

**UNIVERSIDADE FEDERAL DE SÃO CARLOS  
CENTRO DE CIÊNCIAS EXATAS E DE TECNOLOGIA  
PROGRAMA DE PÓS-GRADUAÇÃO EM CIÊNCIA DA  
COMPUTAÇÃO**

**FILTRAGEM DE PROJEÇÕES TOMOGRÁFICAS DA  
CIÊNCIA DO SOLO UTILIZANDO KALMAN  
E REDES NEURAIIS**

**Marcos Antônio de Matos Laia**

**SÃO CARLOS - SP**

**2007**

**Ficha catalográfica elaborada pelo DePT da  
Biblioteca Comunitária da UFSCar**

L185fp

Laia, Marcos Antonio de Matos.

Filtragem de projeções tomográficas da ciência do solo utilizando Kalman e redes neurais / Marcos Antonio de Matos Laia. -- São Carlos : UFSCar, 2007.

131 f.

Dissertação (Mestrado) -- Universidade Federal de São Carlos, 2007.

1. Processamento de sinais - técnicas digitais. 2. Filtros adaptativos. 3. Kalman, Filtragem de. 4. Redes neurais (Computação). 5. Tomografia computadorizada. I. Título.

CDD: 621.3822 (20ª)


**Universidade Federal de São Carlos**  
**Centro de Ciências Exatas e de Tecnologia**  
**Programa de Pós-Graduação em Ciência da Computação**

***“Filtragem de projeções tomográficas da ciência do  
solo utilizando Kalman e redes neurais”***

MARCOS ANTONIO DE MATOS LAIA

Dissertação de Mestrado apresentada ao Programa de Pós-Graduação em Ciência da Computação da Universidade Federal de São Carlos, como parte dos requisitos para a obtenção do título de Mestre em Ciência da Computação.

Membros da Banca:



---

Prof. Dr. Paulo Estevão Cruvinel  
(Orientador – EMBRAPA/Instrumentação  
Agropecuária)



---

Prof. Dr. José Hiroki Saito  
(DC/UFSCar)



---

Prof. Dr. Joaquim Teixeira de Assis  
(UERJ/Rio de Janeiro)

São Carlos  
Agosto/2007

## Agradecimentos

---

Agradeço em primeiro lugar aos meus pais por sempre estarem ao meu lado e contribuírem com confiança e perseverança em todos os momentos da minha vida. Agradeço também ao meu irmão e demais parentes pelo apoio durante todos esses anos. Dedico esse trabalho a eles, pelo carinho, amor e compreensão.

Este trabalho não se tornaria possível sem a orientação do Professor Doutor Paulo Estevão Cruvinel, que sempre depositou sua confiança em mim e a este trabalho, compreendendo as situações difíceis, contribuindo com sua ajuda precisa em todos os momentos.

Agradeço aos meus colegas de departamento, que se tornaram amigos próximos, Luís, Alexandre, Leonardo, Luanna, Val, Amanda, Cristiane, Eduardo, Débora, Eliane, Robson, Danilo e tantos outros pela companhia durante os anos de mestrado nas conversas e por todo apoio nos momentos difíceis.

Agradeço de forma especial aos meus colegas Leonardo e Maurício sobre toda a colaboração durante a realização desse trabalho e pela amizade que se estabeleceu entre nós. Foram contribuições importantes não só no trabalho, mas pelas conversas sobre diversos assuntos e pontos de vistas que foram essenciais.

Agradeço aos Professores e demais funcionários do Departamento de Computação que favoreceram um ambiente de trabalho que favoreceu a confecção deste trabalho.

Não posso deixar de agradecer também aos meus amigos da minha cidade, que sempre me acolheram de braços abertos em minhas viagens de volta a Minas.

## Resumo

---

Neste trabalho é apresentada a filtragem de projeções tomográficas com ruído variantes no espaço com base na filtragem de Kalman. Para o desenvolvimento e seleção dos filtros foram avaliadas diferentes modalidades da configuração de Kalman, incluindo o uso da transformada de Anscombe e redes neurais. Resultados foram analisados com base em medidas da melhoria na relação sinal/ruído (ISNR), as quais foram obtidas em uma região de interesse (ROI) nas imagens resultantes, reconstruídas com o uso do algoritmo de retroprojeção. Neste contexto os resultados qualificaram o filtro de Kalman descentralizado com uma rede neural possuindo três camadas do tipo *perceptron* como a melhor opção para a filtragem de projeções tomográficas da ciência do solo.

## **Abstract**

---

This work presents the space variant noise filtering of tomographic projections based on the Kalman filter. For development and filter selection it was evaluated different modalities of the Kalman filter, as well as included the use of Ascombe transform and neural network. Results were analyzed by means of Improvement in Signal to Noise Ratio (ISNR) measurements, which were obtained in a region of interest (ROI) on the resultant images, reconstructed with the use of a backprojection algorithm. In this context the results qualified the unscented Kalman filter with a neural network as the best configuration for filtering of soil tomographic projections.

## Lista de Figuras

---

Figura 1. Mini-tomógrafo da Embrapa Instrumentação Agropecuária (CRUVINEL 1987). .....	19
Figura 2. Duas projeções paralelas de um objeto expresso por uma função bidimensional. .....	21
Figura 3. O sinal analógico é multiplicado por um trem de impulsos, onde é feita a amostragem em pontos. A reconstrução do sinal é feita com base nas amostras, o que depende diretamente da qualidade amostral. $f(x)$ corresponde a uma função representando o sinal analógico, $s(x)$ um trem de impulsos, $(fs)(x)$ um sinal discretizado, $l(x)$ uma função contínua e $(fsl)(x)$ o sinal contínuo reconstruído.....	24
Figura 4. A distância $d$ (a) é representada de forma inversa no espectro de Fourier (b), O sinal (c) representado no domínio do tempo, e sua representação (d) no domínio da frequência.....	25
Figura 5. (a) $f(x)$ é um sinal a ser amostrado, (b) $F(u)$ é o seu espectro de Fourier equivalente no domínio da frequência (c) $s(x)$ é um trem de impulso e (d) $S(u)$ seu espectro de Fourier equivalente no domínio da frequência, (e) $f(x).s(x)$ é o sinal amostrado, (f) $F(u)*S(u)$ é seu espectro de Fourier equivalente no domínio da frequência. ....	26
Figura 6. (a) O sinal representado por $f(x)$ é amostrado com uma nova função de trem de impulsos com um intervalo maior entre os pontos (c). Isto provoca uma perda de qualidade do sinal (e) e a presença de aliasing (f) no domínio da frequência.....	26
Figura 7. Algoritmo para filtro de Kalman.Com as estimativas iniciais para o primeiro valor da medida e a probabilidade inicial, o algoritmo atua de maneira recursiva, alternando entre a previsão dos dados e a correção. ....	37
Figura 8. Algoritmo do filtro de Kalman Estendido com as funções para problemas não-lineares.....	40
Figura 9. Princípio da transformação descentralizada. ....	43
Figura 10. Algoritmo para filtro de Kalman descentralizado. ....	44
Figura 11. Exemplo de uma transformação descentralizada para propagação de média e covariância. ....	45
Figura 12. A probabilidade inicial é igual a 1 em todos os casos, pois o valor inicial é dado pelo valor do sinal original. O que varia entre os casos é o valor do ruído do ganho de	

Kalman. Em (a), esse valor é igual a 1, em (b), igual a 0,0001, em (c), igual 10 e em (d), igual a 0,01.....	48
Figura 13. Ilustrando as atualizações de tempo do suavizador para (a) retro-filtragem e (b) filtragem adiante. ....	49
Figura 14. Ilustrando a covariância de erro para a filtragem adiante, retro filtragem, e suavizante.....	52
Figura 15. Diagrama do neurônio de McCulloch-Pitts, apresentando as entradas ( $X_1...X_n$ ), os pesos ( $W_1...W_n$ ) e a saída ( $Y$ ) do neurônio.....	55
Figura 16. Tabela comparativa do <i>perceptron</i> unicamada e multicamadas e os problemas que podem ser solucionados (JAIN 1996). ....	57
Figura 17. Modelo de neurônio artificial com uma entrada.....	58
Figura 18. Função de transferência de limite (a), linear (b) e sigmoideal (c). O eixos a e n correspondem a saída e a entrada da rede, respectivamente. Cada entrada é mapeada na função de forma independente, assumindo um valor de acordo com os pesos.....	59
Figura 19. Neurônio com vetor de entradas.....	60
Figura 20. Vários neurônios em uma única camada. ....	60
Figura 21. Uma rede neural de três camadas e a equação para calcular sua saída. ....	62
Figura 22. Saídas das redes neurais com entrada de tamanho 10 e 10 passos para treinamentos.....	64
Figura 23. Saídas das redes neurais com 100 entradas e 100 passos de treinamento.....	64
Figura 24. Saídas das redes neurais com 1000 entradas e 1000 passos de treinamento. Para melhor visualização dos dados, a escala foi ampliada.....	65
Figura 25. Resultado mostrando a alta convergência do algoritmo de treinamento Levenberg-Marquadt.....	66
Figura 26. Sinal tomográfico definido por $P_{\theta_1}[n] = I_1\delta(n+4) + I_2\delta(n+3) + I_3\delta(n+2) + I_4\delta(n+1) + I_5\delta(n) + I_6\delta(n-1) + I_7\delta(n-2) + I_8\delta(n-3) + I_9\delta(n-4)$ .....	69
Figura 27. Diagrama para construção do <i>phantom</i> homogêneo. ....	70
Figura 28. Diagrama para construção do <i>phantom</i> heterogêneo.....	71
Figura 29. <i>Phantoms</i> utilizados para calibração e validação: projeções (a) e imagem reconstruída (b) do <i>phantom</i> homogêneo, projeções (c) e imagem reconstruída (d) do <i>phantom</i> heterogêneo. ....	71
Figura 30. Ferramenta para filtragem de projeções tomográficas. ....	72



Figura 31. Campo para carregar as projeções. Os comentários dos arquivos é exibido logo abaixo do botão. ....	72
Figura 32. Caixa de diálogo para navegar entre os arquivos de projeções. ....	73
Figura 33. Após a escolha do arquivo que contem as projeções a serem tratadas, são exibidas o conjunto de projeções e a reconstrução da projeção no campo <i>Filtro 1</i> . ....	73
Figura 34. Campo para a entrada da variância do ruído nas projeções. ....	73
Figura 35. Campo <i>Filtro 2</i> onde são mostradas a projeção filtrada pelo Kalman discreto e sua resconstrução, após selecionar o botão <i>Aplicar</i> . ....	74
Figura 36. Análise dos sinais comparando as filtragens de Kalman e suas variações. ....	74
Figura 37. Botões para salvar de forma individual a projeção filtrada. ....	75
Figura 38. Caixa de diálogo para navegar pelos diretórios e salvar o conjunto de projeção filtrada. ....	75
Figura 39. Diagramação do sinal para ruídos gaussianos brancos. ....	76
Figura 40. Diagramação do sinal para ruídos <i>Poisson</i> e gaussianos brancos. Nota-se que o ruído da função de transição do sistema depende diretamente do valor de $x$ . ....	76
Figura 41. Diagramas de blocos para o filtro de Kalman discreto. ....	76
Figura 42. Regiões de interesse para avaliar os resultados obtidos: Em (a) uma ROI de 42x28 pixels e em (b) cinco ROIs de 14x11 pixels. ....	78
Figura 43. Resultados obtidos com o filtro de Kalman discreto com a variância do ruído do processo ( $Q$ ) igual ao valor do feixe livre. Resultados obtidos pelo filtro com diferentes valores para o ruído do sistema dos <i>Phantoms</i> (a) homogêneo e (b) heterogêneo. ....	79
Figura 44. Projeções filtradas pelo filtro de Kalman discreto com variância igual ao valor do feixe livre (10000) para o ruído do processo e os melhores ISNR: (a) <i>phantom</i> homogêneo e (b) sua respectiva reconstrução; (c) <i>phantom</i> heterogêneo com sua respectiva reconstrução (d). ....	80
Figura 45. Diagrama de blocos para a filtragem com Kalman estendido e a função de transformada de Anscombe na equação do processo. ....	81
Figura 46. Diagramação do sinal utilizando as Transformadas de Anscombe. ....	81
Figura 47 . Resultados obtidos com o filtro de Kalman estendido com a transformada de Anscombe com a variância do ruído do processo ( $Q$ ) igual a 1. Resultados obtidos pelo filtro com diferentes valores para o ruído do sistema dos <i>Phantoms</i> (a) homogêneo e (b) heterogêneo. ....	83

Figura 48. Projeções filtradas pelo filtro de Kalman estendido com variância igual a 1 para o ruído do processo e os melhores ISNR obtidos: (a) <i>phantom</i> homogêneo e (b) sua respectiva reconstrução; (c) <i>phantom</i> heterogêneo com sua respectiva reconstrução (d). ..	85
Figura 49. Filtro de Kalman estendido com redes neurais.....	86
Figura 50. Resultados obtidos com o filtro de Kalman estendido com redes neurais com a variância do ruído do processo (Q) igual a 1. Resultados obtidos pelo filtro com diferentes valores para o ruído do sistema dos <i>Phantoms</i> (a) homogêneo e (b) heterogêneo.....	88
Figura 51. Melhores resultados obtidos com o filtro de Kalman estendido com redes neurais: (a) (b) <i>phantom</i> homogêneo e (c) (d) <i>phantom</i> heterogêneo.....	90
Figura 52. Diagrama de blocos do filtro de Kalman descentralizado.....	91
Figura 53 – Resultados obtidos com o filtro de Kalman descentralizado na sua forma pura, equivalente ao filtro de Kalman discreto. ....	93
Figura 54. Resultados obtidos com o filtro de Kalman descentralizado para (a)(b) <i>phantom</i> homogêneo e (c)(d) <i>phantom</i> heterogêneo. ....	94
Figura 55. Filtro de Kalman descentralizado com transformada de Anscombe.....	95
Figura 56. Comparação entre os sinais para a filtragem com Kalman descentralizado usando a transformada de Anscombe na equação do processo.....	96
Figura 57. Conjunto de projeções e suas respectivas imagens reconstruídas filtradas pelo filtro de Kalman descentralizado com transformada de Anscombe: (a)(b) <i>phantom</i> homogêneo e (c)(d) <i>phantom</i> heterogêneo. ....	98
Figura 58. Resultados obtidos com o filtro de Kalman descentralizado com transformada de Anscombe com variância do ruído do processo próximo de zero para a filtragem do ruído Poisson.....	99
Figura 59. Conjunto de projeções e suas respectivas reconstruções, obtidos com o filtro de Kalman descentralizado com transformada de Anscombe e ruído do processo próximo a zero.....	101
Figura 60 – Filtro de Kalman descentralizado dual estimation com redes neurais. ....	102
Figura 61. Resultados obtidos com o filtro de Kalman descentralizado com redes neurais: (a) <i>phantom</i> homogêneo e (b) <i>phantom</i> heterogêneo. ....	103
Figura 62. Conjunto de projeções e suas respectivas reconstruções, obtidos com o filtro de Kalman descentralizado e redes neurais para os <i>phantoms</i> (a) (b) homogêneo e (c)(d) heterogêneo. ....	104
Figura 63 - Resultados obtidos com o filtro de Kalman descentralizado com redes neurais com duas camadas ocultas: (a) <i>phantom</i> homogêneo e (b) <i>phantom</i> heterogêneo. ....	106

Figura 64. Conjunto de projeções e suas respectivas reconstruções, obtidos com o filtro de Kalman descentralizado e redes neurais para os <i>phantoms</i> (a) (b) homogêneo e (c)(d) heterogêneo.....	108
Figura 65. Resultado obtido com um <i>phantom</i> real de solo arenoso com o filtro de Kalman descentralizado com redes neurais com três camadas (a) correção da rede neural feita através dos valores observados e (b) correção feita pelos valores estimados.....	109
Figura 66. Amostra de solo: (a) (b) sem filtragem e com redes neurais: (c)(d) correções da rede baseadas nas observações e (e)(f) correções da rede baseada nos valores estimados. .....	110
Figura 67. A função de distribuição de probabilidade Normal ou Gaussiana. ....	121
Figura 68. Dois exemplos (hipotéticos) de funções de autocorrelação $X_1$ e $X_2$ .....	123
Figura 69. Ruído branco mostrado em ambos o domínio de tempo (a) e frequência (b). .	124
Figura 70. Rede neural com três camadas, com duas entradas, uma saída e três neurônios na primeira camada, dois na segunda e um na terceira. ....	126
Figura 71. Detalhe da função soma do neurônio. ....	127
Figura 72. Em (a), os valores da entrada $x_1$ e $x_2$ para o primeiro neurônio da primeira camada, seguido, em (b), para o segundo neurônio e em (c), para o terceiro.....	128
Figura 73. Em (a), a saída dos neurônios da primeira camada são repassados ao primeiro neurônio da segunda camada (ou camada oculta), enquanto em (b), ao segundo neurônio. .....	128
Figura 74. As saídas da camada oculta são repassadas ao neurônio da ultima camada (camada de saída).....	128
Figura 75. Erro sendo propagado de volta ao neurônio de saída. ....	129
Figura 76. Em (a), o erro do neurônio da camada de saída é propagado para o primeiro neurônio da camada oculta. Em (b), o erro é propagado para o segundo neurônio.....	129
Figura 77. Retropropagação dos erros da camada oculta para a camada de entrada (setas tracejadas em vermelho) e erros são somados aos erros anteriores (setas vermelhas). ....	130
Figura 78. Em (a), os pesos de entrada são modificados de acordo com o erro do primeiro neurônio da primeira camada. Em (b) e (c) segue da mesma forma para seus respectivos pesos e neurônios. ....	131
Figura 79. Em (a) e (b), a atualização dos pesos da segunda camada são feitas de acordo com os erros dos neurônios da segunda camada. Em (c), os pesos são atualizados pelo erro do neurônio de saída. ....	131

## Lista de Tabelas

---

Tabela 1- Tabela de erro médio para a função $fx = poisson(x)$ .....	63
Tabela 2- Tabela de tempo de execução para a função $fx = poissonx$ , em segundos. ...	63
Tabela 3- Resultados obtidos com o algoritmo Levenber-Marquadt.....	66
Tabela 4 – Resultados obtidos com o filtro de Kalman discreto para <i>phantoms</i> para calibração. ....	78
Tabela 5 – Resultados obtidos com o filtro de Kalman estendido com transformada de Anscombe para <i>phantoms</i> para calibração.....	84
Tabela 6 – Resultados obtidos com o filtro de Kalman estendido com redes neurais para <i>phantoms</i> para calibração.....	89
Tabela 7 – Resultados obtidos com o filtro de Kalman descentralizado para <i>phantoms</i> para calibração. ....	92
Tabela 8 – Resultados obtidos com o filtro de Kalman descentralizado com a transformada de Anscombe para <i>phantoms</i> para calibração. ....	97
Tabela 9 – Resultados obtidos com o filtro de Kalman descentralizado com a transformada de Anscombe para <i>phantoms</i> para calibração. ....	100
Tabela 10 - Resultados obtidos com o filtro de Kalman descentralizado com redes neurais para <i>phantoms</i> para calibração.....	105
Tabela 11 - Resultados obtidos com o filtro de Kalman descentralizado com redes neurais para <i>phantoms</i> para calibração.....	105

---

## Sumário

---

Introdução .....	14
1 Tomografia reconstitutiva de raios X e $\gamma$ .....	18
1.1 Histórico da tomografia reconstitutiva .....	18
1.2 Características do minitomógrafo .....	20
1.3 Considerações sobre o ruído em tomografia.....	20
1.4 Transformada de Anscombe .....	27
2 Filtro de Kalman .....	29
2.1 Introdução .....	29
2.2 Estimação estocástica.....	30
2.3 Modelos de estado-espacial .....	30
2.4 O problema de observação.....	31
2.5 Ruído da medição e do processo.....	32
2.6 Estudando o filtro de Kalman .....	33
2.6.1 A origem computacional do filtro.....	34
2.6.2 A origem probabilística do filtro.....	35
2.7 O algoritmo do filtro de Kalman discreto .....	36
2.8 Filtro de Kalman aumentado.....	38
2.9 Filtro de Kalman-Bucy .....	39
2.10 Filtro de Kalman estendido .....	40
2.11 Filtro de Kalman descentralizado .....	42
2.11.1 Transformação descentralizada.....	42
2.11.2 Algoritmo do filtro de Kalman descentralizado.....	44
2.12 Aplicação do filtro de Kalman.....	47
2.13 Algoritmo de suavização Rauch-Tung-Striebel.....	48

3	Redes Neurais .....	54
3.1	Histórico das redes neurais artificiais .....	54
3.2	O modelo computacional .....	54
3.3	Função de transferência .....	59
3.4	Neurônio com um vetor de entrada.....	59
3.5	Multicamadas de neurônios .....	61
3.6	Comparação entre os algoritmos de treinamento das redes neurais .....	63
4	Resultados e Conclusões.....	68
4.1	Modelagem do processo .....	68
4.2	Phantom para caracterização e calibração .....	70
4.3	Apresentação da ferramenta para filtragem com Kalman.....	72
4.4	O Filtro de Kalman discreto para projeções tomográficas.....	75
4.5	Filtro de Kalman estendido com transformada de Anscombe para projeções tomográficas.....	80
4.6	Filtro de Kalman estendido com redes neurais para projeções tomográficas.....	84
4.7	Filtro de Kalman descentralizado para projeções tomográficas .....	90
4.8	Aplicação do filtro de Kalman descentralizado com redes neurais em uma amostra de solo 108	
4.9	Conclusões .....	111
	Referências Bibliográficas .....	113
	Apêndice A .....	117
	Probabilidade e Variáveis Aleatórias.....	117
	a. Probabilidade .....	117
	b. Variáveis aleatórias.....	117
	c. Média e Variância .....	118
	d. Distribuição Normal ou Gaussiana .....	120
	Independência contínua e probabilidade condicional .....	121
	a. Teorema de Bayes.....	122

b. Contínuo-Discreto.....	122
c. Características do Sinal Espacial versus Espectral .....	122
d. Ruído Branco .....	124
Apêndice B.....	125
Treinamento de uma rede neural.....	125
Princípios de treinamento de uma rede neural multi-camadas utilizando o algoritmo backpropagation.....	126

## Introdução

---

Neste trabalho é abordado o uso de filtro de Kalman para a filtragem de projeções tomográficas da ciência do solo. Essas projeções são assumidas como sinais variantes no espaço que são corrompidos por ruídos provenientes da fonte de emissão de fótons e ruídos ocasionados pela incerteza da contagem desses fótons pelo detector, além de ruídos devido à movimentação do equipamento e efeito térmico sobre componentes da eletrônica do sistema como um todo.

