

Incerteza de Dados em Fluxo de Potência: uma Abordagem com a Matemática Intervalar do C-XSC

R.R. DE VARGAS¹, C.M. DE FARIAS², L.V. BARBOZA³, G.P. DIMURO⁴,
Programa de Pós-Graduação em Informática, Universidade Católica de Pelotas,
96010-000 Pelotas, RS, Brazil.

Resumo. O artigo apresenta um estudo da Matemática Intervalar aplicada ao problema das incertezas no fluxo de potência em redes de energia elétrica com a utilização da biblioteca C-XSC. O problema se baseia na análise das influências de erros de medição e erros gerados na computação numérica. A Matemática Intervalar é uma técnica eficaz no controle dos diversos erros computacionais presentes em cálculos científicos. Com o auxílio da biblioteca C-XSC foi desenvolvido um software capaz de tratar o problema do fluxo de potência de forma intervalar. Na validação da metodologia proposta, são apresentados resultados para um sistema teste do IEEE e um sistema real representando a região Sul e Sudeste do Brasil. Realizou-se também um comparativo de resultados obtidos com IntLab, para o ambiente MatLab.

Palavras-chave C-XSC, Fluxo de Carga, Matemática Intervalar.

1. Introdução

Para a resolução de problemas físicos em ambientes computacionais, diversas fontes de erro podem contribuir para a obtenção de um resultado não muito exato. Erros de truncamento e arredondamento são muito comuns em operações de ponto flutuante e precisam ser controlados com o intuito de obter a máxima exatidão do ponto de vista computacional. Além disso, a inexatidão na coleta dos dados possui um papel relevante, pois, em sua grande maioria, os dados são adquiridos por meio de medições ou estimativas.

A Matemática Intervalar introduzida por [8] é altamente eficaz no controle dos diversos tipos de erros de operações computacionais, pois controla de maneira automática todos os arredondamentos gerados pelo sistema de ponto flutuante e propicia que o resultado final seja de máxima exatidão do ponto de vista computacional. Atualmente, as linguagens XSC vêm sendo amplamente aplicadas na elaboração de algoritmos auto-validados e com controle interno dos erros computacionais.

¹rogerio@ppgsc.ufrn.br; Universidade Federal do Rio G. do Norte/PPgSC, bolsista CAPES.

²charles@ucpel.tche.br; Universidade Federal de Santa Catarina/PPGEE, bolsista CNPq.

³luciano@ucpel.tche.br; Centro Federal de Educação Tecnológica de Pelotas/RS.

⁴liz@ucpel.tche.br; Universidade Católica de Pelotas/PPGInf.

As linguagens XSC são extensões de programação que trazem consigo os conceitos da Matemática Intervalar e possuem características desejáveis quando aplicadas em métodos numéricos, nos quais a exatidão da resposta é extremamente importante. Arredondamentos direcionados, máxima exatidão nos tipos de dados, alocação dinâmica de memória facilitada e diversos tipos de dados inexistentes na maioria das linguagens de programação tradicionais são algumas das inovações propostas por essas linguagens. Atualmente, existem extensões das linguagens XSC para diversos ambientes de programação.

Neste trabalho, foi utilizada a biblioteca C-XSC que propicia aos programadores em C produzirem métodos numéricos aplicando os conceitos da Matemática Intervalar. Com isso, o problema do fluxo de potência em linhas de transmissão pode ser resolvido de forma intervalar, utilizando uma metodologia alternativa para a resolução das equações não-lineares resultantes da modelagem matemática de sistemas de potência, denominado método de Krawczyk, o qual foi previamente aplicado e discutido em [1]. A utilização da Matemática Intervalar possibilita que sejam considerados no resultado final os diversos erros na obtenção dos dados, oriundos, por exemplo, dos erros relativos de cada instrumento de medição. Além disso, é possível também fazer análises em sistemas de potências tendo como parâmetros de entrada diversas estimativas de potência, na sua grande maioria imprecisas.

