

UNIVERSIDADE FEDERAL DE MINAS GERAIS  
ESCOLA DE ENGENHARIA  
PROGRAMA DE PÓS-GRADUAÇÃO EM ENGENHARIA ELÉTRICA

**DETECÇÃO DE PONTOS DE MUDANÇA EM SÉRIES  
TEMPORAIS UTILIZANDO UMA FORMULAÇÃO  
NEURAL/FUZZY/BAYESIANA: APLICAÇÃO NA  
DETECÇÃO DE FALHAS**

FABIANO DE SOUZA MOREIRA

Belo Horizonte  
01 de junho de 2011

UNIVERSIDADE FEDERAL DE MINAS GERAIS  
ESCOLA DE ENGENHARIA  
PROGRAMA DE PÓS-GRADUAÇÃO EM ENGENHARIA ELÉTRICA

**DETECÇÃO DE PONTOS DE MUDANÇA EM SÉRIES  
TEMPORAIS UTILIZANDO UMA FORMULAÇÃO  
NEURAL/FUZZY/BAYESIANA: APLICAÇÃO NA  
DETECÇÃO DE FALHAS**

Dissertação apresentada ao Programa de Pós-Graduação em Engenharia Elétrica da Universidade Federal de Minas Gerais como requisito parcial para a obtenção do grau de Mestre em Engenharia Elétrica.

FABIANO DE SOUZA MOREIRA  
ORIENTADOR: REINALDO MARTINEZ PALHARES  
CO-ORIENTADOR: MARCOS FLÁVIO SILVEIRA VASCONCELOS D'ANGELO

Belo Horizonte  
01 de junho de 2011

# Agradecimentos

Agradeço primeiramente a minha família, Fernanda, Samuel, Pai e Mãe, pelo apoio sem o qual não conseguiria concluir este projeto. Agradeço aos meus orientadores, professor Reinaldo Martinez Palhares e professor Marcos Flávio Silveira Vasconcelos D'Angelo pela ajuda na definição e desenvolvimento do trabalho. Agradeço também pela paciência e boa vontade em me ajudarem na solução dos problemas que apareceram durante o desenvolvimento do trabalho. Ao professor Walmir pela ajuda na utilização de lógica fuzzy na metodologia. A Gerdau Açominas, meus agradecimentos por me dar condições de frequentar as aulas, possibilitando o desenvolvimento deste trabalho.

# Resumo

Neste trabalho o problema de detecção de até dois pontos de mudança em séries temporais utilizando uma formulação neural/*fuzzy*/Bayesiana foi tratado. Este problema é abordado usando uma formulação de três passos, ou seja: o primeiro passo consiste de um algoritmo de classificação do tipo rede neural de Kohonen que define o modelo a ser usado, de um ponto de mudança ou dois pontos de mudança. O segundo passo consiste em uma clusterização *fuzzy* para transformar a série temporal inicial, com distribuição arbitrária, em uma nova série cuja distribuição de probabilidade pode ser aproximada por uma distribuição beta. Os centros dos clusters *fuzzy* são determinados pelo algoritmo de classificação do primeiro passo. O último passo consiste em usar o algoritmo Metropolis-Hastings para realizar a detecção de até dois pontos de mudança na nova série temporal gerada pelo segundo passo, que tem distribuição beta. A principal contribuição e diferença apresentadas neste trabalho, quando comparado a trabalhos anteriores, é a possibilidade de detectar dois pontos de mudança na série temporal considerada. Resultados simulados são apresentados no decorrer da dissertação para ilustrar a metodologia proposta.

# Abstract

In this work, the problem of detecting till two change points in time series is handled by using a new neural/*fuzzy*/Bayesian technique. This proposed technique is split into a three-step formulation, namely: the first step is performed by a Kohonen neural network classification algorithm that defines the model to be used in the case of one change point or two change points in the time series. The second step consists of a *fuzzy* clustering to transform the initial data in the time series, with arbitrary distribution, into a new one that can be approximated by a beta distribution. Also, the *fuzzy* cluster centers are determined by using the Kohonen neural network classification algorithm used in the first step. The last step consists in using the Metropolis-Hastings algorithm to appropriately perform the detection of the change points in the transformed time series generated by the second step, with beta distribution. The main contribution of the proposed approach in this work, related to previous one in the Literature, is to allow to detect till two change points in time series with the correct model selection. Simulation results are presented in this work to illustrate the effectiveness of the proposed approach.

# Sumário

<b>1</b>	<b>Introdução</b>	<b>1</b>
1.1	Contextualização da detecção de falhas . . . . .	1
1.2	Inspirações para este trabalho . . . . .	5
1.3	Perspectivas histórica em pontos de mudança . . . . .	5
1.4	Objetivo do trabalho . . . . .	7
1.5	Organização do texto . . . . .	7
<b>2</b>	<b>Uma visita a conceitos preliminares</b>	<b>9</b>
2.1	Rede neural de Kohonen . . . . .	9
2.2	Conjuntos Fuzzy . . . . .	10
2.3	Formulação do Metropolis-Hastings . . . . .	13
2.3.1	Cadeias de Markov . . . . .	14
2.3.2	Algoritmo Metropolis-Hastings básico . . . . .	15
<b>3</b>	<b>Metodologia</b>	<b>17</b>
3.1	Rede de Kohonen para clusterização e seleção de modelos . . . . .	17
3.2	Transformação da série temporal - Fuzzificação . . . . .	20
3.3	Formulação do Metropolis Hastings para dois pontos de mudança . . . . .	22
<b>4</b>	<b>Simulações e estudo de casos</b>	<b>32</b>
4.1	Modelo de uma máquina de indução e detecção de pontos de mudança . . . . .	32
4.2	Aplicação da metodologia proposta . . . . .	38
<b>5</b>	<b>Considerações Finais</b>	<b>53</b>
5.1	Conclusão . . . . .	53
5.2	Proposta de trabalhos futuros . . . . .	54
	<b>Referências Bibliográficas</b>	<b>55</b>

# Lista de Figuras

1.1	Supervisão de processos industriais com integração de sistema FDI . . . . .	1
1.2	Comportamento de falha abrupta no tempo . . . . .	2
1.3	Comportamento de falha incipiente no tempo . . . . .	2
1.4	Classificação de métodos de FDI. . . . .	3
1.5	Esquema geral para detecção de falhas através de modelos quantitativos. . . . .	4
1.6	Ponto de mudança detectado - primeiro caso . . . . .	6
1.7	Ponto de mudança detectado - segundo caso . . . . .	7
1.8	Histograma dos pontos de mudança detectados . . . . .	8
1.9	Exemplo de série temporal . . . . .	8
2.1	Rede de Kohonen em arranjo unidimensional . . . . .	10
2.2	Rede de Kohonen em arranjo bidimensional . . . . .	11
2.3	Função de ativação, " <i>chapéu mexicano</i> " . . . . .	11
2.4	Redução da vizinhança durante o treinamento . . . . .	12
2.5	Pertinência segundo a teoria clássica de conjuntos de indivíduos com relação ao <i>predicado velho</i> . . . . .	14
2.6	Pertinência segundo a teoria de conjuntos fuzzy de indivíduos com relação ao <i>predicado velho</i> . . . . .	15
3.1	Diagrama de blocos da metodologia proposta . . . . .	18
3.2	Rede neural auto-organizada para determinar os centros das funções de pertinência . . . . .	18
3.3	Funções de Pertinência . . . . .	19
3.4	Série temporal com $p_1 = 1$ , $p_2 = 10$ , e $p_3 = 20$ fixados, $\epsilon(t) \sim U(0, 1)$ , $m_1 = 20$ , $m_2 = 50$ e 100 amostras. . . . .	20
3.5	Funções de pertinência, $\mu_1(t)(-)$ , $\mu_2(t)(\dots)$ e $\mu_3(t)(- - -)$ . . . . .	21
3.6	Resultado da metodologia proposta para $p_1 = 1$ , $p_2 = 10$ , $p_3 = 20$ $e(t) \sim U(0, 1)$ , e $m_1 = 20$ . . . . .	31
3.7	Resultado da metodologia proposta para $p_1 = 1$ , $p_2 = 10$ , $p_3 = 20$ $e(t) \sim U(0, 1)$ , e $m_2 = 50$ . . . . .	31
4.1	Representação dos enrolamentos . . . . .	33
4.2	Corrente da fase <i>a</i> . . . . .	35
4.3	Corrente da fase <i>b</i> . . . . .	35

4.4	Corrente da fase $c$ .	36
4.5	Corrente da fase $a$ (rms).	36
4.6	Corrente da fase $b$ (rms).	37
4.7	Corrente da fase $c$ (rms).	37
4.8	Corrente da fase $a$ (rms): 2% e 5% de espiras em curto na fase $a$	38
4.9	Evolução do parâmetro $m_1$ para a corrente da fase $a$	39
4.10	Evolução do parâmetro $m_2$ para a corrente da fase $a$	40
4.11	Histograma do primeiro ponto de mudança para a corrente da fase $a$	40
4.12	Histograma do segundo ponto de mudança para a corrente da fase $a$	41
4.13	Corrente da fase $b$ (rms): 2% e 5% de espiras em curto na fase $a$	41
4.14	Evolução do parâmetro $m_1$ para a corrente da fase $b$	42
4.15	Evolução do parâmetro $m_2$ para a corrente da fase $b$	42
4.16	Histograma do primeiro ponto de mudança para a corrente da fase $b$	43
4.17	Histograma do segundo ponto de mudança para a corrente da fase $b$	43
4.18	Corrente da fase $c$ (rms): 2% e 5% de espiras em curto na fase $a$	44
4.19	Evolução do parâmetro $m_1$ para a corrente da fase $c$	44
4.20	Evolução do parâmetro $m_2$ para a corrente da fase $c$	45
4.21	Histograma do primeiro ponto de mudança para a corrente da fase $c$	45
4.22	Histograma do segundo ponto de mudança para a corrente da fase $c$	46
4.23	Corrente da fase $a$ (rms): 0.1% e 0.3% de espiras em curto na fase $a$	47
4.24	Evolução do parâmetro $m_1$ para a corrente da fase $a$	48
4.25	Evolução do parâmetro $m_2$ para a corrente da fase $a$	48
4.26	Histograma do primeiro ponto de mudança para a corrente da fase $a$	49
4.27	Histograma do segundo ponto de mudança para a corrente da fase $a$	49
4.28	Corrente da fase $b$ (rms): 0.1% e 0.3% de espiras em curto na fase $a$	50
4.29	Evolução do parâmetro $m_1$ para a corrente da fase $b$	50
4.30	Evolução do parâmetro $m_2$ para a corrente da fase $b$	51
4.31	Histograma do primeiro ponto de mudança para a corrente da fase $b$	51
4.32	Histograma do segundo ponto de mudança para a corrente da fase $b$	52

# Lista de Tabelas

2.1	Grau de pertinência dos indivíduos representando o <i>predicado velho</i> . . . . .	14
-----	---	----

# Acrônimos

FDI	Isolação e Detecção de Falhas, <i>Fault Detection and Isolation</i>
MCMC	Cadeia de Markov Monte Carlo, <i>Markov Chain Monte Carlo</i>
PCA	Análise de Componentes Principais, <i>Principal Component Analysis</i>

# Capítulo 1

## Introdução

### 1.1 Contextualização da detecção de falhas

Atualmente, existe uma tendência de se concentrar a atenção e recursos para a detecção segura de falhas em sistemas dinâmicos. Isso pode ser explicado pelos ganhos obtidos com o aumento de confiabilidade dos equipamentos, isto é, a previsibilidade sobre as condições de funcionamento dos mesmos o que reduz o risco de paradas não programadas de produção, perdas materiais quantitativas e qualitativas e acidentes de trabalho. Além disso, o custo para solucionar uma falha logo que começa a ocorrer é mais baixo que quando a falha já está em estado avançado ou quando o sistema já perdeu completamente sua função. Essa tendência resultou na necessidade de sistemas de supervisão FDI (Fault Detection and Isolation), conforme figura 1.1.

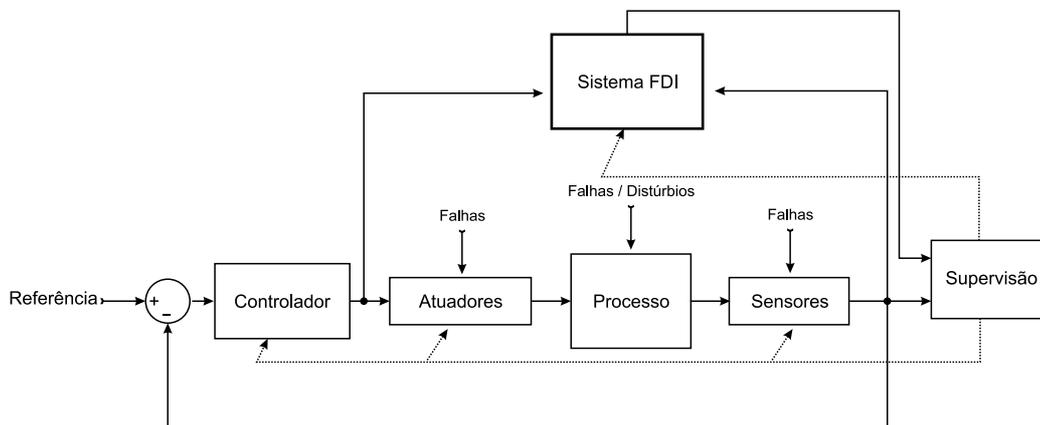


Figura 1.1: Supervisão de processos industriais com integração de sistema FDI

No contexto de sistemas dinâmicos, podemos classificar as falhas em dois grupos (Isermann e Balle, 1997):

1. *Falhas abruptas*: são as que acontecem em um curto espaço de tempo (ver figura 1.2);

2. *Falhas incipientes*: são as que afetam gradualmente o funcionamento normal do processo, e por isso podem levar um tempo maior para serem detectadas (ver figura 1.3).

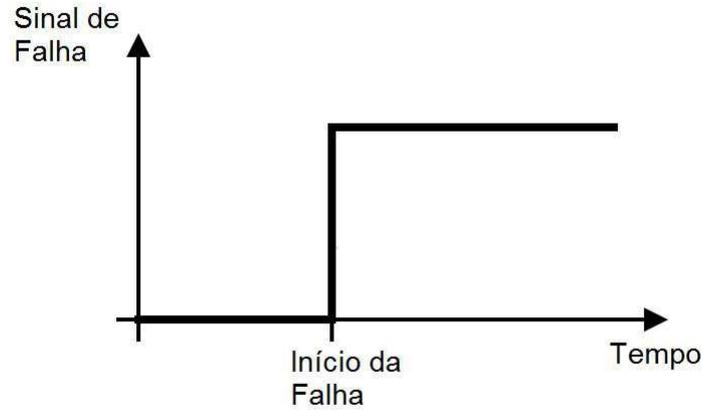


Figura 1.2: Comportamento de falha abrupta no tempo

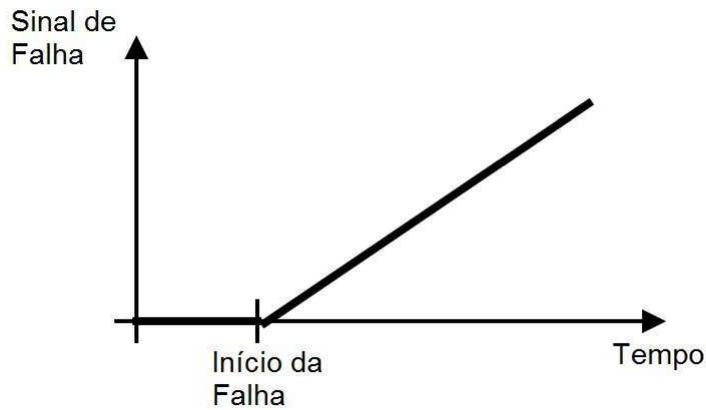


Figura 1.3: Comportamento de falha incipiente no tempo

Conforme descrito em [D'angelo. \(2010\)](#), a classe de sistemas dinâmicos tratada neste trabalho é descrita por:

$$\dot{x}(t) = \xi(x(t), u(t)) + \beta(t - T)f(x(t), u(t)) \quad (1.1)$$

sendo  $x \in \mathbb{R}^n$  o vetor de estados,  $u \in \mathbb{R}^m$  o vetor de entradas,  $\xi, f : \mathbb{R}^{n \times m} \rightarrow \mathbb{R}^n$ ,  $T \geq 0$  o tempo de início da falha, e  $\beta \in \mathbb{R}^{n \times n}$  a matriz que representa o comportamento temporal da falha incipiente que pode ser modelado por:

$$\beta(t - T) = \text{diag}\{\beta_1(t - T), \beta_2(t - T), \dots, \beta_n(t - T)\}$$

sendo

$$\beta_i(\tau) = \begin{cases} 0, & \text{se } \tau < 0, \\ \rho_i \tau, & \text{se } 0 \leq \tau \leq T_{fi} \\ 1, & \text{se } \tau > T_{fi} \end{cases}$$

sendo  $T_{fi}$  o tempo onde a falha atinge seu valor máximo a partir do seu início e  $\rho_i = \frac{1}{T_{fi}}$  uma constante que representa a taxa de desenvolvimento da falha no estado  $i$ . Para falhas incipientes este valor é bastante reduzido.

A literatura apresenta varias estratégias para tratar os problemas de FDI (Fault Detection and Isolation) (Chen e Patton, 1999). Estas estratégias podem ser classificadas em abordagens baseadas em modelos qualitativos e quantitativos conforme figura 1.4.

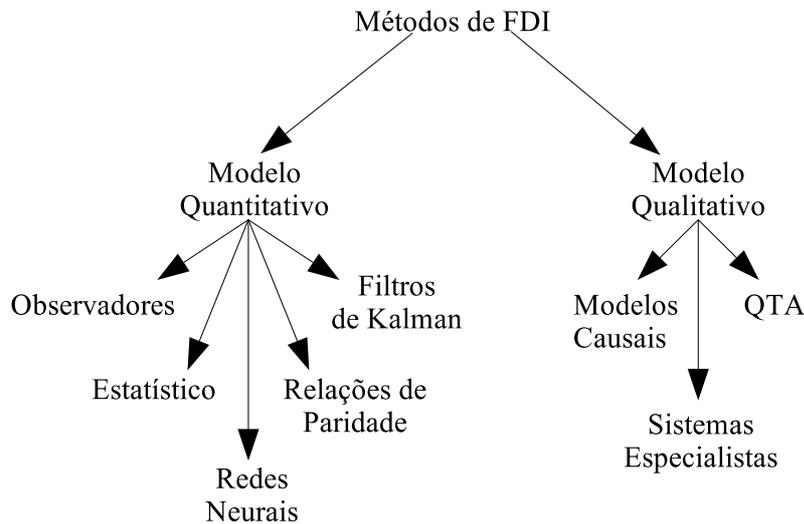


Figura 1.4: Classificação de métodos de FDI.

Dentre estas abordagens, existem métodos que utilizam dados retirados diretamente do sistema, isto é, sem utilização de modelo matemático. Existem também métodos baseados em conhecimento, métodos baseados em modelos e métodos que combinam conhecimento e dados do sistema.

Os métodos baseados em dados exploram somente as informações históricas do sistema e, como exemplo deste grupo, pode-se citar análise de componentes principais (PCA), reconhecimento de padrões e análise de espectro.

Exemplos de métodos baseados em conhecimento são os sistemas especialistas no qual o conhecimento sobre o processo é transformado em um conjunto de regras. Uma falha é apontada segundo a ocorrência de um grupo de sintomas (*SE conjunto de sintomas ENTÃO*

*falha*). Este métodos requerem a manutenção de uma base de conhecimento para a geração das regras o que pode representar um custo elevado.

Os métodos baseados em modelo comparam o comportamento real do sistema com um modelo do mesmo, isto é, compara a saída do processo com a saída do modelo. Neste grupo estão os observadores, filtros, redes neurais, redes neurofuzzy, relações de paridade e estimação de parâmetros usando algoritmos de identificação (Puig et al., 2006), (Douglas e Speyer, 1996), (Chen e Patton, 1999), (Hou e Patton, 1998), (Douglas e Speyer, 1996), (Takahashi et al., 1999), (Takahashi e Peres, 1999), (Jiang e Zhou, 2005), (Dai et al., 2009), (Ploix e Adrot, 2006) e (Wilsky, 1976). Estes métodos necessitam de muita atenção na criação do modelos que devem indicar corretamente inconsistências entre o comportamento real do sistema e o comportamento anormal (com falha). Estas inconsistências são indicadas por sinais chamados resíduos, ver figura 1.5.

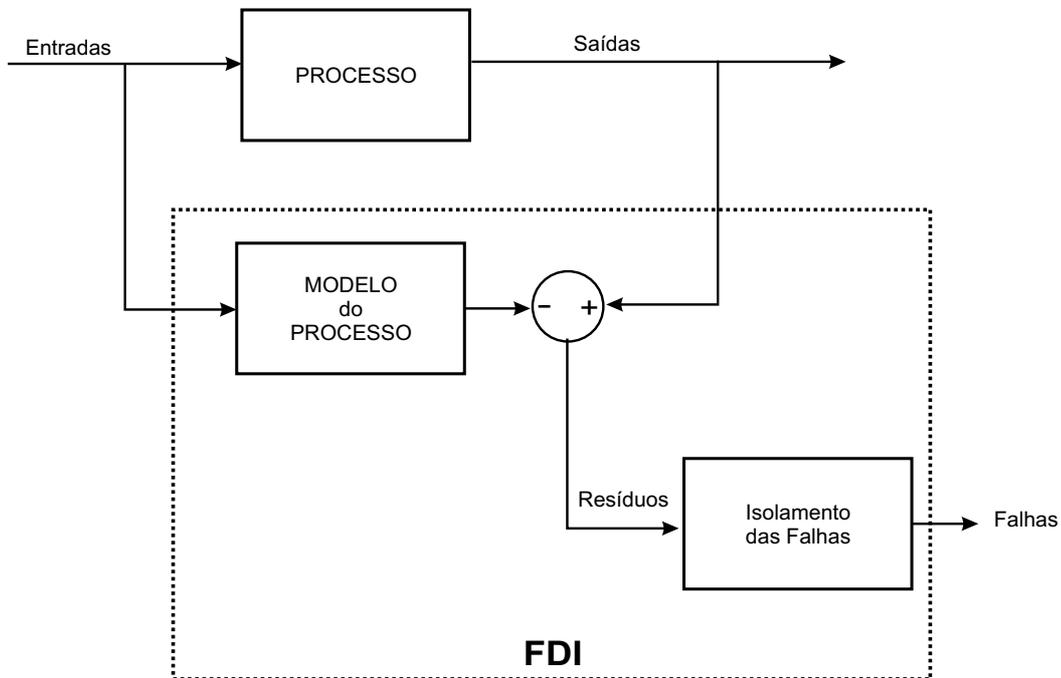


Figura 1.5: Esquema geral para detecção de falhas através de modelos quantitativos.

Um grupo de métodos mais recente trata o problema de detecção de falhas combinando dados e conhecimento sobre o sistema. Neste grupo pode-se destacar as redes neurais, lógica fuzzy, algoritmos genéticos e combinações entre estes métodos (Calado et al., 2001), (Lo et al., 2009), (D'Angelo e Costa, 2001), (Rigatos e Zhang, 2009), (Bartys et al., 2006), (Witczak et al., 2006) e (Bocaniala e da Costa, 2006). Estas técnicas são interessantes em problemas que envolvem sistemas não lineares, pois não requerem modelos matemáticos explícitos que são de difícil obtenção nestes casos.