As projeções tomográficas podem ser entendidas como um conjunto de sinais armazenados em uma matriz, onde pode ser extraídos de suas linhas ou colunas, sinais individuais com variância no espaço e no tempo.

Um sinal pode, assim, ser considerado como uma função unidimensional  $f(x)$ , onde  $x$  corresponde às distâncias relativas ao tempo e sua amplitude indica a intensidade no ponto. Então,  $f(x)$  deve assumir um valor no intervalo  $-\infty < f(x) < \infty$ . Sinais reais são contínuos e podem possuir intensidade também em uma escala contínua. Mas, para haver a possibilidade de análise e tratamento desses sinais, é necessário discretizá-los espacialmente e em intensidade. A este processo chama-se digitalização, tendo como consequência um sinal digital. Conceitos como sinal digital, amostragem e vetor de sinal podem ser encontrados em literatura especializada (GONZALES 1987) (BRIGHAM 1988).

O uso da tomografia computadorizada na ciência do solo permite realizar um estudo não invasivo de uma amostra do solo para obter informações detalhadas de suas características como porosidade, densidade, umidade entre outras. Para a melhoria da qualidade da imagem final reconstruída, o uso da pré-filtragem se mostra importante ao evitar a perda de detalhes e garantia de uma imagem próxima ao corpo real, permitindo que detalhes que são perdidos devido à reconstrução ou tratados como partes do ruído sejam mantidos.

A medida de qualidade das imagens filtradas pode ser realizada com base no índice melhoria na relação sinal/ruído (ISNR - *Improvement in Signal Noise Ratio*) feita em cima de regiões de interesse (ROIs – *Regions of Interest*). A relação sinal/ruído é um termo para a relação em ter a potência de um sinal (informação significativa) e um ruído por trás desse sinal que pode ser dado pela relação entre a potência do sinal e a potência do ruído ou a amplitude dessa relação ao quadrado:



$$SNR = \frac{P_{\text{signal}}}{P_{\text{ruído}}} = \left( \frac{A_{\text{signal}}}{A_{\text{ruído}}} \right)^2 \quad (1)$$

Como alguns sinais têm altos valores, SNR é usualmente expressado em termos da escala logarítmica de decibéis:

$$SNR(\text{dB}) = 10 \log_{10} \left( \frac{P_{\text{signal}}}{P_{\text{ruído}}} \right) = 20 \log_{10} \left( \frac{A_{\text{signal}}}{A_{\text{ruído}}} \right) \quad (2)$$

O ISNR é a relação de melhoria em Sinal/Ruído. É uma relação entre o ruído da imagem degradada e o ruído da imagem restaurada e pode ser escrito como:

$$ISNR(\text{dB}) = 10 \log_{10} \left( \frac{P_{\text{ruído original}}}{P_{\text{ruído filtrado}}} \right) = 20 \log_{10} \left( \frac{A_{\text{ruído original}}}{A_{\text{ruído filtrado}}} \right) \quad (3)$$

Para um ruído variante no tempo a variância do maior valor de ISNR é uma representação mais confiável.

A tomografia computadorizada é apresentada no Capítulo 1 com uma abordagem para tratamento de sinais de com ruído *Poisson* utilizando a transformada de Anscombe. Filtragem se caracteriza por separar algo. No caso de sinais, tem o seu significado em separar algo indesejado do sinal, que se caracteriza pelo ruído presente nesses sinais, com o objetivo de obter um alto valor na relação sinal/ruído.

Existem duas categorias de filtros: adaptativos e não adaptativos. O primeiros filtros utilizados eram não adaptativos e são relacionados à circuitos elétricos, onde uma freqüência do sinal ruidoso poderia ser atenuada ou aumentada, permitindo a diminuição dos ruídos e aumento do sinal, respectivamente.

Esses filtros se caracterizam por filtros passa-baixas, passa-altas e passa-faixas, possíveis de serem implementados facilmente em circuitos elétricos. Pode-se filtrar o sinal de duas maneiras: pré-filtragem, que é feita antes de amostrar, removendo-se as freqüências maiores que a metade da freqüência de amostragem, e pós-filtragem, feita depois, quando se aumenta a freqüência da amostragem.

Para eliminar freqüências altas de um sinal, multiplica-se seu espectro por uma função “caixa”, pois a Transformada de Fourier de uma função caixa é a função *sinc*, cuja convolução do sinal com a função *sinc* resulta num filtro passa-baixa ideal. Mas, na prática, utilizam-se funções com *suporte limitado*, tais como Gaussiana, *sinc2* e triangular.

O filtro passa-baixa é uma solução para resolver o problema de sinal ou ruído, mas há perda de informações cruciais. Sistemas com diferentes fontes de ruído não apresentam uma solução plausível com o uso desse filtro. Há, portanto, necessidade do uso de filtros mais complexos.

O uso de filtros digitais se faz necessário desde que o sinal estudado é representado discretamente e devido à capacidade de tratar uma abordagem adaptativa e elevar a capacidade de filtragem para problemas não lineares para tratar problemas relacionados aos diferentes tipos de ruídos e distribuições das projeções dentro do sinal.

O filtro de Kalman é uma ferramenta matemática desenvolvida com base em conceitos como cadeias (ocultas) de Markov e estimação Bayesiana com a capacidade de obter estados futuros e ocultos dada a uma observação e a capacidade de se aprimorar com o uso de outras técnicas de estimação como redes neurais e algoritmos genéticos. Estes filtros são entendidos como extensões não lineares do filtro e são modificações feitas diretamente nas equações de para medição e correção do filtro.

A característica principal da filtragem linear está na capacidade do filtro de fazer uma predição utilizando alguma função linear conhecida. No caso do filtro discreto foi utilizada a matriz de translação, onde a diferença do estado futuro com o estado atual era estimado e o valor observado passa a ser a soma desses estados após serem corrigidos pelo filtro. A filtragem não linear pode ser feita através do uso de uma função não linear para esta essa estimação. No caso deste trabalho é feito com o uso de redes neurais que promovem um mapeamento não linear e o uso do próprio filtro para estimar esses pesos. O mapeamento não linear possui a capacidade de mapear funções com ordens maiores do que com mapeamento linear (BUCY 1970). O filtro de Kalman é abordado no Capítulo 2 com suas variações para filtragem linear e não linear.

Redes neurais são utilizadas atualmente em várias áreas de pesquisa devido ao seu poder de mapeamento de funções lineares e não lineares através de treinamentos onde se conheça apenas algumas entradas de dados e saídas conhecidas. São baseadas nos agrupamentos de neurônios artificiais (que, por sua vez são baseados nos funcionamentos dos neurônios de animais) que se interligam e se comunicação através de conexões entre eles baseado em camadas, que podem ter vários elementos, independente de cada camada. A possibilidade de conseguir mapear um comportamento ou funções fez com que o uso de redes neurais se popularizasse e que fossem criadas novas maneiras de treinar uma rede, sendo um treinamento supervisionado ou não, com retropropagação do sinal e até mesmo a forma de organizacional das distribuições dos valores de saída nos neurônios. Uma abordagem mais aprofundada de rede neural pode ser visto no Capítulo 3.

No Capítulo 4, os filtros são comparados em relação ao grau de filtragem e melhoria na qualidade da imagem, bem como são tratadas as conclusões deste trabalho.



# 1 Tomografia reconstitutiva de raios X e $\gamma$

---

## 1.1 Histórico da tomografia reconstitutiva

Em Radiologia, uma Tomografia Computadorizada (TC) é uma imagem que deriva do tratamento informatizado dos dados obtidos numa série de projeções angulares de raios X. Isto é, traduz uma secção transversal (uma “fatia”) do objeto em estudo. A TC, tal como a radiologia convencional, baseia-se no fato de que os raios X são apenas parcialmente absorvidos pelos diversos materiais. Enquanto materiais como plástico e água são facilmente atravessados pelos raios X, outros, a exemplo dos metais, não o são.

Largamente utilizada nas áreas médicas, seu uso em ciência do solo foi introduzido por Petrovic (PETROVIC, SIEBERT e RIEKE 1982), Hainswoth e Aylmores (HAINSWOTH e AYLMORE 1983) e por Crestana (CRESTANA 1985). Petrovic demonstrou a possibilidade de usar um tomógrafo computadorizado de raios X para medir a densidade de volumes de solos, enquanto Crestana demonstrou que a TC pode solucionar problemas ligados aos estudos da física da água no solo. A partir desses estudos, originou-se um projeto que envolveu o desenvolvimento de um tomógrafo para a ciência do solo. (P. CRUVINEL 1987) (CRUVINEL, CESAREO, et al. 1990)

O uso do computador na tomografia é essencial para a reconstrução da imagem a partir de projeções. Em 1917, o matemático austríaco Radon foi o primeiro a apresentar uma solução matemática das equações de reconstrução de corpos a partir de projeções determinando a função distribuição de densidade da região estudada através de suas projeções, no campo da gravitação (RADON 1917). Bracewell utilizou a reconstrução tomográfica para construir o mapa das regiões solares emissoras de microondas, pois as antenas de microondas existentes não permitiam focalizar um ponto, apenas pequenas faixas da superfície solar. O embrião da tomografia reconstitutiva de raios X pode ser encontrado nos trabalhos de Takahashi (TAKAHASHI, SHIMODA e TOWNES 1957), em que planos indesejáveis foram eliminados, colocando a fonte de raios X e o filme em um mesmo plano. Oldendorf, em 1961, desenvolveu um aparato rústico para obter imagens através da transmissão de raios  $\gamma$ . Em 1963, Cormack, professor de Física da University of Cape Town, desenvolveu a técnica matemática para reconstruir imagens utilizando o

método da retroprojeção (CORMACK 1963). Em 1956, Cormack, havia sido solicitado para supervisionar o uso de isótopos radioativos no Groote Schuur Hospital devido à demissão do físico do hospital. Durante algumas semanas o pesquisador trabalhou com os isótopos radioativos e acompanhou tratamentos de radioterapia. Com base em experimentos e observações, formulou uma matriz de coeficientes para cortes seccionais que poderia ser obtida pela medida da transmissão de raios X em vários ângulos através de um corpo com a obtenção de imagens de secções transversais, o que foi decisivo para a reconstrução das seções a partir de um número finito de projeções estudadas. Em aplicações médicas, o primeiro tomógrafo computadorizado de raios X de caráter comercial foi apresentado em 1973 pela EMI Ltda. (HOUSNFIELD 1973), causando um grande impacto para diagnóstico radiológico.

A Figura 1 ilustra o tomógrafo dedicado à ciência do solo instalado na Embrapa Instrumentação Agropecuária (São Carlos, SP – Rua XV de novembro 1452), o qual é baseado em fontes de raios X e raios  $\gamma$  para o estudo de amostras de solo e plantas, possibilitando também o uso de diversas fontes de radiação e intensidades energéticas. Vários trabalhos foram desenvolvidos para o aprimoramento da visualização das imagens adquiridas e do algoritmo de reconstrução, bem como do equipamento desenvolvido (VENTURINI 1995) (MINATEL 1997) (GRANATO 1998) (MASCARENHAS, SANTOS e CRUVINEL 1999).



**Figura 1. Mini-tomógrafo da Embrapa Instrumentação Agropecuária (CRUVINEL 1987).**

## 1.2 Características do minitomógrafo

Comparado com métodos clássicos como transmissão direta de raios  $\gamma$  e testes gravimétricos, o minitomógrafo tem a vantagem de medir a) heterogeneidades no interior do solo; b) a densidade do solo e c) o conteúdo de umidade pixel a pixel, além de obter uma imagem bi-dimensional ou tri-dimensional de amostras do solo de forma não invasiva e independente da geometria e forma, utilizando inclusive diferentes energias e fontes radioativas (CRUVINEL, CESAREO, et al. 1990).

Basicamente, uma TC indica a quantidade de radiação absorvida por cada porção da secção analisada, traduz essas variações numa escala de cinza e produz uma imagem. Como a capacidade de absorção de raios X de um material está intimamente relacionada com a sua densidade, zonas com diferentes densidades terão diferentes intensidades monocromáticas que podem ser visualizadas em cores, se for aplicada uma máscara de pseudocor, permitindo distingui-las claramente. Deste modo, cada valor do sinal corresponde à média da absorção dos tecidos nessa zona, expresso em Unidades de Hounsfield (em homenagem ao criador da primeira máquina de TC). A principal vantagem da TC é permitir o estudo de secções transversais de matérias sem intrusão, uma melhoria sem paralelo em relação às técnicas de análise do solo, que, em geral, são invasivas e podem destruir características importantes a serem observadas. A importância de uma qualidade de imagem perfeita previne que materiais encontrados no solo não sejam interpretados de maneira errônea. Essa qualidade é alcançada quando o material em questão é o mesmo encontrado por uma busca manual intrusiva.

## 1.3 Considerações sobre o ruído em tomografia

Tomografia é a ação de iluminar um objeto em várias direções proporcionais, após armazenagem, um conjunto de valores que representam amostras de um conjunto de projeções. Cada dado armazenado é na realidade uma média de alguns parâmetros inerentes do caminho de propagação do raio de iluminação.

As projeções podem ser provenientes de diversas fontes como, por exemplo, dos tradicionais raios X, dos raios  $\gamma$ , da ressonância magnética ou do ultra-som. Com base na intensidade emitida pela fonte de raios X e na intensidade captada por um detector na outra extremidade da reta de propagação, pode-se determinar o peso da atenuação devido ao

objeto que se faz presente entre fonte e o detector. Este peso da atenuação é um dado crucial para o processo de reconstrução, o qual possibilita obter um mapa de coeficientes de atenuação linear da secção transversal do objeto.

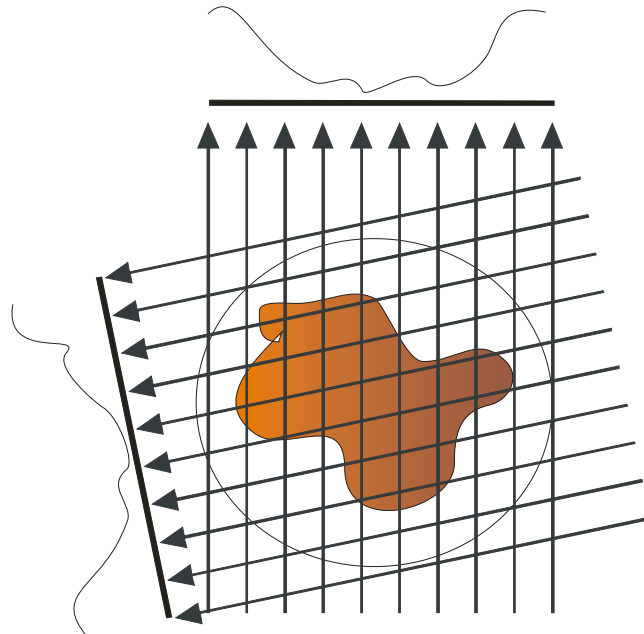
Esse mapa de coeficientes é representado através de pixels cujos valores são dados pelos chamados números CT (decorrente de *Computerized Tomography*). Esses números são normalizados em função do coeficiente de atenuação da água  $\mu_{H_2O}$ . Em outras palavras, definem-se os números CT pela equação 4:

$$\text{Número CT} = \frac{\mu - \mu_{H_2O}}{\mu_{H_2O}} \times 1000 \quad (4)$$

onde  $\mu$  é o coeficiente de atenuação do corpo analisado.

Com esse número é possível a obtenção de um mapa de coeficientes de atenuação, o que permite uma análise mais detalhada do corpo em estudo. Em medicina, convencionou-se que para a água o número CT é igual a 0 (zero).

O raio parte de uma fonte e é detectado por um emissor. A função de integração do objeto ao longo do raio é uma integral de linha. Cada conjunto de integrais de linha dos raios paralelos forma uma projeção paralela (Figura 2), a qual pode ser tratada com base na teoria de sinais.



**Figura 2. Duas projeções paralelas de um objeto expresso por uma função bidimensional.**

Para feixes de raios X na TC, três principais processos de interação da radiação com a amostra influenciam na atenuação do feixe. São eles os efeitos fotoelétricos, efeito Compton e efeito de produção de pares (CRUVINEL, CESAREO, et al. 1990).

Além das questões relacionadas aos efeitos mencionados que são decorrentes da faixa de energia utilizada na fonte, há outras questões que influenciam a medição em tomografia computadorizada. Dentre elas encontram-se a estatística da contagem de fótons.

A probabilidade de detecção de fótons em um intervalo de tempo de exposição  $t$  pode ser estimada pela função distribuição de probabilidade de Poisson, (DEREMACK e CROWE 1984) (CRUVINEL 1987)

$$Ps(\lambda, t) = \frac{(\bar{\lambda})^n}{\lambda! e^n} \quad (5)$$

onde  $\lambda$  é o número de fótons e  $\bar{\lambda}$  é a média de fótons emitidos no intervalo de tempo  $t$  e  $n$  está relacionado à eficiência da fotomultiplicadora, conforme a expressão abaixo:

$$\bar{\lambda} = \xi M t \quad (6)$$

onde  $M$  é a razão média de fótons (fótons/segundo) e  $\xi$  é a eficiência quântica da fotomultiplicadora.

A incerteza ou ruído é dado pelo desvio padrão

$$\sigma = \sqrt{\xi M t} = \sqrt{\bar{\lambda}} \quad (7)$$

Logo, a relação sinal-ruído, SNR, apresentada pelo sinal incidente, é:

$$SNR = \frac{\bar{\lambda}}{\sigma} = \sqrt{\bar{\lambda}} \quad (8)$$

Desta relação, é avaliado que, para um número pequeno de fótons, o ruído pode ser considerável, porém, com o aumento de  $\bar{\lambda}$ , o ruído pode vir a ser desprezível. A emissão de elétrons termiônicos no foto cátodo da multiplicadora causa um aumento nesse ruído.

Considerando que o foto-cátodo emite elétrons aleatoriamente em razão da corrente de cátodo incrementado pela expressão 9, tem-se a nova relação sinal-ruído, que é dada pela equação 10:

$$\sqrt{(\xi M + M)t} \quad (9)$$

$$SNR = \frac{\sqrt{t} \xi M}{\sqrt{\xi M + M}} \quad (10)$$

Na visualização de uma imagem tomográfica há presença de granulosidade, que é significativa para visualização de objetos de baixo contraste. O termo ruído em imagens



tomográficas refere-se à variação dos coeficientes de atenuação sobre o valor médio quando uma imagem é obtida de um objeto uniforme (HENDER 1983). O ruído na imagem pode ser baseado no cálculo do desvio padrão e também no espectro de potência do ruído de Wiener, no qual é visualizado como uma função da frequência espacial, permitindo observar a intensidade e o tipo de ruído envolvendo o sistema, influenciando a imagem obtida.

O ruído presente nas imagens TC inclui erros de arredondamento no programa de reconstrução (ruído do algoritmo), ruído eletrônico e ruído causado pelo sistema de visualização. A principal fonte de ruído em imagens TC é a variação quântica (*quantus mottle*), definida como a variação estatística espacial e temporal no número de fótons de raios X absorvidos no detector.

O ruído do algoritmo depende do tamanho do pixel do dispositivo de visualização e também influencia o ruído da imagem, assim, maiores pixels levam a uma redução do ruído na imagem, porém com perda na resolução. Algoritmos de reconstrução normalmente utilizam filtros de suavização que minimizam o efeito visual do ruído, acompanhado de alguma perda na resolução espacial.

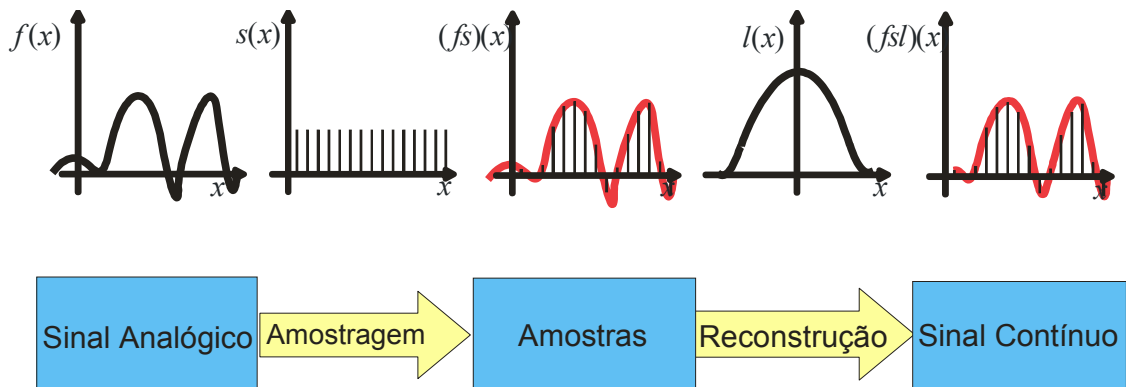
O ruído eletrônico pode ter origem em dispositivos eletrônicos não ideais, tais como resistores e capacitores não puros, terminais de contatos não ideais, corrente de fuga de transistores, efeito Joule, podendo ser também independente do sinal, como interferência externa (elétrica ou até mesmo mecânica) (ZIEL 1976).

Além dos ruídos, as imagens de TC estão sujeitas a diversos artefatos e distorções como fontes poli-cromáticas (ou monocromáticas não-ideais), atribuídos pelos efeitos conhecidos como endurecimento do feixe, *aliasing*, diferentes materiais em um mesmo voxel (volume parcial), deslocamento da amostra ou do equipamento (DUERINCKX e MACOCSKI 1978) (JOSEPH e SPITAL 1978) (IBBOTT 1980) (GRANATO 1998).