2. Matemática Intervalar

A Matemática Intervalar considera um conjunto de métodos para manipulação de intervalos numéricos que aproximam dados incertos. 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 [9]. Estes métodos baseiam-se na definição da Aritmética Intervalar e do produto escalar ótimo [6]. Uma revisão sobre a Matemática Intervalar e Funções Intervalares pode ser encontrada em [7].

2.1. Operações básicas

Sejam, $X, Y \in \mathbb{IR}$ dois intervalos reais, com $X = [x_1; x_2]$ e $Y = [y_1; y_2]$. As operações intervalares de adição, subtração, multiplicação e divisão são definidas na Tabela 1.

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 [3].

3. Fluxo de Potência

O cálculo do fluxo de potência em uma rede de energia elétrica consiste essencialmente na determinação do estado desta rede (tensões complexas em todas as barras) e da distribuição dos fluxos de potências ativa e reativa nos circuitos. A modelagem

Tabela 1: Principais operações aritméticas com intervalos

Descrição	Operações
Adição	$X + Y = [(x_1 + y_1); (x_2 + y_2)]$
Subtração	$X - Y = [(x_1 - y_2); (x_2 - y_1)]$
Multiplicação	$X \times Y = [\min\{x_1 \times y_1, x_1 \times y_2, x_2 \times y_1, x_2 \times y_2\}; \max\{x_1 \times y_1, x_1 \times y_2, x_2 \times y_1, x_2 \times y_2\}]$
Divisão	$\frac{X}{Y} = \left[\min \left\{ \frac{x_1}{y_1}, \frac{x_1}{y_2}, \frac{x_2}{y_1}, \frac{x_2}{y_2} \right\}; \max \left\{ \frac{x_1}{y_1}, \frac{x_1}{y_2}, \frac{x_2}{y_1}, \frac{x_2}{y_2} \right\} \right]$ com $0 \notin [y_1; y_2]$

do sistema é estática, significando que a rede é representada por um conjunto de equações algébricas. Esse tipo de representação é usado em situações nas quais as variações com o tempo são suficientemente lentas para que se possam ignorar os efeitos transitórios. O cálculo do fluxo de carga 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, por exemplo, geradores, cargas, reatores e capacitores;
- os que estão ligados entre duas barras quaisquer da rede (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 carga (balanços de potências) são obtidas impondo-se a conservação das potências ativa e reativa em cada barra da rede, ou seja, 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.

As equações estáticas do fluxo de potência podem ser expressas, de forma genérica, pela equação 3.1.

$$f(x) - S = 0, \tag{3.1}$$

onde $f(x)$ (equação 3.2), em coordenadas polares, são as expressões matemáticas do fluxo de potência e S é o vetor com as injeções de potências especificadas (equação 3.3).

$$f(x) = \begin{cases} V_i \sum_{k \in \Omega_i} V_k (G_{ik} \cos \delta_{ik} + B_{ik} \sin \delta_{ik}) \\ V_i \sum_{k \in \Omega_i} V_k (G_{ik} \sin \delta_{ik} - B_{ik} \cos \delta_{ik}) \end{cases} \tag{3.2}$$

V_i e V_k representam as magnitudes de tensão nas barras i e k , respectivamente; Ω_i refere-se ao conjunto de barras que estão conectadas à barra i . G_{ik} e B_{ik} são elementos da matriz admitância de barras; $\delta_{ik} = \delta_i - \delta_k$ é a abertura angular da

linha $i - k$, com δ_i e δ_k sendo os ângulos de fase das tensões complexas nas barras i e k , respectivamente.

$$S = \begin{cases} P_{G_i} - P_{d_i} \\ Q_{G_k} - Q_{d_k} \end{cases} \quad (3.3)$$

com $i = 1, \dots, nb$ e $i \neq fol$ e $k = 1, \dots, npq$, onde nb é o número de barras do sistema elétrico, npq é o número de barras PQ , fol correspondente à barra de folga e P_{G_i} e P_{d_i} e Q_{G_k} e Q_{d_k} correspondem, respectivamente, à geração e à demanda de potências ativa e reativa.