Tem-se ainda, trabalhos que exploram o uso de estatística *Bayesiana* na solução de problemas de detecção de falhas (Tang, 2000), (Berec, 1998), (Alag, 1996), (Castilloa et al., 1999), (Mast et al., 1999) e (O'Reilly, 1998). Destaca-se nesta linha, o uso de redes *Bayesianas*, que

são grafos orientados que representam a dependência probabilística entre as variáveis como uma forma de modelar a incerteza associada com o modelo.

Porém, no geral, tais linhas de trabalhos lidam com falhas abruptas, mais simples de serem "capturadas" do que falhas incipientes, com evolução lenta no tempo. Considerando este problema, os trabalhos apresentados em [D'Angelo et al. \(2011b\)](#), [D'Angelo et al. \(2011a\)](#), [D'Angelo et al. \(2010\)](#), [D'Angelo et al. \(2008\)](#) e [D'angelo. \(2010\)](#) tiveram como principal contribuição a elaboração de uma abordagem que permitia a detecção de falhas incipientes em sistema dinâmicos, sem a necessidade de modelo matemático, resíduos e geração de padrões.

## 1.2 Inspirações para este trabalho

A inspiração para este trabalho é a detecção de falhas utilizando metodologia baseada em teoria dos conjunto fuzzy associadas a estatística Bayesiana apresentadas em [D'Angelo et al. \(2010\)](#) e [D'Angelo et al. \(2011a\)](#). Tais trabalhos analisam séries temporais com nenhum ou apenas um ponto de mudança, dando margem a seguinte questão:

*Se houver um segundo ponto de mudança na janela móvel, qual o comportamento da metodologia tratada nos trabalhos citados anteriormente?*

Com o objetivo de responder a esta questão, o algoritmo de detecção de um ponto de mudança apresentado em [D'Angelo et al. \(2010\)](#) e [D'Angelo et al. \(2011a\)](#) é aplicado a uma série temporal que explicitamente apresenta dois pontos de mudança. O resultado observado foi que o algoritmo identificou o primeiro ponto de mudança, ver figura [1.6](#), ou o segundo ponto de mudança, ver figura [1.7](#), de forma aleatória.

O algoritmo apresentado em [D'Angelo et al. \(2010\)](#) e [D'Angelo et al. \(2011a\)](#) foi executado 100 vezes, e o resultado, mostrado na figura [1.8](#), não permite concluir se há um ou dois pontos de mudança e pode-se verificar também que a escolha do primeiro ou o segundo ponto de mudança é aleatória.

Com o objetivo de resolver esta deficiência na formulação citada, esta dissertação revisita o método proposto em [D'Angelo et al. \(2010\)](#) e [D'Angelo et al. \(2011a\)](#) e propõe uma adaptação para se detectar até dois pontos de mudança.

## 1.3 Perspectivas histórica em pontos de mudança

A identificação de pontos de mudança é um problema encontrado em diversas áreas como estudos de criminalidade ([Loschi et al., 2005](#)), área financeira ([Oh et al., 2005](#)), ecologia ([Beckage et al., 2007](#)), hidrometeorologia ([Perreault et al., 2000](#)). Dado uma série temporal, ver exemplo na figura [1.9](#), o objetivo é detectar se houve um ponto de mudança filtrando as variações normais e ruído da série.

Várias abordagens para resolver este problema já foram formuladas, e estas podem ser divi-

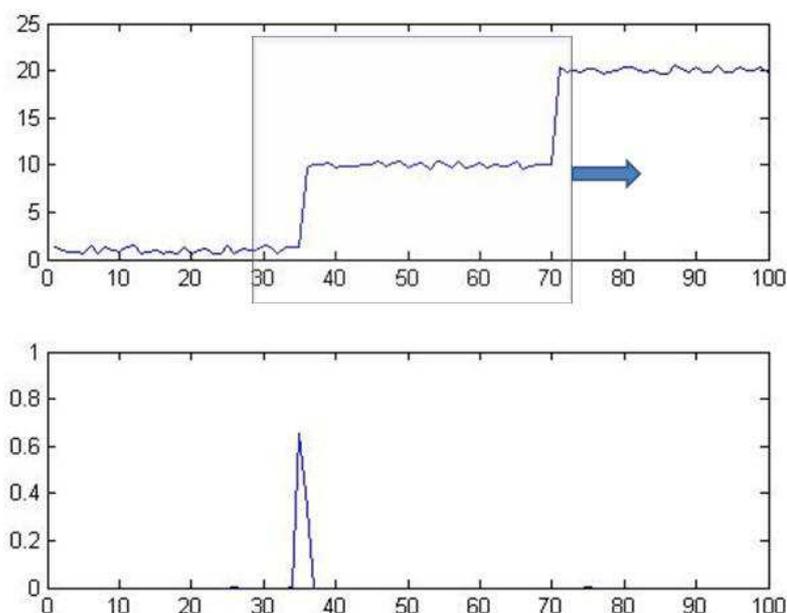


Figura 1.6: Ponto de mudança detectado - primeiro caso

didadas em dois grupos, abordagens Bayesianas e clássicas. Como exemplo de técnica estatística pode-se citar o teste mais comum que é o CUSUM [Hinkey \(1971\)](#). Já para exemplificar as técnicas Bayesianas pode-se citar os métodos MCMC [Barry e Hartigan \(1993\)](#). Todos os métodos mencionados anteriormente dependem de algum conhecimento *a priori* do comportamento estatístico da série temporal, como qual o tipo de distribuição que melhor representa seu comportamento dinâmico.

Para evitar esta dependência, foi proposto em [D'Angelo et al. \(2007\)](#) uma maneira na qual não é preciso nenhum conhecimento *a priori* da série temporal, pois a série original passa por uma transformação definida por operações *fuzzy* e o resultado é uma série que pode ser aproximada por séries com distribuição beta. Utilizando esta metodologia proposta em [D'Angelo et al. \(2007\)](#) pode-se ver uma aplicação na área de detecção de falhas incipientes em [D'Angelo et al. \(2008\)](#). Neste, o método é aplicado para se detectar falhas incipientes no RTN DAMADICS. Já em [D'Angelo et al. \(2011b\)](#) é proposto uma forma de se detectar falhas incipientes no enrolamento de estator de motores de indução por meio de análise de séries temporais de corrente elétrica sem nenhum conhecimento *a priori* da distribuição destas séries. Em [D'Angelo et al. \(2011b\)](#) a metodologia proposta é utilizada em séries temporais reais e pode-se notar que esta é eficiente mesmo para pequenas falhas, isto é, para pequenas variações na série temporal.

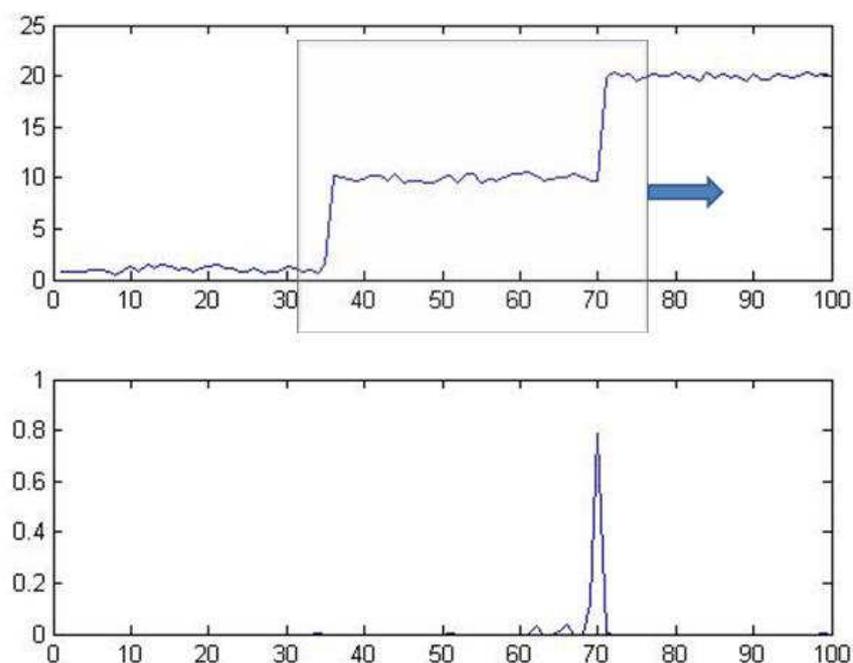


Figura 1.7: Ponto de mudança detectado - segundo caso

## 1.4 Objetivo do trabalho

O objetivo deste trabalho é estender o método proposto em [D'angelo. \(2010\)](#) para detectar até dois pontos de mudança em uma série temporal. Além disso, o método proposto deverá ser capaz de analisar a série temporal e selecionar automaticamente o modelo correto para um ou dois pontos de mudança, ou indicar que não há ponto de mudança.

## 1.5 Organização do texto

O texto está organizado da seguinte forma: o segundo capítulo visita conceitos e técnicas que serão usadas no desenvolvimento do trabalho. O terceiro capítulo apresenta a metodologia desenvolvida e sua fundamentação teórica. O quarto capítulo mostra a aplicação da metodologia em dados simulados de uma máquina de indução utilizando o modelo proposto em [Baccarini et al. \(2004\)](#), bem como uma discussão dos resultados obtidos aplicando a metodologia apresentada. Por último serão apresentadas as conclusões e trabalhos futuros.



## Capítulo 2

# Uma visita a conceitos preliminares

Este capítulo tem o objetivo de revisar alguns conceitos que serão usados no decorrer do trabalho:

1. Rede neural de Kohonen;
2. Conjuntos Fuzzy;
3. Metropolis-Hastings;

### 2.1 Rede neural de Kohonen

O algoritmo de Kohonen pertence a uma classe de redes neurais artificiais que apresentam a capacidade de auto-organização. Estas redes são conhecidas como redes SOM (Self Organizing Maps) [Kohonen \(1990\)](#) e apresentam, como uma de suas principais características, a capacidade de aprender através de exemplos. Elas possuem forte semelhança com estruturas neurofisiológicas, mapa topológico do córtex cerebral, em comparação com outros modelos de redes neurais artificiais. Nas estruturas neurofisiológicas, os neurônios estão espacialmente ordenados, e neurônios próximos tendem a responder a padrões ou estímulos de forma semelhante. A estrutura básica das redes auto-organizáveis é formada, normalmente, de uma camada de entrada e uma camada de saída. Geralmente estas redes tem dimensão 1, ver figura [2.1](#), ou 2, ver figura [2.2](#).

As redes auto-organizáveis possuem um vasto campo de aplicações, sendo o mais comum o reconhecimento de padrões e agrupamento de dados em que classes não são conhecidas *a priori* [Braga et al. \(2007\)](#). Para problemas de reconhecimento de padrões, os padrões que compartilham características comuns devem ser agrupadas, sendo que, cada grupo de padrões representa uma única classe. Para realizar este agrupamento, o algoritmo de aprendizado precisa identificar características significativas nos dados de entrada, e isto só é possível se existir redundância dos dados de entrada. A redundância dos dados fornece informações sobre similaridades e diferenças entre os dados para a rede, sem isso, os dados são como ruído branco para a rede. As redes auto-organizáveis, ou mapas auto-organizáveis, utilizam o algoritmo

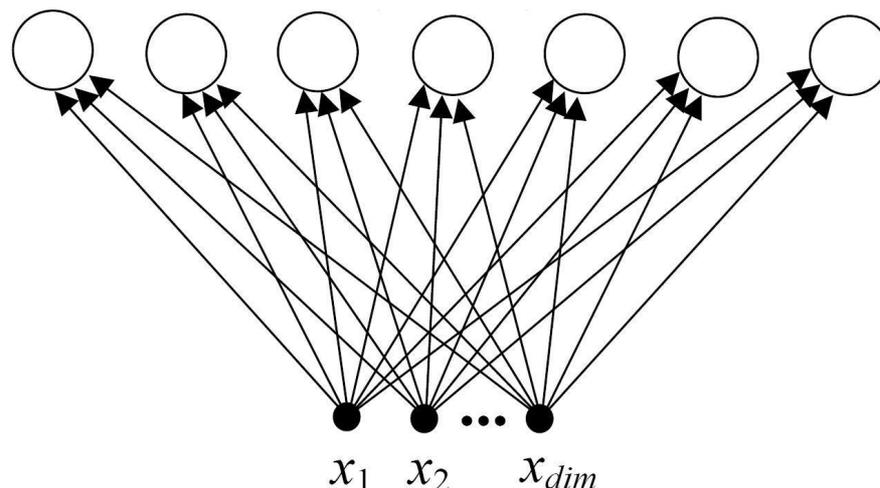


Figura 2.1: Rede de Kohonen em arranjo unidimensional

de treinamento competitivo, os neurônios da rede competem entre si pelo direito de atualizar seus pesos.

Quando uma entrada  $\mathbf{p}$  é apresentada, a rede procura qual o neurônio mais próximo de  $\mathbf{p}$ . Durante o treinamento, a rede aumenta a semelhança do neurônio escolhido e de seus vizinhos ao padrão  $\mathbf{p}$ . Desta forma, a rede constrói um mapa topológico onde neurônios que estão próximos respondem de forma semelhante a padrões de entrada semelhantes. O treinamento ocorre de forma que somente o neurônio vencedor se torna ativo. Uma forma de se implementar essa competição é utilizar conexões laterais inibitórias entre neurônios de saída. Por meio da introdução do conceito de vizinhos topológicos dos neurônios vencedores, o algoritmo de treinamento simula o efeito da função "*chapéu mexicano*" [Braga et al. \(2007\)](#), ver figura 2.3. O efeito em questão é de se ajustar o peso do neurônio vencedor que produziu o maior valor de saída para uma dada entrada, e ajustar também o peso dos neurônios localizados em sua vizinhança.

Para melhorar o desempenho da rede, a rede de Kohonen, reduz a vizinhança dos neurônios vencedores durante o treinamento. A vizinhança define quantos neurônios localizados próximo ao vencedor terão seus pesos ajustados. Nas primeiras iterações a vizinhança é grande, e durante a execução do treinamento, esta região é reduzida até um limite predefinido conforme figura 2.4.

No seção 3.1 serão apresentados mais detalhes sobre a rede de Kohonen e o algoritmo de treinamento usado vai ser explicado.

## 2.2 Conjuntos Fuzzy

Os conjuntos Fuzzy são uma abordagem poderosa para solução de problemas, com uma vasta aplicabilidade, especialmente, nas áreas de controle e tomada de decisão. A utilização desta técnica permite inferir conclusões e gerar respostas a partir de informações incertas.

A teoria dos conjuntos fuzzy é em grande parte uma extensão da teoria clássica dos con-

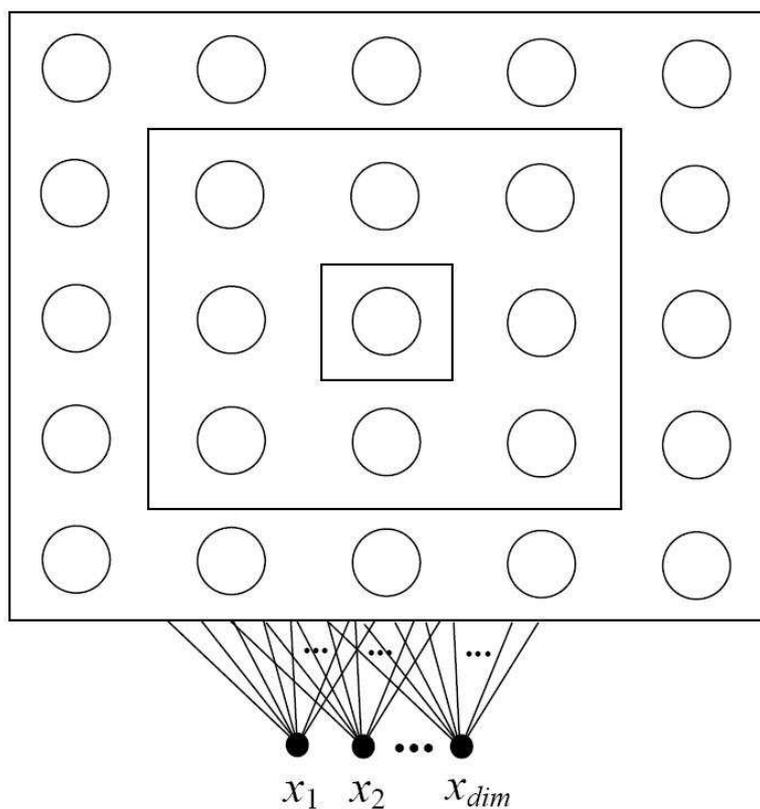


Figura 2.2: Rede de Kohonen em arranjo bidimensional

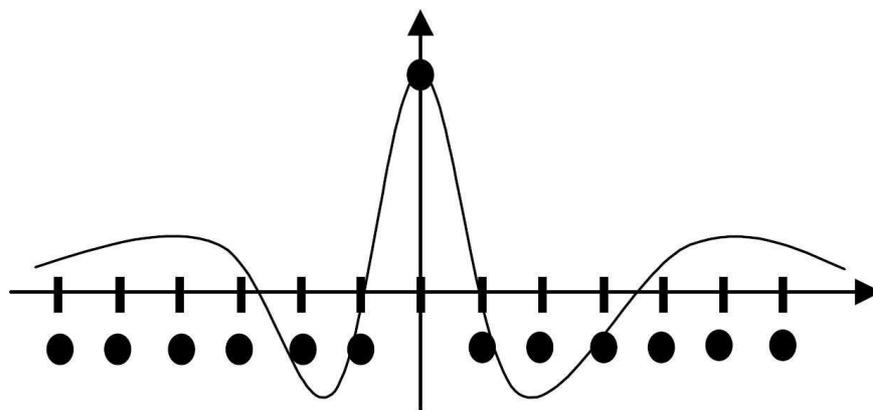


Figura 2.3: Função de ativação, "chapéu mexicano"

juntos. Ele surgiu como uma alternativa para tratar de problemas subjetivos, que necessitem de um raciocínio aproximado, ou ainda para lidar com problemas nos quais há tanto dados numéricos quanto dados na forma lingüística.

Na teoria clássica de conjuntos, uma proposição lógica tem dois extremos: ou é verdadeiro ou é falso. Considerando um conjunto  $A$  e um elemento  $a_1$ , podemos dizer que o elemento

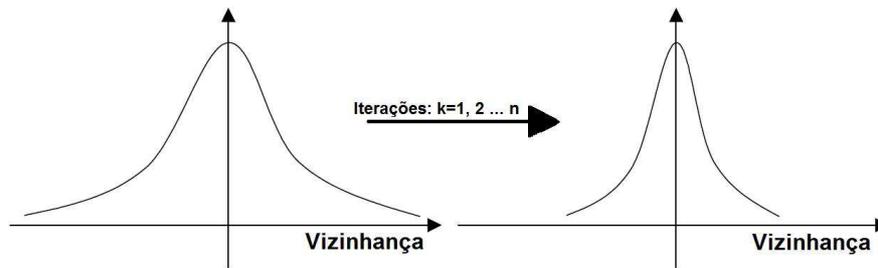


Figura 2.4: Redução da vizinhança durante o treinamento

$(a_1 \in A)$  ou não pertence ( $a_1 \notin A$ ) ao conjunto.

Se o problema é bem definido, como por exemplo, definir no universo  $\mathbb{Z}$  quais são os números primos, o conceito clássico é suficiente. Porém, em problemas subjetivos como, por exemplo, separar dentre um conjunto de pessoas  $P$  definido como  $\mathbb{R}$  os elementos deste conjunto  $p$  em pessoas altas e baixas, temos que uma pessoa com 1,80 metros e outra de 1,75 metros são consideradas altas, porém a pessoa de 1,8 metros é *mais alta* que a de 1,75 metros. Por esse exemplo percebe-se que a definição deste conjunto já não é exata.

Essa subjetividade está ligada ao fato de que o critério que define quais são as pessoas altas e baixas não é bem definido e depende do tipo de problema com que estamos lidando. Os exemplos a seguir, que foram retirados de [Mozelli \(2008\)](#), ilustram essa questão.

Seja um sistema dinâmico linear, estável e de 3ª ordem com a seguinte função de transferência:

$$G(s) = \frac{K}{(s^2 + 2\zeta\omega_n s + 1)(\gamma s + 1)}, \quad (2.1)$$

onde se verifica a existência de um pólo real e outro par de pólos, que podem ser complexos conjugados ou reais dependendo do valor de  $\zeta\omega_n$ .

Na prática, verifica-se que quando

$$|1/\gamma| \geq 10 \cdot |\zeta\omega_n| \quad (2.2)$$

o desempenho do sistema pode ser aproximado pelo desempenho de um sistema de segunda ordem. Neste caso específico, é possível considerar uma relação de dez vezes como muito maior.

Em contrapartida, considere um transistor de junção bipolar. A relação entre as correntes de coletor  $i_c$  e emissor  $i_e$  é dada por:

$$i_c = \frac{\beta}{\beta + 1} i_e,$$

onde  $\beta$  é o ganho de corrente de emissor comum. Valores típicos de  $\beta$  nesses componentes são  $\beta \geq 100$ . De acordo com a aplicação, considerar um ganho  $10\beta$  não será significativo na relação das correntes, i.e., uma ordem de grandeza a mais não será considerada muito maior.

Considerando as particularidades de cada problema que implicam num contexto de grandezas diferentes, a utilização da teoria clássica de conjuntos se torna inviável, uma vez que para cada problema seria necessário definir limiares para separar os elementos em conjuntos.

Ainda considerando o trabalho de [Mozelli \(2008\)](#), voltando ao exemplo do sistema dinâmico mostrado em [2.1](#), Se considerarmos um conjunto de sistemas dinâmicos onde há dominância de pólos, ou seja, cujos elementos satisfazem [\(2.2\)](#). Um sistema somente faz parte deste conjunto quando existe uma relação de pelo menos 10 vezes entre o módulo do pólo dominado e do produto entre coeficiente de amortecimento e frequência natural. Então se um sistema tiver uma relação igual a 9,9 vezes e apresenta um comportamento muito próximo de um sistema com relação igual a 10, ele não fará parte do conjunto o que é uma discrepância.

Os exemplos anteriores mostram claramente as limitações da teoria clássica de conjuntos, indicando a necessidade de uma teoria compatível com os problemas apresentados. Mais exemplos que motivam a utilização de sistemas fuzzy são mostrados em ([Tsoukalas e Uhrig, 1997](#); [Jang et al., 1997](#); [Tanscheit et al., 2007](#)).

Um conjunto fuzzy  $F$  é caracterizado por uma função de pertinência (função característica ou função de compatibilidade)  $f_F(x)$  que associa a cada elemento do conjunto  $F$  um valor de pertinência entre  $[0, 1]$ . Ela pode ser representada por meio de um conjunto de pares ordenados ([Tsoukalas e Uhrig, 1997](#))

$$F = \{(x, f_F(x))\}, \quad x \in X.$$

Assim, esta divisão binária ( $f_1 \in F$ ) ou ( $f_1 \notin F$ ) gerada pela teoria clássica de conjuntos é abandonada e em seu lugar aparece o teoria de conjuntos fuzzy com o conceito de pertinência. Este valor de pertinência representa a relação entre o elemento, por exemplo  $f_1$ , e o conjunto, por exemplo  $F$ , isto é, a medida que o valor se aproxima de 1 significa que maior é o seu grau de pertinência com o conjunto  $F$ .