A amostragem corresponde em discretizar o sinal espacialmente ou por tempo, ou seja, selecionar um limitado número de pontos para representá-lo. Denomina-se quantização o processo de discretizar os valores que representam a intensidade no ponto (VENTURINI 1995).

Qualquer sinal no qual frequências maiores que  $f$  sejam nulas pode ser perfeitamente reconstruído se amostrado em uma frequência igual ou superior a  $2f$ . A frequência limite para amostragem é também conhecida como Limite de Nyquist.

A Figura 3 mostra um fluxograma de como é feito o processo de discretização e reconstrução do sinal (teorema da amostragem).



**Figura 3.** O sinal analógico é multiplicado por um trem de impulsos, onde é feita a amostragem em pontos. A reconstrução do sinal é feita com base nas amostras, o que depende diretamente da qualidade amostral.  $f(x)$  corresponde a uma função representando o sinal analógico,  $s(x)$  um trem de impulsos,  $(fs)(x)$  um sinal discretizado,  $l(x)$  uma função contínua e  $(fsl)(x)$  o sinal contínuo reconstruído.

Em aplicações de processamento de sinais, utiliza-se sua representação no espaço recíproco. A transformada de Fourier fornece a representação do sinal no domínio da frequência. A Transformada de Fourier de uma função contínua  $f(x)$ , onde  $x$  é uma variável real, pode ser simbolizada por  $F(u)$  como sendo:

$$F(u) = \frac{1}{2\pi} \int f(x) e^{-jux} dx \quad (11)$$

$f(x)$  pode ser encontrado a partir de  $F(u)$  pela transformada inversa de Fourier dada por:

$$f(x) = \int F(u) e^{jux} du \quad (12)$$

onde  $j^2 = -1$  e  $e^{-jux} = \cos ux - j \sin ux$ .

Estas duas equações são denominadas Pares de Transformada de Fourier (BRIGHAM 1988) e apenas existem se  $f(x)$  for uma função contínua e integrável e se  $F(u)$  for também integrável. Um sinal qualquer pode, assim, ser decomposto em uma soma (possivelmente infinita) de senóides e cossenóides.

A Transformada de Fourier viabiliza computar a distribuição (amplitudes, frequências e fases) dessas senóides e cossenóides (Figura 4). Uma das propriedades da transformada de Fourier é a convolução para duas funções  $f(x)$  e  $g(x)$  que são definidas pela integral:

$$f(x) * g(x) = \int f(x') g(x - x') dx' \quad (13)$$

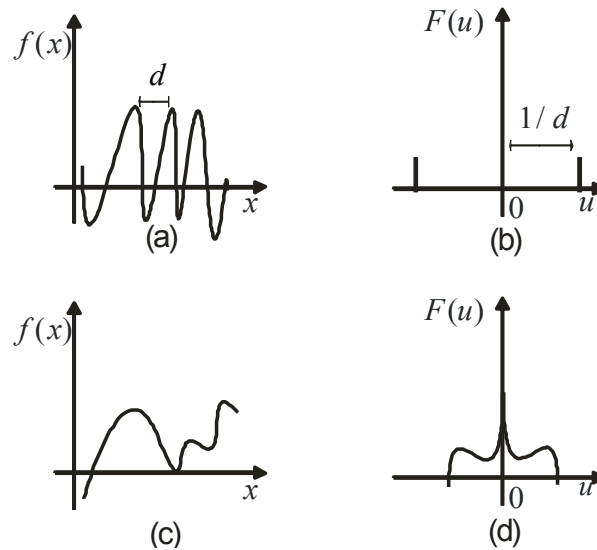
onde  $x'$  é a variável de integração. No domínio da frequência, pode-se escrever que

$$f(x) * g(x) \Leftrightarrow F(u) \cdot G(u) \quad (14)$$

ou ainda:

$$f(x) \cdot g(x) \Leftrightarrow F(u) * G(u) \quad (15)$$

pois convolução e multiplicação de funções são operações relacionadas.



**Figura 4.** A distância  $d$  (a) é representada de forma inversa no espectro de Fourier (b), O sinal (c) representado no domínio do tempo, e sua representação (d) no domínio da frequência.

Considerando um sinal contínuo unidimensional, sua amostragem e representação no espectro de Fourier são obtidas multiplicando o sinal por um trem de impulsos, igualmente espaçados  $s(x)$ , resultando  $f(x)s(x)$ , como ilustra a Figura 5.

Um sinal digital não pode assumir qualquer valor real, apenas valores discretos pré-definidos, uma vez que o espaço de memória disponível para armazená-lo é limitado. É feita então uma quantização, sendo que valores intermediários são aproximados. Considerando ainda o sinal  $f(x)$  e seu equivalente no domínio da frequência  $F(u)$ ,  $s'(x)$  representa a multiplicação do sinal por uma função de trem de impulsos de intervalos mais espaçados, isto é, um número menor de amostragem. No domínio da frequência, o intervalo entre cada ponto de  $S'(u)$  é menor do que  $S(u)$  (Figura 6).

A região de  $u$  onde  $F(u)$  possui valores significativos é limitada e quando é feita a amostragem, há uma sobreposição de altas e baixas frequências de  $F(u)$ , o que leva a uma distorção do sinal original, que se denomina por *aliasing* (do inglês *alias*, referencia ao sinal de “sombras”) (VENTURINI 1995). Para evitar o *aliasing*, adota-se a eliminação das altas frequências, ou seja, filtragem. A palavra filtragem (ou filtro) é utilizada pela sua idéia de remoção da parte indesejável de uma mistura e, no caso deste trabalho, é aplicada ao ruído presente em um sinal.

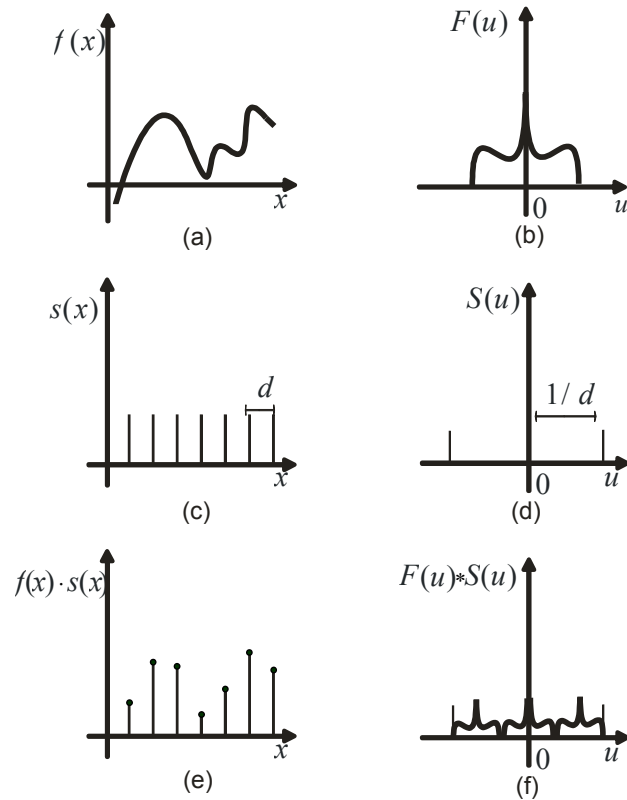


Figura 5. (a)  $f(x)$  é um sinal a ser amostrado, (b)  $F(u)$  é o seu espectro de Fourier equivalente no domínio da frequência (c)  $s(x)$  é um trem de impulso e (d)  $S(u)$  seu espectro de Fourier equivalente no domínio da frequência, (e)  $f(x) \cdot s(x)$  é o sinal amostrado, (f)  $F(u) * S(u)$  é seu espectro de Fourier equivalente no domínio da frequência.

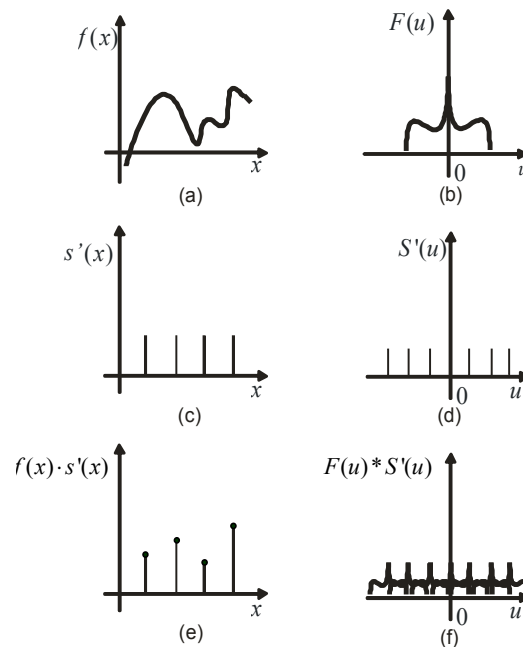


Figura 6. (a) O sinal representado por  $f(x)$  é amostrado com uma nova função de trem de impulsos com um intervalo maior entre os pontos (c). Isto provoca uma perda de qualidade do sinal (e) e a presença de aliasing (f) no domínio da frequência.

## 1.4 Transformada de Anscombe

A Transformada de Anscombe consiste em uma transformação não linear do ruído Poisson, que é dependente do sinal, em um ruído independente do sinal, com (aproximadamente) média nula e variância unitária (INOUE 1971) seguindo uma distribuição gaussiana (ANSCOMBE 1948) (MASCARENHAS, SANTOS e CRUVINEL 1999).

A Transformada de Anscombe de uma variável aleatória  $x$  com distribuição de Poisson é dada por:

$$z = 2\sqrt{x + \frac{3}{8}} \quad (16)$$

E ainda pode ser representada por

$$z = 2\sqrt{x + \frac{1}{8}} + n = s + n \quad (17)$$

onde  $n$  representa um ruído aproximadamente independente de  $s$ , descrito por uma distribuição gaussiana  $n \sim N(0,1)$ .

Assim sendo, é possível demonstrar que a média de  $x$  é aproximadamente

$$E[\bar{z}] = 2\sqrt{E[\bar{x}] + \frac{1}{8}} \quad (18)$$

e sua variância é dada por

$$\sigma_z^2 = \sigma_s^2 + 1. \quad (19)$$

Podendo usar uma abordagem para definir o valor do sinal, como

$$\sigma_s^2 = \sigma_z^2 - 1. \quad (20)$$

Mas essa abordagem não é muito confiável devido à possibilidade de se obter valores negativos para a variância. Após a redução do ruído deve-se fazer uma transformada inversa de Anscombe, para obter, desse modo uma estimativa do sinal com o ruído Poisson reduzido. Essa transformada pode ser obtida por:

$$b = \frac{1}{4}\hat{s}^2 - \frac{1}{8}. \quad (21)$$

Essa abordagem é pontual, não tendo correlação com *pixels* e *voxels*.

Neste trabalho se estuda uma solução para essa filtragem que leva em consideração as possíveis variabilidades temporais da relação Sinal/Ruído (S/R) das projeções tomográficas com base no filtro de Kalman utilizado em conjunto com redes neurais artificiais, ambos apresentados nos próximos capítulos.

## 2 Filtro de Kalman

---

### 2.1 Introdução

O filtro de Kalman é uma ferramenta matemática largamente utilizada para problemas estatísticos e é considerado um estimador ótimo para uma larga classe de problemas e um efetivo e útil estimador para outras classes.

Criado há mais de 30 anos, recentemente tem sido utilizado em computação gráfica para aplicações que envolvem simulação de instrumentos musicais em realidade virtual (VR), extração de movimentos dos lábios de locutores em seqüências de vídeo entre outras aplicações (PEREIRA 2000). Em 1960, Rudolf Emil Kalman publicou um artigo descrevendo uma solução recursiva para o problema de filtragem linear de dados discretos (KALMAN 1960).

Sua primeira aplicação prática foi encontrada por Stanley F. Shimidt, que trabalhava no projeto Apollo da NASA e tinha como objetivo levar uma nave à Lua e trazê-la de volta. Para isso, ele tinha que resolver problemas na estimação de trajetórias e controle. Shimidt trabalhou no que seria a primeira implementação completa do Filtro de Kalman e tornou o mesmo integrante do sistema de controle da Apollo, e, partir desta experiência, ele vem sendo usado em grande parte dos sistemas *on-boards* de estimação de trajetória e controle em aeronaves.

O filtro de Kalman é considerado um avanço da teoria de estimação do séc. XX. Ele é utilizado, por exemplo, para estimar um problema linear-quadrático gaussiano, que trata do problema de estimação de estados instantâneos de um sistema linear dinâmico perturbado por ruído Gaussiano branco usando-se medições linearmente relacionadas aos estados e também corrompidas por esse ruído. Este filtro se caracteriza por um conjunto de equações matemáticas que provê uma significativa solução computacional eficiente (recursivamente) para estimar o estado de um processo, de modo que minimiza a média do erro quadrático. O filtro permite que estados passados, presentes e até mesmo futuros possam ser estimados e pode fazê-lo até mesmo quando a natureza precisa do sistema modelado é desconhecida (WELCH e BISHOP 2004). Noções sobre probabilidades e variáveis aleatórias podem ser vistas no apêndice A, no final deste trabalho.

## 2.2 Estimação estocástica

Enquanto existem várias aplicações específicas que se aproximam para estimar um estado desconhecido de um conjunto de medidas de processos, vários destes métodos não inerentemente levam em consideração a natureza típica do ruído das medidas. Por exemplo, considere-se trabalhar em mapeamento para uma computação gráfica interativa. Enquanto os requerimentos para a informação variam com aplicação, a fonte fundamental de informação é a mesma: estimar poses que são derivadas de medidas elétricas ruidosas de sensores mecânicos, inertes, ópticos, acústicos ou magnéticos. Este ruído é estatisticamente típico na natureza (ou pode ser efetivamente modelado assim), que leva para métodos estocásticos para endereçar problemas.

## 2.3 Modelos de estado-espacial

Modelos de Estado-Espacial são essencialmente conveniências notacionais para problemas de estimação e controle, desenvolvidos para fazer tratável o que poderia, por outro lado, ser uma análise notacionalmente intratável. Considere um processo dinâmico descrito por uma equação diferencial de  $n$ -ésima ordem (similarmente uma equação diferencial) de forma

$$y_{i+1} = a_{0,i}y_i + \dots + a_{n-1,i}y_{i-n+1} + u_i, i \geq 0, \quad (22)$$

onde  $\{u_i\}$  é um processo aleatório ruidoso branco (espectralmente) com média zero (estatisticamente) com autocorrelação

$$E(u_i, u_j) = R_u = Q_i \delta_{ij}, \quad (23)$$

e os valores iniciais  $\{y_0, y_{-1}, \dots, y_{-n+1}\}$  são variáveis aleatórias com média zero com uma conhecida matriz de covariância  $n \times n$

$$P_0 = E(y_{-j}, y_{-k}), j, k \in \{0, n-1\} \quad (24)$$

Também se assume que

$$E(u_i, u_j) = 0 \quad (25)$$

para  $-n+1 \leq j \leq 0$  e  $i \geq 0$ , que assegura que

$$E(u_i, u_j) = 0, i \geq j \geq 0. \quad (26)$$



Em outras palavras, este sinal é estatisticamente independente do processo a ser estimado. Sob qualquer outra condição básica, esta equação diferencial pode ser reescrita como

$$\bar{x}_{i+1} \equiv \begin{bmatrix} y_{i+1} \\ y_i \\ y_{i-1} \\ \vdots \\ y_{i-n+2} \end{bmatrix} = \underbrace{\begin{bmatrix} a_0 & a_1 & \cdots & a_{n-2} & a_{n-1} \\ 1 & 0 & \cdots & 0 & 0 \\ 0 & 1 & \cdots & 0 & 0 \\ \vdots & \vdots & \cdots & \vdots & \vdots \\ 0 & 0 & \cdots & 1 & 0 \end{bmatrix}}_A \underbrace{\begin{bmatrix} y_i \\ y_{i-1} \\ y_{i-2} \\ \vdots \\ y_{i-n+1} \end{bmatrix}}_{\bar{x}_i} + \underbrace{\begin{bmatrix} 1 \\ 0 \\ 0 \\ \vdots \\ 0 \end{bmatrix}}_G u_i \quad (27)$$

que leva ao modelo estado-espacial, ou seja:

$$\bar{x}_{i+1} = A\bar{x}_i + Gu_i \quad (28)$$

$$\bar{y}_i = [1 \ 0 \ \dots \ 0]\bar{x}_i \quad (29)$$

ou a forma mais geral

$$\bar{x}_{i+1} = A\bar{x}_i + Gu_i \quad (30)$$

$$\bar{y}_i = H_i\bar{x}_i. \quad (31)$$

A equação 30 representa o caminho de um novo estado  $\bar{x}_{i+1}$  que é modelado com uma combinação linear de ambos os estados anteriores  $\bar{x}_i$  e algum ruído processual  $u_i$ . A equação 31 descreve o modo como as medições do processo ou observações  $\bar{y}_i$  são derivadas do estado interno  $\bar{x}_i$ . Estas duas equações sempre se referem respectivamente a modelos de processos e modelos de medição, e elas servem como a base para todos os métodos de estimação, como o Filtro de Kalman descreve abaixo (WELCH e BISHOP 2004).

## 2.4 O problema de observação

Existe um problema geral relatado na área da teoria de sistemas lineares geralmente chamados de problema de observação (*observer design problem*). O problema básico é determinar (estimar) os estados internos de um sistema linear, tendo acesso somente às saídas do sistema. É informalmente conhecido como o problema da “caixa preta”, em que se acessam alguns sinais vindos da caixa (as saídas), mas não se pode diretamente observar

o que há dentro. As várias aproximações para este problema básico são tipicamente baseadas no modelo estado-espacial apresentado anteriormente. Existe um modelo de processo que modela a transformação dos estados do processo. Isto pode ser usualmente representado como uma equação diferencial estocástica linear similar à equação 30:

$$x_k = Ax_{k-1} + Bu_k + w_{k-1}. \quad (32)$$

Em adição existe alguma forma de modelo de medição que descreve um relacionamento entre o estado do processo e as medidas. Isto pode ser usualmente representado com uma expressão linear semelhante à equação 31:

$$z_k = Hx_k + v_k. \quad (33)$$

Os termos  $w_{k-1}$  e  $v_k$  são variáveis aleatórias representando o ruído do processo e das medidas respectivamente. Note que na equação 33 foi modificada a variável dependente para  $z_k$  ao invés de  $y_k$  como na equação 31, para reforçar a noção de que as medidas não têm que ser elementos dos estados especificamente, mas podem ser qualquer combinação linear dos elementos do estado.

## 2.5 Ruído da medição e do processo

Considere o caso comum de medidas num sensor ruidoso. Existem várias fontes de ruídos nestas medidas. Por exemplo, cada tipo de sensor tem limitações fundamentais relacionadas ao meio físico associado e, quando empurramos o invólucro destas limitações, os sinais são tipicamente degradados. Assim, algum amontoado de ruídos elétricos aleatórios é adicionado ao sinal via sensor e circuitos elétricos.

A taxa variação de tempo do sinal “puro” para o ruído elétrico continuamente afeta a quantidade e a qualidade da informação.

O resultado é que a informação obtida de qualquer um dos sensores pode ser qualificada e interpretada como parte de uma seqüência de estimativas globais e modelos de medidas analíticas tipicamente incorporam alguma noção de ruídos aleatórios medidos ou incertos, como mostrado acima.

Existe o problema adicional de que o modelo de transformação do estado atual é completamente desconhecido, enquanto podem-se fazer previsões sobre pequenos

intervalos relativos usando modelo baseado em transformações recentes de estado, como predições que assumem as transformações que são previsíveis, o que não é sempre o caso.

O resultado é como a informação do sensor, em que estimativas dos estados podem ser qualificadas como elas são combinadas com as medidas da seqüência de estimativas globais.

Assim, os modelos processuais tipicamente incorporam alguma noção de movimentação aleatória ou incertezas, como mostrado acima.

## 2.6 Estudando o filtro de Kalman

Existem vários modelos, como o aumentado, estendido, estendido com estimação dual, descentralizado etc. (PEREIRA 2000) (WELCH e BISHOP 2004). O filtro de Kalman é essencialmente um conjunto de equações matemáticas que implementa um estimador do tipo corrige-prevê, que é ótimo no sentido de minimizar a covariância de erro estimada – quando alguma condição presumida é encontrada.

O processo a ser estimado resolve o problema geral de estimar um estado  $x \in \mathfrak{R}^n$  de um processo controlado em um tempo discreto que é gerado por uma equação linear de diferenças estocástica, ou seja:

$$x_k = Ax_{k-1} + Bu_k + w_{k-1}, \quad (34)$$

com uma medição  $z \in \mathfrak{R}^m$  que é

$$z_k = Hx_k + v_k. \quad (35)$$

As variáveis aleatórias  $w_{k-1}$  e  $v_k$  representam o ruído do processo e da medida (respectivamente). Elas assumem ser independentes (entre si), brancas e com a distribuição de probabilidade normal:

$$P(w) \sim N(0, Q), \quad (36)$$

$$P(v) \sim N(0, R). \quad (37)$$

Na prática, as matrizes da covariância do ruído do processo  $Q$  e da covariância do ruído da medição  $R$  podem mudar com cada passo de tempo ou medida, contudo aqui assumimos que elas são constantes.

A matriz  $A_{n \times n}$  na equação diferencial 34 relata o estado do tempo anterior do passo (k-1) para o estado atual do passo (k), na ausência de qualquer função direcionada ou ruído

do processo. De acordo com Welch, (WELCH e BISHOP 2004), na prática,  $A_{n \times n}$  pode mudar a cada estado, mas assume-se que  $A_{n \times n}$  é constante. A matriz  $B_{n \times 1}$  relata a entrada opcional do controle  $u \in \mathfrak{R}^l$  para o estado  $z_k$ . A matriz  $H_{m \times n}$  é a equação de medição 35 que relata o estado para a medida  $z_k$ . Na prática,  $H_{m \times n}$  pode mudar, mas também se assume como uma constante.