As equações estáticas para a resolução do problema do fluxo de carga, neste estudo, são modeladas em coordenadas retangulares, pois, neste formato, evita-se a utilização das funções *seno* e *cos seno*. Neste sistema de coordenadas, o vetor $f(x)$, equação 3.2, é escrito como mostrado nas equações 3.4 e 3.5.

$$\begin{aligned} & \sum_{k=1}^n [e_i(e_k G_{ik} - f_k B_{ik}) + f_i(f_k G_{ik} - e_k B_{ik})] \\ &= (e_i^2 + f_i^2) G_{ii} + \sum_{\substack{k=1 \\ k \neq i}}^n [e_i(e_k G_{ik} - f_k B_{ik}) + \\ & \quad f_i(f_k G_{ik} + e_k B_{ik})] \end{aligned} \quad (3.4)$$

$$\begin{aligned} & \sum_{k=1}^n [f_i(e_k G_{ik} - f_k B_{ik} - e_i(f_k G_{ik} - e_k B_{ik}))] \\ &= -(e_i^2 + f_i^2) B_{ii} + \sum_{\substack{k=1 \\ k \neq i}}^n [f_i(e_k G_{ik} - f_k B_{ik}) - \\ & \quad e_i(f_k G_{ik} + e_k B_{ik})] \end{aligned} \quad (3.5)$$

Os resíduos de potência ativa, equação 3.6, para as barras PV e PQ são

$$\begin{aligned} \Delta P_i &= (e_i^2 + f_i^2) G_{ii} - \sum_{\substack{k=1 \\ k \neq i}}^n [e_i(e_k G_{ik} - f_k B_{ik}) + \\ & \quad f_i(f_k G_{ik} - e_k B_{ik})] - P_{G_i} + P_{d_i} \end{aligned} \quad (3.6)$$

e os resíduos de potência reativa, equação 3.7, para as barras PQ são

$$\begin{aligned} \Delta Q_i &= -(e_i^2 + f_i^2) B_{ii} - \sum_{\substack{k=1 \\ k \neq i}}^n [f_i(e_k G_{ik} - f_k B_{ik}) \\ & \quad - e_i(f_k G_{ik} + e_k B_{ik})] - Q_{G_i} + Q_{d_i} \end{aligned} \quad (3.7)$$

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

$$e_i^2 + f_i^2 = (V_i^{esp})^2,$$

onde V_i^{esp} é a magnitude de tensão especificada na barra i e, portanto, os resíduos dos quadrados das magnitudes das tensões nas barras PV são

$$\Delta|V_i^{esp}|^2 = e_i^2 + f_i^2 - |V_i^{esp}|^2. \tag{3.8}$$

O conjunto de equações algébricas não-lineares, equações 3.6, 3.7 e 3.8, formam um sistema que representa o comportamento estático do fluxo de potência em redes de energia elétrica.

4. Fluxo de Potência Intervalar

Para a resolução do sistema de equações $f(x) = 0$ que representa o fluxo de potência intervalar foi utilizado o método de Krawczyk, onde o sistema não-linear é resolvido somente por produto de matrizes.

A vantagem deste operador é que nenhum sistema de equações lineares necessita ser resolvido para determinar a atualização da próxima iteração [1]. O operador de Krawczyk é definido por: $K : \mathbb{R}^n \times \mathbb{IR}^n \rightarrow \mathbb{IR}^n$ [5].

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

onde C é uma matriz de pré-condicionamento, a inversa do ponto médio da matriz Jacobiana $J(X)$, I é a matriz identidade e $\tilde{x} \in X$.

Um estudo sobre o operador de Krawczyk pode ser encontrado em [13].

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

$$X^{k+1} = X^k \cap K(\tilde{x}^k, X^k) \text{ com } k = 0, 1, \dots \tag{4.1}$$

É realizada uma análise no sistema de equações não-lineares intervalares buscando verificar qual a maior distância entre o ínfimo e o supremo de cada intervalo. Este valor é comparado com um valor de tolerância pré-especificado pelo usuário, permitindo determinar a convergência ou não do método.