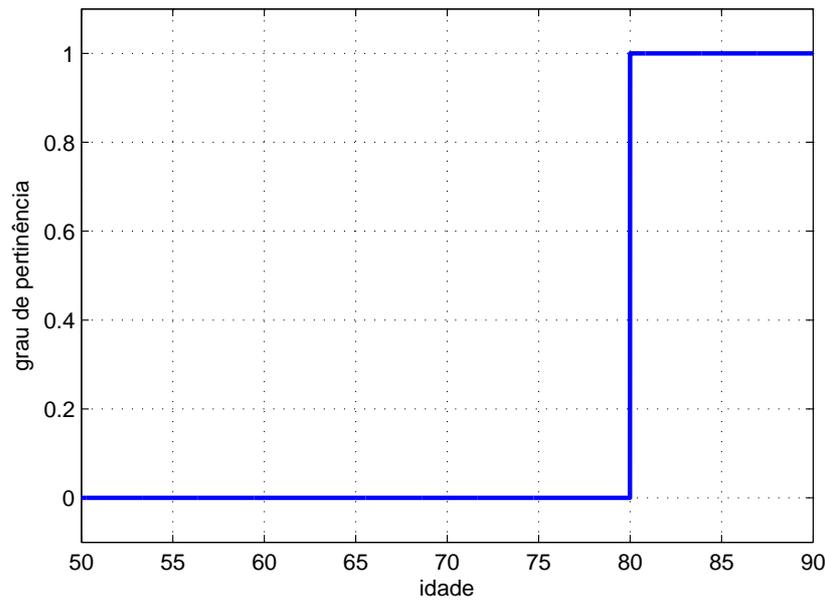
Considere a tarefa de classificar um grupo de indivíduos em relação a sua idade, sendo que a característica observada é se a pessoa é velha. Desta forma, com base na teoria clássica dos conjuntos, podemos considerar que pessoas velhas são somente aquelas com mais de 80 anos. A função característica teria o padrão mostrado na [Figura 2.5](#). Seguindo esta classificação, um individuo com 81 anos faz parte do conjunto enquanto que um individuo com 79 anos não faz.

Por outro lado, com base nos conjuntos fuzzy e analisando o grau de pertinência, pode-se calcular valores para cada individuo que aumentam a medida que a idade aumenta. Estes dados são mostrados na [tabela 2.2](#).

## 2.3 Formulação do Metropolis-Hastings

O algoritmo Metropolis-Hastings pode ser visto como um dos algoritmos de cadeia de Markov (MCMC) mais gerais.

Idade	Grau de pertinência
100	1
80	1
66	0.92
56	0.73
46	0.44
36	0.17
16	0

Tabela 2.1: Grau de pertinência dos indivíduos representando o *predicado velho*Figura 2.5: Pertinência segundo a teoria clássica de conjuntos de indivíduos com relação ao *predicado velho*

### 2.3.1 Cadeias de Markov

Uma cadeia de Markov  $\{X^t\}$  é uma sequência de variáveis aleatórias dependentes

$$X^0, X^1, X^2, \dots, X^t, \dots$$

tal que a distribuição de probabilidade de  $\{X^t\}$ , dada as variáveis passadas, depende somente de  $\{X^{t-1}\}$ . Esta distribuição de probabilidade condicional é chamada de *núcleo de transição* da cadeia de Markov  $K$ , isto é

$$X^{t+1} | X^0, X^1, X^2, \dots, X^t \sim K(X^t, X^{t+1})$$

Por exemplo, um simples passo da cadeia de Markov satisfaz

$$X^{t+1} = X^t + \epsilon_t$$

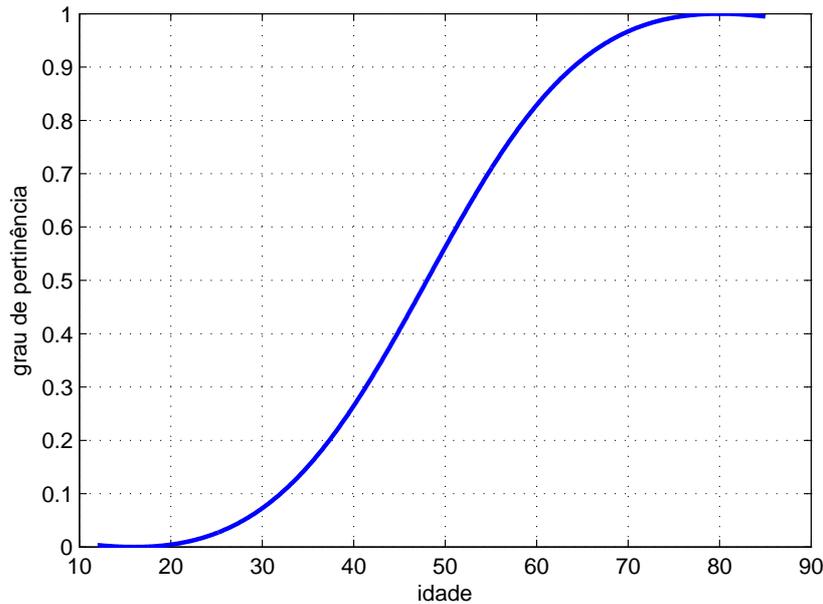


Figura 2.6: Pertinência segundo a teoria de conjuntos fuzzy de indivíduos com relação ao predicado *velho*

onde  $\epsilon_t \sim N(0, 1)$  independente de  $X^t$ .

A cadeia de Markov encontrada no método MCMC possui uma propriedade de estabilidade muito forte. De fato, existe uma distribuição de probabilidade estacionária para estas cadeias, isto é, existe uma distribuição de probabilidade  $f$  tal que se  $X^t \sim f$ , então  $X^{t+1} \sim f$ . Portanto, formalmente, o núcleo e a distribuição estacionária satisfazem a equação 2.3.

$$\int_{\mathcal{X}} K(x, y) f(x) dx = f(y) \quad (2.3)$$

A existência de uma distribuição estacionária coloca uma restrição a  $K$  chamada *irredutibilidade* na teoria das cadeias de Markov, que permite ao núcleo  $K$  movimentos livres em todo o espaço de estados, sendo que não importa o valor inicial  $X^0$ , a sequência  $\{X^t\}$  tem uma probabilidade positiva de eventualmente atingir qualquer região do espaço de estados. A existência de distribuição estacionária tem outras consequências no comportamento da cadeia  $\{X^t\}$ , sendo uma delas que a cadeia converge e que a maioria das cadeias dos algoritmos MCMC são *recorrentes*, isto é, elas voltam a um conjunto arbitrário qualquer infinitas vezes. No caso de cadeias recorrentes, a distribuição estacionária é também uma *distribuição limitadora* no sentido que a distribuição limitadora de  $X^t$  é  $f$  para quase todos os valores iniciais  $X^0$ .

### 2.3.2 Algoritmo Metropolis-Hastings básico

O princípio de trabalho do método MCMC é simples de descrever. Dado uma densidade alvo  $f$ , constrói-se um núcleo de Markov  $K$  com distribuição estacionária  $f$  e então gera-se

uma cadeia de Markov  $(X^t)$  usando este núcleo de tal forma que a distribuição limitadora de  $(X^t)$  é  $f$ . A dificuldade é construir um núcleo  $K$  que é associado com uma densidade arbitrária  $f$ . Porém, existem métodos que geram tais núcleos que são universais e, além disso, são teoricamente válidos para qualquer densidade  $f$ . O algoritmo Metropolis-Hastings é um desses métodos. Dada a densidade alvo  $f$ , ela é associada com uma densidade condicional  $q(x | y)$  que, na prática, é fácil de simular. Mais,  $q$  pode ser quase arbitrário, pois, o único requisito teórico é que a taxa  $f(y)/q(x | y)$  seja uma constante conhecida que não dependa de  $x$  e que  $q(\cdot | x)$  tenha dispersão suficiente para permitir uma exploração total de  $f$ . Desta forma, pode-se dizer que o algoritmo Metropolis-Hastings possui uma característica muito interessante, que para qualquer  $q$ , é possível construir um núcleo Metropolis-Hastings tal que  $f$  seja sua distribuição estacionária.

O algoritmo Metropolis-Hastings associado com a densidade alvo  $f$  e a densidade condicional  $q$  produz uma cadeia de Markov  $(X^t)$  seguindo os passos do núcleo de transição, dado  $x^t$ :

1. Gere um valor candidato  $Y_t \sim q(y | x^t)$ ;
2. Escolha

$$x^{t+1} = \begin{cases} X^{t+1} = Y_t, & \text{com probabilidade } \rho(x^t, Y_t), \\ X^{t+1} = x^t, & \text{com probabilidade } 1 - \rho(x^t, Y_t), \end{cases}$$

onde

$$\rho(x, y) = \min \left\{ \frac{f(y) q(x|y)}{f(x) q(y|x)}, 1 \right\}.$$

A distribuição  $q$  é a distribuição do valor candidato (ou instrumental), e a probabilidade  $\rho(x, y)$  é a probabilidade de aceitação do algoritmo Metropolis-Hastings.

## Capítulo 3

# Metodologia

Neste capítulo, a formulação de três passos para o problema de detecção de um e dois pontos de mudança é detalhada. Considere uma série temporal na qual serão identificados um ou dois pontos de mudança. No primeiro passo, é escolhido qual é o melhor modelo para a série temporal analisada, um ponto de mudança ou dois pontos de mudança. A seleção é feita com um algoritmo de clusterização cujos centros são centros das funções de pertinência. O algoritmo usado é baseado numa rede neural auto-organizável, uma rede de Kohonen. Esta é inicializada com três estruturas, cada uma correspondendo a uma função de pertinência que, por sua vez, é relacionada a um centro do cluster. Quando este algoritmo é executado, tem-se que as funções de pertinência com maiores valores representam os centros da série temporal e as outras funções tem valores muito pequenos, próximos de zero. Se existem duas funções de pertinência, então o modelo para um ponto de mudança é usado. Se existem três, então o modelo para dois pontos de mudança é considerado. Este algoritmo é uma das diferenciações deste trabalho em relação ao trabalho anterior [D'Angelo et al. \(2011b\)](#).

Considerando dois pontos de mudança, o segundo passo consiste em transformar a série temporal dada em outra série com distribuição beta usando uma técnica de conjuntos *fuzzy* [Zadeh \(1965\)](#). Uma vez que o algoritmo gerou uma série temporal com função de distribuição de probabilidade beta, essa série pode ser usada na formulação Bayesiana para detectar os pontos de mudança. Neste trabalho, o algoritmo Metropolis-Hastings é usado devido a sua estratégia simples e poderosa. O objetivo do algoritmo Metropolis-Hastings [Gamerman \(1997\)](#) é construir uma cadeia de Markov com distribuição de equilíbrio  $\pi$  especificada. Um diagrama de blocos mostrando a metodologia é apresentado em [3.1](#).

### 3.1 Rede de Kohonen para clusterização e seleção de modelos

O algoritmo de clusterização é baseado numa rede neural auto-organizada [Kohonen \(1990\)](#) como mostrado na figura [3.2](#). Na rede de Kohonen foi utilizada somente uma entrada pelo fato da série temporal analisada ser de uma dimensão. Os pesos da rede,  $a_r$ , correspondem aos valores dos centros das funções de pertinência, figura [3.3](#), que por sua vez correspondem aos centros dos clusters. Estes centros determinam qual o tipo de modelo e onde estão os

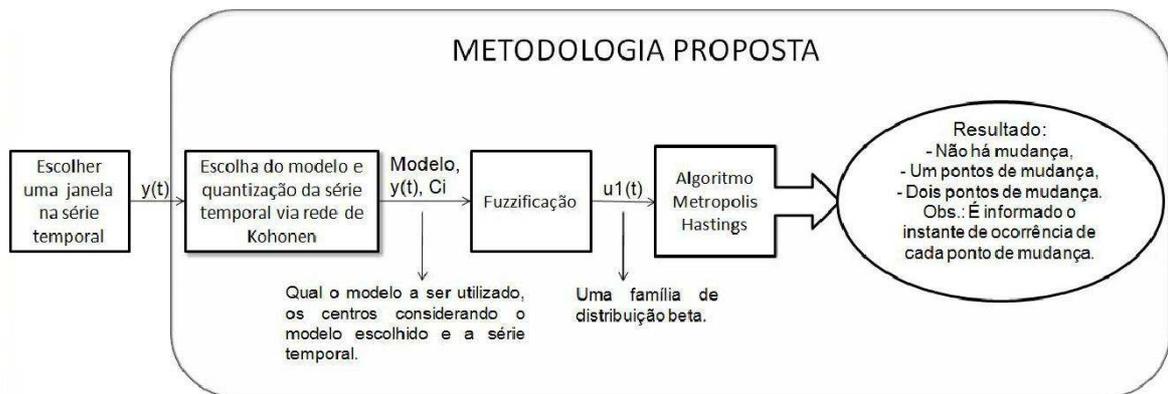


Figura 3.1: Diagrama de blocos da metodologia proposta

centros da série temporal analisada. O número de neurônios na segunda camada  $N_p$ , que é definido *a priori*, corresponde ao número de subconjuntos *fuzzy*. Neste trabalho são usados três neurônios igualmente espaçados entre os valores máximo e mínimo da série temporal modificada.

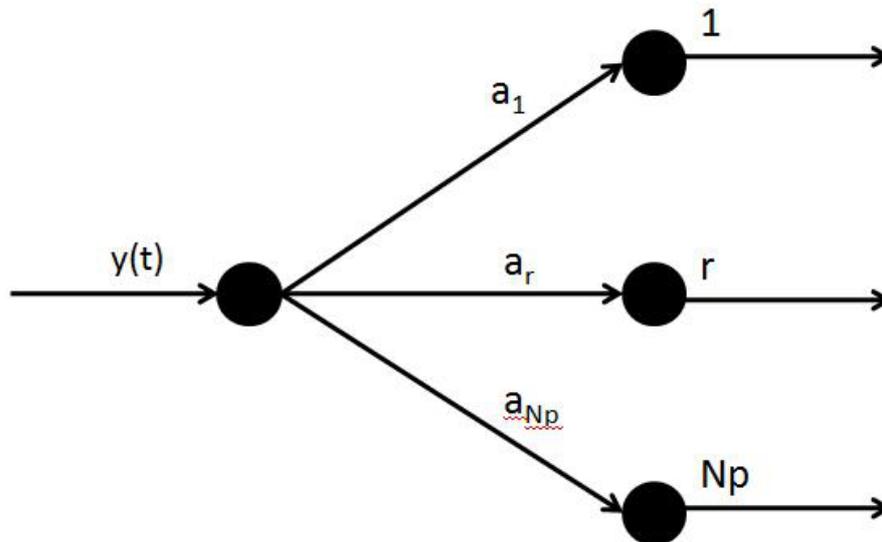


Figura 3.2: Rede neural auto-organizada para determinar os centros das funções de pertinência

O treinamento da rede neural auto-organizada, figura 3.2, é não supervisionado e competitivo Kohonen (1990). Somente o peso do neurônio vencedor é ajustado. Ao final do treinamento, aqueles neurônios que tiveram um baixo índice de desempenho, isto é, que venceram poucas vezes, são desconsiderados. Aqueles neurônios que restaram, determinam o número de funções de pertinência adequado,  $N_p$ , e os respectivos centros.

O treinamento da rede neural é feito conforme os passos abaixo:

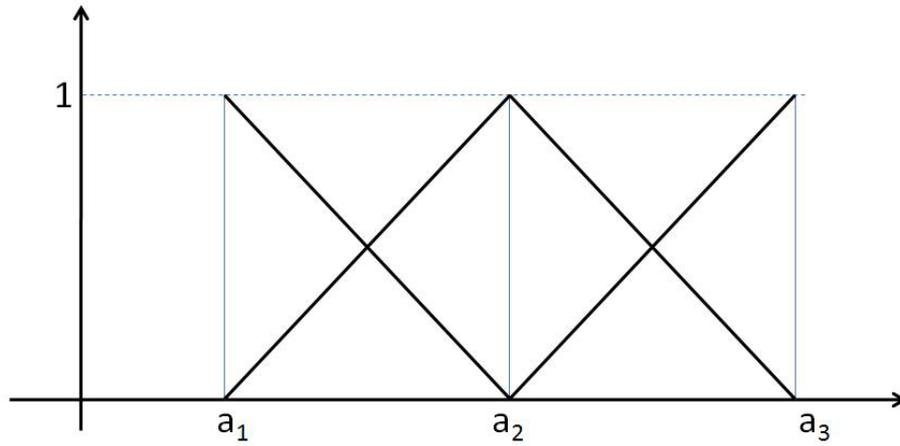


Figura 3.3: Funções de Pertinência

## 1. Inicialização:

a) pesos de  $a_r$ :

$$a_1 = \min_i x_i \quad (3.1)$$

$$a_r = a_{(r-1)} + \Delta_i \quad (3.2)$$

b) Índice de desempenho:  $Id(r) = 0$ , para  $r = 1, 2, 3, \dots, Np$ . Este procedimento de inicialização geralmente proporciona uma convergência mais rápida que a inicialização aleatória;

## 2. Processo iterativo:

Para  $epoca = 1$  até  $N_{max}$ Para  $t = 1$  até tamanho da janela

a) Apresente um padrão  $t$  à rede e atualize o peso da conexão do neurônio vencedor da seguinte forma:

$$a_L(t+1) = a_L(t) + \alpha(t) \cdot [y(t) - a_L(t)] \quad (3.3)$$

onde  $L$  é o índice do neurônio vencedor, que é aquele cujo peso da conexão possui o valor mais próximo de  $y(t)$ , ou seja:

$$L = \arg \left\{ \min_r \left| y(t) - a_r \right| \right\} \quad (3.4)$$

b) reduzir o passo  $\alpha(t)$  linearmente por um fator multiplicativo 0.01;

c) atualize o índice de desempenho do neurônio vencedor, fazendo:

$$Id(L) = Id(L) + 1 \quad (3.5)$$

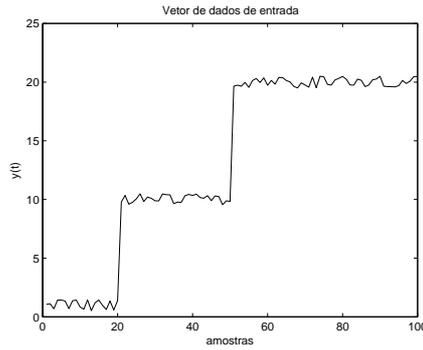


Figura 3.4: Série temporal com  $p_1 = 1$ ,  $p_2 = 10$ , e  $p_3 = 20$  fixados,  $\epsilon(t) \sim U(0, 1)$ ,  $m_1 = 20$ ,  $m_2 = 50$  e 100 amostras.

3. Elimine todos os neurônios cujo valor de  $Id$  seja menor que um limiar  $N$ , inteiro positivo definido a priori. Seja  $Nne$  o número de neurônios eliminados. Portanto, o número de subconjuntos *fuzzy* é:

$$Np = Np^0 - Nne \quad (3.6)$$

4. Fim.

Terminado a execução do algoritmo anterior, pode-se definir o número de funções de pertinência e seus respectivos centros. Isto é, definir qual o modelo será usado e os centros dos clusters para o passo 2 do algoritmo de detecção de ponto de mudança.

### 3.2 Transformação da série temporal - Fuzzificação

O segundo passo consiste em transformar a série temporal dada numa série com distribuição beta usando uma técnica de conjuntos fuzzy [Zadeh \(1965\)](#). Para ilustrar como isso é feito, considere a série temporal:

$$y(t) = \begin{cases} p_1 + 0.1 * \epsilon(t) - 0.1 * \epsilon(t - 1), & \text{se } t \leq m_1 \\ p_2 + 0.1 * \epsilon(t) - 0.1 * \epsilon(t - 1), & \text{se } t \geq m_2 \\ p_3 + 0.1 * \epsilon(t) - 0.1 * \epsilon(t - 1), & \text{se } t > m_2 \end{cases} \quad (3.7)$$

onde  $p_1$  é o primeiro ponto de operação (o valor médio antes do primeiro ponto de mudança),  $p_2$  é o segundo ponto de operação (o valor médio depois do primeiro ponto de mudança),  $p_3$  é o terceiro ponto de operação (o valor médio depois do segundo ponto de mudança)  $\epsilon(t)$  é um sinal de ruído com distribuição  $\pi(\cdot)$  e  $m_1$  e  $m_2$  são os pontos de mudança. A figura 3.4 mostra a série temporal  $y(t)$  com  $p_1 = 1$ ,  $p_2 = 10$ , e  $p_3 = 20$  fixos,  $\epsilon(t) \sim U(0, 1)$  (distribuição uniforme no intervalo  $[0, 1]$ ),  $m_1 = 20$ ,  $m_2 = 50$  e 100 amostras.

O algoritmo fuzzy proposto para transformar a série temporal original na série modificada é descrito abaixo:

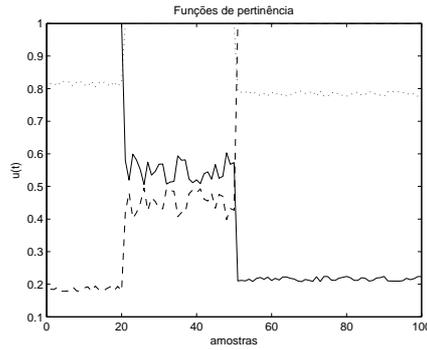


Figura 3.5: Funções de pertinência,  $\mu_1(t)$ (—),  $\mu_2(t)$ (⋯) e  $\mu_3(t)$ (- - -).

1. Entre com a série temporal  $y(t)$ ;
2. Encontre  $C_i, i = 1, 2, 3$ , os elementos do conjunto de centros de clusters para  $y(t)$  usando a rede Kohonen (considerando, por exemplo, a série temporal em (3.7)).
3. Calcule o grau de pertinência fuzzy para cada amostra da série temporal conforme equações 3.8, 3.9 e 3.10,  $y(t)$ , em relação a cada centro  $C_i$  (como ilustrado na figura 3.5 considerando, por exemplo, a série temporal em (3.7)).

$$\mu_1(t) = 1 - \frac{(y(t) - C_1)^2}{(y(t) - C_1)^2 + (y(t) - C_2)^2 + (y(t) - C_3)^2} \quad (3.8)$$

$$\mu_2(t) = 1 - \frac{(y(t) - C_2)^2}{(y(t) - C_1)^2 + (y(t) - C_2)^2 + (y(t) - C_3)^2} \quad (3.9)$$

$$\mu_3(t) = 1 - \frac{(y(t) - C_3)^2}{(y(t) - C_1)^2 + (y(t) - C_2)^2 + (y(t) - C_3)^2} \quad (3.10)$$

Além disso, é claro que as distribuições de  $\mu_1(t)$ , como mostrado em [D'Angelo et al. \(2011b\)](#), são limitadas no intervalo  $[0, 1]$ , e usando a divergência de Kullback-Leibler [Kullback e Leibler \(1951\)](#) pode-se concluir que as distribuições de  $\mu_1(t)$  formam uma família de distribuição beta com parâmetros de entrada diferentes: para  $\mu_1(t)$ ,  $t \leq m_1$ , obtêm-se uma distribuição beta  $(a, b)$ , para  $0 \leq t < m_1 = 20$ , distribuição beta  $(c, d)$ , para  $m_1 = 20 \leq t < m_2 = 50$ , ou distribuição beta  $(e, f)$ , para  $t \geq m_2 = 50$ . Este teste empírico foi feito para várias séries temporais com diferentes distribuições de probabilidade, sempre levando à mesma família de distribuições beta depois da técnica de clusterização.

