a revista TEMA (Tendências em Matemática Aplicada e Computacional), com algumas modificações no artigo apresentado no XXX CNMAC;
- dois resumos para congressos internacionais foram aceitos. Um para o 16th PSCC (*Power Systems Computation Conference*), onde foi apresentado a abordagem do fluxo de potência em um determinado período de tempo. E outro resumo foi aceito para o 10th PMAPS (*Probabilistic Methods Applied to Power Systems*), onde é abordado o fluxo de potência usando a Matemática Intervalar e Números Fuzzy.

REFERÊNCIAS

- ACIÓLY, B. **Fundamentação, Computacional da Matemática Intervalar**. 1991. Tese (Doutorado em Ciência da Computação) — Universidade Federal do Rio Grande do Sul, Porto Alegre, Brasil.
- ALTROCK, C. **Fuzzy Logic and NeuroFuzzy Applications in Business and Finance**. Upper Saddle River, NJ, USA: Prentice-Hall, Inc., 1997.
- ANILE, A.; DEODATO, S.; PRIVITERA, G. Implementing Fuzzy Arithmetic. **Fuzzy Sets Syst.**, Amsterdam, The Netherlands, The Netherlands, v.72, n.2, p.239–250, 1995.
- BARBOZA, L. **Sistemas de Potência I**. Notas de aula do curso de Engenharia Elétrica - Universidade Católica de Pelotas 2001.
- BARBOZA, L.; DIMURO, G.; REISER, R. Towards Interval Analysis of the Load Uncertainty in Power Electric Systems. **8th International Conference on Probabilistic Methods Applied to Power System**, Ames, Iowa, 2004.
- BARBOZA, L.; DIMURO, G.; REISER, R. Interval Mathematics Applied to the Load Flow Analysis. **17th IMACS Word Congress Scientific Computation, Applied Mathematics and Simulations**, Paris, France, 2005.
- BARÁN, B.; KASZCUREWICZ, E.; FALCÃO, D. In: 1995, Londres, Gran Bretaña. **Anais...** Transmission and Distribution, 1995. p.583–588.
- BEDREGAL, B. **Sistemas de Informação Contínuos**: Uma Abordagem Lógica e Computacional para a Matemática Intervalar. 1996. Tese (Doutorado em Ciência da Computação) — Universidade Federal de Pernambuco, Pernambuco, Brasil.
- BURKILL, J. Functions of Intervals. **Proc. London Math. Soc.**, [S.l.], v.22, p.275–336, 1924.
- CISNEROS, L. **Sistemas de Potencia**. Notas de Clase del curso de Ingenieria Electrica - Instituto Tecnológico de Morelia 2006.

COX, E. The Fuzzy Systems Handbook: A Practitioner's Guide to Building, Using, and Maintaining Fuzzy Systems. **Academic Press Professional Inc.**, [S.l.], 1994.

COX, E. Fuzzy Logic for Business and Industry. **Charles River Media Inc.**, [S.l.], 1995.

CRUZ, M. **Equivalência e Consistência entre Funções Intervalares**. 2000. Dissertação (Mestrado em Ciência da Computação) — Universidade Federal do Rio Grande do Norte, Natal, Brasil.

Digital Computer Solution of Power-Flow Problems. **AIEE Transaction**, [S.l.], v.5, p.298–404, 1956.

DIMURO, G. **Domínios Intervalares da Matemática Computacional**. 1991. Dissertação (Mestrado em Ciência da Computação) — Universidade Federal do Rio Grande do Sul, Porto Alegre, Brasil.

DUSTAN, L. Machine Computations of Power Network Performance. **AIEE Transaction**, [S.l.], 1947.

DUTRA, J. **JAVA-XSC**: Uma Biblioteca Java para Computações Intervalares. 2000. Dissertação (Mestrado em Ciência da Computação) — Universidade Federal do Rio Grande do Norte, Natal, Brasil.

DWYER, P. Computation with Approximate Numbers. In: LINEAR COMPUTATIONS, 1951, New York. **Anais...** Wiley & Sons Inc, 1951. p.11–35.

FANG, J.; CHEN, H. Uncertainties are Better Handled by Fuzzy Arithmetic. **AAPG Bulletin**, [S.l.], v.74, n.8, p.1228–1233, 1990.