A Figura 1 mostra o diagrama do processo computacional, partindo dos cálculos pontuais e finalizando no processo intervalar, com a obtenção das magnitudes de tensão e ângulos de fase intervalares.

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 de equações não-lineares em análise não possui solução e, portanto, é impossível obter um ponto de operação factível para a rede elétrica.

As etapas desenvolvidas 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;

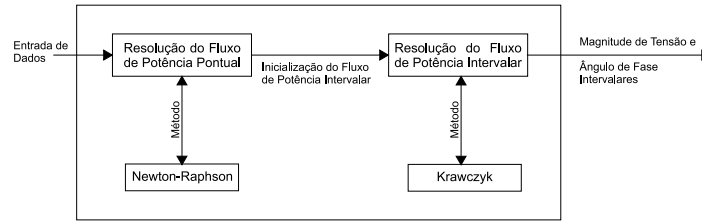


Figura 1: Diagrama de Resolução do Problema

(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 é realizada com base no perfil de tensões pontual determinado anteriormente. Estes valores são utilizados como pontos médios para as magnitudes de tensão e ângulos de fase intervalares. As etapas do processo 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. Havendo convergência do processo de solução intervalar, o perfil de tensões intervalares do sistema é determinado.

5. Resultados

A implementação do algoritmo foi realizada no ambiente C++ (compilador g++ 4.0), para o sistema operacional Linux (Ubuntu 6.06) e utilizando a biblioteca C-XSC (versão 2.0). Para testar o desempenho e a confiabilidade da metodologia proposta, foram analisados os resultados para diversos sistemas elétricos com diferente número de barras. Neste trabalho, serão apresentados e discutidos resultados para dois sistemas teste: um do IEEE contendo 6 barras e outro equivalente da região Sul-Sudeste do Brasil com 352 barras. Em trabalhos anteriores, as análises foram realizadas no ambiente Matlab usando o toolbox IntLab [1, 2]. A documentação para a biblioteca IntLab pode ser encontrada em [10].

A tabela 2 apresenta as principais características dos sistemas simulados.

Tabela 2: 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 que existe em cada sistema.

5.1. Sistema IEEE-6 barras

Este sistema teste está mostrado na Figura 2. Ele é composto por 6 barras e 7 circuitos, sendo que, destes, 5 são linhas de transmissão (circuitos 1–4, 1–6, 2–3, 2–5 e 4–6) e dois transformadores (circuitos 3–4 e 5–6). Além disso, observa-se duas barras de geração (barras 1 e 2) e três barra de geração (barras 3, 5 e 6). Há, ainda, a existência de uma barra de injeção nula (barra 3), a qual não possui nem geração nem demanda.

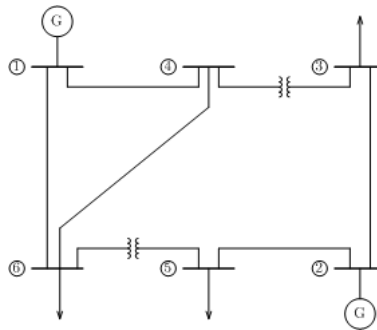


Figura 2: Diagrama Unifilar da Rede Elétrica

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.

A tabela 3 refere-se aos resultados obtidos para o fluxo de potência pontual. Estes valores foram utilizados para a inicialização do perfil de tensões intervalares.

Tabela 3: Sistema de 6 barras pontual

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

As tabelas 4 e 5 apresentam, respectivamente, os resultados obtidos pelo MatLab e pelo C++ . Esses resultados correspondem às magnitudes e ângulos de fase das tensões complexas em todas as barras da rede elétrica. A Tabela 4 foi obtida com a implementação da metodologia proposta no ambiente de programação do Matlab, utilizando a biblioteca intervalar IntLab. Por outro lado, os resultados da Tabela 5 foram obtidos a partir da implementação da abordagem em linguagem C++ com o uso da biblioteca intervalar C-XSC.