### 3.3 Formulação do Metropolis Hastings para dois pontos de mudança

Uma vez que a técnica de clusterização transforma a série original, com uma distribuição de probabilidade arbitrária, em uma nova série temporal  $\mu_1(t)$  com uma função de distribuição de probabilidade beta, este modelo estatístico fixado pode ser considerado na formulação bayesiana para detectar os pontos de mudança na série temporal transformada (terceiro passo). O algoritmo Metropolis-Hastings [Gamerman \(1997\)](#) é usado para realizar a detecção dos pontos de mudança devido a sua estratégia simples e eficiente. Este algoritmo constrói uma cadeia de Markov que tem uma distribuição de equilíbrio especificada  $\pi$ .

Define-se uma cadeia de Markov da seguinte forma: Se  $X_i = x_i$ , então escolha um valor candidato  $Y$  de uma distribuição com densidade  $f_{Y|X}(y) = q(x_i, y)$ . A função  $q$  é conhecida como o núcleo de transição da cadeia. É uma função de duas variáveis, o estado atual da cadeia  $x_i$  e o valor candidato  $y$ . Para cada  $x_i$ , a função  $q(x_i, y)$  é uma densidade que uma função de  $y$ .

O valor candidato  $y$  é então aceito ou rejeitado. A probabilidade de aceite é

$$\alpha(x, y) = \min \left( 1, \frac{\pi(y) q(y, x_i)}{\pi(x_i) q(x_i, y)} \right) \quad (3.11)$$

Se o valor candidato é aceito, então  $X_{i+1} = Y$ , de outra forma  $X_{i+1} = X_i$ . Deste modo, se o valor candidato é rejeitado, a cadeia de Markov tem uma *repetição* na sequência. É possível mostrar que sobre condições gerais a sequência  $X_0, X_1, X_2, \dots$  é uma cadeia de Markov com distribuição de equilíbrio  $\pi$ .

Em termos práticos, o algoritmo Metropolis-Hastings pode ser especificado pelos seguintes passos:

---

#### Algoritmo Metropolis-Hastings

1. Escolha um valor inicial  $x_0$ , o numero de iterações,  $R$ , e faça o contador de iterações  $r = 0$ ;
2. Gere um valor candidato  $y$  usando a distribuição de referência dada por  $q(x_r, y)$ ;
3. Calcule a probabilidade de aceite in [\(3.11\)](#) e gere  $u \sim U(0, 1)$ ;
4. Calcule o novo valor para o estado atual:

$$x^{t+1} = \begin{cases} y, & \text{if } \alpha(x, y) \geq u, \\ x^t, & \text{senão} \end{cases}$$

5. If  $r < R$ , retorne para o passo 2. Senão pare.
-

Note que, como discutido previamente, a técnica de clusterização gera uma série temporal transformada com a seguinte distribuição:

$$\begin{aligned} y(t) &\sim \text{beta}(a, b), \text{ for } t = 1, \dots, m_1 \\ y(t) &\sim \text{beta}(c, d), \text{ for } t = m_1 + 1, \dots, m_2 \\ y(t) &\sim \text{beta}(e, f), \text{ for } t = m_2 + 1, \dots, n \end{aligned}$$

Os parâmetros a serem estimados para o algoritmo Metropolis-Hastings são  $a, b, c, d, e, f$  e os pontos de mudança  $m_1$  e  $m_2$ . Neste tipo de algoritmo, a escolha dos valores iniciais é feita, normalmente, usando distribuições pouco informativas, por exemplo:

$$\begin{aligned} a &\sim \text{gamma}(0.1, 0.1) \\ b &\sim \text{gamma}(0.1, 0.1) \\ c &\sim \text{gamma}(0.1, 0.1) \\ d &\sim \text{gamma}(0.1, 0.1) \\ e &\sim \text{gamma}(0.1, 0.1) \\ f &\sim \text{gamma}(0.1, 0.1) \\ m_1 &\sim U\{1, 2, \dots, m_2 - 1\}, \text{ com } p(m) = \frac{1}{m_2 - 1} \\ m_2 &\sim U\{m_1 + 1, \dots, n\}, \text{ com } p(m) = \frac{1}{n - m_2 + 1} \end{aligned}$$

Estas distribuições, com parâmetros 0.1, foram escolhidas com o propósito de expandir o espaço paramétrico.

A função de verossimilhança de  $y$  em relação aos parâmetros  $m, a, b, c, d, e$  e  $f$  é dada por:

$$\begin{aligned} f(y | a, b, c, d, e, f) &\sim \prod_{i=1}^{m_1} \frac{\Gamma(a+b)}{\Gamma(a)\Gamma(b)} y_i^{a-1} (1-y_i)^{b-1} \\ &\quad \prod_{i=m_1+1}^{m_2} \frac{\Gamma(c+d)}{\Gamma(c)\Gamma(d)} y_i^{c-1} (1-y_i)^{d-1} \\ &\quad \prod_{i=m_2+1}^n \frac{\Gamma(e+f)}{\Gamma(e)\Gamma(f)} y_i^{e-1} (1-y_i)^{f-1} \end{aligned}$$

Sendo  $G$  a função *Gamma*,  $G(k) \sim \text{gamma}(k, 1)$ .

Os parâmetros  $a, b, c, d, e, f, m_1$  e  $m_2$  são gerados pelas seguintes funções:

1. Para o parâmetro  $a$ :

$$\begin{aligned}
&= \frac{\Pi(a^*)q(a^*, a)}{\Pi(a)q(a, a^*)} \\
&= \frac{\Pi(a^*)}{\Pi(a)} \frac{f(y|a^*, b, c, d, e, f, m_1, m_2)\Pi(a^*)\Pi(b)\Pi(c)\Pi(d)\Pi(e)\Pi(f)\Pi(m_1)\Pi(m_2)}{f(y|a, b, c, d, e, f, m_1, m_2)\Pi(a)\Pi(b)\Pi(c)\Pi(d)\Pi(e)\Pi(f)\Pi(m_1)\Pi(m_2)} \\
&= \frac{\Pi(a^*)\Pi(a^*)}{\Pi(a)\Pi(a)} \frac{f(y|a^*, b, c, d, e, f, m_1, m_2)}{f(y|a, b, c, d, e, f, m_1, m_2)} \\
&= \frac{[\Pi(a^*)]^2 \prod_{i=1}^{m_1} \frac{\Gamma(a^*+b)}{\Gamma(a^*)\Gamma(b)} y_i^{a^*-1} (1-y_i)^{b-1} \prod_{i=m_1+1}^{m_2} \frac{\Gamma(c+d)}{\Gamma(c)\Gamma(d)} y_i^{c-1} (1-y_i)^{d-1}}{[\Pi(a)]^2 \prod_{i=1}^{m_1} \frac{\Gamma(a+b)}{\Gamma(a)\Gamma(b)} y_i^{a-1} (1-y_i)^{b-1} \prod_{i=m_1+1}^{m_2} \frac{\Gamma(c+d)}{\Gamma(c)\Gamma(d)} y_i^{c-1} (1-y_i)^{d-1}} \\
&\times \frac{\prod_{i=m_2+1}^n \frac{\Gamma(e+f)}{\Gamma(e)\Gamma(f)} y_i^{e-1} (1-y_i)^{f-1}}{\prod_{i=m_2+1}^n \frac{\Gamma(e+f)}{\Gamma(e)\Gamma(f)} y_i^{e-1} (1-y_i)^{f-1}} \\
&= \frac{[\Pi(a^*)]^2 \prod_{i=1}^{m_1} \frac{\Gamma(a^*+b)}{\Gamma(a^*)\Gamma(b)} y_i^{a^*-1}}{[\Pi(a)]^2 \prod_{i=1}^{m_1} \frac{\Gamma(a+b)}{\Gamma(a)\Gamma(b)} y_i^{a-1}} \\
&= \frac{\left\{ 0.1^{0.1} [\Gamma(0.1)]^{-1} a^{*0.1-1} e^{-0.1a^*} \right\}^2 \prod_{i=1}^{m_1} \frac{\Gamma(a^*+b)}{\Gamma(a^*)\Gamma(b)} y_i^{a^*-1}}{\left\{ 0.1^{0.1} [\Gamma(0.1)]^{-1} a^{0.1-1} e^{-0.1a} \right\}^2 \prod_{i=1}^{m_1} \frac{\Gamma(a+b)}{\Gamma(a)\Gamma(b)} y_i^{a-1}} \\
&= \left\{ \left[ \frac{a}{a^*} \right]^{0.9} e^{-0.1(a^*-a)} \right\}^2 \frac{\left[ \frac{\Gamma(a^*+b)}{\Gamma(a^*)\Gamma(b)} \right]^{m_1} \prod_{i=1}^{m_1} y_i^{a^*-1}}{\left[ \frac{\Gamma(a+b)}{\Gamma(a)\Gamma(b)} \right]^{m_1} \prod_{i=1}^{m_1} y_i^{a-1}} \\
&= \left\{ \left[ \frac{a}{a^*} \right]^{0.9} e^{-0.1(a^*-a)} \right\}^2 \left[ \frac{\Gamma(a^*+b)\Gamma(a)}{\Gamma(a+b)\Gamma(a^*)} \right]^{m_1} \prod_{i=1}^{m_1} y_i^{a^*-a}
\end{aligned} \tag{3.12}$$

2. Para o parâmetro  $b$ :

$$\begin{aligned}
&= \frac{\Pi(b^*)q(b^*, b)}{\Pi(b)q(b, b^*)} \\
&= \frac{\Pi(b^*)}{\Pi(b)} \frac{f(y|a, b^*, c, d, e, f, m_1, m_2)\Pi(a)\Pi(b^*)\Pi(c)\Pi(d)\Pi(e)\Pi(f)\Pi(m_1)\Pi(m_2)}{f(y|a, b, c, d, e, f, m_1, m_2)\Pi(a)\Pi(b)\Pi(c)\Pi(d)\Pi(e)\Pi(f)\Pi(m_1)\Pi(m_2)} \\
&= \frac{\Pi(b^*)\Pi(b^*)}{\Pi(b)\Pi(b)} \frac{f(y|a, b^*, c, d, e, f, m_1, m_2)}{f(y|a, b, c, d, e, f, m_1, m_2)} \\
&= \frac{[\Pi(b^*)]^2}{[\Pi(b)]^2} \frac{\prod_{i=1}^{m_1} \frac{\Gamma(a+b^*)}{\Gamma(a)\Gamma(b^*)} y_i^{a-1} (1-y_i)^{b^*-1} \prod_{i=m_1+1}^{m_2} \frac{\Gamma(c+d)}{\Gamma(c)\Gamma(d)} y_i^{c-1} (1-y_i)^{d-1}}{\prod_{i=1}^{m_1} \frac{\Gamma(a+b)}{\Gamma(a)\Gamma(b)} y_i^{a-1} (1-y_i)^{b-1} \prod_{i=m_1+1}^{m_2} \frac{\Gamma(c+d)}{\Gamma(c)\Gamma(d)} y_i^{c-1} (1-y_i)^{d-1}} \\
&\times \frac{\prod_{i=m_2+1}^n \frac{\Gamma(e+f)}{\Gamma(e)\Gamma(f)} y_i^{e-1} (1-y_i)^{f-1}}{\prod_{i=m_2+1}^n \frac{\Gamma(e+f)}{\Gamma(e)\Gamma(f)} y_i^{e-1} (1-y_i)^{f-1}} \\
&= \frac{[\Pi(b^*)]^2}{[\Pi(b)]^2} \frac{\prod_{i=1}^{m_1} \frac{\Gamma(a+b^*)}{\Gamma(a)\Gamma(b^*)} (1-y_i)^{b^*-1}}{\prod_{i=1}^{m_1} \frac{\Gamma(a+b)}{\Gamma(a)\Gamma(b)} (1-y_i)^{b-1}} \\
&= \frac{\left\{0.1^{0.1} [\Gamma(0.1)]^{-1} b^{*0.1-1} e^{-0.1b^*}\right\}^2 \prod_{i=1}^{m_1} \frac{\Gamma(a+b^*)}{\Gamma(a)\Gamma(b^*)} (1-y_i)^{b^*-1}}{\left\{0.1^{0.1} [\Gamma(0.1)]^{-1} b^{0.1-1} e^{-0.1b}\right\}^2 \prod_{i=1}^{m_1} \frac{\Gamma(a+b)}{\Gamma(a)\Gamma(b)} (1-y_i)^{b-1}} \\
&= \left\{ \left[ \frac{b}{b^*} \right]^{0.9} e^{-0.1(b^*-b)} \right\}^2 \frac{\left[ \frac{\Gamma(a+b^*)}{\Gamma(a)\Gamma(b^*)} \right]^{m_1} \prod_{i=1}^{m_1} (1-y_i)^{b^*-1}}{\left[ \frac{\Gamma(a+b)}{\Gamma(a)\Gamma(b)} \right]^{m_1} \prod_{i=1}^{m_1} (1-y_i)^{b-1}} \\
&= \left\{ \left[ \frac{b}{b^*} \right]^{0.9} e^{-0.1(b^*-b)} \right\}^2 \left[ \frac{\Gamma(a+b^*)\Gamma(b)}{\Gamma(a+b)\Gamma(b^*)} \right]^{m_1} \prod_{i=1}^{m_1} (1-y_i)^{b^*-b}
\end{aligned} \tag{3.13}$$

3. Para o parâmetro  $c$ :

$$\begin{aligned}
&= \frac{\Pi(c^*)q(c^*, c)}{\Pi(c)q(c, c^*)} \\
&= \frac{\Pi(c^*)}{\Pi(c)} \frac{f(y|a, b, c^*, d, e, f, m_1, m_2)\Pi(a)\Pi(b)\Pi(c^*)\Pi(d)\Pi(e)\Pi(f)\Pi(m_1)\Pi(m_2)}{f(y|a, b, c, d, e, f, m_1, m_2)\Pi(a)\Pi(b)\Pi(c)\Pi(d)\Pi(e)\Pi(f)\Pi(m_1)\Pi(m_2)} \\
&= \frac{\Pi(c^*)\Pi(c^*)}{\Pi(c)\Pi(c)} \frac{f(y|a, b, c^*, d, e, f, m_1, m_2)}{f(y|a, b, c, d, e, f, m_1, m_2)} \\
&= \frac{[\Pi(c^*)]^2}{[\Pi(c)]^2} \frac{\prod_{i=1}^{m_1} \frac{\Gamma(a+b)}{\Gamma(a)\Gamma(b)} y_i^{a-1} (1-y_i)^{b-1} \prod_{i=m_1+1}^{m_2} \frac{\Gamma(c^*+d)}{\Gamma(c^*)\Gamma(d)} y_i^{c^*-1} (1-y_i)^{d-1}}{\prod_{i=1}^{m_1} \frac{\Gamma(a+b)}{\Gamma(a)\Gamma(b)} y_i^{a-1} (1-y_i)^{b-1} \prod_{i=m_1+1}^{m_2} \frac{\Gamma(c+d)}{\Gamma(c)\Gamma(d)} y_i^{c-1} (1-y_i)^{d-1}} \\
&\times \frac{\prod_{i=m_2+1}^n \frac{\Gamma(e+f)}{\Gamma(e)\Gamma(f)} y_i^{e-1} (1-y_i)^{f-1}}{\prod_{i=m_2+1}^n \frac{\Gamma(e+f)}{\Gamma(e)\Gamma(f)} y_i^{e-1} (1-y_i)^{f-1}} \\
&= \frac{[\Pi(c^*)]^2}{[\Pi(c)]^2} \frac{\prod_{i=m_1+1}^{m_2} \frac{\Gamma(c^*+d)}{\Gamma(c^*)\Gamma(d)} y_i^{c^*-1}}{\prod_{i=m_1+1}^{m_2} \frac{\Gamma(c+d)}{\Gamma(c)\Gamma(d)} y_i^{c-1}} \\
&= \frac{\left\{ 0.1^{0.1} [\Gamma(0.1)]^{-1} c^{*0.1-1} e^{-0.1c^*} \right\}^2 \prod_{i=m_1+1}^{m_2} \frac{\Gamma(c^*+d)}{\Gamma(c^*)\Gamma(d)} y_i^{c^*-1}}{\left\{ 0.1^{0.1} [\Gamma(0.1)]^{-1} c^{0.1-1} e^{-0.1c} \right\}^2 \prod_{i=m_1+1}^{m_2} \frac{\Gamma(c+d)}{\Gamma(c)\Gamma(d)} y_i^{c-1}} \\
&= \left\{ \left[ \frac{c}{c^*} \right]^{0.9} e^{-0.1(c^*-c)} \right\}^2 \frac{\left[ \frac{\Gamma(c^*+d)}{\Gamma(c^*)\Gamma(d)} \right]^{m_2-m_1-1} \prod_{i=m_1+1}^{m_2} y_i^{c^*-1}}{\left[ \frac{\Gamma(c+d)}{\Gamma(c)\Gamma(d)} \right]^{m_2-m_1-1} \prod_{i=m_1+1}^{m_2} y_i^{c-1}} \\
&= \left\{ \left[ \frac{c}{c^*} \right]^{0.9} e^{-0.1(c^*-c)} \right\}^2 \left[ \frac{\Gamma(c^*+d)\Gamma(c)}{\Gamma(c+d)\Gamma(c^*)} \right]^{m_2-m_1} \prod_{i=m_1+1}^{m_2} y_i^{c^*-c}
\end{aligned} \tag{3.14}$$

4. Para o parâmetro  $d$ :

$$\begin{aligned}
&= \frac{\Pi(d^*)q(d^*, d)}{\Pi(d)q(d, d^*)} \\
&= \frac{\Pi(d^*)}{\Pi(d)} \frac{f(y|a, b, c, d^*, e, f, m_1, m_2)\Pi(a)\Pi(b)\Pi(c)\Pi(d^*)\Pi(e)\Pi(f)\Pi(m_1)\Pi(m_2)}{f(y|a, b, c, d, e, f, m_1, m_2)\Pi(a)\Pi(b)\Pi(c)\Pi(d)\Pi(e)\Pi(f)\Pi(m_1)\Pi(m_2)} \\
&= \frac{\Pi(d^*)\Pi(d^*)}{\Pi(d)\Pi(d)} \frac{f(y|a, b, c, d^*, e, f, m_1, m_2)}{f(y|a, b, c, d, e, f, m_1, m_2)} \\
&= \frac{[\Pi(d^*)]^2}{[\Pi(d)]^2} \frac{\prod_{i=1}^{m_1} \frac{\Gamma(a+b)}{\Gamma(a)\Gamma(b)} y_i^{a-1} (1-y_i)^{b-1} \prod_{i=m_1+1}^{m_2} \frac{\Gamma(c+d^*)}{\Gamma(c)\Gamma(d^*)} y_i^{c-1} (1-y_i)^{d^*-1}}{\prod_{i=1}^{m_1} \frac{\Gamma(a+b)}{\Gamma(a)\Gamma(b)} y_i^{a-1} (1-y_i)^{b-1} \prod_{i=m_1+1}^{m_2} \frac{\Gamma(c+d)}{\Gamma(c)\Gamma(d)} y_i^{c-1} (1-y_i)^{d-1}} \\
&\times \frac{\prod_{i=m_2+1}^n \frac{\Gamma(e+f)}{\Gamma(e)\Gamma(f)} y_i^{e-1} (1-y_i)^{f-1}}{\prod_{i=m_2+1}^n \frac{\Gamma(e+f)}{\Gamma(e)\Gamma(f)} y_i^{e-1} (1-y_i)^{f-1}} \\
&= \frac{[\Pi(d^*)]^2}{[\Pi(d)]^2} \frac{\prod_{i=m_1+1}^{m_2} \frac{\Gamma(c+d^*)}{\Gamma(c)\Gamma(d^*)} (1-y_i)^{d^*-1}}{\prod_{i=m_1+1}^{m_2} \frac{\Gamma(c+d)}{\Gamma(c)\Gamma(d)} (1-y_i)^{d-1}} \\
&= \frac{\left\{0.1^{0.1} [\Gamma(0.1)]^{-1} d^{*0.1-1} e^{-0.1d^*}\right\}^2 \prod_{i=m_1+1}^{m_2} \frac{\Gamma(c+d^*)}{\Gamma(c)\Gamma(d^*)} (1-y_i)^{d^*-1}}{\left\{0.1^{0.1} [\Gamma(0.1)]^{-1} d^{0.1-1} e^{-0.1d}\right\}^2 \prod_{i=m_1+1}^{m_2} \frac{\Gamma(c+d)}{\Gamma(c)\Gamma(d)} (1-y_i)^{d-1}} \\
&= \left\{ \left[ \frac{d}{d^*} \right]^{0.9} e^{-0.1(d^*-d)} \right\}^2 \frac{\left[ \frac{\Gamma(c+d^*)}{\Gamma(c)\Gamma(d^*)} \right]^{m_2-m_1-1} \prod_{i=m_1+1}^{m_2} (1-y_i)^{d^*-1}}{\left[ \frac{\Gamma(c+d)}{\Gamma(c)\Gamma(d)} \right]^{m_2-m_1-1} \prod_{i=m_1+1}^{m_2} (1-y_i)^{d-1}} \\
&= \left\{ \left[ \frac{d}{d^*} \right]^{0.9} e^{-0.1(d^*-d)} \right\}^2 \left[ \frac{\Gamma(c+d^*)\Gamma(d)}{\Gamma(c+d)\Gamma(d^*)} \right]^{m_2-m_1} \prod_{i=m_1+1}^{m_2} (1-y_i)^{d^*-d}
\end{aligned} \tag{3.15}$$