### 2.6.1 A origem computacional do filtro

Quando se define  $\hat{x}^- \in \mathfrak{R}^n$  (note o -) para ser o estado *a priori* estimado no passo  $k$  dado conhecimento do processo prévio para o passo  $k$ , e  $\hat{x} \in \mathfrak{R}^n$  para ser o estado *a posteriori* estimado no passo  $k$  dada a medida  $z_k$ , pode ser então definida uma estimativa de erros *a priori* e *a posteriori* como:

$$e_k^- \equiv x_k - \hat{x}_k^- \quad (38)$$

e

$$e_k \equiv x_k - \hat{x}_k . \quad (39)$$

A covariância da estimativa de erros *a priori* é então

$$P_k^- = E[e_k^- e_k^{-T}], \quad (40)$$

e o covariância da estimativa de erros *a posteriori* é

$$P_k = E[e_k e_k^T]. \quad (41)$$

Na derivação destas equações para o filtro de Kalman, começa-se com o objetivo de achar uma equação que compute um estado de estimativa *a posteriori*  $\hat{x}_k$  como uma combinação linear de uma estimativa *a priori*  $\hat{x}_k^-$  e uma diferença pesada entre a atual medida  $z_k$  e uma predição de medida  $H\hat{x}_k^-$  como mostrado abaixo na equação 42. Alguma justificativa para esta equação é dada na próxima seção.

$$\hat{x}_k = \hat{x}_k^- + K(z_k - H\hat{x}_k^-) \quad (42)$$

A diferença  $(z_k - H\hat{x}_k^-)$  na equação é chamada de medida de inovação, ou residual. O residual reflete a discrepância entre a medida prevista  $H\hat{x}_k^-$  e a medida atual  $z_k$ . O residual de zero significa que as duas estão em completo aceitação.

A matriz  $K_{n \times m}$  na equação 42 é escolhida para ser o ganho ou fator de mistura que minimiza a equação de erro da covariância a posteriori da equação 41. Esta minimização pode ser acoplada pela equação de substituição primária na definição acima para  $e_k$ , substituindo-o nesta última equação, atendendo as expectativas indicadas, levando a derivada para a descoberta do resultado respectivo a  $K_{n \times m}$ , colocando este resultado igual a zero, e então o resolvendo para  $K_{n \times m}$ . Uma forma do resultado de  $K_{n \times m}$  que minimiza esta equação é dada por

$$K_k = P_k^- H^T (H P_k^- H^T + R)^{-1}. \quad (43)$$

Tomando esta última equação, vê-se que as medidas da covariância de erro R aproximam de zero, o ganho de K pesa o residual mais pesado. Especificamente,

$$\lim_{R_k \rightarrow 0} K_k = H^{-1}. \quad (44)$$

Por outro lado, como a covariância de estimativa de erro  $P_k^-$  aproxima de zero, o ganho  $K_k$  pesa o residual menos pesado. Especificamente

$$\lim_{P_k^- \rightarrow 0} K_k = 0. \quad (45)$$

Outro modo de pensar sobre pesagem por  $K_k$  é que como a medida da covariância de erro R aproxima de zero, a medida atual  $z_k$  é cada vez mais confiável, enquanto a medida prevista (predita)  $H\hat{x}_k^-$  é cada vez menos confiável. Por outro lado, como a covariância de estimação de erro a priori  $P_k^-$  aproxima de zero a medida atual  $z_k$  é cada vez menos confiável, enquanto a medida prevista  $H\hat{x}_k^-$  é confiável cada vez mais.

### 2.6.2 A origem probabilística do filtro

A justificativa para a equação 42 é originada na probabilidade da estimativa do *a priori*  $\hat{x}_k^-$  condicionado para todas as medidas anteriores  $z_k$  (teorema de Bayes). Neste

momento, se aceita este princípio como suficiente para apontar que o filtro mantém os dois momentos da distribuição de estados,

$$E[x_k] = \hat{x}_k \quad (46)$$

$$E[(x_k - \hat{x}_k)(x_k - \hat{x}_k)^T] = P_k. \quad (47)$$

A equação estimativa do estado a posteriori reflete a média (o primeiro momento) da distribuição do estado – ela é normalmente distribuída se as condições da equação  $P(w) \sim N(0, Q)$  e  $P(v) \sim N(0, R)$  forem encontradas. A equação de estimativa da covariância de erro a posteriori equação 31 reflete a variância da distribuição do estado (o segundo momento não central). Em outras palavras,

$$p(x_x | z_k) \sim N(E[x_k], E[(x_k - \hat{x}_k)(x_k - \hat{x}_k)^T]) = N(\hat{x}_k, P_k). \quad (48)$$

## 2.7 O algoritmo do filtro de Kalman discreto

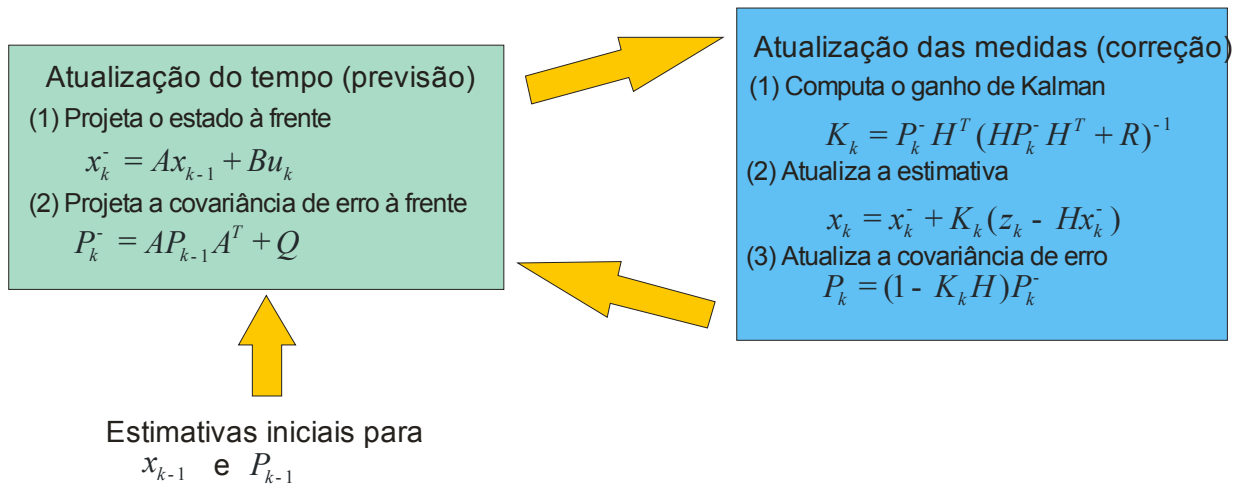
O objeto de interesse deste trabalho é entender o uso do filtro de Kalman e o uso do algoritmo para separar um ruído de um sinal, mais adiante. Será utilizada uma visão de alto nível e depois será especificada cada equação do algoritmo.

O filtro de Kalman discreto (FKD) estima um processo utilizando a forma de realimentação (*feedback*). O filtro estima o estado do processo em algum tempo e então obtém realimentação na forma de medidas (ruidosas). Assim, as equações podem ser divididas em dois momentos: As equações para a atualização do tempo e as equações para a atualização das medidas.

As equações para a atualização do tempo são responsáveis para a projeção *a priori* (no tempo) das estimativas do estado atual e da covariância de erros para obter as estimativas *a priori* para o passo do próximo tempo.

As equações para a atualização das medidas ou correção são responsáveis para a realimentação – por exemplo, incorporação de uma nova medida na estimativa *a priori* para obter uma estimativa *a posteriori*.

As equações para atualização do tempo ou previsão podem ser também pensadas como equações preditoras, enquanto as equações para atualização das medidas podem ser tratadas como equações corretoras. Com isso, a estimação final do algoritmo junta o que é de um algoritmo preditor-corretor para resolver problemas numéricos como mostra a Figura 7.



**Figura 7. Algoritmo para filtro de Kalman.** Com as estimativas iniciais para o primeiro valor da medida e a probabilidade inicial, o algoritmo atua de maneira recursiva, alternando entre a previsão dos dados e a correção.

Equações de atualização do tempo (preditor)

$$\hat{x}_k^- = A\hat{x}_{k-1} + Bu_k \quad (49)$$

$$P_k^- = AP_{k-1}A^T + Q. \quad (50)$$

Nota-se que estas equações projetam as estimativas do tempo e da covariância através do tempo no passo  $k-1$  para o passo  $k$ .

Equações de atualização das medidas (corretor)

$$K_k = P_k^- H^T (HP_k^- H^T + R)^{-1} \quad (51)$$

$$\hat{x}_k = \hat{x}_k^- + K_k (z_k - H\hat{x}_k^-) \quad (52)$$

$$P_k = (I - K_k H)P_k^- \quad (53)$$

A primeira tarefa durante a atualização da medida é computar o ganho de Kalman,  $K_k$ . O próximo passo é atualizar para medir o processo e obter  $z_k$ , e então gerar um estado estimado a posteriori por incorporação à medida na equação 52. O passo final é obter uma estimada covariância de erro via a equação 53.

## 2.8 Filtro de Kalman aumentado

Em (PEREIRA 2000) está definida outra variação para o filtro de Kalman, o filtro de Kalman aumentado (FKA). Como no Filtro de Kalman discreto, assume-se que os modelos do processo e de observação estavam corrompidos apenas por ruído branco. Entretanto, muitos sensores produzem observações que estão correlacionadas no tempo. Ruídos deste tipo são chamados ruídos coloridos. O efeito desta correlação é a violação das considerações feitas para o funcionamento do Filtro de Kalman, que causará polarização nos estados estimados.

Entretanto, é possível modelar a dinâmica do processo de medição para evitar que isto aconteça.

Se o ruído colorido pode ser gerado passando-se um ruído branco por um sistema linear, pode-se utilizar um método em que a entrada do filtro é um ruído branco e sua saída um ruído colorido, logo, esse método pode ser utilizado para modelar um erro de medição. O sistema total deverá ser aumentado com um modelo de observação formando um novo modelo que satisfaça as condições impostas pelo Filtro de Kalman (WELCH e BISHOP 2004).

O modelo proposto por Kalman é

$$x_k = Ax_{k-1} + Bu_k + w_{k-1} \quad (54)$$

onde  $w_{k-1}$  é ruído branco.

O modelo de observações agora é dado por

$$z_k = Hx_k + v_k + n_k \quad (55)$$

onde  $v_k$  é ruído branco mas  $n_k$  é uma seqüência aleatória colorida. Estas observações não satisfazem os critérios exigidos pelo filtro de Kalman. Mas pode-se assumir que  $n_k$  é a saída de um sistema linear do tipo:

$$x_{ck} = A_c x_{ck-1} + w_{ck-1} \quad (56)$$

$$n_k = H_c x_{ck} \quad (57)$$

onde o termo  $w_c$  é ruído branco.

Agrupando estas equações às anteriores tem o sistema aumentado:



$$\begin{bmatrix} x_k \\ x_{ck} \end{bmatrix} = \begin{bmatrix} A & 0 \\ 0 & A_c \end{bmatrix} \begin{bmatrix} x_{k-1} \\ x_{ck-1} \end{bmatrix} + \begin{bmatrix} B \\ 0 \end{bmatrix} u_{k-1} + \begin{bmatrix} w_{k-1} \\ w_{ck-1} \end{bmatrix} \quad (58)$$

$$z_k = \begin{bmatrix} H & H_c \end{bmatrix} \begin{bmatrix} x_k \\ x_{ck} \end{bmatrix} + v_k \quad (59)$$

que satisfaz as considerações feitas para o Filtro de Kalman discreto. Um FKA pode ser agora implementado e sintonizado como um filtro convencional. Nota-se que o sistema pode ser utilizado para qualquer tipo de ruído representado pelas equações 56 e 57.

## 2.9 Filtro de Kalman-Bucy

Pode-se utilizar o Filtro de Kalman-Bucy (FKB), que é uma forma análoga ao caso discreto, como estimador para os estados de um processo contínuo no tempo. Neste caso, o sistema pode ser modelado como

$$\dot{x}_t = Ax_t + Bu_t + Gw_t \quad (60)$$

$$z_t = Hx_t + v_t \quad (61)$$

$$E[w_t] = E[v_t] = 0 \quad (62)$$

$$E[w_{t_1} w_{t_2}^T] = Q_t \delta_{t_2-t_1} \quad (63)$$

$$E[v_{t_1} v_{t_2}^T] = R_t \delta_{t_2-t_1} \quad (64)$$

onde  $A_{n \times n}$ ,  $B_{n \times 1}$ ,  $G_{n \times n}$ ,  $H_{m \times n}$ ,  $Q_{n \times n}$  e  $R_{m \times m}$ .

Deseja-se encontrar uma estimação para o vetor de estados  $x_t \in \mathfrak{R}^n$ , representado por  $\hat{x}_t$ , que minimize a equação:

$$E\{[x_t - \hat{x}_t]^T [x_t - \hat{x}_t]\} \quad (65)$$

Como as equações para o Filtro de Kalman discreto são conhecidas, utiliza-se um método de linearização para a determinação das equações:

$$K_t = P_t H^T R_t^{-1} \quad (66)$$

$$\dot{\hat{x}}_t = A\hat{x}_t + Bu_t + K_t [z_t - H\hat{x}_t] \quad (67)$$

$$\dot{P}_t = AP_t + P_t A^T - K_t H P_t + G Q G^T \quad (68)$$

O FKB é implementado, geralmente, em computadores digitais, assim há necessidade de algoritmos de integração numérica, como os métodos de Runge-Kutta, para a implementação deste estimador. Assim como os filtros de Kalman estendido e aumentado para o caso discreto, se o sistema a ser estudado não for linear, o mesmo processo adotado no filtro de Kalman estendido pode ser aplicado no FKB, obtendo-se assim um Filtro de Kalman aumentado contínuo (PEREIRA 2000).

## 2.10 Filtro de Kalman estendido

Uma solução para sistemas não lineares é o filtro de Kalman estendido (FKE) (WELCH e BISHOP 2004). Analisando a função de predição do filtro de Kalman, equações 54 e 56, é possível observar que o filtro se comporta de forma linear. Aplicando uma função não-linear, pode-se obter uma predição ótima para os próximos estados, o filtro de Kalman Estendido. Este algoritmo aplica o filtro de Kalman para sistemas não lineares simplesmente linearizando todos os modelos não-lineares, assim as equações do filtro tradicional podem ser aplicadas. O sistema não linear pode ser reescrito na forma:

$$x_k = F(x_{k-1}, u_k, n_{k-1}), \quad (69)$$

$$z_k = H(x_k, v_k). \quad (70)$$

O algoritmo também é adaptado para resolver problemas não-lineares, como visto na Figura 8

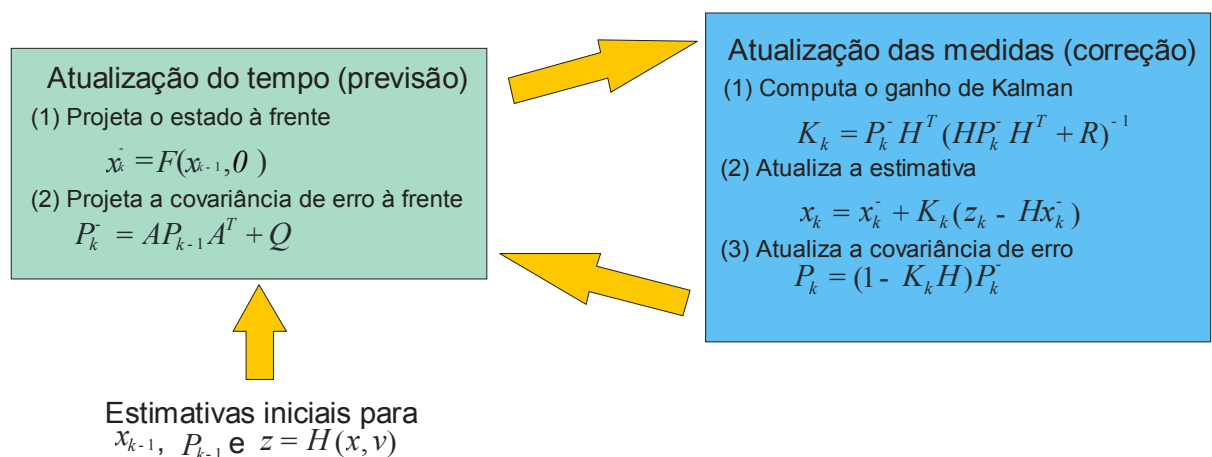


Figura 8. Algoritmo do filtro de Kalman Estendido com as funções para problemas não-lineares.

Para a propagação das variâncias, deve-se conhecer as matrizes jacobianas ou hessianas das funções de transição e observação de estado.

O principal problema é quando se deve identificar as funções necessárias para estimação dos estados e dos parâmetros. As equações para predição passam a ser descritas como:

$$x_k = F(x_{k-1}, w_k) \quad (71)$$

$$z_k = H(x_k, v_k) \quad (72)$$

A estimação de parâmetros envolve a determinação de um mapeamento não linear

$$y_k = G(x_k, W) \quad (73)$$

onde  $x_k$  é a entrada,  $W$  é o peso e  $y_k$  é a saída e o mapeamento não linear  $G$  é parametrizado pelo vetor  $W$ . O mapeamento não linear pode ser feito por uma rede neural artificial, onde  $W$  é o peso. O aprendizado corresponde à estimação dos parâmetros de  $W$ . O treinamento pode ser feito com pares de amostras, consistindo em uma entrada conhecida e saída desejadas.  $(x_k, d_k)$ . O erro de máquinas é definido pela equação 74. O objetivo do aprendizado é minimizar o erro quadrático esperado.

$$e_k = d_k - G(x_k, W) \quad (74)$$

Usando um modelo para treinamento da rede como feito neste trabalho, o FKE pode ser usado para estimar os parâmetros escrevendo uma nova representação estado-espacial:

$$w_k = w_{k-1} + v_k \quad (75)$$

$$y_k = G(x_k, w_k) + e_k \quad (76)$$

onde os parâmetros  $w_k$  correspondem a um processo estacionário com uma matriz identidade de transição de estado, governado por um ruído processual  $v_k$  (a escolha da variância determina a performance da filtragem). A saída  $y_k$  corresponde a uma observação não linear em  $w_k$ . O filtro de Kalman estendido pode ser aplicado diretamente como uma eficiente técnica de segunda ordem para correção dos parâmetros.

Como o problema em questão consiste em trabalhar sobre uma entrada  $x_k$  não observada e requer uma união entre a estimação dos estados e parâmetros, deve-se considerar um problema de estimação dual, considerando uma dinâmica discreto-temporal do sistema não linear:

$$x_{k+1} = F(x_k, v_k, W) \quad (77)$$

$$z_k = H(x_k, n_k, W) \quad (78)$$

onde ambos os estados dos sistema  $x_k$  e o conjunto dos parâmetros  $w$  para o sistema dinâmico devem ser estimados somente do sinal ruidoso  $y_k$ .

O sistema dinâmico pode ser entendido como uma rede neural, onde  $W$  é o conjunto de pesos. Assim, aplicando essas equações ao filtro de Kalman estendido, tem-se uma nova função para estimação e para observação.

## 2.11 Filtro de Kalman descentralizado

O filtro de Kalman descentralizado é semelhante à versão estendida (JULIER e UHLMANN 1997). A distribuição de estados é ainda representada por uma variável aleatória gaussiana, mas agora é especificada usando um mínimo de conjuntos de pontos amostrados escolhidos cuidadosamente. Os pontos amostrados capturam a verdadeira média e covariância da variável aleatória, e quando propagados através de um sistema verdadeiramente não linear, capturam a média e covariância posteriores de forma exata para terceira ordem para qualquer não linearidade. Para tanto, se utilizam da transformação descentralizada.

### 2.11.1 Transformação descentralizada

A transformação descentralizada (*Unscented Transformation*) é um método para calcular as estatísticas de uma variável aleatória que se submete a uma transformação não linear (WAN e MERWE 2000). Considerando uma propagação de uma variável aleatória  $x$  (com dimensão  $L$ ) através de uma função não linear  $y = g(x)$ . Assumindo que  $x$  tem média  $\bar{x}$  e covariância  $P_x$ . Para calcular as estatísticas de  $y$ , deve-se formar uma matriz  $X$  de  $2L+1$  vetores *sigma*  $x_i$  (com pesos correspondentes  $W_i$ ), de acordo o que segue:

$$X_i = \bar{x} \quad (79)$$

$$X_i = \bar{x} + (\sqrt{(L + \lambda)P_x})_i, \text{ para } i = 1, \dots, L \quad (80)$$

$$X_i = \bar{x} - (\sqrt{(L + \lambda)P_x})_i, \text{ para } i = L + 1, \dots, 2L \quad (81)$$

$$W_o^{(m)} = \lambda / (L + \lambda) \quad (82)$$

$$W_o^{(c)} = \lambda / (L + \lambda) + (1 - \alpha^2 + \beta) \quad (83)$$

$$W_i^{(m)} = W_i^{(c)} = 1 / \{2(L + \lambda)\}, \text{ para } i = 1, \dots, 2L \quad (84)$$

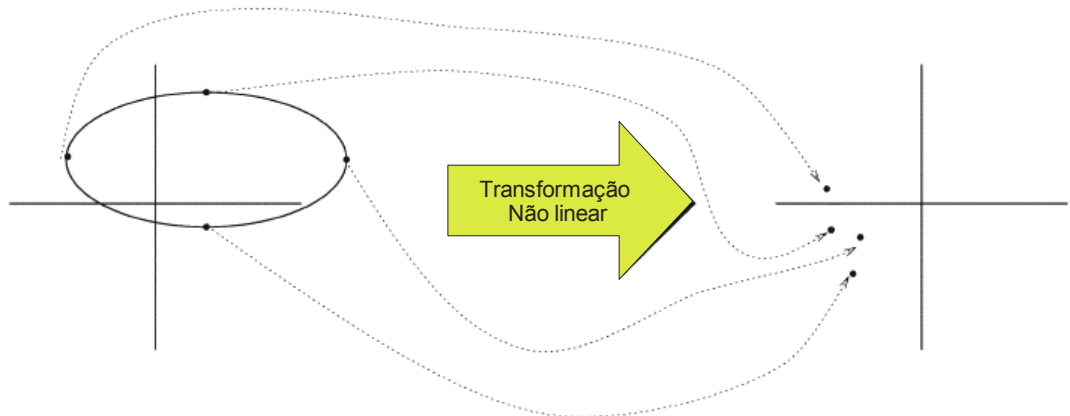
onde  $\lambda = \alpha^2(L+k) - L$  é um parâmetro escalar. A variável  $\alpha$  determina o espalhamento dos pontos sigma em torno da média  $\bar{x}$  e é sempre um valor mínimo positivo.  $k$  é um parâmetro escalar secundário que é sempre 0 e  $\beta$  é usado para incorporar o conhecimento a priori da distribuição de  $x$  (para distribuições gaussianas,  $\beta = 2$  é ótimo).  $(\sqrt{(L + \lambda)P_x})_i$  é a  $i$ -ésima linha da matriz de raiz quadrada. Estes vetores sigmas são propagados através da função não linear

$$y_i = g(X_i) \text{ para } i = 0, \dots, 2L \quad (85)$$

e a média e covariância de  $y_i$  são aproximadas utilizando média e covariância da amostra (Figura 9) dos pontos sigma posteriores

$$\bar{y} \approx \sum_{i=0}^{2L} W_i^{(m)} y_i \quad (86)$$

$$P_y \approx \sum_{i=0}^{2L} W_i^{(c)} \{y_i - \bar{y}\} \{y_i - \bar{y}\}^T. \quad (87)$$



**Figura 9. Princípio da transformação descentralizada.**

Este método difere dos métodos gerais de amostragem (os métodos Monte-Carlo tal como filtros de partículas) que requerem ordens de magnitude com mais pontos amostrados numa tentativa de propagar e definir as distribuições (possivelmente não-gaussianas) do estado. O tratamento feito pela transformação descentralizada resulta em aproximações com mais acertos para a terceira ordem para entradas gaussianas para todas as não linearidades.