Tabela 4: MatLab: Sistema de 6 barras

Barra	Magnitude(pu)	Ângulo(°)
1	[1, 0999; 1, 1001]	[0, 0000; 0, 0000]
2	[1, 0988; 1, 1012]	[-3, 1954; -3, 1826]
3	[0, 8930; 0, 9174]	[-12, 0675; -11, 3934]
4	[0, 9983; 1, 0195]	[-9, 0777; -8, 5992]
5	[0, 9226; 0, 9542]	[-11, 8858; -11, 1450]
6	[0, 9749; 1, 0011]	[-11, 4116; -10, 7724]

Tabela 5: C++: Sistema de 6 barras

Barra	Magnitude(pu)	Ângulo(°)
1	[1, 0999; 1, 1000]	[0, 0000; 0, 0000]
2	[1, 0996; 1, 1003]	[-3, 1898; -3, 1880]
3	[0, 9024; 0, 9077]	[-11, 7543; -11, 7014]
4	[1, 0066; 1, 0112]	[-8, 8535; -8, 8203]
5	[0, 9351; 0, 9416]	[-11, 5415; -11, 4811]
6	[0, 9849; 0, 9909]	[-11, 1163; -11, 0625]

Analisando a tabela 4, para a barra dois, a magnitude de tensão no MatLab está compreendida entre [1, 0988; 1, 1012] e no C++, este valor, na tabela 5, é [1, 0996; 1, 1003]. Observe que o diâmetro da magnitude de tensão do intervalo 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-XSC também é menor do que o do MatLab ([-3, 1954; -3, 1826] no MatLab, [-3, 1898; -3, 1880] no C++).

Para a barra cinco, a magnitude de tensão no MatLab está compreendida entre [0, 9226; 0, 9542] e no C++ entre [0, 9351; 0, 9416]. O intervalo do ângulo de fase da tensão complexa nesta barra no MatLab está entre [-11, 8858; -11, 1450] e no C++, entre [-11, 5415; -11, 4811]. 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++.

Por fim, analisando a barra 6, tem-se os intervalos [0, 9749; 1, 0011] e [0, 9849; 0, 9909], para o valor da magnitude de tensão, respectivamente, no MatLab e no C++. Para os intervalos do ângulo de fase da tensão complexa, obteve-se no MatLab [-11, 4116; -10, 7724] e no C++, [-11, 1163; -11, 0625]. Assim, as mesmas conclusões anteriores são obtidas novamente.

5.2. Região Sul e Sudeste 352 barras

O sistema de potência de 352 barras é um equivalente real da região Sul-Sudeste do Brasil e as demandas totais de potências ativa e reativa são, respectivamente,

4.404,90 MW e 1.558,70 MVar. Para a geração dos resultados intervalares, foi considerada uma variação em torno do ponto médio das demandas de potências ativa e reativa, em cada barra de carga, de 1%. Esse valor percentual é devido à consideração de erros de leitura nas potências ativas e reativas nas barras de carga.

A tabela 6 apresenta os resultados correspondentes aos limites mínimos e máximos das magnitudes de tensão e ângulos de fase nas barras 6, 340, 938 e 1069. Como a simulação está sendo realizada em um sistema real, optou-se por utilizar o número real da barra no sistema interligado. Essas quatro barras selecionadas são as mais problemáticas, pois apresentam as menores magnitudes de tensão.

Tabela 6: C++: Sistema de 352 barras

Barra	Magnitude(pu)	Ângulo(°)
6	[0, 9194; 0, 9498]	[-44, 7835; -43.2754]
340	[0, 8935; 0, 9227]	[-40, 3153; -39.4332]
938	[0, 9186; 0, 9308]	[-23, 0928; -22.8096]
1069	[0, 9873; 1, 0017]	[-20.0055; -19.7954]

Da tabela 6, observa-se que a barra 6 possui o maior diâmetro para a magnitude de tensão e também que a barra 938 apresenta o menor diâmetro para a magnitude de tensão.