5. Para o parâmetro  $e$ :

$$\begin{aligned}
&= \frac{\Pi(e^*)q(e^*, e)}{\Pi(e)q(e, e^*)} \\
&= \frac{\Pi(e^*) f(y|a, b, c, d, e^*, f, m_1, m_2)\Pi(a)\Pi(b)\Pi(c)\Pi(d)\Pi(e^*)\Pi(f)\Pi(m_1)\Pi(m_2)}{\Pi(e) f(y|a, b, c, d, e, f, m_1, m_2)\Pi(a)\Pi(b)\Pi(c)\Pi(d)\Pi(e)\Pi(f)\Pi(m_1)\Pi(m_2)} \\
&= \frac{\Pi(e^*)\Pi(e^*) f(y|a, b, c, d, e^*, f, m_1, m_2)}{\Pi(e)\Pi(e) f(y|a, b, c, d, e, f, m_1, m_2)} \\
&= \frac{[\Pi(e^*)]^2 \prod_{i=1}^{m_1} \frac{\Gamma(a+b)}{\Gamma(a)\Gamma(b)} y_i^{a-1} (1-y_i)^{b-1} \prod_{i=m_1+1}^{m_2} \frac{\Gamma(c+d)}{\Gamma(c)\Gamma(d)} y_i^{c-1} (1-y_i)^{d-1}}{[\Pi(e)]^2 \prod_{i=1}^{m_1} \frac{\Gamma(a+b)}{\Gamma(a)\Gamma(b)} y_i^{a-1} (1-y_i)^{b-1} \prod_{i=m_1+1}^{m_2} \frac{\Gamma(c+d)}{\Gamma(c)\Gamma(d)} y_i^{c-1} (1-y_i)^{d-1}} \\
&\times \frac{\prod_{i=m_2+1}^n \frac{\Gamma(e^*+f)}{\Gamma(e^*)\Gamma(f)} y_i^{e^*-1} (1-y_i)^{f-1}}{\prod_{i=m_2+1}^n \frac{\Gamma(e+f)}{\Gamma(e)\Gamma(f)} y_i^{e-1} (1-y_i)^{f-1}} \\
&= \frac{[\Pi(e^*)]^2 \prod_{i=m_2+1}^n \frac{\Gamma(e^*+f)}{\Gamma(e^*)\Gamma(f)} y_i^{e^*-1}}{[\Pi(e)]^2 \prod_{i=m_2+1}^n \frac{\Gamma(e+f)}{\Gamma(e)\Gamma(f)} y_i^{e-1}} \\
&= \frac{\left\{ 0.1^{0.1} [\Gamma(0.1)]^{-1} e^{*0.1-1} e^{-0.1e^*} \right\}^2 \prod_{i=m_2+1}^n \frac{\Gamma(e^*+f)}{\Gamma(e^*)\Gamma(f)} y_i^{e^*-1}}{\left\{ 0.1^{0.1} [\Gamma(0.1)]^{-1} e^{0.1-1} e^{-0.1e} \right\}^2 \prod_{i=m_2+1}^n \frac{\Gamma(e+f)}{\Gamma(e)\Gamma(f)} y_i^{e-1}} \\
&= \left\{ \left[ \frac{e}{e^*} \right]^{0.9} e^{-0.1(e^*-e)} \right\}^2 \frac{\left[ \frac{\Gamma(e^*+f)}{\Gamma(e^*)\Gamma(f)} \right]^{n-m_2-1} \prod_{i=m_2+1}^n y_i^{e^*-1}}{\left[ \frac{\Gamma(e+f)}{\Gamma(e)\Gamma(f)} \right]^{n-m_2-1} \prod_{i=m_2+1}^n y_i^{e-1}} \\
&= \left\{ \left[ \frac{e}{e^*} \right]^{0.9} e^{-0.1(e^*-e)} \right\}^2 \left[ \frac{\Gamma(e^*+f)\Gamma(e)}{\Gamma(e+f)\Gamma(e^*)} \right]^{n-m_2} \prod_{i=m_2+1}^n y_i^{e^*-e}
\end{aligned} \tag{3.16}$$

6. Para o parâmetro  $f$ :

$$\begin{aligned}
&= \frac{\Pi(f^*)q(f^*, f)}{\Pi(f)q(f, f^*)} \\
&= \frac{\Pi(f^*)}{\Pi(f)} \frac{f(y|a, b, c, d, e, f^*, m_1, m_2)\Pi(a)\Pi(b)\Pi(c)\Pi(d)\Pi(e)\Pi(f^*)\Pi(m_1)\Pi(m_2)}{f(y|a, b, c, d, e, f, m_1, m_2)\Pi(a)\Pi(b)\Pi(c)\Pi(d)\Pi(e)\Pi(f)\Pi(m_1)\Pi(m_2)} \\
&= \frac{\Pi(f^*)\Pi(f^*)}{\Pi(f)\Pi(f)} \frac{f(y|a, b, c, d, e, f^*, m_1, m_2)}{f(y|a, b, c, d, e, f, m_1, m_2)} \\
&= \frac{[\Pi(f^*)]^2 \prod_{i=1}^{m_1} \frac{\Gamma(a+b)}{\Gamma(a)\Gamma(b)} y_i^{a-1} (1-y_i)^{b-1} \prod_{i=m_1+1}^{m_2} \frac{\Gamma(c+d)}{\Gamma(c)\Gamma(d)} y_i^{c-1} (1-y_i)^{d-1}}{[\Pi(f)]^2 \prod_{i=1}^{m_1} \frac{\Gamma(a+b)}{\Gamma(a)\Gamma(b)} y_i^{a-1} (1-y_i)^{b-1} \prod_{i=m_1+1}^{m_2} \frac{\Gamma(c+d)}{\Gamma(c)\Gamma(d)} y_i^{c-1} (1-y_i)^{d-1}} \\
&\times \frac{\prod_{i=m_2+1}^n \frac{\Gamma(e+f^*)}{\Gamma(e)\Gamma(f^*)} y_i^{e-1} (1-y_i)^{f^*-1}}{\prod_{i=m_2+1}^n \frac{\Gamma(e+f)}{\Gamma(e)\Gamma(f)} y_i^{e-1} (1-y_i)^{f-1}} \\
&= \frac{[\Pi(f^*)]^2 \prod_{i=m_2+1}^n \frac{\Gamma(e+f^*)}{\Gamma(e)\Gamma(f^*)} (1-y_i)^{f^*-1}}{[\Pi(f)]^2 \prod_{i=m_2+1}^n \frac{\Gamma(e+f)}{\Gamma(e)\Gamma(f)} (1-y_i)^{f-1}} \\
&= \frac{\left\{ 0.1^{0.1} [\Gamma(0.1)]^{-1} f^{*0.1-1} e^{-0.1f^*} \right\}^2 \prod_{i=m_2+1}^n \frac{\Gamma(e+f^*)}{\Gamma(e)\Gamma(f^*)} (1-y_i)^{f^*-1}}{\left\{ 0.1^{0.1} [\Gamma(0.1)]^{-1} f^{0.1-1} e^{-0.1f} \right\}^2 \prod_{i=m_2+1}^n \frac{\Gamma(e+f)}{\Gamma(e)\Gamma(f)} (1-y_i)^{f-1}} \\
&= \left\{ \left[ \frac{f}{f^*} \right]^{0.9} e^{-0.1(f^*-f)} \right\}^2 \frac{\left[ \frac{\Gamma(e+f^*)}{\Gamma(e)\Gamma(f^*)} \right]^{n-m_2-1} \prod_{i=m_2+1}^n (1-y_i)^{f^*-1}}{\left[ \frac{\Gamma(e+f)}{\Gamma(e)\Gamma(f)} \right]^{n-m_2-1} \prod_{i=m_2+1}^n (1-y_i)^{f-1}} \\
&= \left\{ \left[ \frac{f}{f^*} \right]^{0.9} e^{-0.1(f^*-f)} \right\}^2 \left[ \frac{\Gamma(e+f^*)\Gamma(f)}{\Gamma(e+f)\Gamma(f^*)} \right]^{n-m_2} \prod_{i=m_2+1}^n (1-y_i)^{f^*-f}
\end{aligned} \tag{3.17}$$

7. Para o parâmetro  $m_1$ :

$$\begin{aligned}
&= \frac{\Pi(m_1^*)q(m_1^*, m_1)}{\Pi(m_1)q(m_1, m_1^*)} \\
&= \frac{\Pi(m_1^*)}{\Pi(m_1)} \frac{f(y|a, b, c, d, e, f, m_1^*, m_2)\Pi(a)\Pi(b)\Pi(c)\Pi(d)\Pi(e)\Pi(f)\Pi(m_1^*)\Pi(m_2)}{f(y|a, b, c, d, e, f, m_1, m_2)\Pi(a)\Pi(b)\Pi(c)\Pi(d)\Pi(e)\Pi(f)\Pi(m_1)\Pi(m_2)} \\
&= \frac{\Pi(m_1^*)\Pi(m_1^*)}{\Pi(m_1)\Pi(m_1)} \frac{f(y|a, b, c, d, e, f, m_1^*, m_2)}{f(y|a, b, c, d, e, f, m_1, m_2)} \\
&= \frac{[\Pi(m_1^*)]^2 \prod_{i=1}^{m_1^*} \frac{\Gamma(a+b)}{\Gamma(a)\Gamma(b)} y_i^{a-1} (1-y_i)^{b-1} \prod_{i=m_1^*+1}^{m_2} \frac{\Gamma(c+d)}{\Gamma(c)\Gamma(d)} y_i^{c-1} (1-y_i)^{d-1}}{[\Pi(m_1)]^2 \prod_{i=1}^{m_1} \frac{\Gamma(a+b)}{\Gamma(a)\Gamma(b)} y_i^{a-1} (1-y_i)^{b-1} \prod_{i=m_1+1}^{m_2} \frac{\Gamma(c+d)}{\Gamma(c)\Gamma(d)} y_i^{c-1} (1-y_i)^{d-1}} \times \\
&\quad \frac{\prod_{i=m_2+1}^n \frac{\Gamma(e+f)}{\Gamma(e)\Gamma(f)} y_i^{e-1} (1-y_i)^{f-1}}{\prod_{i=m_2+1}^n \frac{\Gamma(e+f)}{\Gamma(e)\Gamma(f)} y_i^{e-1} (1-y_i)^{f-1}} \\
&= \frac{\left[ \frac{\Gamma(a+b)}{\Gamma(a)\Gamma(b)} \right]^{m_1^*} \left[ \frac{\Gamma(c+d)}{\Gamma(c)\Gamma(d)} \right]^{m_2-m_1^*} \prod_{i=1}^{m_1^*} y_i^{a-1} (1-y_i)^{b-1}}{\left[ \frac{\Gamma(a+b)}{\Gamma(a)\Gamma(b)} \right]^{m_1} \left[ \frac{\Gamma(c+d)}{\Gamma(c)\Gamma(d)} \right]^{m_2-m_1} \prod_{i=1}^{m_1} y_i^{a-1} (1-y_i)^{b-1}} \\
&\quad \times \frac{\prod_{i=m_1^*+1}^{m_2} y_i^{c-1} (1-y_i)^{d-1}}{\prod_{i=m_1+1}^{m_2} y_i^{c-1} (1-y_i)^{d-1}} \tag{3.18}
\end{aligned}$$

8. Para o parâmetro  $m_2$ :

$$\begin{aligned}
&= \frac{\Pi(m_2^*)q(m_2^*, m_1)}{\Pi(m_2)q(m_2, m_2^*)} \\
&= \frac{\Pi(m_2^*)}{\Pi(m_2)} \frac{f(y|a, b, c, d, e, f, m_1, m_2^*)\Pi(a)\Pi(b)\Pi(c)\Pi(d)\Pi(e)\Pi(f)\Pi(m_1)\Pi(m_2^*)}{f(y|a, b, c, d, e, f, m_1, m_2)\Pi(a)\Pi(b)\Pi(c)\Pi(d)\Pi(e)\Pi(f)\Pi(m_1)\Pi(m_2)} \\
&= \frac{\Pi(m_2^*)\Pi(m_2^*)}{\Pi(m_2)\Pi(m_2)} \frac{f(y|a, b, c, d, e, f, m_1, m_2^*)}{f(y|a, b, c, d, e, f, m_1, m_2)} \\
&= \frac{[\Pi(m_2^*)]^2 \prod_{i=1}^{m_1} \frac{\Gamma(a+b)}{\Gamma(a)\Gamma(b)} y_i^{a-1} (1-y_i)^{b-1} \prod_{i=m_1+1}^{m_2^*} \frac{\Gamma(c+d)}{\Gamma(c)\Gamma(d)} y_i^{c-1} (1-y_i)^{d-1}}{[\Pi(m_2)]^2 \prod_{i=1}^{m_1} \frac{\Gamma(a+b)}{\Gamma(a)\Gamma(b)} y_i^{a-1} (1-y_i)^{b-1} \prod_{i=m_1+1}^{m_2} \frac{\Gamma(c+d)}{\Gamma(c)\Gamma(d)} y_i^{c-1} (1-y_i)^{d-1}} \times \\
&\quad \frac{\prod_{i=m_2^*+1}^n \frac{\Gamma(e+f)}{\Gamma(e)\Gamma(f)} y_i^{e-1} (1-y_i)^{f-1}}{\prod_{i=m_2+1}^n \frac{\Gamma(e+f)}{\Gamma(e)\Gamma(f)} y_i^{e-1} (1-y_i)^{f-1}} \\
&= \frac{\left[ \frac{\Gamma(c+d)}{\Gamma(c)\Gamma(d)} \right]^{m_2^*-m_1} \left[ \frac{\Gamma(e+f)}{\Gamma(e)\Gamma(f)} \right]^{n-m_2^*}}{\left[ \frac{\Gamma(c+d)}{\Gamma(c)+\Gamma(d)} \right]^{m_2-m_1} \left[ \frac{\Gamma(e+f)}{\Gamma(e)\Gamma(f)} \right]^{n-m_2}} \\
&\quad \times \frac{\prod_{i=m_1+1}^{m_2^*} y_i^{c-1} (1-y_i)^{d-1} \prod_{i=m_2^*+1}^n y_i^{e-1} (1-y_i)^{f-1}}{\prod_{i=m_1+1}^{m_2} y_i^{c-1} (1-y_i)^{d-1} \prod_{i=m_2+1}^n y_i^{e-1} (1-y_i)^{f-1}} \tag{3.19}
\end{aligned}$$

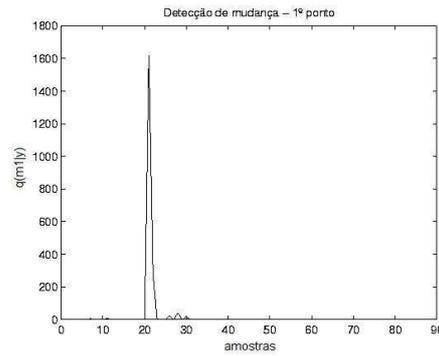


Figura 3.6: Resultado da metodologia proposta para  $p_1 = 1$ ,  $p_2 = 10$ ,  $p_3 = 20$   $e(t) \sim U(0, 1)$ , e  $m_1 = 20$ .

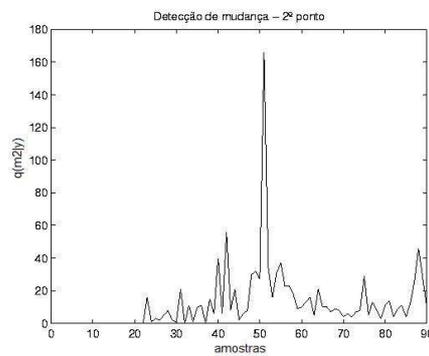


Figura 3.7: Resultado da metodologia proposta para  $p_1 = 1$ ,  $p_2 = 10$ ,  $p_3 = 20$   $e(t) \sim U(0, 1)$ , e  $m_2 = 50$ .

A análise final é executada da seguinte forma: os pontos de mudança,  $m_1$  e  $m_2$ , são obtidos pela verificação de onde o máximo de  $q(m_1, m_2 | y)$  ocorre, com exceção dos pontos extremos da distribuição (se o máximo ocorre em tais pontos, então não existe um ponto de mudança). A figura 3.6 e 3.7 mostram os resultados quando da aplicação da metodologia proposta para  $p_1 = 1$ ,  $p_2 = 10$ ,  $p_3 = 20$ ,  $e(t) \sim U(0, 1)$ ,  $m_1 = 20$  e  $m_2 = 50$ . A função  $q$ , na figura 3.6, pode ser interpretada como um *histograma da mudança* na série temporal no instante  $m_1$  e, na figura 3.7,  $m_2$ .

## Capítulo 4

# Simulações e estudo de casos

Neste capítulo, o algoritmo proposto será usado para detectar falhas numa aplicação prática que é o motor de indução. Utilizando um modelo de motor de indução proposto em [Baccarini et al. \(2004\)](#) simula-se falhas de curto-circuito incipiente no enrolamento de estator com diferentes gravidades e o algoritmo é usado em cada caso para detectar o ponto onde ocorreu o problema.

### 4.1 Modelo de uma máquina de indução e detecção de pontos de mudança

Os motores de indução são as máquinas elétricas mais importantes em aplicações industriais. Nestes equipamentos, a maior parte das falhas que surgem durante a sua vida útil são falhas relacionadas ao enrolamento de estator ([O'Donnell, 1985](#)), ([Albrecht et al., 1987](#)), ([Bonnett e Soukup, 1992](#)), ([Thorsen e Dalva, 1999](#)). Esta situação se torna ainda pior quando os motores são acionados por inversor ([Cruz e Cardoso, 2004](#)). O estator de uma máquina de indução está sujeito a desgastes causados por diversos fatores como sobrecarga térmica, vibrações mecânicas, e picos de tensão causados por variadores de velocidade. A deterioração do isolamento normalmente começa como uma falha de curto-circuito entre espiras do enrolamento do estator.

A detecção prévia da falha de curto-circuito no enrolamento de estator é muito importante, pois permite realizar um reparo mais rápido e com menor custo antes de se danificar completamente as bobinas do estator ([W.T.Thomson e Fenger, 2001](#)) ([Boqiang et al., 2003](#)).

Para simular as falhas de curto-circuito incipientes será usado um modelo genérico do motor de indução ([Baccarini et al. \(2004\)](#)), aplicável para qualquer velocidade dos eixos  $dq$ . Representando as correntes, tensões e fluxos pelas letras  $i$ ,  $v$  e  $\lambda$ , as resistências, indutâncias de dispersão e mútuas por  $r$ ,  $L_l$  e  $L_m$ , as fases  $a$ ,  $b$  e  $c$  pelos subíndices  $a$ ,  $b$  e  $c$ , os enrolamentos do estator e do rotor pelos subíndices  $s$  e  $r$ , tem-se as equações de tensão para o estator e o rotor:

$$[v_s] = [r_s][i_s] + \frac{d[\lambda_s]}{dt} \quad (4.1)$$

$$[v_r] = [r_r][i_r] + \frac{d[\lambda_r]}{dt} \quad (4.2)$$

Sendo

$$[v_s] = [v_{as1} \ v_{as2} \ v_{bs} \ v_{cs}]^T$$

$$[v_r] = [v_{ar} \ v_{br} \ v_{cr}]^T$$

$$[i_s] = [i_{as} \ i_{as} - i_f \ i_{bs} \ i_{cs}]^T$$

$$[i_r] = [i_{ar} \ i_{br} \ i_{cr}]^T$$

$$[\lambda_s] = [\lambda_{as1} \ \lambda_{as2} \ \lambda_{bs} \ \lambda_{cs}]^T$$

$$[\lambda_r] = [\lambda_{ar} \ \lambda_{br} \ \lambda_{cr}]^T$$

A Figura 4.1 representa os enrolamentos do estator do motor de indução, sendo  $as_2$  o número de espiras da fase  $a$  que estão curto-circuitadas.

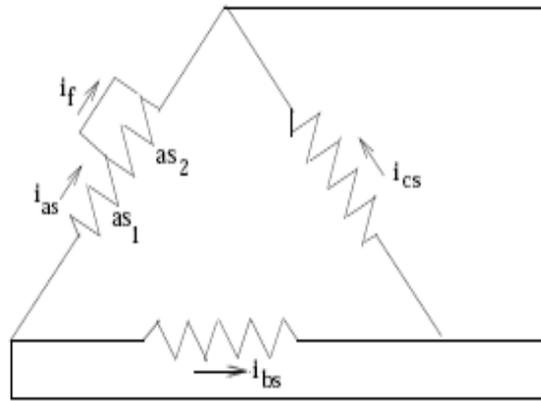


Figura 4.1: Representação dos enrolamentos

No modelo proposto por [Baccarini et al. \(2004\)](#) as tensões dos enrolamentos do estator são dadas por:

$$v_{ds} + \frac{2}{3}\mu r_s i_f \cos\theta = r_s i_{ds} + \frac{d\lambda_{ds}}{dt} - \omega \lambda_{qs} \quad (4.3)$$

$$v_{qs} + \frac{2}{3}\mu r_s i_f \sin\theta = r_s i_{qs} + \frac{d\lambda_{qs}}{dt} + \omega \lambda_{ds} \quad (4.4)$$

$$v_{0s} + \frac{1}{3}\mu r_s i_f = r_s i_{0s} + \frac{d\lambda_{0s}}{dt} \quad (4.5)$$

As equações do circuito do rotor são iguais às do modelo tradicional simétrico. Os fluxos de estator e de rotor nos eixos  $dq$ , são dados por:

$$\lambda_{ds} = L_s i_{ds} + L_m i_{dr} - \frac{2}{3}\mu L_s i_f \cos\theta \quad (4.6)$$

$$\lambda_{qs} = L_s i_{qs} + L_m i_{qr} - \frac{2}{3} \mu L_s i_f \text{sen}\theta \quad (4.7)$$

$$\lambda_{0s} = L_{ls} i_{0s} + \frac{\mu}{3} L_{ls} i_f \text{sen}\theta \quad (4.8)$$

$$\lambda_{dr} = L_r i_{dr} + L_m i_{ds} - \frac{2}{3} \mu L_m i_f \text{cos}\theta \quad (4.9)$$

$$\lambda_{qr} = L_r i_{qr} + L_m i_{qs} - \frac{2}{3} \mu L_m i_f \text{sen}\theta \quad (4.10)$$

A tensão e o fluxo induzidos nas espiras curtocircuitadas são dadas por:

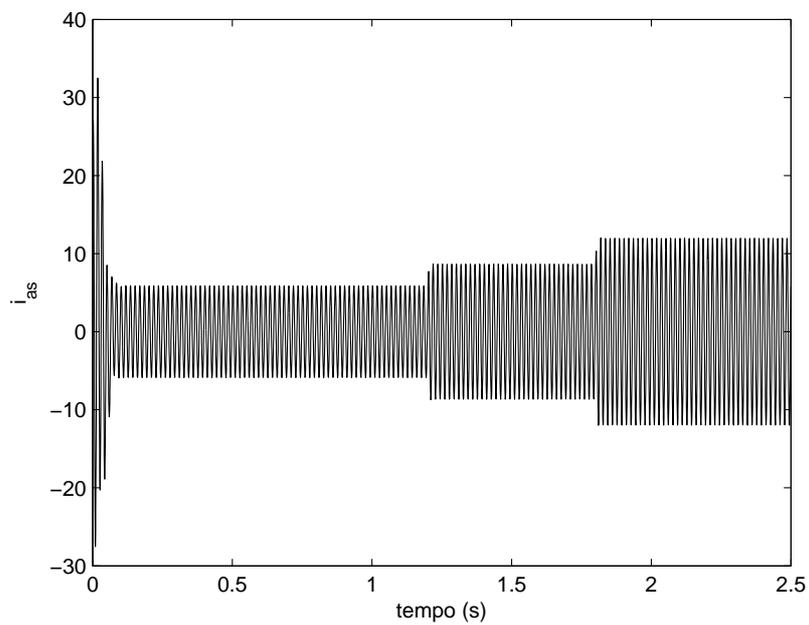
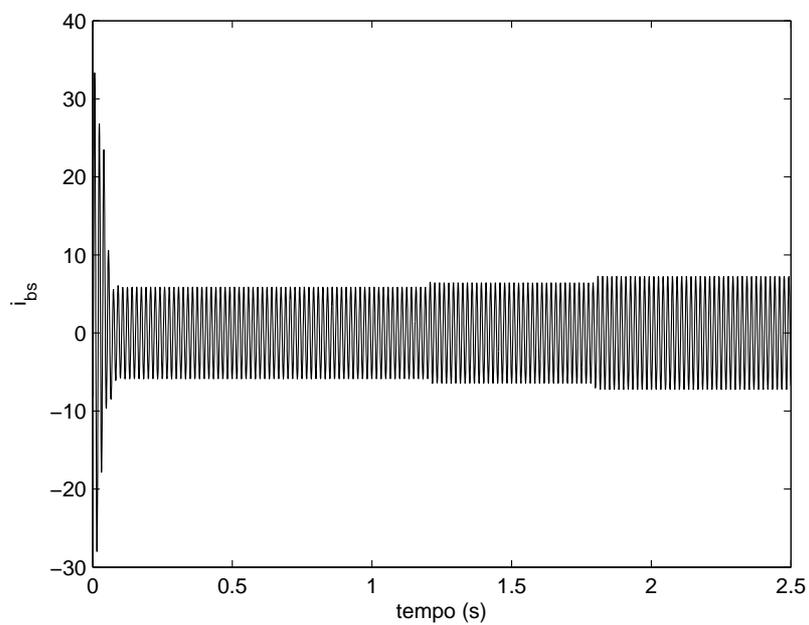
$$v_{as2} = \mu r_s (i_{ds} \text{cos}\theta + i_{qs} \text{sen}\theta - i_f) + \frac{d\lambda_{as2}}{dt} \quad (4.11)$$