GIACHETTI, R.; YOUNG, R. A Parametric Representation of Fuzzy Numbers and Their Arithmetic Operators. **Fuzzy Sets Syst.**, Amsterdam, The Netherlands, The Netherlands, v.91, n.2, p.185–202, 1997.

GIARRATANO, J.; RILEY, G. **Expert Systems**: Principles and Programming. 2nd.ed. Boston: PWS Publishing Company, 1994.

GOTTGTROY, M. **Aplicação de Técnicas de Engenharia do Conhecimento na Análise do Risco em Sistemas Estruturais**. 1996. Tese (Doutorado em Ciência da Computação) — Universidade Federal do Rio de Janeiro, Rio de Janeiro, Brasil.

GRIMMER, M.; PETRAS, K.; REVOL, N. Multiple Precision Interval Packages: Comparing Different Approaches. **Lecture Notes in Computer Science**, [S.l.], v.2991, p.64–90, 2004.

HAMMER, R.; HOCKS, M.; RATZ, D. C++ Toolbox for Verified Computing I - Basic Numerical Problems. **Springer-Verlag**, Berlin, Heidelberg, 1995.

HöLBIG, C. **Ambiente de Alto Desempenho com Alta Exatidão para a Resolução de Problemas**. 2005. Tese (Doutorado em Ciência da Computação) — Universidade Federal do Rio Grande do Sul, Porto Alegre, Brasil.

HOFSCHUSTER, W.; KRÄMER, W.; WIETHOFF, A. C-XSC 2.0 A C++ Library for Extended Scientific Computing. **Lecture Notes in Computer Science**, Universitat Wuppertal, p.15–35, 2001.

KASABOV, N. Foundations of Neural Networks, Fuzzy Systems, and Knowledge Engineering. **Ed. The MIT Press**, Massachusetts Institute of Technology, 1996.

KLIR, G.; YUAN, B. **Fuzzy Sets and Fuzzy Logic; Theory and Applications**. Upper Saddle River, N. Y.: Prentice Hall, 1995.

KNÜPPEL, O. **BIAS – Basic Interval Arithmetic Subroutines**. [S.l.]: Inst. f. Informatik III, TU Hamburg–Harburg, 1993. Technical Report. (93.3).

KOLBERG, M. **Redefinição de Expressões Exatas em C-XSC**. 2002. Dissertação (Mestrado em Ciência da Computação) — Pontifícia Universidade Católica do Rio Grande do Sul, Porto Alegre, Brasil.

KOSKO, B. **Neural Networks and Fuzzy Systems. A Dynamical Systems Approach to Machine Intelligence**. Englewood Cliffs: Prentice–Hall, 1992.

KRAWCZYK, R. Newton-Algorithmen zur Bestimmung von Nullstellen mit Fehlerchranken. (German) [Newton algorithms for evaluation of roots with error bounds]. **j-COMPUTING**, [S.l.], v.4, n.3, p.187–201, 1969.

KRÄMER, W. Advanced Software Tools for Validated Computing. In: WILEY & SONS INC, 2002, Borovets. **Anais...** [S.l.: s.n.], 2002. p.344–355. (Applied optimization).

KULISCH, U. **Advanced Arithmetic for the Digital Computer: Design of Arithmetic Units**. Verlag: Softcover, 2002. xii - 141p.

LYRA, A. **Computabilidade no Espaço dos Intervalos Reais: Um Modelo BSS Inter-
valar**. 1999. Dissertação (Mestrado em Ciência da Computação) — Universidade Federal do Rio Grande do Norte, Natal, Brasil.

MARCHI, M.; CARVALHO, S.; MORAIS, P. Um Modelo Estocástico para a Manutenção de um Equipamento Baseado na Inspeção das Peças Produzidas. **Gestão & Produção**, São Carlos, Brasil, v.8, p.115–127, 2001.

MELEK, A. **Avaliação Econômica em Sistemas Elétricos de Potência**. <http://www.aneel.gov.br/aplicacoes/audiencia/arquivo/2006/012/contribuicao/> - Acesso em 20 setembro de 2007.

MONTICELLI, A. **Fluxo de Carga em Redes de Energia Elétrica**. São Paulo, Brasil: Edgard Blucher LTDA, 1983.

MOORE, R. **Automatic Error Analysis in Digital Computation**. Sunnyvale, CA, USA: Lockheed Missiles and Space Co., 1959. Technical Report. (Space Div. Report LMSD84821).

MOORE, R. **Interval Analysis**. Philadelphia, PA, USA: pub-PH, 1966. p.xi-145.