Para entradas não-gaussianas, aproximações são mais confiáveis para ao menos uma segunda ordem, com o acerto de momentos de terceira ou maiores ordens determinadas pela escolha de  $\alpha$  e  $\beta$ .

### 2.11.2 Algoritmo do filtro de Kalman descentralizado

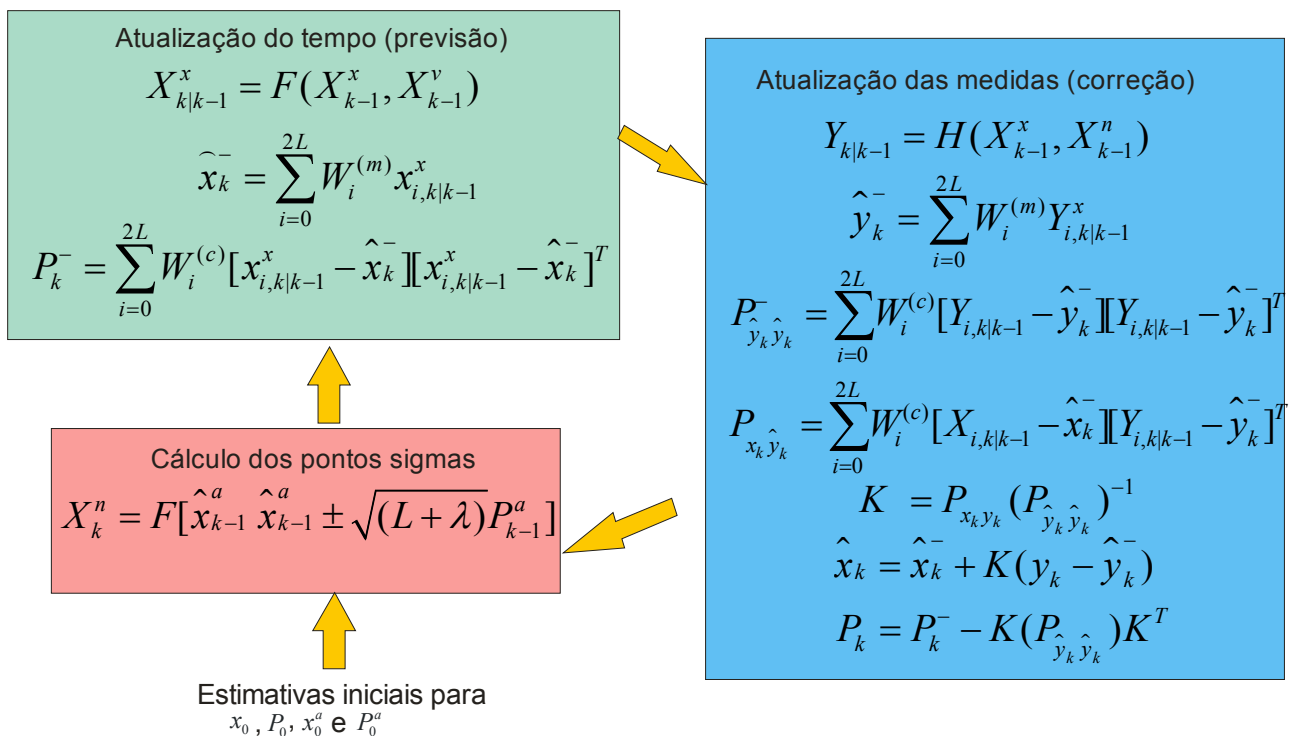
O filtro de Kalman descentralizado é uma extensão direta da transformação descentralizada para estimação recursiva da equação

$$\hat{x}_k = (\text{predição de } x_k) + K_k \cdot [y_k - (\text{predição de } y_k)] \quad (88)$$

onde o estado da variável aleatória é redefinido como a concatenação das variáveis do estado original e do ruído:

$$x_k^a = [x_k^T v_k^T n_k^T]^T. \quad (89)$$

A seleção dos pontos sigmas é aplicada para um novo estado de variável aleatória para calcular a matriz sigma correspondente  $x_k^a$ . As equações do filtro de Kalman descentralizado são dadas no algoritmo da Figura 10.

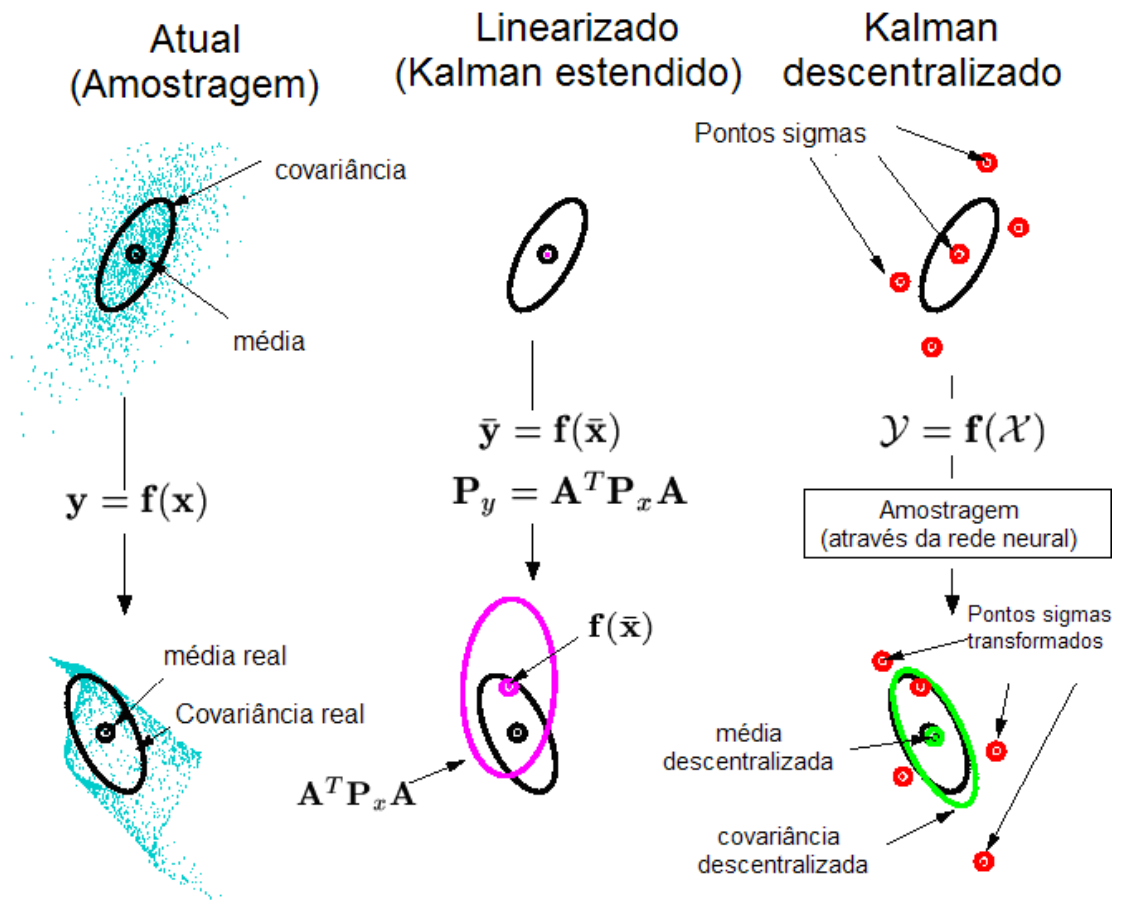


**Figura 10. Algoritmo para filtro de Kalman descentralizado.**

Não há necessidade do cálculo de matrizes jacobianas ou hessianas, além do mais, o número total de cálculos é da mesma ordem que o do filtro estendido. O filtro de Kalman descentralizado foi originalmente desenhado para um problema de estimação de estado, e

tem sido utilizado em aplicações para controles não lineares que requerem uma realimentação dos estados. Nestas aplicações o modelo dinâmico representa um modelo paramétrico fisicamente baseado, o que é assumidamente conhecido.

A Figura 11 ilustra uma comparação entre o filtro de Kalman estendido e o descentralizado. O exemplo é de um sistema bidimensional. A plotagem da esquerda mostra a propagação da média e da covariância reais usando amostragem Monte-Carlo; a plotagem central mostra os resultados obtidos por uma aproximação linearizada como seria feita no filtro de Kalman estendido; a plotagem da direita mostra o desempenho das transformações descentralizadas com o uso de cinco pontos sigmas.



**Figura 11. Exemplo de uma transformação descentralizada para propagação de média e covariância.**

Devido à instabilidade numérica do filtro relacionado ao ruído, como no uso da fatoração de Cholesky para determinar a raiz quadrada da matriz de probabilidade, Rudolph van der Merwe e Eric A. Wan desenvolveram o filtro de Kalman de raiz quadrada descentralizado (square-root unscented Kalman filter) (VAN DER MERVE e WAN 2001), que permite um melhor controle dos valores da matriz de variâncias, contornando o problema de se tornar uma matriz negativa ou indefinida.

Esse novo filtro também apresenta uma melhoria no desempenho, deixando o filtro de Kalman descentralizado com a mesma ordem de complexidade do filtro estendido.

Como o filtro de Kalman descentralizado original, o filtro de raiz quadrada é inicializado calculando-se a raiz quadrada da matriz da covariância dos estados pela fatoração de Cholesky:

$$S_0 = \text{chol}\{E[(x_0 - \hat{x}_0)(x_0 - \hat{x}_0)^T]\} \quad (90)$$

Contudo, o fator propagado e atualizado de Cholesky é então feito em iterações subseqüentes para diretamente formar os pontos sigmas.

Na equação 91, a atualização temporal do fator de Cholesky,  $S_k^-$ , é calculada usando uma decomposição QR da matriz composta contendo o peso dos pontos sigmas propagados e a raiz quadrada da matriz de covariância do ruído aditivo do processo:

$$S_k^- = \text{qr}\{[\sqrt{W_1^{(c)}}(X_{1:2L,k|k-1}^* - \hat{x}_k^-) \sqrt{R^v}]\} \quad (91)$$

A subseqüente atualização de Cholesky (ou regressão) na equação 92 é necessária desde que o peso zero,  $W_0^{(c)}$ , talvez seja negativo:

$$S_k^- = \text{cholupdate}\{S_k^-, X_{0,k}^* - \hat{x}_k^-, W_0^{(c)}\}. \quad (92)$$

Estes dois passos substituem a atualização temporal de  $P^-$ . Eles também são aplicados no cálculo do fator de Cholesky,  $S_{\mathcal{Y}}$ , da covariância do erro da observação:

$$S_{\mathcal{Y}k}^- = \text{qr}\{[\sqrt{W_1^{(c)}}(Y_{1:2L,k} - \hat{y}_k^-) \sqrt{R^a}]\} \quad (93)$$

$$S_{\mathcal{Y}k}^- = \text{cholupdate}\{S_{\mathcal{Y}k}^-, Y_{0,k} - \hat{y}_k^-, W_0^{(c)}\}. \quad (94)$$

Diferente do modo com que o ganho de Kalman é calculado no filtro padrão descentralizado, utilizam-se duas inversões:

$$K_k(S_{\mathcal{Y}k}^- S_{\mathcal{Y}k}^{-T}) = P x_k y_k \quad (95)$$

Desde que  $S_{\mathcal{Y}}$  seja quadrada e triangular, substituições eficientes podem ser usadas para resolvê-la diretamente sem a necessidade de uma inversão de matriz.

Finalmente, a atualização da medida posterior do fator de Cholesky da covariância do estado é calculada na equação 96 por aplicação de regressões sequenciais de Cholesky para regredir  $S_k^-$ .



$$S_k = cholupdata\{S_k^-, U, -1\}. \quad (96)$$

Os vetores de regressão são as colunas da equação 97. Isto substitui atualização posterior de  $P_k$  na equação 96.

$$U = K_k S_{Fk} \quad (97)$$

## 2.12 Aplicação do filtro de Kalman

A utilização do Filtro de Kalman em sinal tem como objetivo separar o ruído do sinal, ou seja, recuperar o sinal que foi corrompido por um ruído.

Como exemplo de uma aplicação prática encontra-se a simulação de uma tensão elétrica medida por um conversor Analógico-Digital não muito preciso. Para a incerteza da precisão do conversor, utiliza-se um ruído branco.

Foram feitos vários testes variando o valor de R, o ruído do ganho de Kalman, e o valor de P, probabilidade inicial, obtendo os seguintes resultados, onde a linha vermelha representa o sinal original, a faixa verde representa o sinal corrompido por um ruído branco e a linha azul representa o sinal filtrado pelo filtro de Kalman tradicional.

Na Figura 12 a probabilidade inicial é igual a 1, devido ao fato do valor inicial a ser estimado dado ao filtro ser o valor do sinal sem ruído.

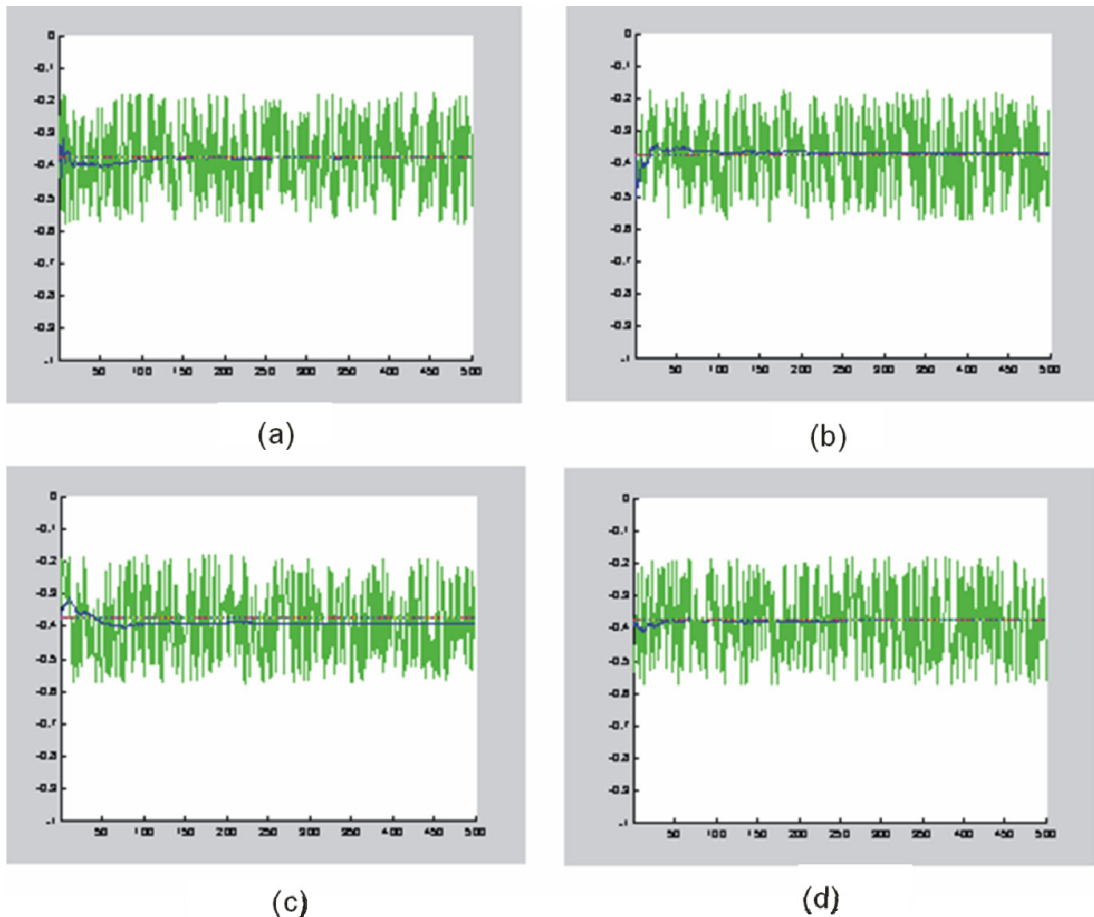
A variação do valor do ruído para o ganho de Kalman afeta diretamente a estimativa. Um valor muito alto pode fazer com que o filtro convirja mais rápido, mas há um risco de não atingir uma resposta ótima. Já com o valor baixo, o resultado é garantido, mas dependerá diretamente do tamanho da amostragem.

A utilização do Filtro de Kalman discreto provou ser eficiente na tentativa de recuperar um sinal em seu estado original. Como todos os equipamentos, por mais modernos que sejam, têm a probabilidade de projetar um ruído, o uso desse filtro pode ser uma alternativa para retornar ao estado original.

No trabalho alvo, que se caracteriza por filtrar sinais para montar uma imagem através de projeções tomográficas do solo, pode-se utilizar o filtro de Kalman para recuperar as imagens separando o ruído presente no tomógrafo, tendo como saída do sistema uma relação sinal-ruído considerada ótima.

## 2.13 Algoritmo de suavização Rauch-Tung-Striebel

A solução para o problema de predição linear segue uma maneira direta da teoria básica da propagação da covariância em relação à média da variável. Nesta seção, será considerado o problema da suavização ótima.



**Figura 12.** A probabilidade inicial é igual a 1 em todos os casos, pois o valor inicial é dado pelo valor do sinal original. O que varia entre os casos é o valor do ruído do ganho de Kalman. Em (a), esse valor é igual a 1, em (b), igual a 0,0001, em (c), igual 10 e em (d), igual a 0,01.

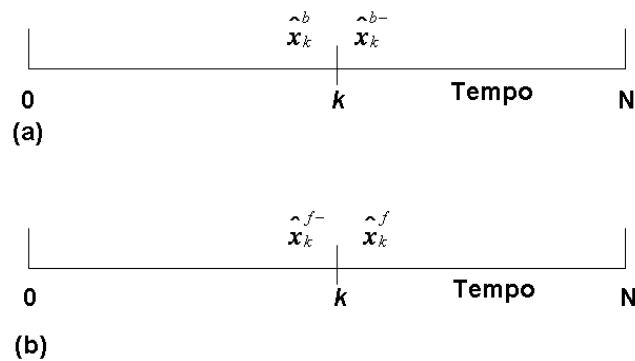
Supondo que exista um conjunto de dados sobre o intervalo do tempo  $0 < k \leq N$ . A suavização é uma operação offline que envolve a estimação do estado  $x_k$  para  $0 < k \leq N$ , usando todos os dados disponíveis, tanto passado como futuro. Assume-se que o tempo final  $N$  é fixo para determinar a estimativa do estado ótimo  $x_k$  para  $0 < k \leq N$ , e a data futura  $y_j$  definida por  $k < j \leq N$ . A estimação pertinente ao dado passado é conhecida como a teoria da filtragem adiante. Para estar de acordo com a emissão de uma estimação pertinente para o estado futuro, usa-se uma retro-filtragem, que começa no final do tempo

$N$  e executa para trás. Deixando como  $\hat{x}_k^f$  e  $\hat{x}_k^b$  denotam-se as estimativas dos estados obtidos das recursões para frente e para trás, respectivamente. Dadas estas duas estimativas, o próximo passo a ser considerado é como combiná-los em uma estimativa suavizada total  $\hat{x}_k$ , que conta com os dados sobre o intervalo de tempo inteiro. O  $\hat{x}_k$  é o símbolo utilizado para a estimativa suavizada e não pode ser confundido com o estimado filtrado (*a posteriori*).

Escrevendo a equação de processo como uma recursão para o decrescente  $k$ , temos

$$x_k = F_{k+1}^{-1} x_{k+1} - F_{k+1,k}^{-1} w_k \quad (98)$$

onde  $F_{k+1}^{-1}$  é o inverso para a matriz de transição  $F_{k+1,k}$ . O raciocínio para a retro filtragem começa no tempo final  $N$ , o que é demonstrado na Figura 13 (a), e a ser contrastada com a filtragem adiante na Figura 13 (b). Notando-se que a estimativa a priori  $\hat{x}_k^{b-}$  e a estimativa a posteriori  $\hat{x}_k^b$  para a retro filtragem ocorre para a direita e para a esquerda do tempo  $k$  respectivamente na Figura 13 (a). Essa situação é o oposto exato do que ocorre no caso da filtragem adiante mostrada na Figura 13 (b).