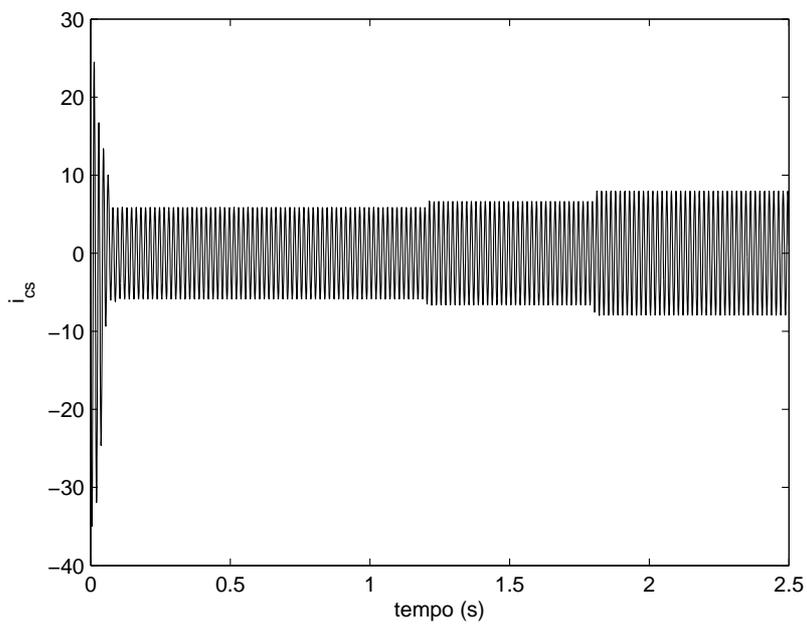
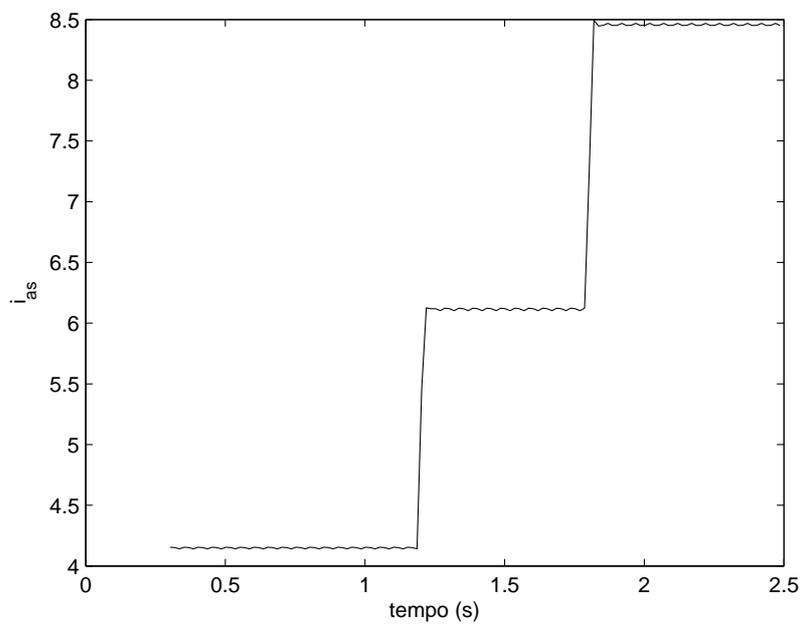
$$\begin{aligned} \lambda_{as2} = & \mu L_{ls} (i_{qs} \text{sen}\theta + i_{ds} \text{cos}\theta - i_f) + \mu L_m (i_{qs} \text{sen}\theta + \\ & + i_{ds} \text{cos}\theta + i_{qr} \text{sen}\theta + i_{dr} \text{cos}\theta - \frac{2}{3} \mu i_f) \end{aligned} \quad (4.12)$$

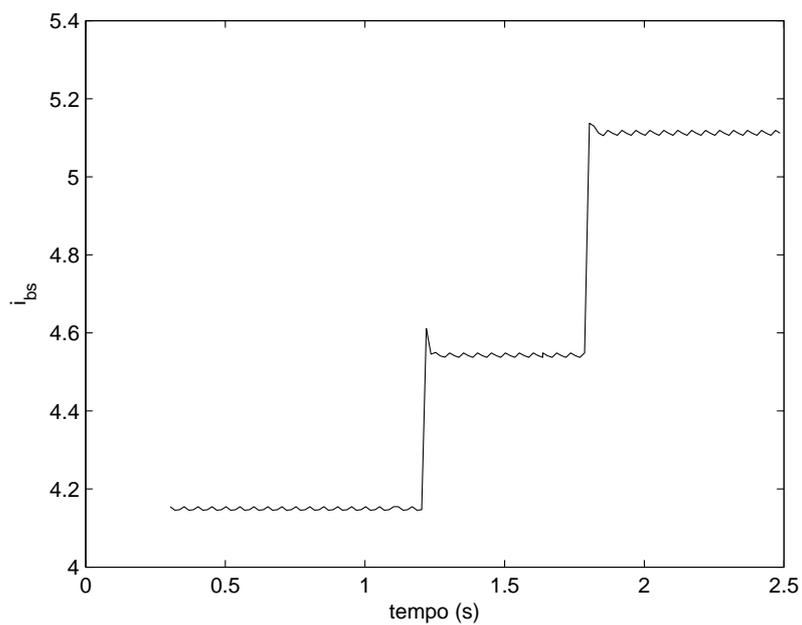
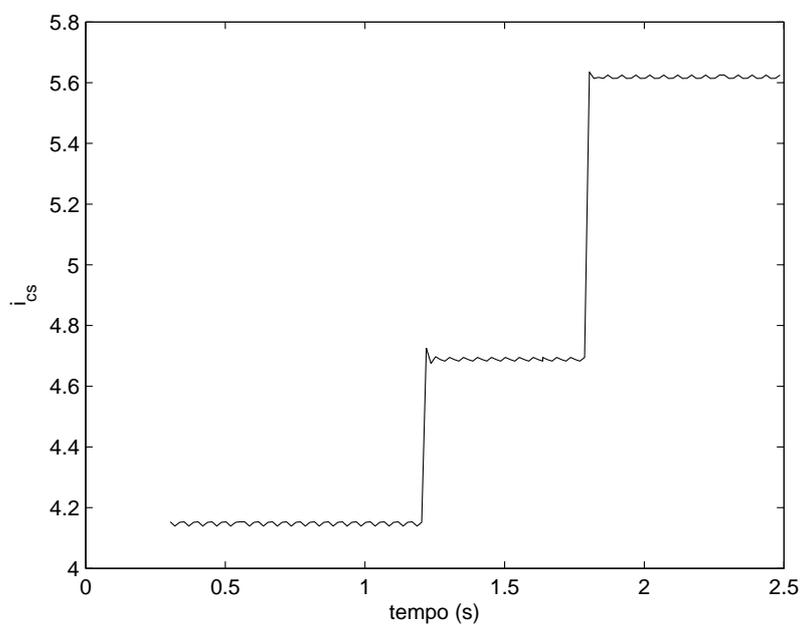
O conjugado é dado por:

$$T = \frac{3p}{2} L_m (i_{qs} i_{dr} - i_{ds} i_{qr}) - \frac{p}{2} \mu L_m i_f i_{qr} \quad (4.13)$$

Os resultados de simulação das correntes do estator da máquina de indução com 5% de espiras da fase *a* em curto-circuito para o primeiro ponto de falha, inseridos após 1.2s de simulação, e 10% de espiras da fase *a* em curto-circuito para o segundo ponto de falha, inseridos após 1.8s de simulação. Estes resultados são ilustrados nas Figuras 4.2–4.4. Os valores *rms* das correntes são ilustrados nas Figuras 4.5–4.7. Observe que quando ocorre um curto-circuito na fase *a*, o aumento da corrente de fase *a* é maior que as correntes das fases *b* e *c*.

Figura 4.2: Corrente da fase  $a$ .Figura 4.3: Corrente da fase  $b$ .

Figura 4.4: Corrente da fase  $c$ .Figura 4.5: Corrente da fase  $a$  (rms).

Figura 4.6: Corrente da fase *b* (rms).Figura 4.7: Corrente da fase *c* (rms).

## 4.2 Aplicação da metodologia proposta

Nesta seção apresentam-se os resultados obtidos com as correntes geradas pelo modelo simulando uma falha de curto circuito entre espiras na fase  $a$ . As correntes nas três fases,  $i_{as}$ ,  $i_{bs}$ ,  $i_{cs}$ , serão analisadas e para determinar em qual dos enrolamentos está o defeito deve-se usar as regras abaixo:

**SE**  $m(\bar{i}_{as}) > m(\bar{i}_{bs}) \cong m(\bar{i}_{cs})$  **ENTÃO** falha na fase  $a$ ;

**SE**  $m(\bar{i}_{bs}) > m(\bar{i}_{as}) \cong m(\bar{i}_{cs})$  **ENTÃO** falha na fase  $b$ ;

**SE**  $m(\bar{i}_{cs}) > m(\bar{i}_{as}) \cong m(\bar{i}_{bs})$  **ENTÃO** falha na fase  $c$ ;

**SE**  $m(\bar{i}_{bs}) \cong m(\bar{i}_{cs}) \cong m(\bar{i}_{as})$  **ENTÃO** não há falha.

Sendo  $m(\nu)$  a probabilidade de mudança na série temporal da variável  $\nu$ . Estas regras, além de definir em qual enrolamento ocorreu a falha, também identificam casos em que houve apenas mudança de ponto de operação do motor, onde a mudança das correntes são balanceadas.

Foi simulado um motor de indução conectado em estrela com 2% de espiras da fase  $a$  do enrolamento de estator em curto para a primeira falha e 5% das espiras da fase  $a$  do enrolamento de estator em curto para a segunda falha. A simulação foi obtida usando o modelo descrito na seção 4.1. Os resultados da simulação são mostrados nas figuras 4.8 – 4.22.

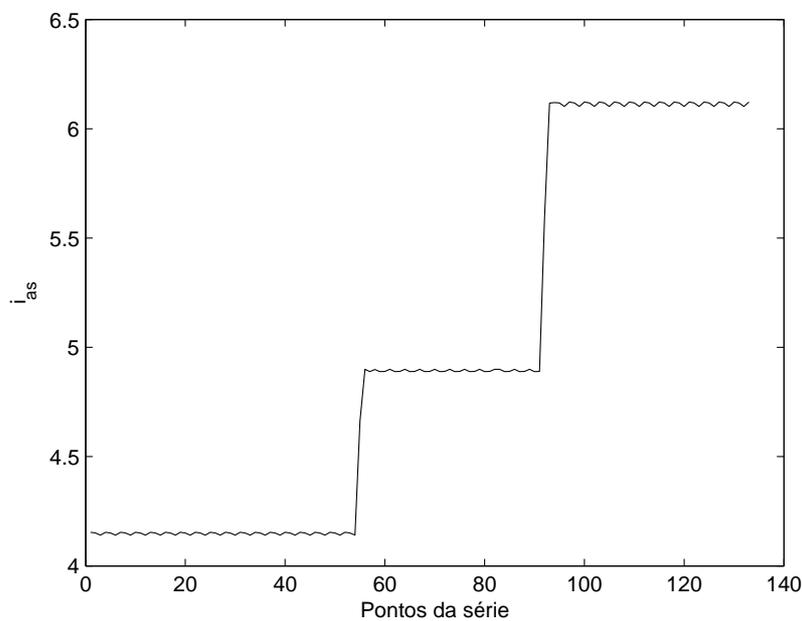


Figura 4.8: Corrente da fase  $a$  (rms): 2% e 5% de espiras em curto na fase  $a$

Pode-se notar que as falhas foram detectadas nos momentos corretos. Para a primeira

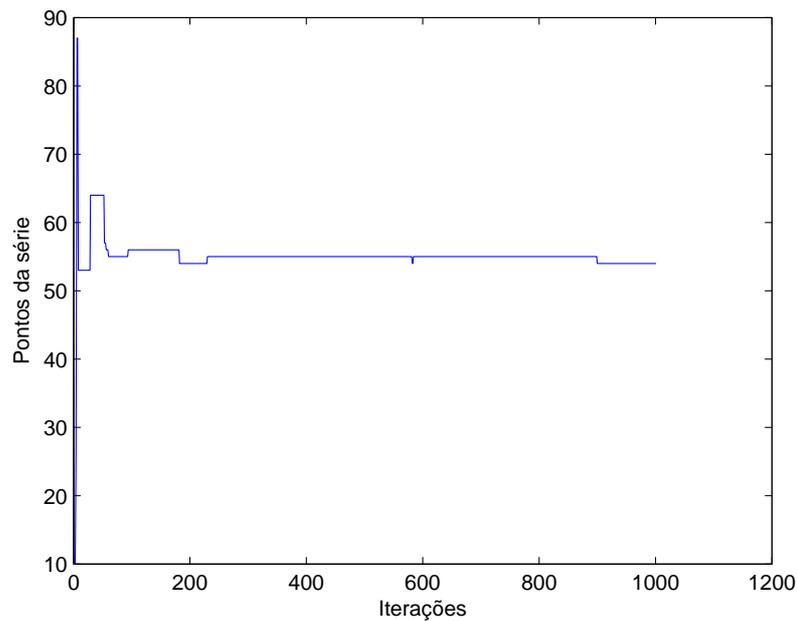


Figura 4.9: Evolução do parâmetro  $m_1$  para a corrente da fase  $a$

falha que ocorre por volta da amostra de número 55 da série, vê-se claramente que o número de ocorrências, ou probabilidade, indicando este ponto no pareto da fase A é muito maior que o numero de ocorrências para as fases B e C. Desta forma utilizando-se as regras descritas no início desta seção conclui-se que a falha ocorreu no fase A. Analisando agora o segundo ponto de mudança, percebe-se que esta diferença entre o numero de ocorrências para as fases A, B e C existem também, sendo o numero de ocorrências no pareto da fase A maior que o pareto das outras fases. Observa-se também que esta diferença é menor. O caso em que não existe falha também foi simulado e durante a etapa de identificação dos centros pelo algoritmo de Kohonen, foi detectado corretamente que não havia pontos de mudança na série e a rotina específica de detecção de falhas não foi usada. Durante os testes com diferentes intensidades de falha, isto é, com diferentes números de espiras em curto, foi percebido que para valores muito pequenos, cerca de 0.1% de espiras em curto, a metodologia proposta não apresentou um bom resultado como é mostrado nas figuras 4.23 – 4.32. Neste caso, nota-se que houve a identificação correta dos pontos de mudança para o corrente da fase A, porém o mesmo não aconteceu para a fase B em que foi identificado somente o segundo ponto de mudança, figuras 4.28 – 4.32. Os resultados da análise da corrente da fase C foram análogos aos da fase B. Mesmo para a fase A, em que os pontos de falha foram identificados corretamente, se observarmos a figura que mostra a convergência do parametro  $m_1$  pode-se notar que existe grande variação, o que indica dificuldade na identificação do ponto de falha.

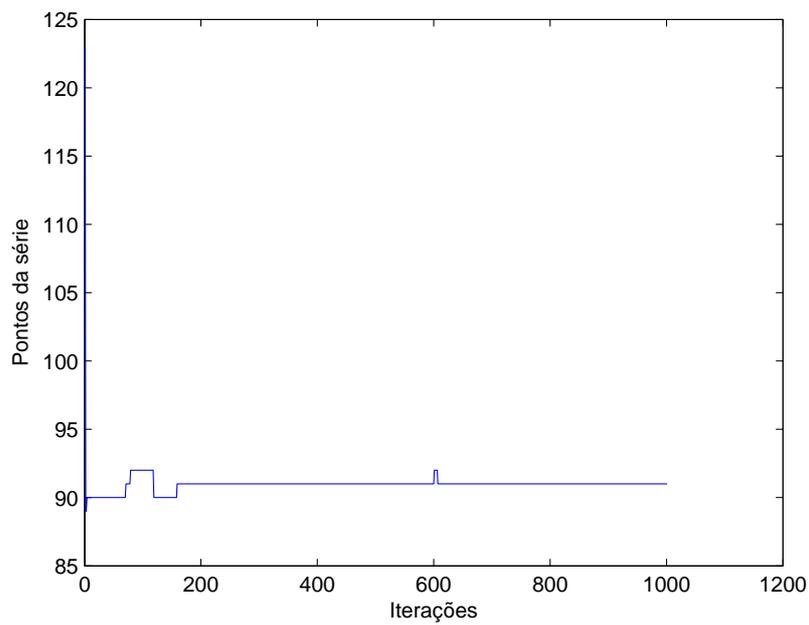


Figura 4.10: Evolução do parâmetro  $m_2$  para a corrente da fase  $a$

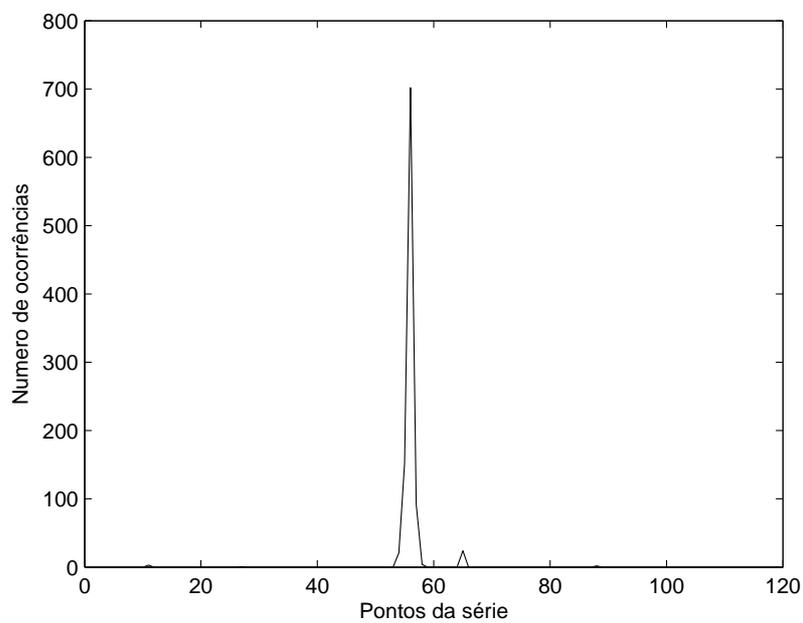


Figura 4.11: Histograma do primeiro ponto de mudança para a corrente da fase  $a$

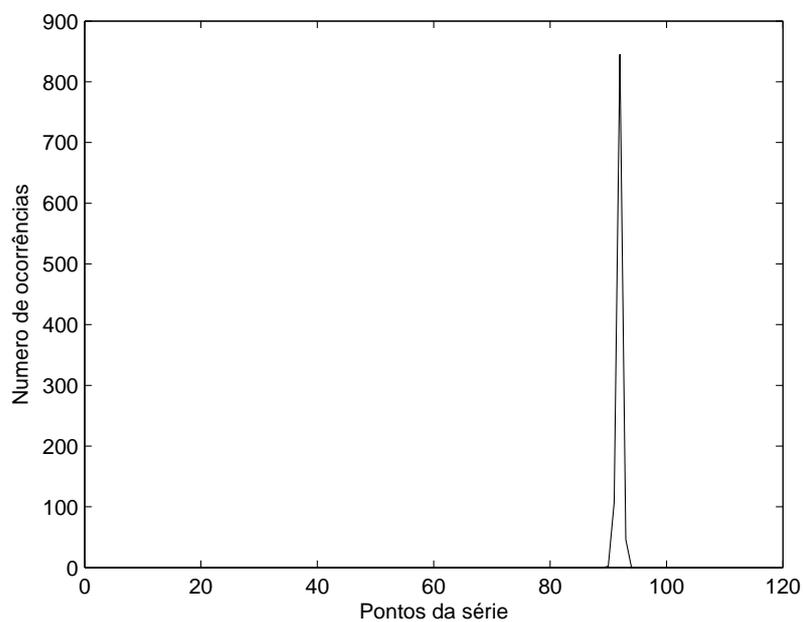


Figura 4.12: Histograma do segundo ponto de mudança para a corrente da fase  $a$

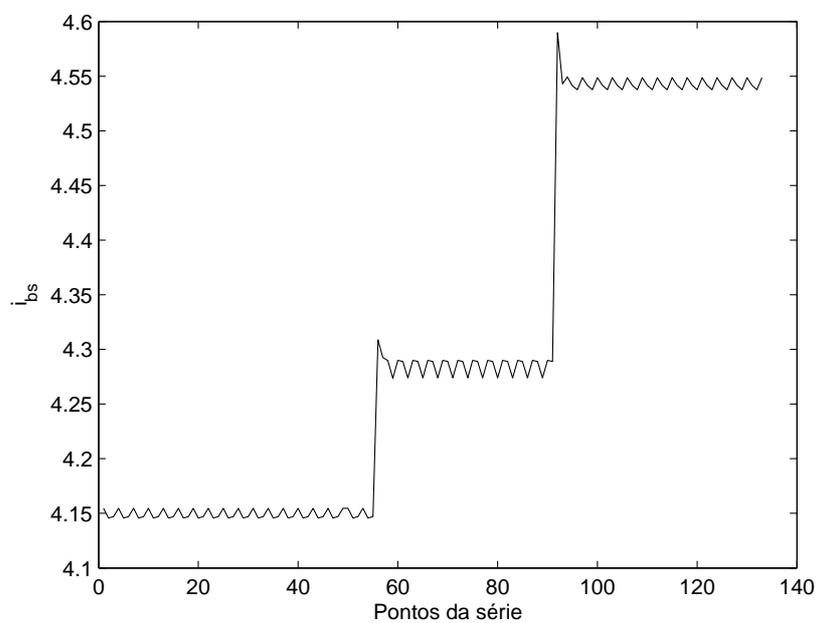
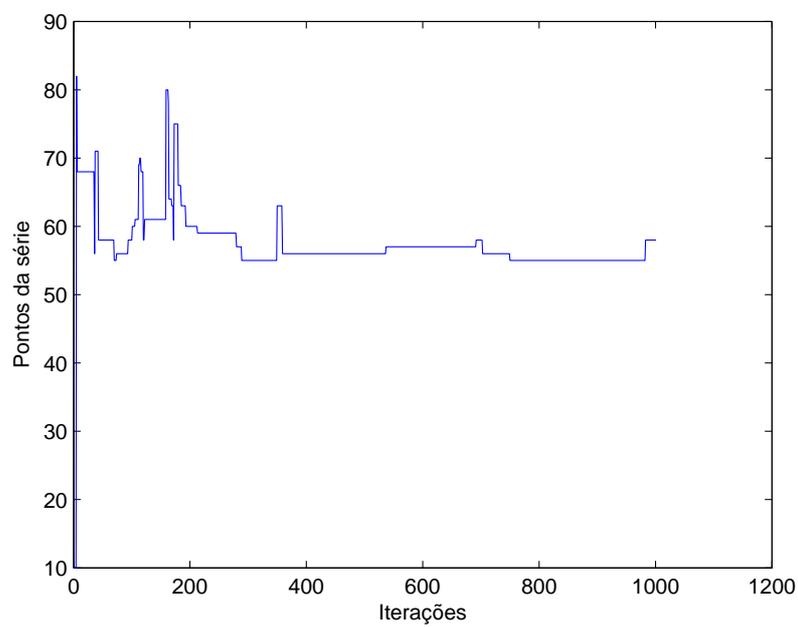
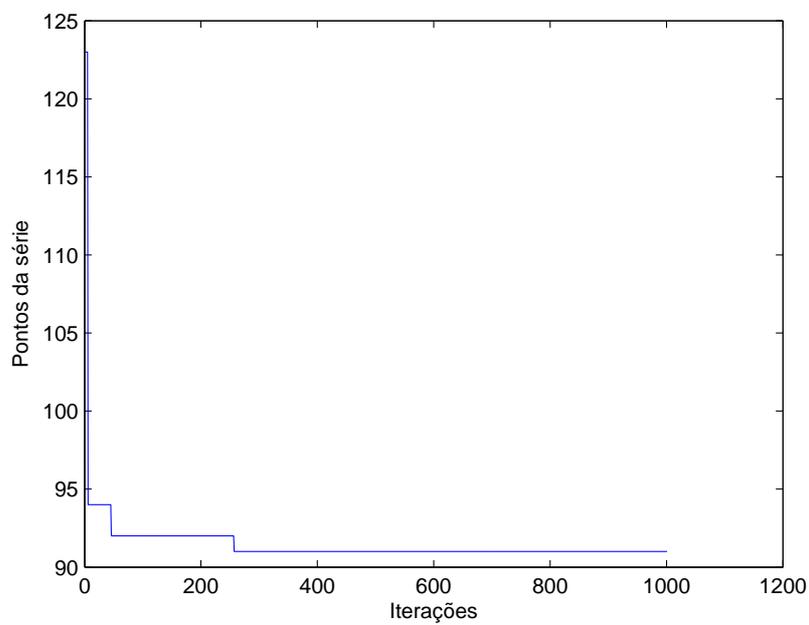