MOORE, R. **Methods and Applications of Interval Analysis**. Philadelphia, PA, USA: Society for Industrial and Applied Mathematics, 1979. xi-190p.

OLIVEIRA, R.; DIVERIO, T.; CLAUDIO, D. **Fundamentos da Matemática Intervalar**. Instituto de Informática da UFRGS, Porto Alegre, Brasil: Editora Sagra Luzzato, 2001. xi-90p.

PEDRYCZ, W.; GOMIDE, F. **An Introduction to Fuzzy Sets**. MIT Press, Cambridge, England, 1998.

PEREIRA, S. **Tratamento de Incertezas em Modelagem de Bacias**. 2002. Tese (Doutorado em Ciência da Computação) — Universidade Federal do Rio de Janeiro, Rio de Janeiro, Brasil.

PUEYO, R. **Procesado y Optimización de Espectros Raman mediante Técnicas de Lógica Difusa**: Aplicación a la identificación de Materiales Pictóricos. 2005. Tese (Doutorado em Ciência da Computação) — Universitat Politècnica de Catalunya, Barcelona, Espanha.

RUMP, S. **Fast and Parallel Interval Arithmetic**. [S.l.]: BIT, 1999. 539-560p.

RUSS, E.; SIMPSON, P.; DOBBINS, R. **Computational Intelligence PC Tools**. San Diego, CA, USA: Academic Press Professional, Inc., 1996.

SANTIAGO, R. **Teoria das Equações Intervalares Locais**. 1999. Tese (Doutorado em Ciência da Computação) — Universidade Federal de Pernambuco, Recife, Brasil.

SCHULTE, M.; SWARTZLANDER, E. **Software and Hardware Techniques for Accurate, Self-Validating Arithmetic**. In: APPLICATIONS OF INTERVAL COMPUTATIONS:

PAPERS PRESENTED AT AN INTERNATIONAL WORKSHOP IN EL PASO, TEXAS, FEBRUARY 23–25, 1995, 1996, Norwell, MA, USA, and Dordrecht, The Netherlands. **Anais...** Kluwer Academic Publishers Group, 1996. p.381–404. (Applied optimization, v.3).

SHAW, I.; SIMÕES, M. **Controle e Modelagem Fuzzy**. São Paulo, Brasil: Edgard Blucher, 1999.

SILVEIRA, M. **Teoria Fuzzy Intervalar**: Uma Proposta de Integração da Matemática Intervalar à Teoria Fuzzy. 2002. Dissertação (Mestrado em Ciência da Computação) — Universidade Federal do Rio Grande do Norte, Natal, Brasil.

SUNAGA, T. Theory of an Interval Algebra and its Application to Numerical Analysis. **RAAG Memoirs**, [S.l.], v.2, p.29–46, 547–564, 1958.

TERANO, T.; ASAI, K.; SUGENO, M. **Fuzzy Systems Theory and its Applications**. San Diego: Academic Press, 1992.

TSOUKALAS, L.; UHRIG, R. **Fuzzy and neural approaches in engineering**. pub-WILEY:adr: Wiley, 1997. xix + 587p.

WARMUS, M. Calculus of Approximations. **Bull. Acad. Polon. Sci., Cl. III**, [S.l.], v.IV, n.5, p.253–259, 1956.

WIENER, N. A Contribution to the Theory of Relative Position. **Cambr. Phil. Soc. Proc.**, [S.l.], v.17, p.441–449, 1914.

WIENER, N. **A New Theory of Measurement: A Study in the Logic of Mathematics**. [S.l.]: Lond. M. S. Proc. (2) 19, 181–205, 1920.

YOUNG, R. The Algebra of Many-Valued Quantities. **Math. Ann.**, [S.l.], v.104, p.260–290, 1931.

ZADEH, L. Fuzzy Sets. **Inf. Control**, New York-San Francisco-London-San Diego, v.8, p.338–353, 1965.

ZEMKE, J. **b4m - BIAS for Matlab**. [S.l.]: Inst. f. Informatik III, Technische Universität Hamburg-Harburg, 1998. Technical Report.

ZIMMERMANN, H. **Fuzzy Set Theory and its Applications (3rd ed.)**. Norwell, MA, USA: Kluwer Academic Publishers, 1996.