**Figura 13. Ilustrando as atualizações de tempo do suavizador para (a) retro-filtragem e (b) filtragem adiante.**

Para simplificar a apresentação, nós introduzimos as duas definições:

$$S_k = [P_k^b]^{-1}, \quad (99)$$

$$S_k^- = [P_k^{b-}]^{-1} \quad (100)$$

e as duas variáveis intermediárias

$$\hat{z}_k = [P_k^b]^{-1} \hat{x}_k = S_k \hat{x}_k \quad (101)$$

$$\hat{z}_k^- = [P_k^{b-}]^{-1} \hat{x}_k^{b-} = S_k^- \hat{x}_k^{b-}. \quad (102)$$

Então, construindo o raciocínio da figura, tem-se a derivação das seguintes atualizações para o retro filtro:

## 1. Atualização da medida

$$S_k = S_k^- + H_k R_k^{-1} H_k \quad (103)$$

$$z_k = \hat{z}_k^- + H_k^T R_k^{-1} y_k \quad (104)$$

onde  $y_k$  é o observável definido pela equação de medição,  $H_k$  é a matriz de medição e  $R_k^{-1}$  é a inversa da matriz de covariância do ruído de medição  $v_k$ .

## 2. Atualização do tempo

$$G_k^b = S_{k+1} [S_{k+1} + Q_k^{-1}]^{-1} \quad (105)$$

$$S_k^- = F_{k+1,k}^T (I - G_k^b) S_{k+1} F_{k+1,k} \quad (106)$$

$$\hat{z}_k^- = F_{k+1,k}^T (I - G_k^b) \hat{z}_{k+1} \quad (107)$$

onde  $G_k^b$  é o ganho de Kalman para retro-filtragem e  $Q_k^{-1}$  é o inverso da matriz de covariância do ruído do processo  $w_k$ . O retro-filtro definido pelas equações de medição e atualização do tempo é a formulação de informação do filtro de Kalman. O filtro de informação é distinguido do filtro básico de Kalman na maneira em que ele propaga a inversa da matriz de covariância de erro, melhor do que a matriz de covariância de erro normal.

Sejam os dados observados no intervalo  $0 < k \leq N$  para o fixo  $N$ , supõe-se que se obtêm as duas seguintes estimativas:

- A estimativa a posteriori adiante  $\hat{x}_k^f$  pela operação do filtro de Kalman nos dados  $y_j$  para  $0 < j \leq k$ .
- A estimativa a priori para trás  $\hat{x}_k^{b-}$  pela operação do filtro de informação nos dados  $y_j$  para  $k < j \leq N$ .

Com estas duas estimativas e suas respectivas matrizes de covariância de erro conhecidas, o próximo passo de interesse é como determinar a estimativa suavizada de  $\hat{x}_k$  e sua matriz de covariância de erro, que incorpora os dados totais sobre o tempo inteiro no intervalo  $0 < k \leq N$ .

Reconhecendo que o ruído do processo  $w_k$  e o ruído de medição  $v_k$  são independentes, formula-se a matriz de covariância de erro da estimativa a posteriori suavizada  $\hat{x}_k$  como se segue:

$$P_k = [[P_k^f]^{-1} + [P_k^{b-}]^{-1}]^{-1} \quad (108)$$

$$= [[P_k^f]^{-1} + S_k^-]^{-1} \quad (109)$$

Mais adiante, invoca-se o teorema da inversão de matriz, que pode ser mostrado a seguir. Deixando A e B serem duas matrizes positiva-definidas relacionadas por

$$A = B^{-1} + CD^{-1}C^T, \quad (110)$$

onde D é outra matriz positiva-definida e C é uma matriz com dimensões compatíveis. O teorema da matriz de inversão demonstra que a inversa da matriz A pode ser expressa como:

$$A^{-1} = B - BC[D + C^T BC]^{-1} C^T B \quad (111)$$

Para o problema que se trabalha, atribuem-se:

$$A = P_k^{-1} \quad (112)$$

$$B = P_k^f \quad (113)$$

$$C = I \quad (114)$$

$$D = [S_k^-]^{-1} \quad (115)$$

onde I é a matriz identidade. Então, aplicando o teorema de inversão de matriz na equação da matriz de covariância de erro, obtém-se:

$$P_k = P_k^f - P_k^f [P_k^{b-} + P_k^f]^{-1} P_k^f \quad (116)$$

$$= P_k^f - P_k^f S_k^- [I + P_k^f S_k^-]^{-1} P_k^f. \quad (117)$$

Desta equação, encontra-se que a matriz de covariância de erro a posteriori suavizada  $P_k$  é menor ou igual do que a matriz de covariância de erro a posteriori  $P_k^f$  produzida pelo filtro de Kalman, o que é naturalmente devido ao fato de que esta suavização utiliza informação adicional contida nos dados futuros.

Este ponto é apresentado na Figura 14, que descreve as variações de  $P_k$ ,  $P_k^f$  e  $P_k^{b-}$  com  $k$  para uma situação unidimensional.

$$\hat{x}_k = P_k ([P_k^f]^{-1} \hat{x}_k^f + [P_k^{b-}]^{-1} \hat{x}_k^{b-}) \quad (118)$$

Usando as equações anteriores para simplificar, obtém-se:

$$\hat{x}_k = \hat{x}_k^f + (P_k z_k^- - G_k \hat{x}_k^f), \quad (119)$$

onde o ganho suavizador é definido por

$$G_k = P_k^f S_k^- [I + P_k^f S_k^-]^{-1}, \quad (120)$$

que não pode ser confundido com o ganho de Kalman.

O suavizador ótimo já derivado consiste de três componentes:

- Um filtro adiante na forma de um filtro de Kalman;
- Um retro-filtro na forma de um filtro de informação;
- Um suavizador separado, que combina resultados encorpados nos filtros adiante e retrocesso.

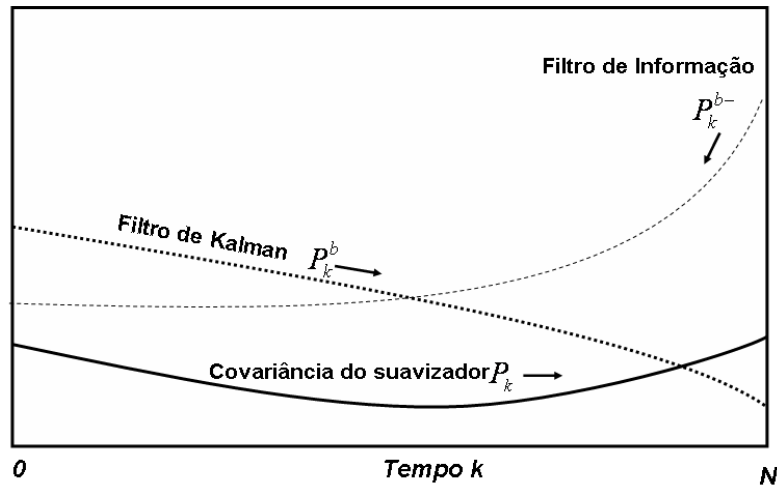


Figura 14. Ilustrando a covariância de erro para a filtragem adiante, retro filtragem, e suavizante.

O suavizador *Rach-Tung-Striebel* (RTS), contudo, é mais eficiente do que um suavizador de três partes que incorpora o retro filtro-e um suavizador separado em uma entidade simples. Especificamente, a atualização da medição do suavizador RTS é definida por:

$$P_k = P_k^f - A_k(P_{k+1}^{f-} - P_{k+1}^f)A_k^T \quad (121)$$

onde  $A_k$  é a nova matriz de ganho:

$$A_k = P_k^f F_{k+1,k}^T [P_{k+1}^{f-}]^{-1} . \quad (122)$$

A atualização de tempo correspondente é definida por

$$\hat{x}_k = \hat{x}_k^f + A_k(\hat{x}_{k+1} - \hat{x}_{k+1}^{f-}) . \quad (123)$$

O suavizador RTS procede como a seguir:

1. O filtro de Kalman é aplicado para os dados observáveis em uma maneira adiante, que é  $k = 0, 1, 2, \dots$ , de acordo com a teoria básica.
2. O suavizador recursivo é aplicado para os dados observados de uma maneira retrocessa, que é  $k = N - 1, N - 2, \dots$ , de acordo com as equações do RTS.
3. As condições iniciais são definidas por:

$$P_N = P_N^f \quad (124)$$

$$\hat{x}_k = \hat{x}_k^f \quad (125)$$



## 3 Redes Neurais

---

### 3.1 Histórico das redes neurais artificiais

Na década de 40, McCulloch e Pitts introduziram os estudos sobre redes neurais, apresentando hipóteses sobre o funcionamento dos neurônios em 1943 (MCCULLOCH e PITTS,1943) apresentaram detalhes neurofisiológicos e modelos para organização e operação do *Colliculus Superior* (MCCULLOCH e PITTS,1947). Em 1949, as redes neurais foram aprimoradas pelo trabalho de Donald O. Hebb (HEBB 1949) com a teoria de que o aprendizado ocorria nas sinapses (conexões entre neurônios). Em 1954, Farley e Clark (FARLEY e CLARK 1954) realizaram uma simulação bem sucedida do sistema nervoso. Em 1956, N. Rochester, J. H. Holland, L. H. Haibt e W. L. Duda (ROCHESTER, et al. 1956) executaram uma simulação de uma teoria bem formulada e detalhada de rede neural em um computador IBM. No final da década de 50, a área foi impulsionada com o trabalho de Rosenblatt (F. ROSENBLATT 1958), propondo o modelo perceptron de uma camada baseado no modelo de aprendizagem de Hebb, capaz de solucionar problemas simples de reconhecimento de padrões. No final da década de 60, Marvin Minsky e Seymour Papert (MINSKY e PAPERT 1968) realizaram um trabalho demonstrando as limitações do modelo de uma camada, apresentando o problema do OU EXCLUSIVO, que não é linearmente separável, e, portanto, não pode ser resolvido por modelos de única camada, havendo, assim, um recesso nas pesquisas desta área, sendo retomadas apenas no início da década de 80, com trabalhos de Hopfield (HOPFIELD 1982), Hinton (HINTON e SEJNOWSKI 1986), Rumelhart (RUMELHART, HINTON e WILLIAMS 1986), Sejnowski (SEJNOWSKI e ROSENBERG 1986) e outros, que propuseram teorias e modelos direcionados à computação (PIZZOLATO 1992).

### 3.2 O modelo computacional

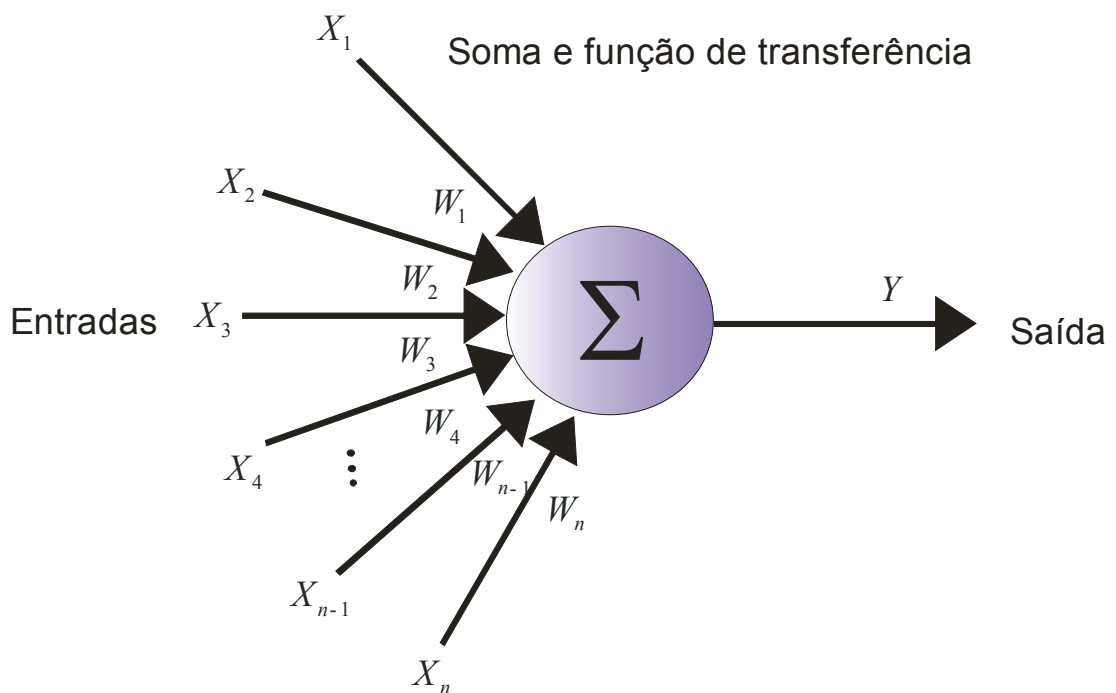
Um neurônio, dada a sua natureza complexa, ainda não é totalmente modelável, mas, devido ao seu comportamento, há possibilidade de formular modelos computacionais que



satisfazem alguns objetivos os quais são difíceis de serem alcançados com técnicas convencionais de computação (TREALEVEN, PACHECO e VELLASCO 1988). O neurônio computacional é um modelo criado a partir de análises do modelo biológico considerando seu aspecto funcional.

Uma tabela com pesos é responsável pela associação de um peso a um determinado impulso recebido de um neurônio específico. Cada neurônio trata o impulso recebido de um determinado neurônio emissor de forma diferente, atribuindo-lhe um valor, definindo uma tabela de pesos para si (PIZZOLATO 1992). Esses pesos são aplicados a uma soma escalar pelos impulsos de entrada aos outros pesos proporcionando a saída. A saída do neurônio é determinada por uma função de transferência (ou função limite), determinando a transmissão do neurônio quando excitado.

Esse modelo foi proposto por McCulloch e Pitts (MCCULLOCH e PITTS, 1943) e denominado, também, de neurônio de McCulloch-Pitts (Figura 15).



**Figura 15. Diagrama do neurônio de McCulloch-Pitts, apresentando as entradas ( $X_1...X_n$ ), os pesos ( $W_1...W_n$ ) e a saída ( $Y$ ) do neurônio.**

É uma forma bem simples de computar algo. Os pesos do neurônio são ajustados para realizar funções lógicas simples, tendo diferentes neurônios realizando diferentes funções (FAUSETT 1994). Através do arranjo dos neurônios, a rede é capaz de produzir qualquer resposta que pode ser representada como a combinação de funções lógicas. O fluxo de informação através dos neurônios da rede apresenta um “*delay*”, permitindo a

modelagem de processos fisiológicos como quente e frio, e é aplicado um *threshold* (função limite) funcionando como uma condição de disparo (DUARTE 1996).

Donald Hebb concebeu a primeira lei de aprendizagem para as redes neurais artificiais (HEBB 1949). Ele estabeleceu que, quando dois neurônios estão simultaneamente ativos, o poder de conexão entre eles deve aumentar. McClellan e Rumelhart (MCCLELLAN e RUMELHART 1988) desenvolveram uma forma expandida do aprendizado de Hebb, em que, mesmo quando as unidades estão simultaneamente inativas, os pesos também são reforçados nas conexões.

Após a bem sucedida simulação do sistema nervoso por Farley e Clark (FARLEY e CLARK 1954) e Rochester e outros (ROCHESTER, et al. 1956), a década de 50 foi considerada a idade de ouro para as pesquisas em redes neurais, marcada pelo desenvolvimento de técnicas de simulação que atraiu a atenção de Von Neumann (VON NEUMANN 1958), o “pai da computação moderna”, prometendo ser uma alternativa para a computação tradicional proposta (FAUSETT 1994).

Durante os anos 60, Rosenblatt e seu grupo desenvolveram o modelo *Perceptron* (F. ROSENBLATT 1958) (F. ROSENBLATT 1962), introduzindo o desenvolvimento de vários tipos de redes neurais. Este modelo apresentava três camadas, uma de entrada, uma de saída e uma camada intermediária, conhecida como camada de associação. Esse sistema de três camadas pode aprender a conectar (associar) uma entrada dada com uma unidade de saída aleatória (MAREN, HARSTON e PAP 1990). Como os neurônios de McCulloch-Pitts e Hebb, os *perceptrons* utilizam elementos *threshold* para o controle de disparo (FAUSETT 1994), com uma regra de aprendizagem melhor do que o aprendizado hebbiano, fundamentando-se no ajuste iterativo dos pesos. O *Perceptron* era computacionalmente preciso e exibia uma correta regra de aprendizagem, porém, não conseguia resolver problemas complexos (não separáveis linearmente, como a função lógica XOR). No final da década de 60, Marvin Minsky e Seymour Pappert (MINSKY e PAPER 1968) demonstram matematicamente as limitações dos *perceptrons*, mas suas demonstrações valiam apenas para o modelo de unicamadas, enquanto as redes multicamadas mostraram-se capazes de resolver problemas não lineares (RUMELHART, HINTON e WILLIAMS 1986).

Paralelamente, estava sendo desenvolvida, por Bernard Widrow e seu aluno Ted Hoff, uma regra de aprendizagem (WIDROW e HOFF 1960) que é exatamente relacionada ao aprendizado do *perceptron*. A similaridade dos modelos desenvolvidos por Rosenblatt

através da psicologia e por Widrow e Hoff através da engenharia elétrica é uma evidência da interdisciplinaridade natural das redes neurais (FAUSETT 1994). O termo *ADALINE*, *Adaptive Linear Neuron*, foi utilizado por ser reportado como da área de sistemas adaptativos lineares. A regra de aprendizagem utilizada para uma rede unicamada é a precursora da regra *backpropagation* para redes multicamadas (DUARTE 1996).

A incapacidade de resolver problemas simples (mapeamento) e a falta de um método de treinamento para redes multicamadas foi responsável para o desaquecimento da área. Na Figura 16, podem-se perceber os problemas resolvidos pelo perceptron unicamada, o de camada dupla e de três camadas.


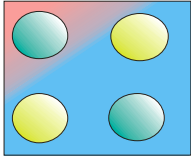
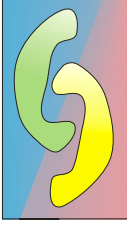
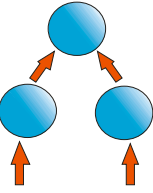
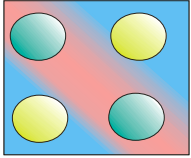

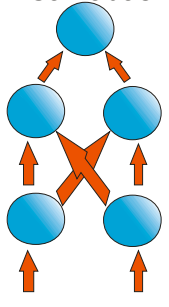
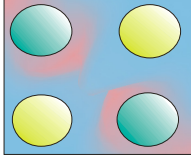
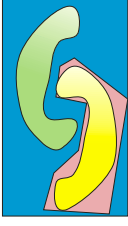
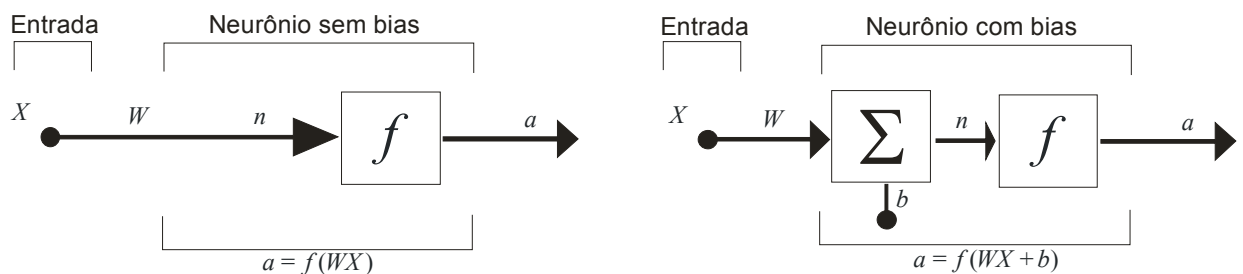
Estrutura	Regiões	Problema de Ou-Exclusivo	Reconhecimento de padrões
<p>Camada Simples</p> 	<p>Meio plano limitado por um hiper-plano</p>		
<p>Duas Camadas</p> 	<p>Regiões convexas ou fechadas</p>		
<p>Três Camadas</p> 	<p>Arbitrário (limitado pelo número de nós)</p>		

Figura 16. Tabela comparativa do *perceptron* unicamada e multicamadas e os problemas que podem ser solucionados (JAIN 1996).

Quanto maior o número de camadas, problemas mais complexos podem ser resolvidos, sendo que o perceptron de três camadas está limitado pelo número de neurônios (ou nós) para resolver determinados problemas, como reconhecimento de padrões, isto é, quanto maior o número de nós, maior é a separabilidade entre classes. O interesse pelas redes neurais obteve uma queda significativa durante os anos 70, sendo reativado nos anos 80.

Paul Werbos chegou a desenvolver um método para a propagação da informação acerca dos erros, das unidades de saída para as unidades escondidas, em 1970, não alcançando a devida publicidade e sendo republicado em 1989 (WERBOS 1989). O método *backpropagation* foi desenvolvido independentemente por Le Cun e David Parker (LE CUN 1986). O trabalho de Parker recebeu refinamentos e a publicidade merecida devido ao grupo de processamento distribuído e paralelo liderados por David Rumelhart e James McClellan, relatados em (RUMELHART, HINTON e WILLIAMS 1986) (MCCLELLAN e RUMELHART 1988).

Outros tipos de redes neurais artificiais, como Hopfield, Neocognitron, Máquinas de Boltzmann e implementação em *hardware*, são citados em (DUARTE 1996). A Figura 17 mostra dois neurônios com uma entrada escalar, sendo um com bias e o outro sem. A entrada escalar  $X$  é transmitida através de uma conexão que multiplica seu valor por um peso  $w$ , para formar o produto  $WX$ , um escalar. Este produto é um argumento da função de transferência  $f$ , que produz uma saída escalar  $a$ . O *bias* é adicionado ao produto  $WX$  como se fosse uma junção de soma ou com função de *shift* para função  $f$ . O bias é como um peso, e sua entrada tem o valor constante de 1.