Utilizando um fluxo de potência pontual, o valor da magnitude da tensão na barra 340 é de 0,9081 pu e $-39,8402^\circ$ para o ângulo de fase. Observe que os valores pontuais pertencem aos intervalos obtidos com a biblioteca intervalar C-XSC.

Da tabela 6, pode-se verificar que a magnitude de tensão e o ângulo de fase da barra 1069 valem, respectivamente, [0,9873; 1,0017] e [-20,0055; -19,7954]. O fluxo de potência pontual apresenta como resultado 0,9944 pu e $-20,1105^\circ$ para a magnitude de tensão e para o ângulo de fase, respectivamente. Novamente, constata-se que os valores pontuais estão compreendidos dentro dos intervalos gerados pelo algoritmo proposto, o que valida os resultados utilizando a biblioteca C-XSC.

A tabela 7 apresenta os pontos médios da magnitude de tensão e ângulo de fase das tensões intervalares para as barras 6, 340, 938 e 1069 e também o erro destes valores em relação ao ponto médio do intervalo. Por exemplo, a estimativa de erro da magnitude de tensão na barra 340 é de 1,60% e para o ângulo de fase nesta mesma barra é de 1,11%.

6. Conclusões

O uso da biblioteca C-XSC é uma boa alternativa para a aplicação da Matemática Intervalar ao 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.

Tabela 7: Influência dos erros

Barra	Magnitude(pu)		Ângulo(°)	
	Ponto Médio	Erro %	Ponto Médio	Erro %
6	0,9346	1,62	-44,0294	1,71
340	0,9001	1,60	-40,3153	1,11
938	0,9237	0,65	-23,0928	0,66
1069	0,9945	0,72	-20,0055	0,53

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.

Os resultados obtidos mostram que é possível aplicar a Matemática Intervalar ao problema do fluxo de potência com a utilização da biblioteca C-XSC a sistemas elétricos reais de grande porte para produzir resultados mais realistas.

Em contraste com o C e C++, todos os operadores aritméticos predefinidos em C-XSC desenvolvem 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.

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 compatíveis em computadores com diferentes compiladores C++.

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 elétricos reais não foram possíveis com o uso do software computacional MatLab. Por ser uma linguagem interpretada, torna-se muito lenta 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.

Em relação aos diâmetros menores para as magnitudes e ângulos de fase das tensões complexas nas barras da rede elétrica obtidos com o uso da biblioteca C-XSC, se comparados com a biblioteca Intlab no Matlab, cumpre esclarecer o que segue. Primeiro que, implementando o algoritmo com o C-XSC de forma correta, se tem a garantia de encontrar o resultado correto contido no intervalo resultante devido ao uso de intervalos, arredondamentos direcionados e aritmética de alta exatidão. O intervalo no C-XSC é de melhor qualidade do que no IntLab pois o

IntLab optou em seu desenvolvimento por ter um melhor desempenho em termos de velocidade com uma respectiva perda de exatidão (em comparação com o C-XSC) com o uso de funções da BLAS.

Por outro lado, para se obter um intervalo de menor diâmetro, porém ainda correto, isto não está relacionado ao uso da biblioteca C-XSC ou da Intlab. Este fato está diretamente relacionado ao algoritmo utilizado para a modelagem matemática do problema e do algoritmo numérico utilizado para resolver as equações algébricas não-lineares resultantes dessa modelagem. Isto passaria pela utilização de controles adicionais de erro que são gerados durante a execução do programa. Normalmente, quando se implementa algoritmos intervalares é comum se ter, embutido nesses, técnicas para o controle de erro. Entretanto, existem algumas técnicas que poderiam ser utilizadas para melhorar o resultado gerado (gerando um intervalo com diâmetro menor). Poderia, por exemplo, ser citada a técnica “error free transformations” [4], que atualmente se encontra em estudo pelo grupo do Prof. Rump na Universidade de Tecnologia de Hamburgo, Alemanha [11, 12]. Ela é utilizada para o controle de erro, principalmente, na função somatório. Com ela, pode-se estimar o “erro” do “erro” de uma forma mais precisa, de modo a incluir essa informação no intervalo resultante, gerando um intervalo resposta com menor diâmetro.