ANEXO A TUTORIAL DE INSTALAÇÃO

Devido à inexistência de um tutorial para a instalação da biblioteca C-XSC e de um ambiente para programação utilizando essa biblioteca, a seguir, serão explicados os passos para esta instalação.

A.1 Configuração do equipamento de teste

- **sistema operacional:** Ubuntu 6.06 LTS (Dapper Drake)
- **processador:** Celeron (R) 2,8 GHz
- **memória RAM:** 1 GB
- **disco rígido:** 80 GB

A.2 Instalação da IDE Anjuta

A versão instalada do Anjuta IDE foi a 1.2.4a. Para esta instalação, siga os seguintes passos:

Abra um terminal e digite:

```
sudo apt-get install anjuta
```

Responda **S** para as questões solicitadas.

A execução é dada pelo seguinte comando:

```
anjuta
```

O Anjuta IDE também pode ser executado a partir do Menu: / *Aplicações* / *Programação* / *Anjuta IDE*.

Ao inicializar o Anjuta IDE, irá carregar uma janela conforme a Figura A.1.

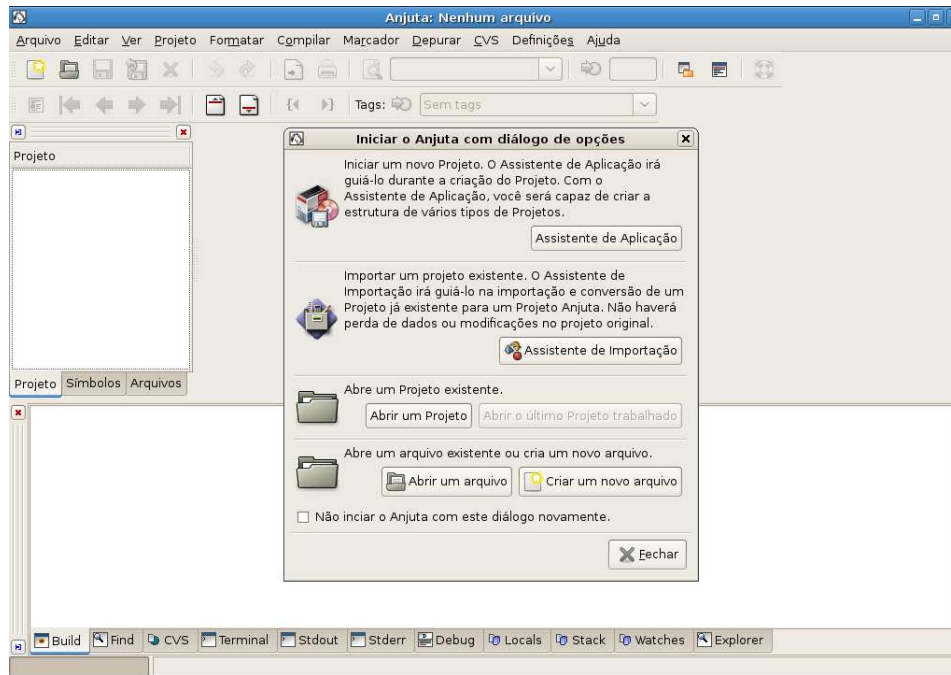


Figura A.1: Tela Inicial de execução do Anjuta

Clique no botão *Assistente de Aplicação* Essa aplicação abrirá uma janela conforme a Figura A.2.

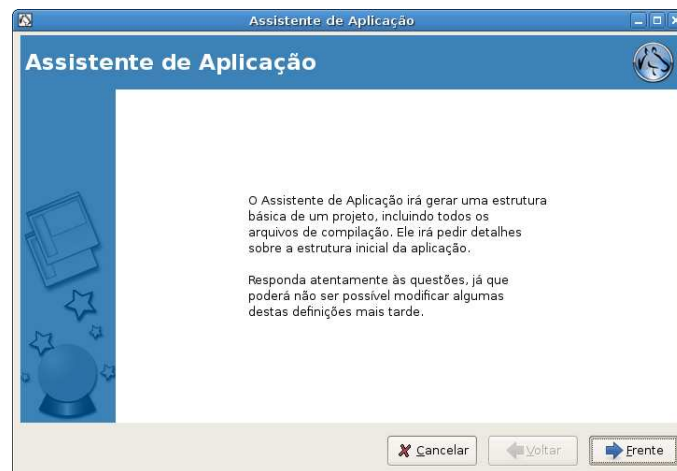


Figura A.2: Assistente de Aplicação

Clique em *Frente* e carregará a Figura A.3.