**Figura 17. Modelo de neurônio artificial com uma entrada.**

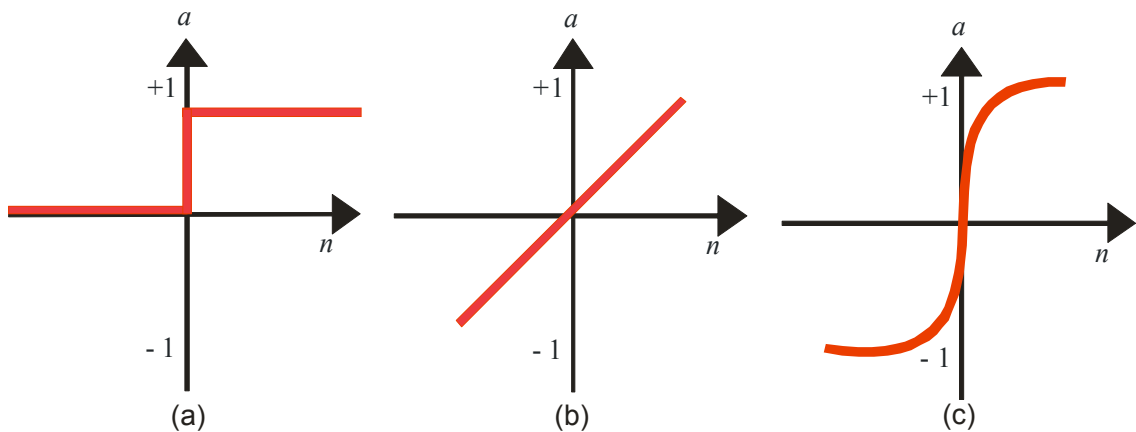
A função de transferência recebe um valor  $n$  e retorna uma saída  $a$ . Essa função pode ser do tipo linear ou sigmoideal. Os valores do peso e do bias podem ser ajustáveis, assim a rede neural exibe um comportamento desejado ou próximo.

Existe possibilidade de treinar a rede para fazer um determinado trabalho ajustando o peso e o bias ou deixar que a rede ajuste-os para uma saída desejada.

Algumas vezes, o uso do bias é omitido no diagrama dos neurônios. Ele é um parâmetro ajustável, não uma saída. Seu valor constante faz com que o bias seja uma saída e deva ser tratado como uma dependência linear nos vetores de entradas.

### 3.3 Função de transferência

Existem vários tipos de funções de transferência, as mais comuns são a função de transferência de limite, linear ou sigmoidal, como poder ser visto na Figura 18.



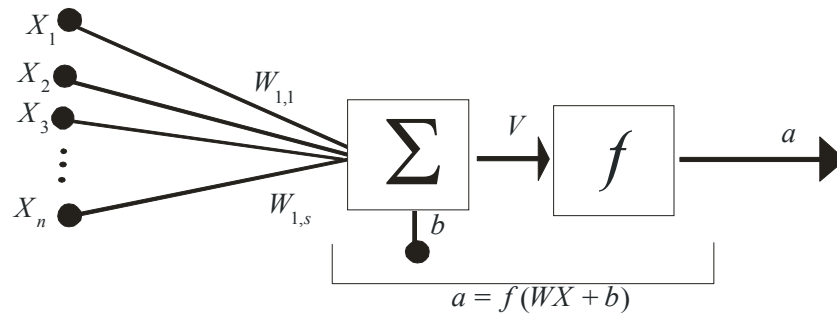
**Figura 18. Função de transferência de limite (a), linear (b) e sigmoidal (c). O eixos  $a$  e  $n$  correspondem a saída e a entrada da rede, respectivamente. Cada entrada é mapeada na função de forma independente, assumindo um valor de acordo com os pesos.**

A função de limite (Hard-Limit) limita a saída do neurônio para 0 (se a entrada  $n$  for menor que 0) ou 1 (se a entrada  $n$  for maior ou igual a 1). Esse tipo de função é utilizado para fazer decisões em classificação. A função linear é utilizada por neurônios como aproximadores lineares em Filtros Lineares. A função sigmoidal recebe a entrada e qualquer valor entre mais ou menos infinito, e retorna a saída entre 0 e 1. Ela é mais usada em redes *backpropagation*, porque é diferenciável.

### 3.4 Neurônio com um vetor de entrada

Um neurônio com um vetor de entradas  $X$  de tamanho  $n$  é multiplicado pelos pesos  $W$  e os valores dos produtos são somados individualmente. O neurônio tem um bias  $b$ , que

é somado com o produto das entradas e dos pesos para formar uma entrada da rede  $V$ . Nesta soma,  $V$  é o argumento da função de transferência  $f$  (Figura 19).



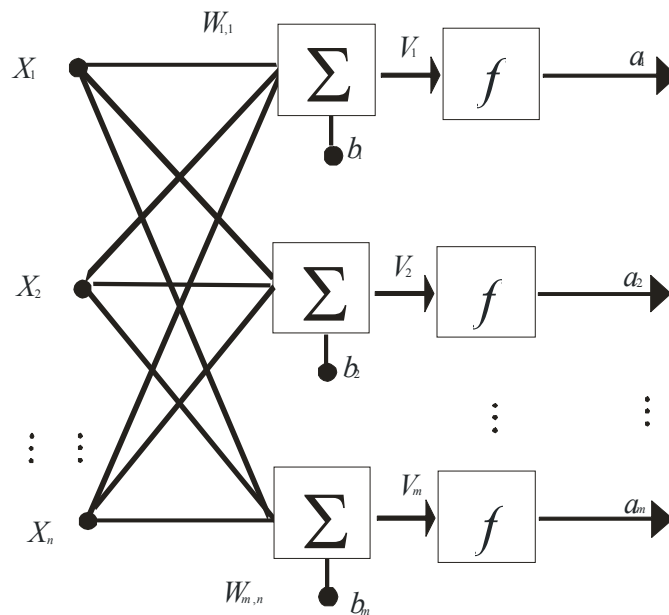
**Figura 19.** Neurônio com vetor de entradas.

Assim, o valor de  $V$  é obtido pela expressão

$$V = W_{1,1}X_1 + W_{1,2}X_2 + \dots + W_{1,n}X_n + b. \quad (126)$$

Dois ou mais neurônios podem ser combinados em uma camada, assim como uma rede neural pode ter mais de uma camada.

Considere uma unicamada de neurônios com  $n$  elementos de entrada e  $m$  neurônios (Figura 20).



**Figura 20.** Vários neurônios em uma única camada.

Cada elemento do vetor de entrada  $X$  é conectado a cada entrada do neurônio através da matriz  $W$ . O  $i$ -ésimo neurônio tem uma soma que seu próprio peso e seu próprio bias para formar uma saída escalar  $V(i)$ . Estes, juntamente com um neurônio  $m$  do vetor de entrada  $V$ . Assim, a saída da camada forma um vetor coluna  $a$ .

O número de neurônios pode ser diferente do número de entradas numa camada, assim a camada não fica restrita. A matriz  $W$  ficaria composta como mostra a equação 127.

$$W = \begin{bmatrix} W_{1,1} & W_{1,2} & \cdots & W_{1,n} \\ W_{2,1} & W_{2,2} & \cdots & W_{2,n} \\ \vdots & \vdots & \vdots & \vdots \\ W_{m,1} & W_{m,2} & \cdots & W_{m,n} \end{bmatrix} \quad (127)$$

### 3.5 Multicamadas de neurônios

Uma rede pode ter várias camadas e cada camada tem uma matriz de peso  $W$ , um vetor bias  $b$  e cada uma destas camadas tem sobrescrita a variável de interesse. Pode-se ver na Figura 21 o diagrama para três camadas e sua equação. Para cada camada, há um vetor de peso, um vetor de entrada, um vetor de bias e um vetor de saída, ambos diferentes.

A rede possui, assim,  $n$  entradas,  $m^1$  neurônios na primeira camada,  $m^2$  neurônios na segunda camada etc. É comum para as diferentes camadas ter diferentes números de neurônios. A entrada constante 1 alimenta cada neurônio no seu respectivo bias.

As saídas de cada camada intermediária são as entradas para a próxima camada. Assim, a segunda camada pode ser interpretada como uma rede de uma camada com  $m^1$  entradas e  $m^2$  neurônios, e uma matriz de peso  $W^2$  com um tamanho  $m^2 \times m^1$ . A entrada para a segunda camada é  $a^1$  e a saída é  $a^2$ .

As redes neurais multicamadas são muito poderosas. Por exemplo, uma rede de duas camadas, onde a primeira é sigmoideal e a segunda linear, pode ser treinada para aproximar qualquer função arbitrária, com um número finito, de descontinuidades. Este tipo de duas camadas é usado extensivamente em *backpropagation*.

A saída da última camada da rede neural é a saída de interesse. Para fácil entendimento, esta saída geral é mostrada como  $Y$ . O treinamento de uma rede neural pode ser visto de forma mais detalhada no apêndice B.

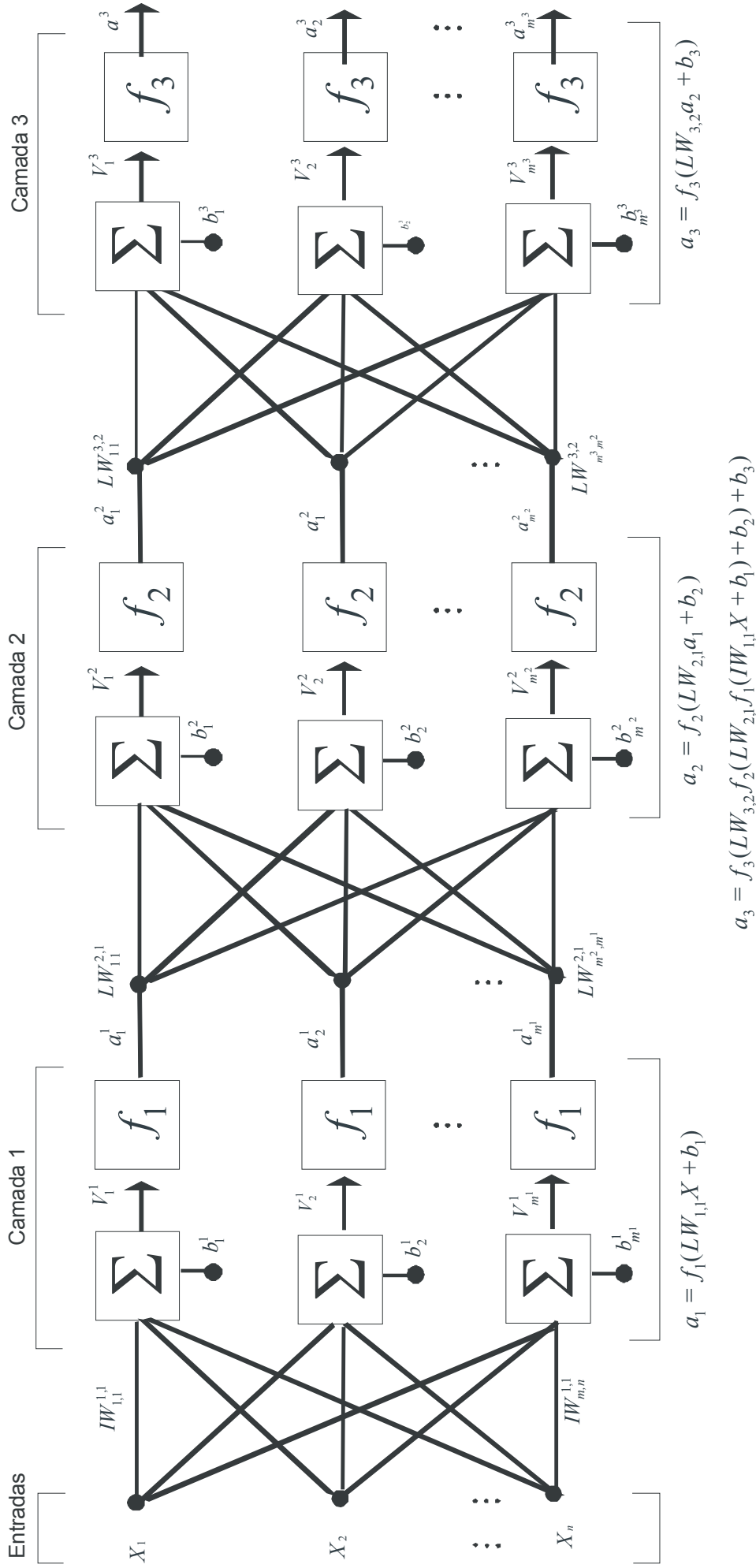


Figura 21. Uma rede neural de três camadas e a equação para calcular sua saída.



### 3.6 Comparação entre os algoritmos de treinamento das redes neurais

Nesta seção serão comparados os algoritmos de treinamentos utilizados para treinar uma rede neural do tipo *perceptron* multicamadas com duas camadas, sendo que a camada intermediária possui cinco neurônios e a camada de saída apenas um neurônio.

Os algoritmos para treinamento foram os de *backpropagation* com gradiente decrescente e taxa de aprendizagem adaptativa, uma variante desse algoritmo com “momentum”, o algoritmo de Levenberg-Marquardt, o algoritmo de filtro de Kalman estendido e uma variação com a variante do ruído do processo adaptativa, e, por último, o filtro de Kalman Descentralizado.

O teste de desempenho dos algoritmos foi realizado de acordo com a capacidade de convergência em menor iteração e o tempo para o treinamento. A função do tipo  $f(x) = poisson(x)$  com  $x=[1,10]$ ,  $x=[1,100]$  e  $x=[1,1000]$ , que introduz um ruído *Poisson* na entrada em  $x$ .  $x$  é a entrada da rede e a função  $f(x)$  a saída desejada. A Tabela 1 mostra o erro médio obtido por cada algoritmo e a Tabela 2, o tempo de execução de cada algoritmo.

**Tabela 1- Tabela de erro médio para a função  $f(x) = poisson(x)$ .**

Entradas e Iterações	Gradiente Decrescente	Grad. Dec. Momentum	Levenberg - Marquardt	Kalman Estendido	Kalman Est. Com Q	Kalman Descent.
10	1,526	2,522	1,092	2,557	2,720	11498,985
100	19,780	25,590	4,899	12,425	6,544	576,759
1000	302,238	251,219	16,493	20,586	20,195	46049,933

**Tabela 2- Tabela de tempo de execução para a função  $f(x) = poisson(x)$ , em segundos.**

Entradas e Iterações	Gradiente Decrescente	Grad. Dec. Momentum	Levenberg- Marquardt	Kalman Estendido	Kalman Est. Com Q	Kalman Descent.
10	0,471	0,118	0,128	0,002	0,133	0,102
100	0,386	0,276	0,399	0,019	0,017	0,713
1000	0,640	4,266	9,401	0,114	0,171	10,932

A Figura 22, a Figura 23 e a Figura 24 ilustram as saídas obtidas pelas redes neurais com seus respectivos treinamentos.

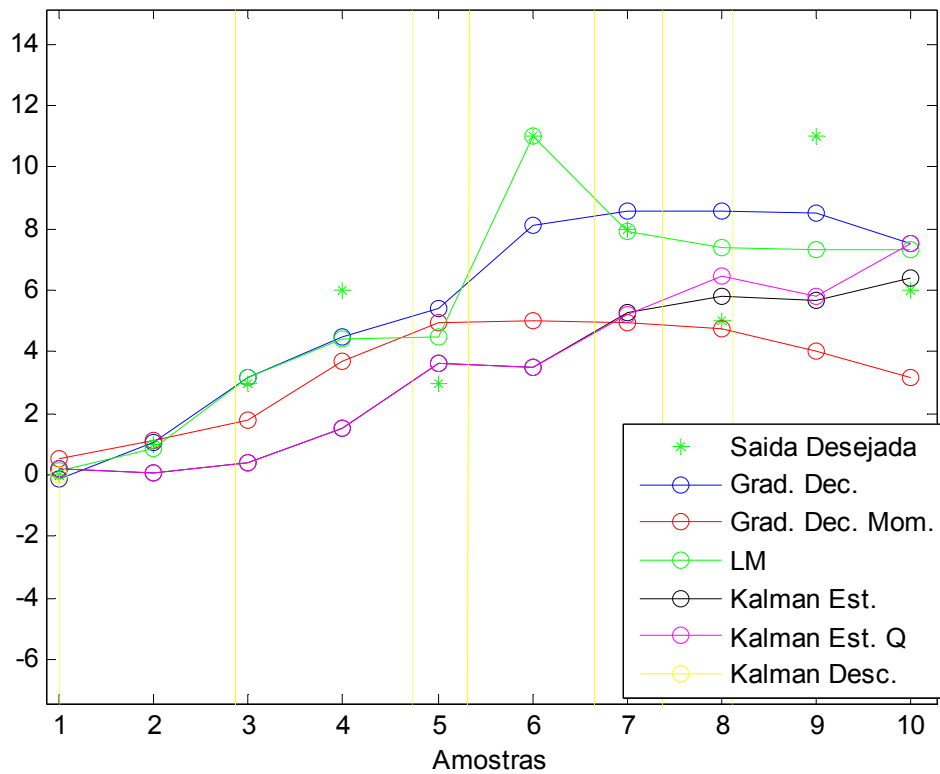


Figura 22. Saídas das redes neurais com entrada de tamanho 10 e 10 passos para treinamentos

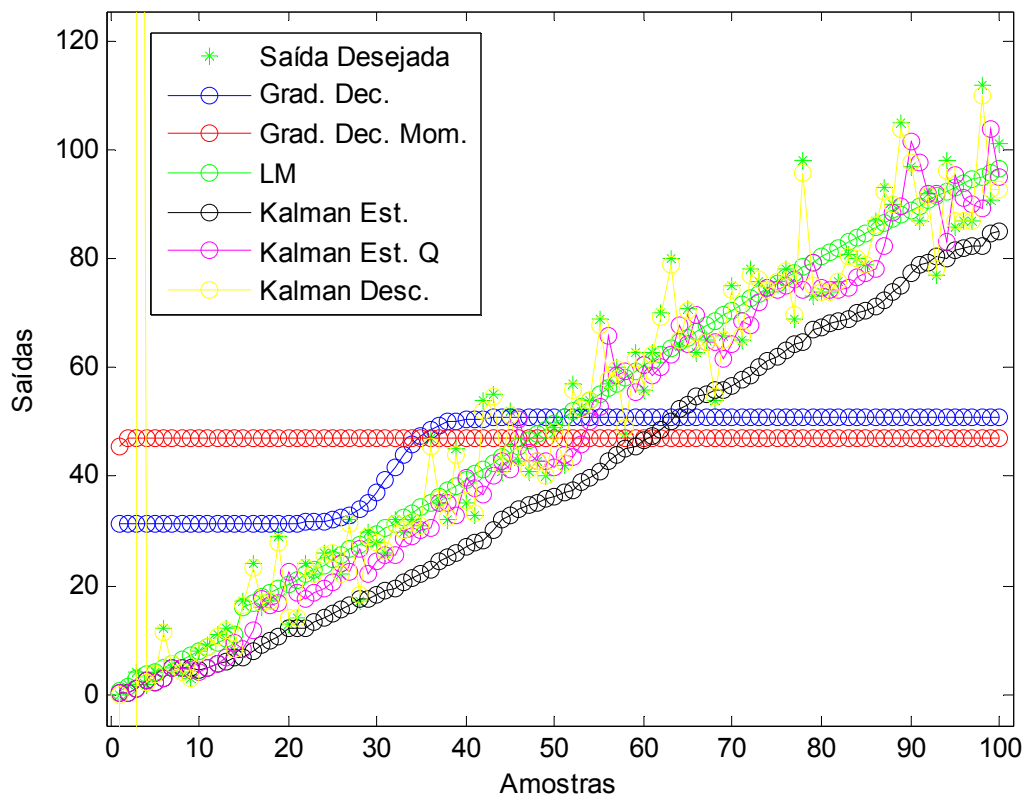
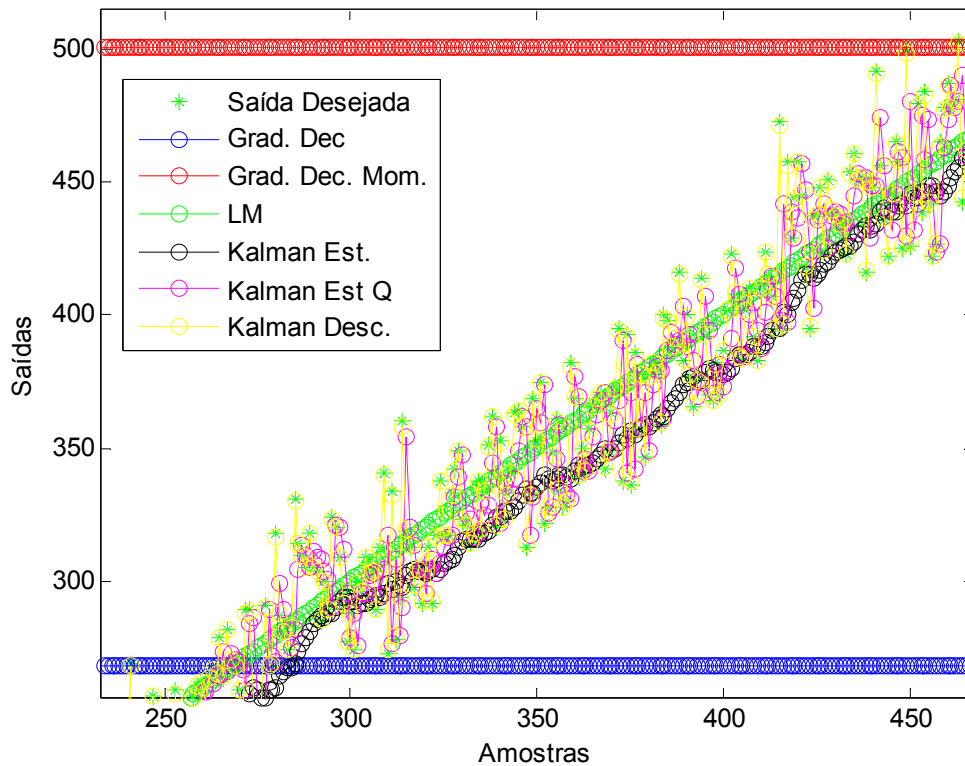


Figura 23. Saídas das redes neurais com 100 entradas e 100 passos de treinamento



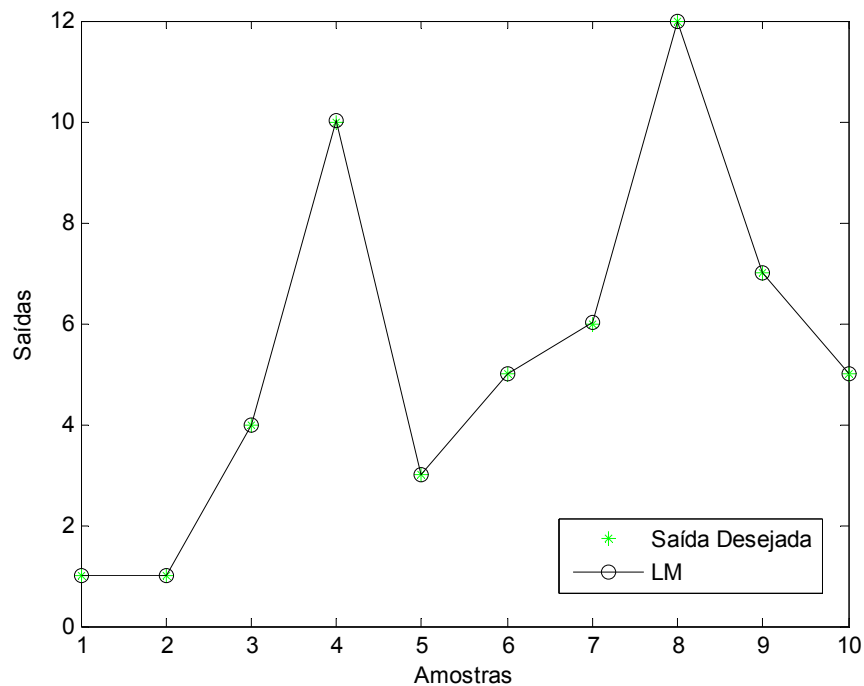
**Figura 24. Saídas das redes neurais com 1000 entradas e 1000 passos de treinamento. Para melhor visualização dos dados, a escala foi ampliada.**

Enquanto os algoritmos de treinamento *on-line* possuem uma taxa de convergência de acordo com o aprendizado no tempo, dependendo diretamente do tamanho das amostras, os algoritmos *off-line* dependem diretamente dos números de passos a serem aplicados. Outra diferença é a capacidade de convergência em um menor tempo, aliado a um menor número de neurônios, promovendo um tempo menor de processamento.