Abstract. This paper presents Interval Mathematics applied to the uncertainty problems in electric power systems using the C-XSC library. The proposed approach is based on the analysis of the influence of the both measurement errors and numerical computation errors on the results obtained. Interval Mathematics is an effective technique on the control of several kinds of computational errors presented in scientific calculations. A software using the C-XSC library was developed in order to deal with the power flow problem in an interval manner. In order to assess the performance of the proposed approach, numerical results for both a hypothetical electric system from IEEE and a real equivalent power system from Brazilian South-Southeast region are presented and discussed. A comparative study between results obtained by our methodology and another software using the Intlab library in Matlab environment are also presented.

Referências

- [1] L.V. Barboza, G.P. Dimuro, R.H.S. Reiser, Towards interval analysis of the load uncertainty in power electric systems, in “Proc. 8th Probabilistic Methods Applied to Power System Conference”, pp. 12-16, Ames, USA, 2004.
- [2] L.V. Barboza, G.P. Dimuro, R.H.S. Reiser, Interval mathematics applied to the load flow analysis, in “Proc. 17th IMACS Word Congress Scientific Computation, Applied Mathematics and Simulations”, pp. 1-5, Paris, France, 2005.
- [3] G.P. Dimuro, “Domínios Intervalares da Matemática Computacional”, Dissertação de Mestrado, Universidade Federal do Rio Grande do Sul, Porto Alegre, 1991.

- [4] S. Graillat, V.M. Morain, Error-free transformations in real and complex floating point arithmetic, in “Proc. 2007 International Symposium on Nonlinear Theory and its Applications”, pp. 341-344, Vancouver, Canada, 2007.
- [5] R. Krawczyk, Newton-algorithmen zur bestimmung von nullstellen mit fehlerschranken, *Computing*, **4** (1969), 187-201.
- [6] U. Kulisch, “Advanced Arithmetic for the Digital Computer: Design of Arithmetic Units”, Springer, 2003.
- [7] F.W. Macedo, “Análise de Erros”, Vila Real, Departamento de Matemática, Universidade de Trás-os-Montes e Alto Doro, Portugal, 1992. Publicação técnica, 63p. Disponível em <http://home.utad.pt/wmacedo/publicacoes/Publicacoes.html> (20/11/2008)
- [8] R. Moore, “Interval Analysis”, Philadelphia, 1966.
- [9] R.E. Oliveira, A.T. Diverio, M.D. Claudio, “Fundamentos da Matemática Intervalar”, Porto Alegre, Instituto de Informática da UFRGS, Brasil: Editora Sagra Luzzato, 2001.
- [10] S.M. Rump, “INTLAB - INTerval LABoratory”, Dordrecht, Institute for Reliable Computing, Alemanha: Kluwer Academic Publishers, 1999. 77 - 104p. Disponível em <http://www.ti3.tu-harburg.de/rump/intlab/> (20/11/2008)
- [11] S.M. Rump, T. Ogita, S. Oishi, Accurate floating-point summation Part I: faithful rounding, *SIAM Journal on Scientific Computing*, **31**, No. 1 (2008), 189-224.
- [12] S.M. Rump, T. Ogita, S. Oishi, Accurate floating-Point summation Part II: sign, K-fold faithful and rounding to nearest, *SIAM Journal on Scientific Computation*, to appear.
- [13] S.P. Shary, “Krawczyk Operator Revised”, Novosibirsk, Institute of Computational Technologies, Rússia, 2004. Workshop on Interval Mathematics and Interval Constraint Programming, 307 - 313p. Disponível em <http://old.ict.nsc.ru/interval/shary/Papers/IMRO-04.pdf> (20/11/2008)