Figura A.3: Tipo de Projeto

Selecione *Projeto Genérico/Console* e clique no botão *Frente*. A seguir digite o nome do projeto e selecione *C++* e clique no botão *Frente*, conforme Figura A.4.

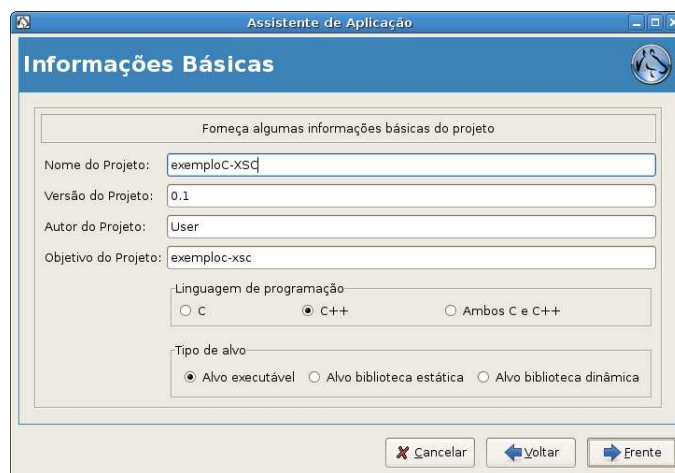


Figura A.4: Informações Básicas

Para as próximas duas telas, clique no botão *Frente* e por fim no botão *Aplicar*.

A Figura A.5 mostra o ambiente de trabalho do Anjuta IDE. Note a seguir que ocorreram alguns erros e que a compilação não obteve sucesso.

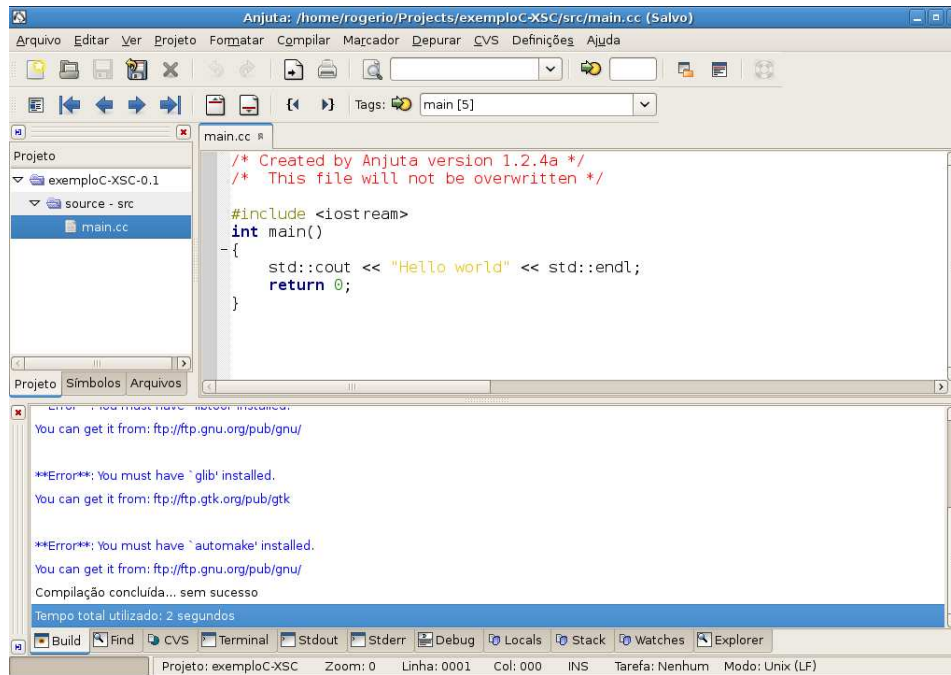


Figura A.5: Erros na compilação

Para solucionar os erros de compilação, abra o terminal e digite os seguintes comandos:

```
sudo apt-get install autoconf
sudo apt-get install automake
sudo apt-get install libtool
sudo apt-get install gettext
sudo apt-get install g++
```

Responda **S** para as questões.

É necessário a instalação da biblioteca *glib*. Para fazer isso, abra o navegador e digite:

<ftp://ftp.gtk.org/pub/glib>

selecione a pasta que contenha a versão mais atual, neste caso é a *2.14* e após selecione o nome do arquivo contendo o nome *glib*.