Figura 4.13: Corrente da fase  $b$  (rms): 2% e 5% de espiras em curto na fase  $a$

Figura 4.14: Evolução do parâmetro  $m_1$  para a corrente da fase  $b$ Figura 4.15: Evolução do parâmetro  $m_2$  para a corrente da fase  $b$

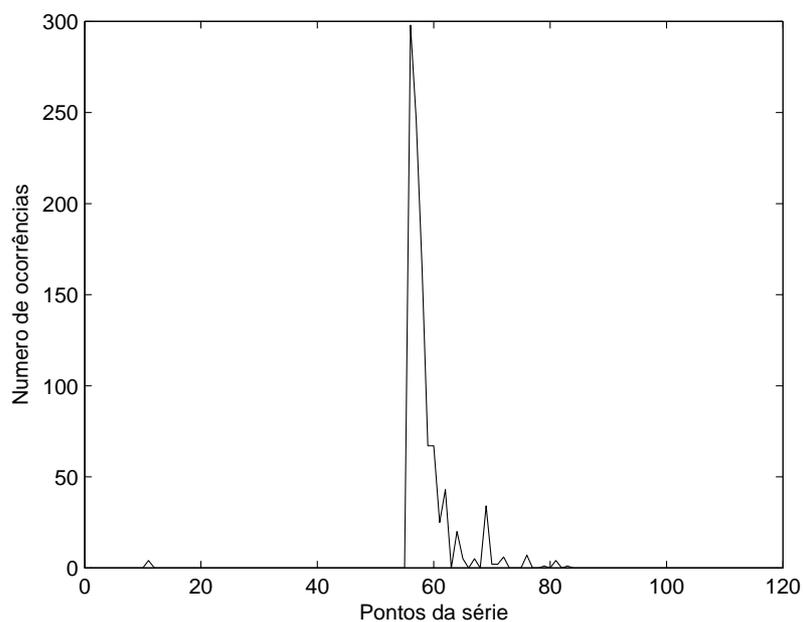


Figura 4.16: Histograma do primeiro ponto de mudança para a corrente da fase  $b$

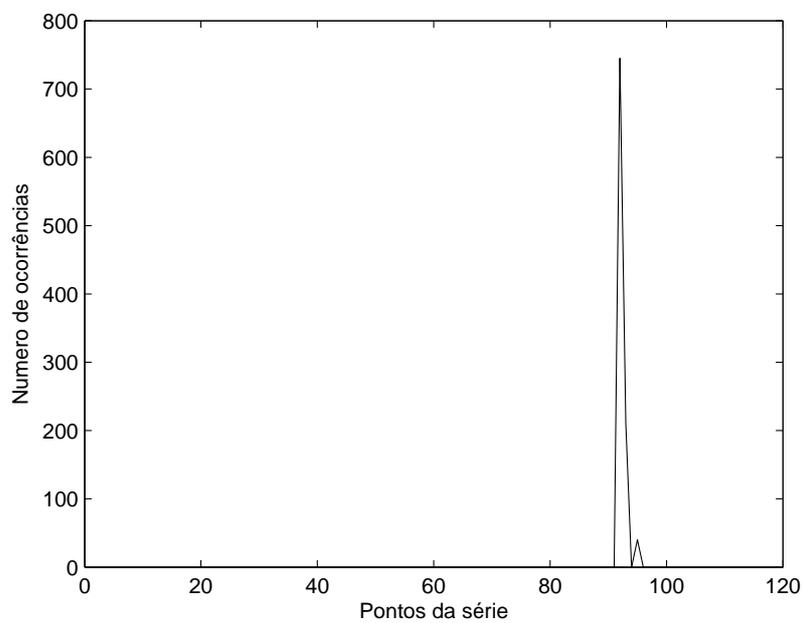


Figura 4.17: Histograma do segundo ponto de mudança para a corrente da fase  $b$

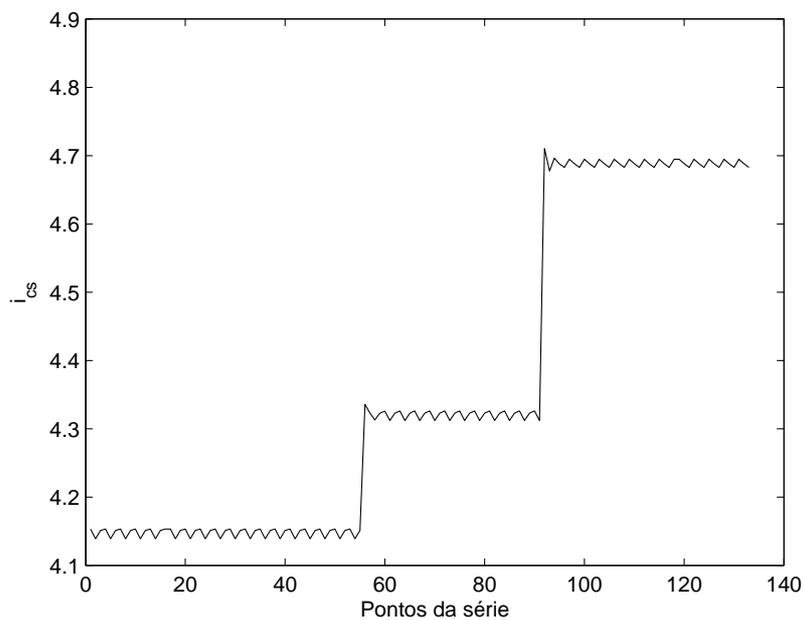


Figura 4.18: Corrente da fase  $c$  (rms): 2% e 5% de espiras em curto na fase  $a$

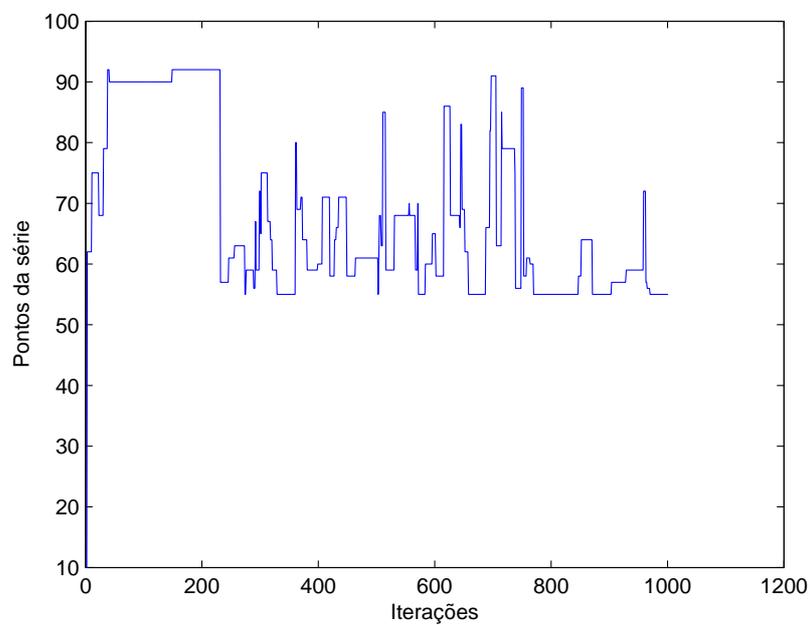


Figura 4.19: Evolução do parâmetro  $m_1$  para a corrente da fase  $c$

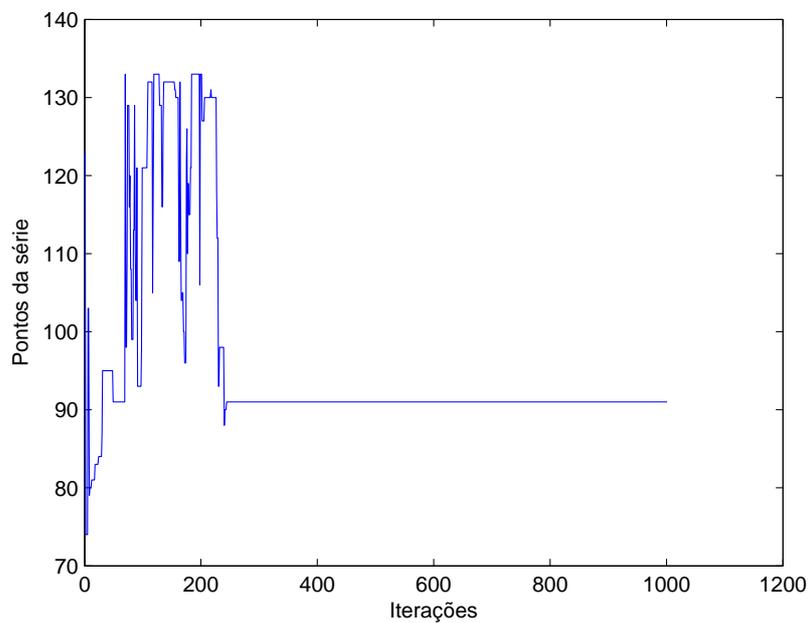


Figura 4.20: Evolução do parâmetro  $m_2$  para a corrente da fase  $c$

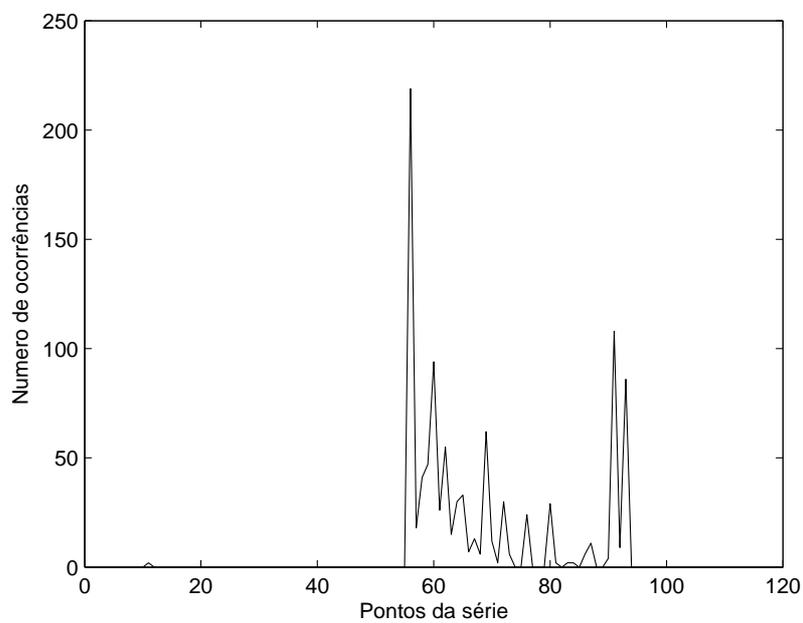


Figura 4.21: Histograma do primeiro ponto de mudança para a corrente da fase  $c$

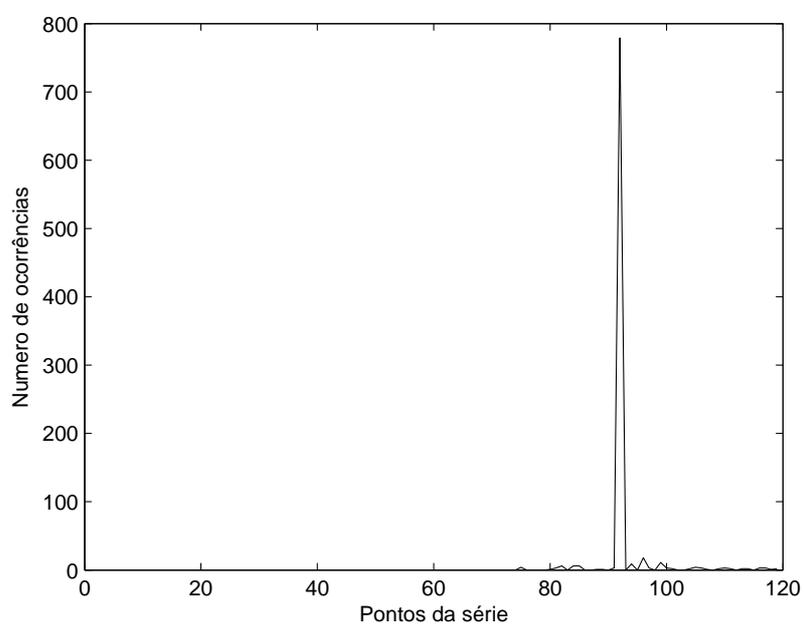


Figura 4.22: Histograma do segundo ponto de mudança para a corrente da fase  $c$

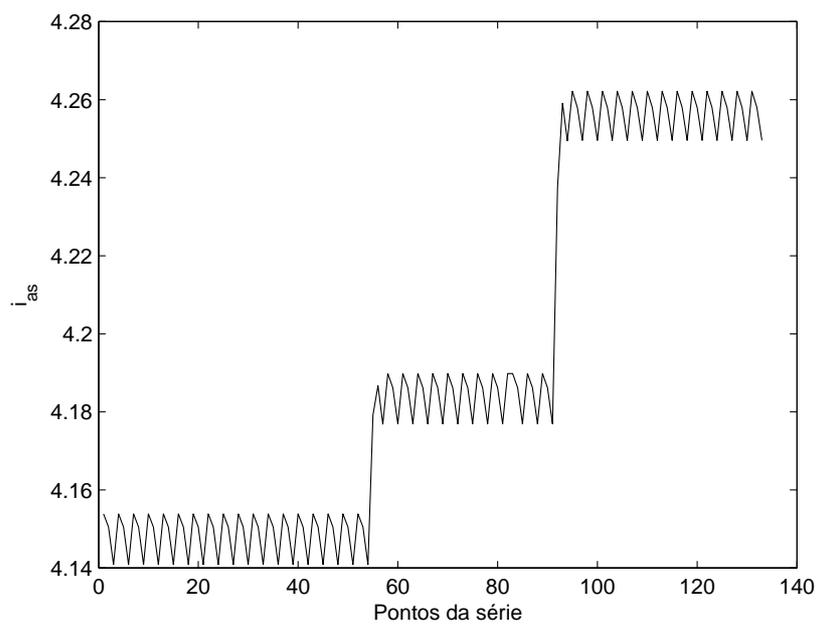


Figura 4.23: Corrente da fase  $a$  (rms): 0.1% e 0.3% de espiras em curto na fase  $a$

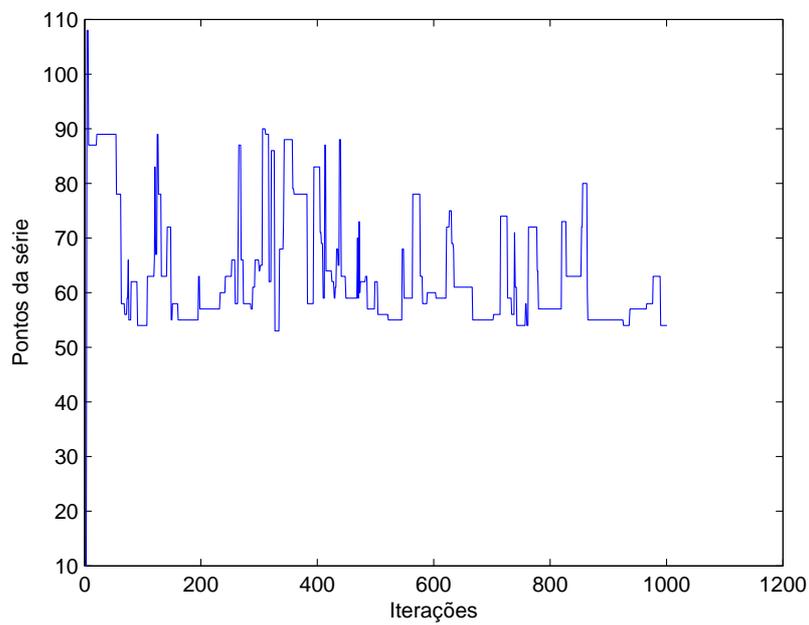


Figura 4.24: Evolução do parâmetro  $m_1$  para a corrente da fase  $a$

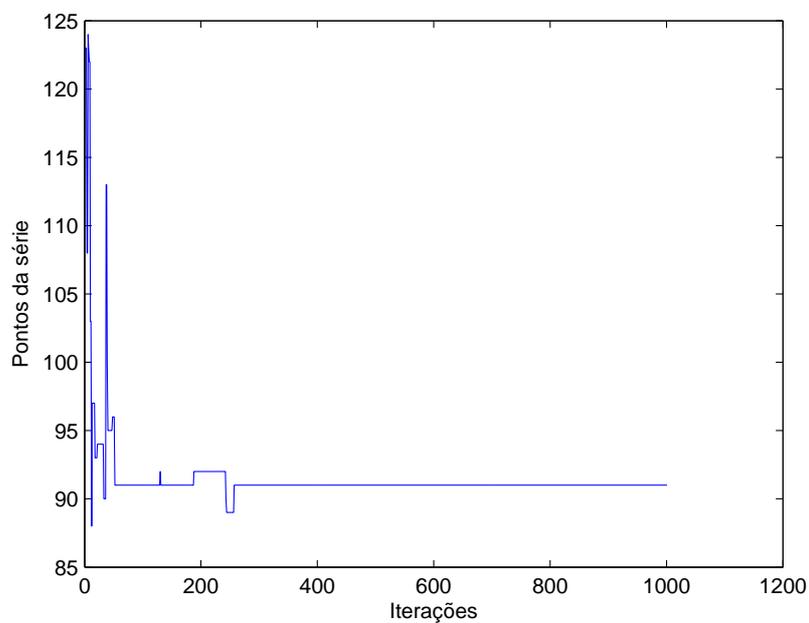


Figura 4.25: Evolução do parâmetro  $m_2$  para a corrente da fase  $a$

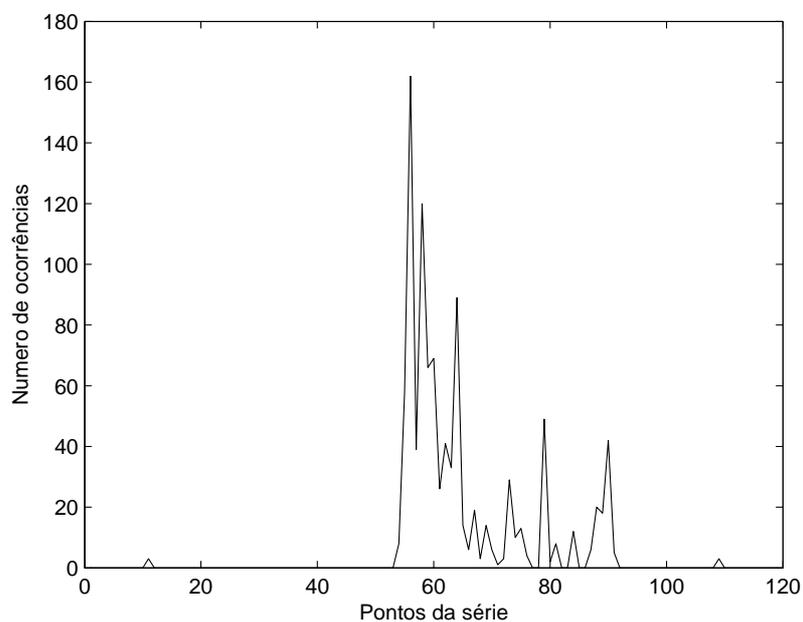


Figura 4.26: Histograma do primeiro ponto de mudança para a corrente da fase  $a$