Agora descompacte o arquivo. Neste exemplo, foi feito no *Desktop* (*botão direito do mouse* / *Extrair Aqui*). Abra um terminal e digite:

```
cd Desktop/glib-2.14.1
sudo ./configure
sudo make
sudo make instal
```

A instalação do ambiente de desenvolvimento Anjuta IDE está concluído. Agora, será realizado a instalação da biblioteca C-XSC.

A.3 Instalação da Biblioteca C-XSC

A versão utilizada para este tutorial é a 2.0. A instalação da Biblioteca deve seguir os passos abaixo:

Faça o download da biblioteca em:

<http://www.math.uni-wuppertal.de/xsc/xsc/cxsc.html>

A instalação foi realizada usando a versão 2.0 obtida em

<http://www.math.uni-wuppertal.de/xsc/xsc/cxsc/cxsc-2-0.tar.gz>

Descompacte o arquivo, abra um terminal e vá para a pasta da biblioteca já descompactada. Digite:

sudo ./install

Para as seguintes questões responda:

```
Do you accept the terms of either license? [no] (digite: yes)
Which C++ compiler? [gnu] (pressione ENTER)
Would you like to select a special GNU C/C++ Compiler-Version? [no]
(pressione ENTER)
Which method? [hard] (pressione ENTER)
safe or quick? [safe] (pressione ENTER)
```

A mensagem a seguir é muito importante, é onde se instalará a biblioteca. Para essa instalação aceitou-se o endereço sugerido pela instalação. Para tal, pressione *ENTER*.

```

Section: Prefix ===== (un-)installation prefix
e.g. /usr/local/cxsc or local home directory
Prefix [/home/rogerio/cxsc] (pressione ENTER)

Do you want to create a dynamic or static library? [dynamic]/static/both
(pressione ENTER)

Proceed with automatic Code generation? [yes]/no (pressione ENTER)

```

O processo de instalação é bastante demorado. Caso ocorra algum problema, remova a instalação e instale novamente. Se a instalação foi realizada com sucesso irá aparecer uma mensagem similar a esta:

```

*****

* All results are equal to the expected solutions. *
* It seems the libraries (C-XSC and CToolbox) are OK! *
*****

Section: CXSC Results
=====
Library successfully created
Examples successfully created
Library successfully installed

Section: CToolbox Results
=====
Examples successfully created
Library and Programs successfully installed and tested

```

Ainda é necessário configurar o Anjuta IDE para funcionar com a biblioteca C-XSC. Para isso, siga os passos da próxima seção.

A.4 Configuração do Anjuta para o C-XSC

Os próximos passos mostram a configuração do Anjuta junto à biblioteca C-XSC.

- Execute o anjuta

- Crie uma nova aplicação
- No menu *Definições / Definições de Compilação e Link*, na aba Caminhos dos Includes, digite o endereço de onde foi instalada a biblioteca juntamente com endereço do include, conforme a Figura A.6.

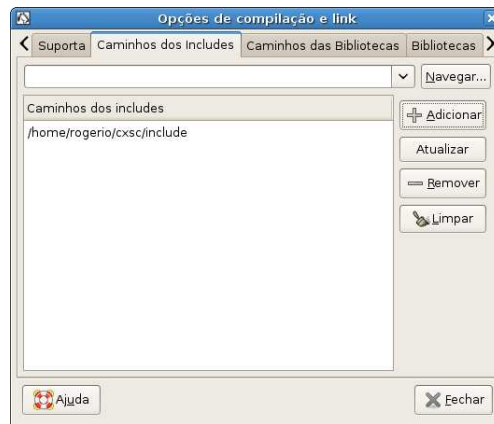


Figura A.6: Configurando o Anjuta (include)

Faça o mesmo para a aba Caminhos das Bibliotecas, adicionando o caminho da biblioteca (*lib*), Figura A.7.



Figura A.7: Configurando o Anjuta (lib)

E, por fim, na aba Bibliotecas, digite cxsc e clique em adicionar, veja a Figura A.8.



Figura A.8: Configurando o Anjuta (bibliotecas)

Para testar o funcionamento, pode-se digitar o código apresentado na Figura A.9.

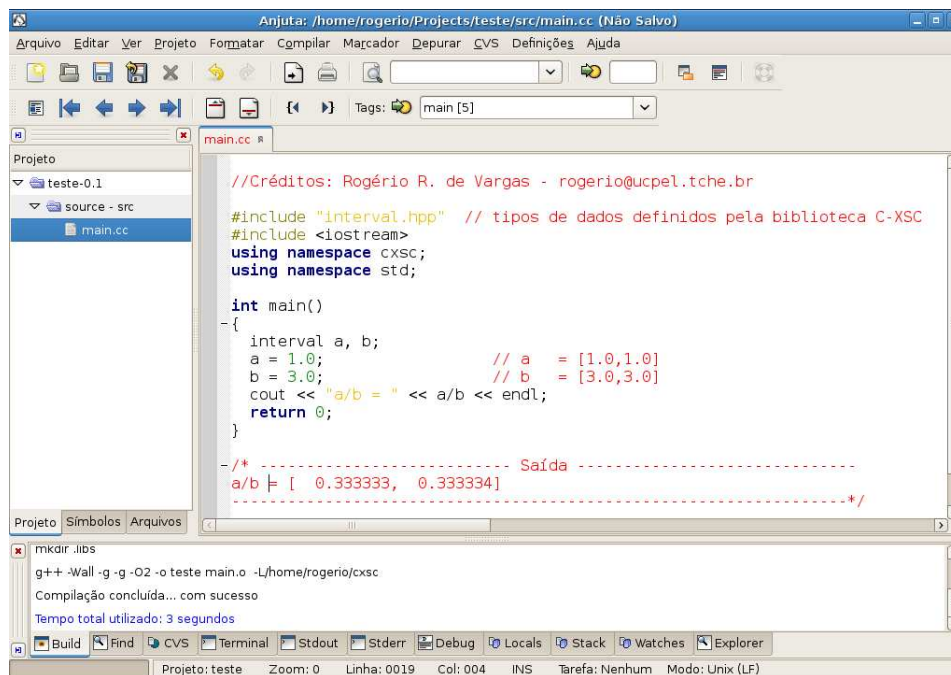


Figura A.9: Codificação no Anjuta usando o C-XSC

Para compilar o programa, vá no menu *Compilar* / *Compilar* (F9). Novamente vá no menu *Compilar* (F11) e por fim execute o programa no menu *Compilar* / *Executar* (F3).

Livros Grátis

(<http://www.livrosgratis.com.br>)

Milhares de Livros para Download:

[Baixar livros de Administração](#)

[Baixar livros de Agronomia](#)

[Baixar livros de Arquitetura](#)

[Baixar livros de Artes](#)

[Baixar livros de Astronomia](#)

[Baixar livros de Biologia Geral](#)

[Baixar livros de Ciência da Computação](#)

[Baixar livros de Ciência da Informação](#)

[Baixar livros de Ciência Política](#)

[Baixar livros de Ciências da Saúde](#)

[Baixar livros de Comunicação](#)

[Baixar livros do Conselho Nacional de Educação - CNE](#)

[Baixar livros de Defesa civil](#)

[Baixar livros de Direito](#)

[Baixar livros de Direitos humanos](#)

[Baixar livros de Economia](#)

[Baixar livros de Economia Doméstica](#)

[Baixar livros de Educação](#)

[Baixar livros de Educação - Trânsito](#)

[Baixar livros de Educação Física](#)

[Baixar livros de Engenharia Aeroespacial](#)

[Baixar livros de Farmácia](#)

[Baixar livros de Filosofia](#)

[Baixar livros de Física](#)

[Baixar livros de Geociências](#)

[Baixar livros de Geografia](#)

[Baixar livros de História](#)

[Baixar livros de Línguas](#)

[Baixar livros de Literatura](#)
[Baixar livros de Literatura de Cordel](#)
[Baixar livros de Literatura Infantil](#)
[Baixar livros de Matemática](#)
[Baixar livros de Medicina](#)
[Baixar livros de Medicina Veterinária](#)
[Baixar livros de Meio Ambiente](#)
[Baixar livros de Meteorologia](#)
[Baixar Monografias e TCC](#)
[Baixar livros Multidisciplinar](#)
[Baixar livros de Música](#)
[Baixar livros de Psicologia](#)
[Baixar livros de Química](#)
[Baixar livros de Saúde Coletiva](#)
[Baixar livros de Serviço Social](#)
[Baixar livros de Sociologia](#)
[Baixar livros de Teologia](#)
[Baixar livros de Trabalho](#)
[Baixar livros de Turismo](#)