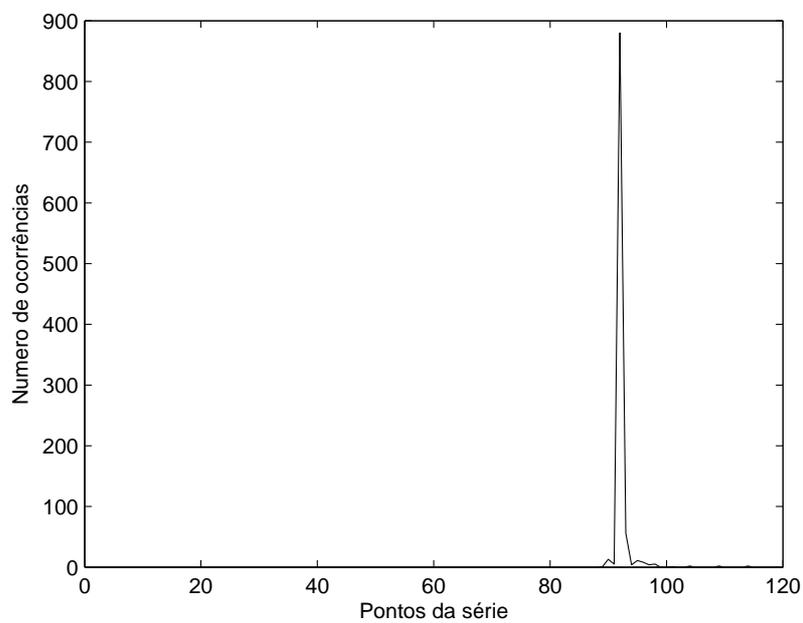


Figura 4.27: Histograma do segundo ponto de mudança para a corrente da fase  $a$

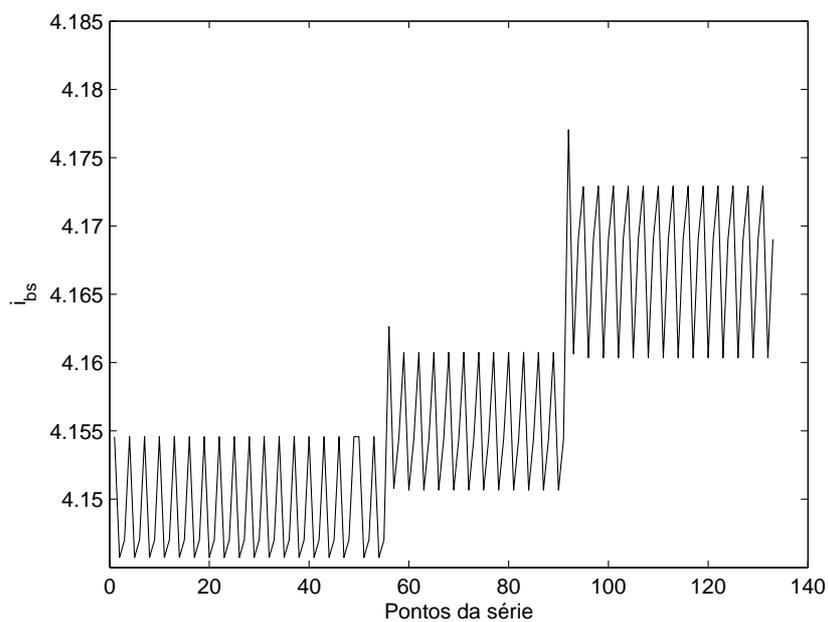


Figura 4.28: Corrente da fase  $b$  (rms): 0.1% e 0.3% de espiras em curto na fase  $a$

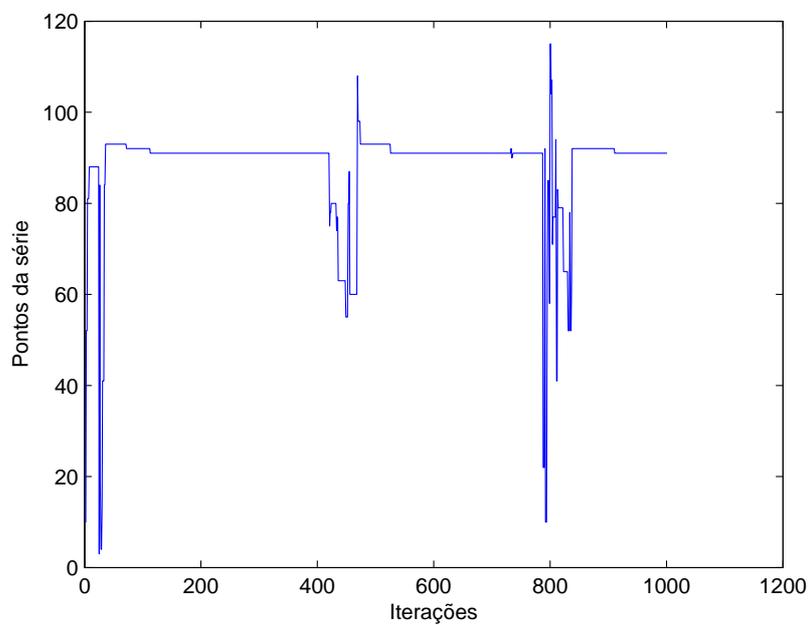


Figura 4.29: Evolução do parâmetro  $m_1$  para a corrente da fase  $b$

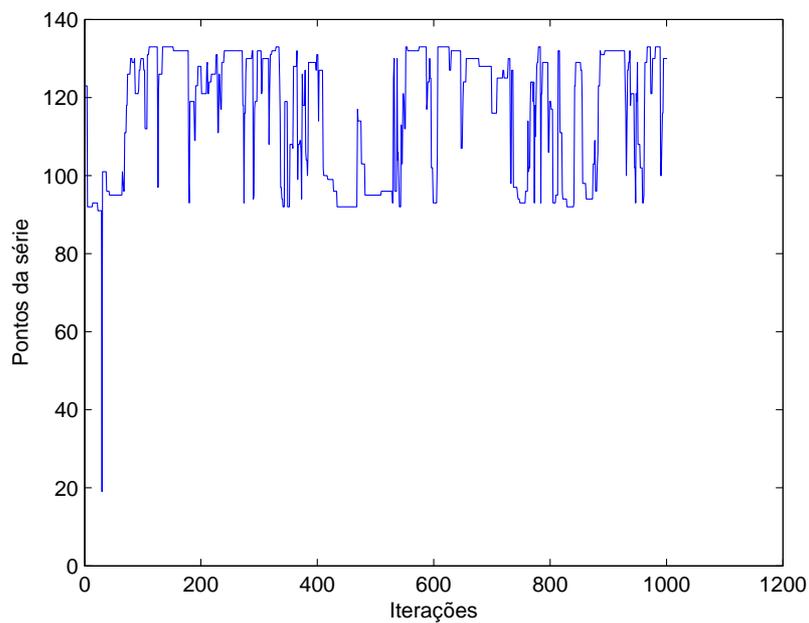


Figura 4.30: Evolução do parâmetro  $m_2$  para a corrente da fase  $b$

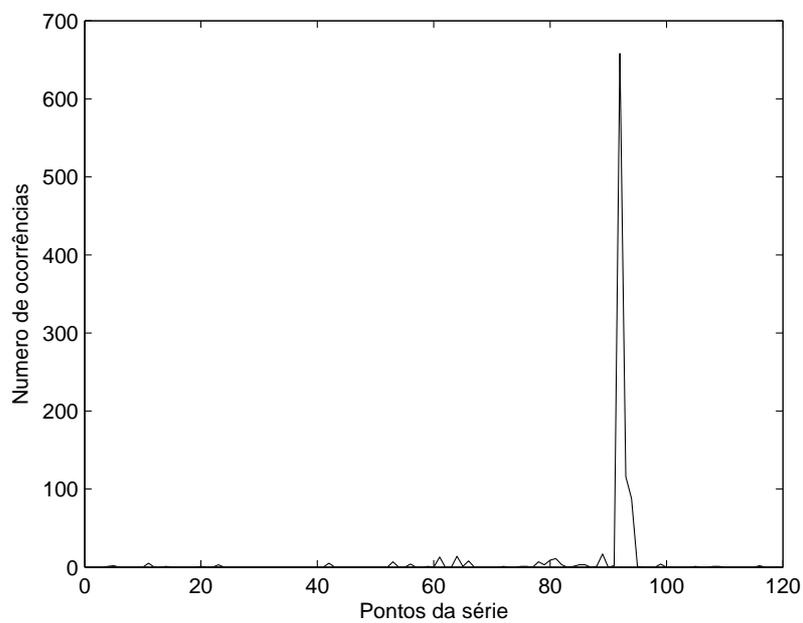


Figura 4.31: Histograma do primeiro ponto de mudança para a corrente da fase  $b$

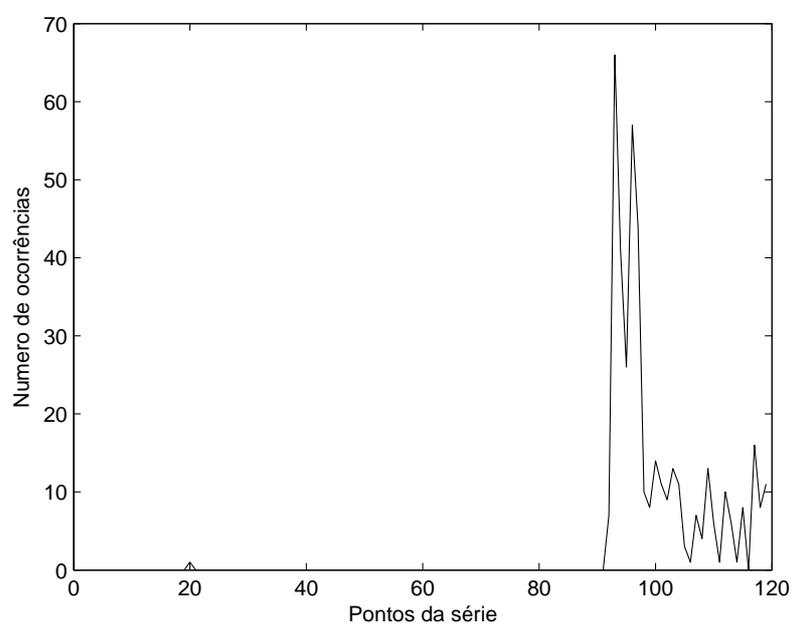


Figura 4.32: Histograma do segundo ponto de mudança para a corrente da fase  $b$

## Capítulo 5

# Considerações Finais

Este trabalho foi a extensão do método de detecção de falhas incipientes apresentado em [D'angelo. \(2010\)](#) para o caso de dois pontos de mudança. Esta extensão compreende o desenvolvimento de um algoritmo capaz de identificar na série temporal o número de pontos de mudança existentes e determinar qual o melhor modelo de detecção de falhas, 1 ou 2 pontos de falhas, deve ser usado na identificação destes pontos. Além disso, foi desenvolvido o modelo para dois pontos de mudança.

### 5.1 Conclusão

O algoritmo proposto apresentou bons resultados para o problema de detecção de falhas no enrolamento estático da máquina de indução para falhas de magnitude superior a 0.2% de espiras em curto. Para o caso de falhas com magnitude inferior a este valor, foram feitas simulações com falhas de 0.1% para o primeiro ponto de mudança e 0.2% para o segundo ponto de mudança e foi observado que o algoritmo não indicou os dois pontos de mudança em todas as simulações para a fase A e não indicou os dois pontos de mudança para as fases B e C. Desta forma, foram considerados pelo algoritmo como pontos de falha as mudanças acima de 0.2%. Esta medida foi feita com base nos centros obtidos pelo algoritmo Kohonen usado para identificar os centros.

Nos casos em que não existe falha, esta condição é identificada quando é executado o algoritmo que identifica o número de centros e seus valores de média, algoritmo Kohonen, e neste caso a função de detecção de falhas nem é executada.

Pode-se concluir também que na medida em que a falha se torna mais severa, o algoritmo identifica mais rapidamente os pontos de mudança. Isso pode ser visto analisando as figuras com os parâmetros  $m_i$ .

Considerações sobre o tamanho da série temporal, janela temporal, e sobre a transformação da série original em uma série com distribuição *beta* foram feitas em [D'angelo. \(2010\)](#).

## 5.2 Proposta de trabalhos futuros

Como dito anteriormente, este trabalho é uma extensão do trabalho apresentado em [D'angelo \(2010\)](#) e apresenta a possibilidade de identificar até dois pontos de mudança na série temporal analisada. Este trabalho pode ser aplicado em diversos casos em que seja necessário identificar a ocorrência de falhas a partir de série temporais. Um exemplo de aplicação interessante é implementar este algoritmo num sistema de controle para analisar valores de variáveis de processo importantes que devem manter valores estáveis, isto é, com pequenas variações. Neste caso, o algoritmo proposto funciona como uma ferramenta de apoio ao operador que vai avisá-lo quando algum parâmetro apresentar mudança.

# Referências Bibliográficas

- Alag, S. S. S. (1996). *A Bayesian Decision-Theoretic Framework for Real-Time Monitoring and Diagnosis of Complex Systems: Theory and Application*. PhD thesis, University of California at Berkeley.
- Albrecht, P. F.; Appiarius, J. C. e Sharma, D. K. (1987). Assessment of reliability of motors in utility applications. *IEEE Transactions On Energy Conversion.*, EC-2(3):396 – 406.
- Baccarini, L. M. R.; de Menezes, B. R.; Guimarães, H. N. e Caminhas, W. M. (2004). Modelagem, simulação e detecção de curto-circuito entre espiras nos enrolamentos do estator de motores de indução. In *Anais do XV Congresso Brasileiro de Automática*, pp. 1930 – 1935, Gramado.
- Barry, D. e Hartigan, J. A. (1993). A bayesian analysis for change point problems. *Journal of the American Statistical Association*, 88(421):309 – 319.
- Bartys, M.; Patton, R.; Syfert, M.; de las Heras, S. e Quevedo, J. (2006). Introduction to the DAMADICS actuator FDI benchmark study. *Control Engineering Practice*, 14(6):577 – 596.
- Beckage, B.; Joseph, L.; Belisle, P.; Wolfson, D. B. e Platt, W. J. (2007). Bayesian change-point analysis in ecology. *New Phytologist*, 2(174):456 – 467.
- Berec, L. (1998). A multi-model method to fault detection and diagnosis: Bayesian solution. an introductory treatise. *International Journal of Adaptive Control and Signal Processing*, 12:81 – 92.
- Bocaniala, C. D. e da Costa, J. S. (2006). Application of a novel fuzzy classifier to fault detection and isolation of the DAMADICS benchmark problem. *Control Engineering Practice*, 14(6):653 – 669.
- Bonnett, A. H. e Soukup, G. C. (1992). Causes and analysis of stator and rotor failures in three-phase induction motors. *IEEE Transactions On Industry Applications*, 28(4):921 – 937.
- Boqiang, X.; Heming, L. e LilingJ, S. (2003). Apparent impedance angle based detection of stator winding interturn short circuit fault in induction motors. In *Proceedings of the Industry Application Conference*, pp. 1118 – 1125.

- Braga, A. P.; Carvalho, A. C. P. L. F. e Ludemir, T. B. (2007). *Redes neurais artificiais: teoria e aplicações*. LTC, 2a edição.
- Calado, J. M. F.; Korbicz, J.; Pattan, K.; Patton, R. J. e da Costa, J. M. G. S. (2001). Soft computing approaches to fault diagnosis for dynamic systems. *European Journal Control*, 7(2 - 3):248 – 286.
- Castilloa, E.; Sarabiab, J. M.; Solaresa, C. e Gómez, P. (1999). Uncertainty analyses in fault trees and bayesian networks using form/sorm methods. *Reliability Engineering and System Safety*, 65:29 – 40.
- Chen, J. e Patton, R. J. (1999). *Robust model-based fault diagnosis for dynamic systems*. Dordrecht: Kluwer Academic Publishers, 1 edição.
- Cruz, S. M. A. e Cardoso, J. (2004). Diagnosis of stator interturn short circuits in dte induction motor drives. *IEEE Transactions on Industry Applications*, 40(5):1349 – 1360.
- Dai, X.; Gao, Z.; Breikin, T. e Wang, H. (2009). Zero assignment for robust  $H_2/H_\infty$  fault detection filter design. *IEEE Transactions on Signal Processing*, 57(4):1363 – 1372.
- D'angelo., M. F. S. V. (2010). *Uma Nova Formulação Fuzzy/Bayesiana para Detecção de Ponto de Mudança em Séries Temporais: Aplicações na Detecção de Falhas Incipientes*. PhD thesis, Universidade Federal de Minas Gerais. Tese (Doutorado em Engenharia Elétrica). Orientador: Reinaldo Martinez Palhares.
- D'Angelo, M. F. S. V. e Costa, P. P. (2001). Detection of shorted turns in the field winding of turbogenerators using the neural network mlp. In *Proceedings of the IEEE International Conference on Systems, Man, and Cybernetics*, pp. 1930 – 1935, Tucson.
- D'Angelo, M. F. S. V.; Palhares, R. M.; Caminhas, W. M.; Takahashi, R. H. C.; Maia, R. D. e Lemos, A. P. (2010). Detecção de falhas: Uma revisão com aplicações. In Lázaro, A. P. F. C. R. M. M. C. M. T. R. A. R., editor, *Tutorias - XVIII Congresso Brasileiro de Automática*, volume 1, pp. 1 – 47. São Paulo: Cultura Acadêmica Editora.
- D'Angelo, M. F. S. V.; Palhares, R. M.; Takahashi, R. H. C. e Loschi, R. H. (2007). Uma abordagem fuzzy/bayesiana para o problema de detecção de pontos de mudança em séries temporais. In *XXXIX Simpósio Brasileiro de Pesquisa Operacional*, pp. 2530 – 2541.
- (2008). Uma abordagem fuzzy/bayesiana para o problema de detecção de falhas incipientes. In *XVII Congresso Brasileiro de Automática*, pp. 1 – 6.
- (2011a). Fuzzy/bayesian change point detection approach to incipient fault detection. *IET Control Theory and Applications (Print)*.
- D'Angelo, M. F. S. V.; Palhares, R. M.; Takahashi, R. H. C.; Loschi, R. H.; Baccarini, L. M. R. e Caminhas, W. M. (2011b). Incipient fault detection in induction machine stator-winding using a fuzzy-bayesian change point detection approach. *Applied Soft Computing*, 11:179 – 192.

- Douglas, R. K. e Speyer, J. L. (1996). Robust fault detection filter design. *Journal of Guidance, Control, and Dynamics*, 19(1):214 – 218.
- Gamerman, D. (1997). Markov chain monte carlo: stochastic simulation for Bayesian inference. *Chapman & Hall*.
- Hinkey, D. V. (1971). Inference about the change point from cumulative sum test. *Biometria*, 26:279 – 284.
- Hou, M. e Patton, R. J. (1998). Optimal filtering for systems with unknown inputs. *IEEE Transactions on Automatic Control*, 43(3):445 – 449.
- Isermann, R. e Balle, P. (1997). Trends in the application of model-based fault detection and diagnosis of technical processes. *Control Engineering Practice*, 5(5):707 – 719.
- Jang, J.-S. R.; Sun, C.-T. e Mizutani, E. (1997). *Neuro-fuzzy and soft computing : a computational approach to learning and machine intelligence*. Prentice Hall.
- Jiang, C. e Zhou, D. H. (2005). Fault detection and identification for uncertain linear time-delay systems. *Computers & Chemical Engineering*, 30:228 – 242.
- Kohonen, T. (1990). The self-organizing map. *Proceedings of the IEEE*, 78:1464 – 1480.
- Kullback, S. e Leibler, R. A. (1951). On information and sufficiency. *Annals of Mathematical Statistics*, 22(1):79 – 86.
- Lo, C. H.; Fung, E. H. K. e Wong, Y. K. (2009). Intelligent automatic fault detection for actuator failures in aircraft. *IEEE Transactions on Industrial Informatics*, 5(1):50 – 55.
- Loschi, R. H.; Gonçalves, F. B. e Cruz, F. B. R. (2005). Avaliação de medida de evidência de um ponto de mudança e sua utilização na identificação de mudanças na taxa de criminalidade em belo horizonte. *Pesquisa Operacional*, 3(25):459 – 463.
- Mast, T. A.; Reed, A. T.; Yurkovich, S.; Ashby, M. e Adibhatla, S. (1999). Bayesian belief networks for fault identification in aircraft gas turbine engines. In *Proceedings of the 1999 IEEE International Conference on Control Applications*, pp. 39 – 44, Hawaii, USA.
- Mozelli, L. A. (2008). *Controle Fuzzy para Sistemas Takagi-Sugeno: Condições Aprimoradas e Aplicações*. PhD thesis, Universidade Federal de Minas Gerais. Dissertação (Mestrado em Engenharia Elétrica). Orientador: Reinaldo Martinez Palhares.
- O'Donnell, P. (1985). Report of large motor reliability survey of industrial and commercial installations: Part i. *IEEE Transactions On Industry Applications*, IA-21(4):853 – 864.
- Oh, K. J.; Roh, T. H. e Moon, M. S. (2005). Developing time-based clustering neural networks to use change-point detection: Application to financial time series. *Asia-Pacific Journal of Operational Research*, 1(22):51 – 70.

- O'Reilly, P. G. (1998). Trends in the application of model-based fault detection and diagnosis of technical processes. In *UKACC International Conference on CONTROL '98*, pp. 247 – 251.
- Perreault, L.; Bernier, J.; Bobée, B. e Parent, E. (2000). Bayesian change-point analysis in hydrometeorological time series. part 1. the normal model revisited. *Journal of Hydrology*, 3 - 4(235):221 – 241.
- Ploix, S. e Adrot, O. (2006). Parity relations for linear uncertain dynamic systems. *Automatica*, 42(9):1553 – 1562.
- Puig, V.; Stancu, A.; Escobet, T.; Nejjari, F.; Quevedo, J. e Patton, R. (2006). Passive robust fault detection using interval observers: Application to the DAMADICS benchmark problem. *Control Engineering Practice*, 14(6):621 – 633.
- Rigatos, G. e Zhang, Q. (2009). Fuzzy model validation using the local statistical approach. *Fuzzy Sets and Systems*, 160(7):882 – 904.
- Takahashi, R. H. C.; Palhares, R. M. e Peres, P. L. D. (1999). Discrete-time singular observers:  $\mathcal{H}_2/\mathcal{H}_\infty$  optimality and unknown inputs. *International Journal of Control*, 72(6):481 – 492.
- Takahashi, R. H. C. e Peres, P. L. D. (1999). Unknown input observers for uncertain systems: A unifying approach. *European Journal of Control*, 5(2 - 4):261 – 275.
- Tang, F. (2000). *A Model-Based Bayesian Fault Diagnostic Systems – With Applications to Semiconductor Manufacturing Processes*. PhD thesis, Carnegie Mellon University.
- Tanscheit, R.; Gomide, F. e Teixeira, M. C. M. (2007). Modelagem e controle nebuloso. In Aguirre, L. A., editor, *Enciclopédia de Automática: Controle & Automação*, volume 3, pp. 283–324. Blucher.
- Thorsen, O. V. e Dalva, M. (1999). Failure identification and analysis for high voltage induction motors in the petrochemical industry. *IEEE Transactions On Industry Applications*, 35(4):810 – 818.
- Tsoukalas, L. H. e Uhrig, R. E. (1997). *Fuzzy and Neural approaches in engineering*. John Wiley & Sons.
- Wilsky, A. (1976). A survey of design methods for failure detection in dynamic systems. *Automatica*, 12:601 – 611.
- Witczak, M.; Korbicz, J.; Mrugalski, M. e Patton, R. J. (2006). A gmdh neural network-based approach to robust fault diagnosis: Application to the DAMADICS benchmark problem. *Control Engineering Practice*, 14(6):671 – 683.
- W.T.Thomson e Fenger, M. (2001). Current signature analysis to detect induction motor faults. *IEEE Industry Applications Magazine*, 7:26 – 34.

Zadeh, L. A. (1965). Fuzzy sets. *Information and Control*, 8(3):338 – 353.