Os treinamentos baseados em *backpropagation* com gradiente decrescente e com momentum geraram resultados satisfatórios quando utilizados *off-line* e com vários passos para treinamento. Os passos do treinamento para a convergência dependerão do número de neurônios na camada e da complexidade das funções exigidas. Devido ao alto tempo para convergir, pode levar horas para que o treinamento obtenha uma melhor resposta.

Os treinamentos baseados em Levenberg-Marquardt possuem uma alta convergência com passos de treinamento do tamanho das entradas, desde que o número de neurônios seja o suficiente. É apresentado em (HAGAN e MENHAJ 1994) o primeiro desenvolvimento deste algoritmo para redes neurais. São descritas, também, a teoria e aplicação deste algoritmo, que treina redes neurais em uma taxa de 10 a 100 vezes mais rápido do que os métodos baseados em *backpropagation* com gradientes decrescentes. Nos

resultados anteriores, o número de neurônios foi insuficiente para uma melhor convergência. Um novo treinamento com um número maior de neurônios obteve resultados melhores que os anteriormente apresentados (Figura 25 e Tabela 3). Possui um ótimo desempenho, mas um alto custo devido às entradas e ao tamanho dos neurônios nas camadas, sendo inviável aplicá-lo num filtro *on-line*.



**Figura 25. Resultado mostrando a alta convergência do algoritmo de treinamento Levenberg-Marquadt**

**Tabela 3- Resultados obtidos com o algoritmo Levenber-Marquadt**

Entradas e Iterações	Erro médio	Tempo (em segundos)	Neurônios
10	0	0.20	20

O filtro de Kalman estendido com Q variante é ideal para aplicações como visão computacional, onde há uma mudança nos valores observados, e o filtro deve acompanhar essas mudanças. Sua taxa de variância depende exclusivamente de uma janela que faz as medidas temporais. A escolha da janela de observação para determinar o valor de Q apresenta melhores resultados, mas uma maior complexidade, além das presentes no filtro em questão.

O filtro de Kalman descentralizado é uma forma totalmente não linear que utiliza aspectos dos filtros de partículas (como os métodos de Monte Carlo). Possui um melhor resultado *on-line* desde que seja bem sintonizado com o problema.

A vantagem em relação aos dois filtros anteriormente apresentados é a possibilidade de determinar as transferências de covariância ser feita através de mapeamentos de partículas (pontos Sigma), não havendo necessidade de determinar matrizes jacobianas ou hessianas. A complexidade e o tempo de resposta também são altos, comparados com os filtros anteriores, devido ao número de partículas criadas e ao número de variáveis para sintonia do filtro. Como um filtro *on-line*, seu tempo de convergência depende diretamente da configuração do filtro.

## 4 Resultados e Conclusões

### 4.1 Modelagem do processo

O processo de aquisição de dados do minitomógrafo fornece uma matriz com os valores das amostras que compõem as projeções. Para a modelagem do processo, considera-se uma linha da matriz que, por convenção, denomina-se raio-soma. Esse sinal é composto de várias incidências com valores variáveis e não determinísticos obtidos pela equação 5 do capítulo 1, cuja amplitude  $I_m^{[n]}$  é dada por:

$$I_m^{[n]} = I_0 e^{-\mu d}, \quad (128)$$

onde  $d$  é a distância percorrida pelo feixe de fótons dentro do corpo em prova e  $\mu$  é o coeficiente de atenuação.

A variável  $I_0$  representa o valor do feixe livre, sendo afetado por um ruído *Poisson* que é definido por:

$$P(n, t) = \frac{(I_0 t)^n}{n! e^{-I_0 t}} \quad (129)$$

onde  $n$  representa o número de elementos da fotomultiplicadora e  $t$  é o tempo de exposição. Assim, a equação 129 pode ser escrita como

$$I_m^{[n]} = \left[ I_0 + \frac{(I_0)^n}{n! e^{-I_0 t}} \right] \quad (130)$$

O raio-soma pode ser representado por sua seqüência finita  $P_0^{[n]}$  dada como se pode ver na Figura 26.

Para uma projeção  $P^m(\theta)$ , pode-se agora escrever que (OPPENHEIM e SCHAFER 1975):

$$P_{\theta_1}^{[n]} = \sum_{j=1}^m I_j \delta(n-j). \quad (131)$$

Esta equação é considerada parte do processo e pode ser escrita na forma matricial:

$$\begin{bmatrix} P_{\theta_1}^{[n]} \\ I_2 \delta(n-2) \\ \vdots \\ I_m \delta(n-m) \end{bmatrix} = \begin{bmatrix} 1 & 1 & 1 & \dots & 1 \\ 0 & 1 & 0 & \dots & 0 \\ 0 & 0 & 1 & & 0 \\ \vdots & \vdots & & \ddots & \vdots \\ 0 & 0 & 0 & \dots & 1 \end{bmatrix} \begin{bmatrix} I_1 \delta(n-1) \\ \vdots \\ I_m \delta(n-m) \end{bmatrix} \quad (132)$$

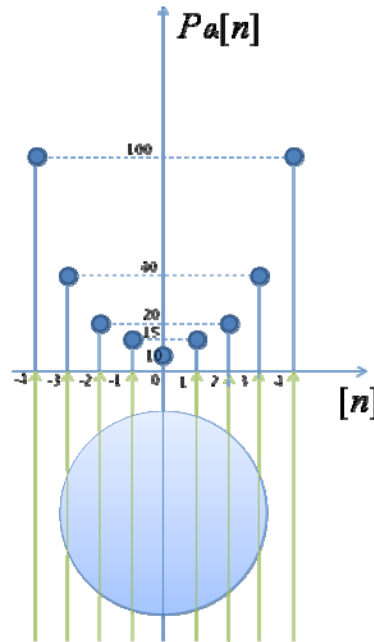


Figura 26. Sinal tomográfico definido por  $P_{\theta_t}[n] = I_1 \delta(n+4) + I_2 \delta(n+3) + I_3 \delta(n+2) + I_4 \delta(n+1) + I_5 \delta(n) + I_6 \delta(n-1) + I_7 \delta(n-2) + I_8 \delta(n-3) + I_9 \delta(n-4)$ .

Como o valor medido é dado por  $P_{\theta_t}[n]$ , a equação do sistema define o que é apenas observado:

$$P_{\theta_t}[n] = [1 \quad 0 \quad \dots \quad 0] \begin{bmatrix} P_{\theta_t}[n] \\ I_2 \delta(n-2) \\ \vdots \\ I_m \delta(n-m) \end{bmatrix} \quad (133)$$

Como o processo pode ser tratado de forma recursiva, ou seja, podemos estimar  $P_{\theta_t}[n]$ , conhecendo  $P_{\theta_t}^{m-1}$  e  $I_{m-1} \delta(n-m-1)$ , na forma da equação:

$$P_{\theta_t}[n] = P_{\theta_t}^{m-1} + I_{m-1} \delta(n-m-1), \quad (134)$$

podem-se reduzir as matrizes para um instante anterior para que o filtro estime o valor do próximo instante como:

$$\begin{bmatrix} P_{\theta_t}[n] \\ I_2 \delta(n-2) \\ \vdots \\ I_m \delta(n-m) \end{bmatrix} = \begin{bmatrix} 1 & 1 & | & 1 & \dots & 1 \\ 0 & 1 & | & 0 & \dots & 0 \\ \hline 0 & 0 & 0 & 1 & \dots & 0 \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & 0 & \dots & 1 \end{bmatrix} \begin{bmatrix} I_1 \delta(n-1) \\ \vdots \\ I_m \delta(n-m) \end{bmatrix}. \quad (135)$$

Isso permitirá uma filtragem com conhecimento *a priori* apenas da variável anterior. Assim, as matrizes de transferência e do sistema ficam reduzidas como:

$$\begin{bmatrix} P_{\theta_1}[n] \\ L_{m-1}\delta(n-m-1) \end{bmatrix} = \begin{bmatrix} 1 & 1 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} L_m\delta(n-m) \\ L_{m-1}\delta(n-m-1) \end{bmatrix} \quad (136)$$

$$P_{\theta_1}[n] = [1 \quad 0] \begin{bmatrix} P_{\theta_1}[n] \\ L_{m-1}\delta(n-m-1) \end{bmatrix}. \quad (137)$$

A matriz  $[1 \ 0]$  corresponde à matriz  $H$  da equação do sistema que permite potencializar ou ocultar os estados observados de acordo com as cadeias ocultas de Markov (Hidden Markov chains). Isso permite ao filtro estimar estados que não estão visíveis à saída do sistema.

## 4.2 Phantom para caracterização e calibração

Um corpo de prova ou phantom é um objeto confeccionado especialmente para caracterização e calibração do tomógrafo ou outros experimentos. Assim, para a caracterização dos resultados da pesquisa, um phantom homogêneo cilíndrico foi construído com um material composto de Nylon (Figura 27).

O phantom heterogêneo foi construído com uma base de plexiglass em forma cilíndrica, com apenas quatro comprimentos cilíndricos para inserção e materiais.

Os phantoms têm 60 mm de diâmetro, sendo que as amostras de cálcio e água possuem 19 mm de diâmetro e as amostras de fósforo e alumínio, 22 mm de diâmetro, como ilustrado na Figura 28 . As tomografias destes corpos de prova foram imageadas utilizando o minitomógrafo de primeira geração citado anteriormente.

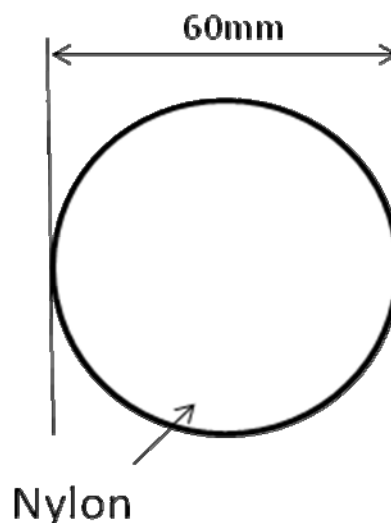


Figura 27. Diagrama para construção do *phantom* homogêneo.



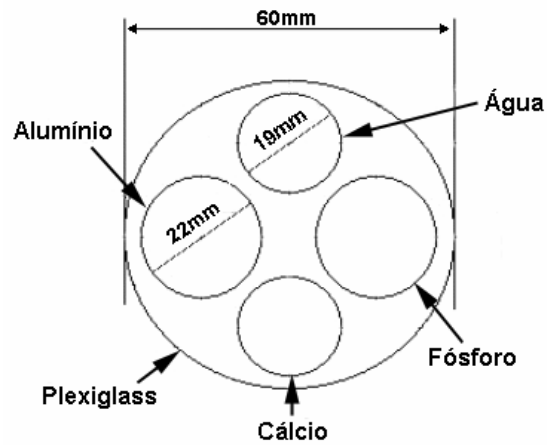


Figura 28. Diagrama para construção do *phantom* heterogêneo.

Para avaliação da filtragem, utilizaram-se *phantoms* e janela (ROI – *Region of interest*). Assim, pôde-se analisar não só o sinal da matriz de projeções como também a imagem reconstruída. As projeções e a imagem reconstruída para os *phantoms* homogêneo e heterogêneo sem filtragem são apresentadas na Figura 29.

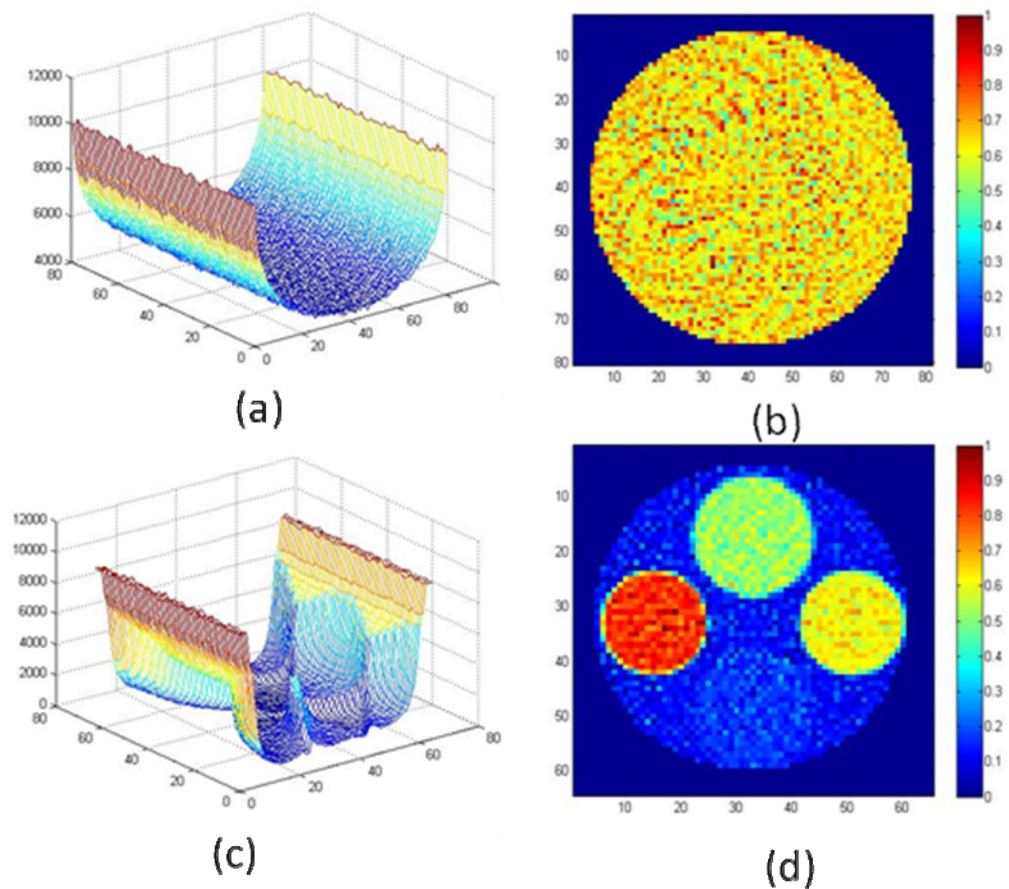


Figura 29. *Phantoms* utilizados para calibração e validação: projeções (a) e imagem reconstruída (b) do *phantom* homogêneo, projeções (c) e imagem reconstruída (d) do *phantom* heterogêneo.

### 4.3 Apresentação da ferramenta para filtragem com Kalman

Na Figura 30, é apresentada a ferramenta para auxiliar na filtragem de projeções tomográficas. As funções utilizadas nesta ferramenta são as mesmas utilizadas para gerar os resultados deste trabalho.

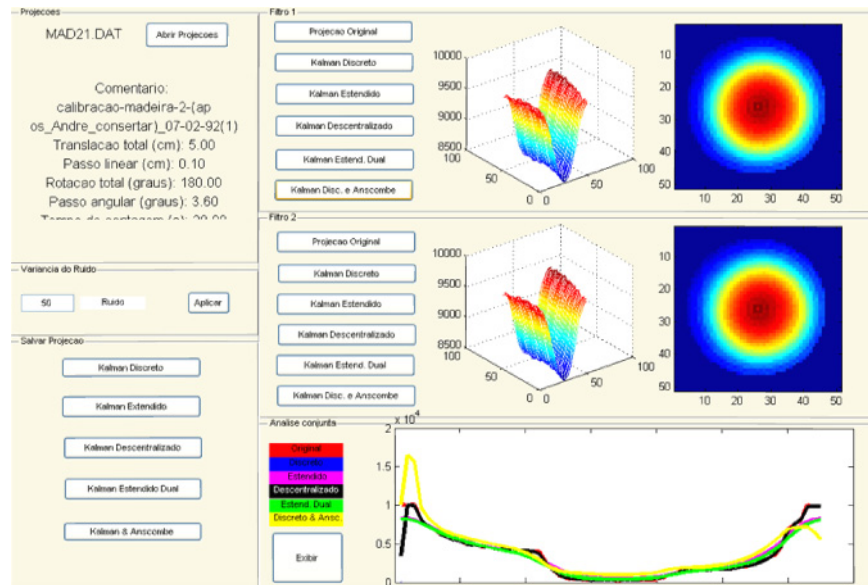


Figura 30. Ferramenta para filtragem de projeções tomográficas.

Na Figura 31, é apresentado o botão para carregar as projeções.



Figura 31. Campo para carregar as projeções. Os comentários dos arquivos é exibido logo abaixo do botão.

Quando selecionado, é aberta uma janela (Figura 32) para navegar entre os arquivos.

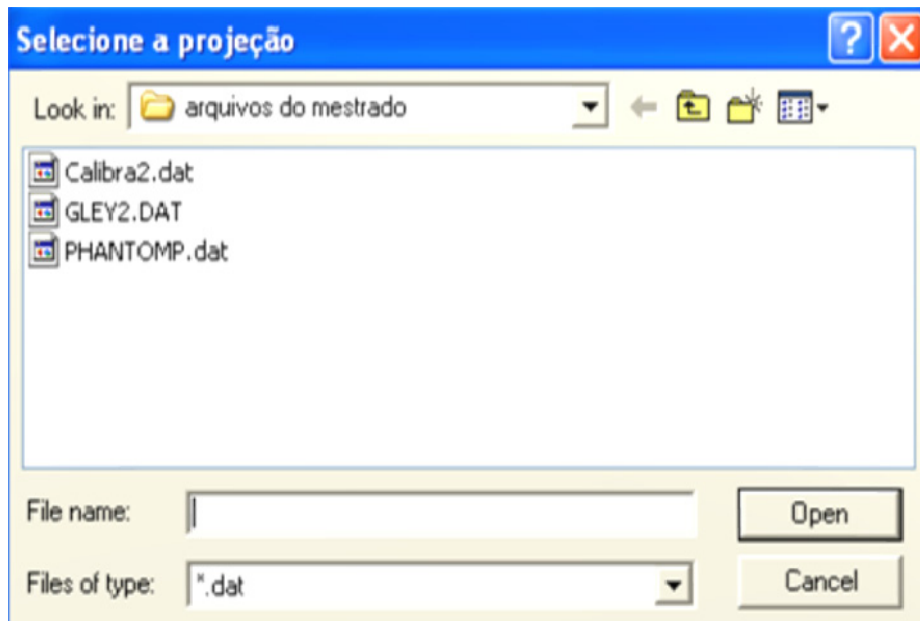


Figura 32. Caixa de diálogo para navegar entre os arquivos de projeções.

Após a abertura, o campo *Filtro 1* exibirá as projeções e sua respectiva reconstrução pelo algoritmo de retro-projeção filtrada (Figura 33).

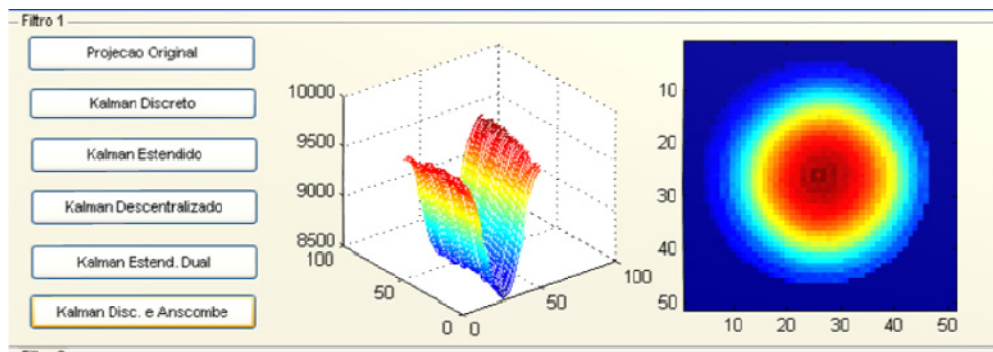


Figura 33. Após a escolha do arquivo que contem as projeções a serem tratadas, são exibidas o conjunto de projeções e a reconstrução da projeção no campo *Filtro 1*.

Na Figura 34, é apresentada o campo onde os dados da variância do ruído são escritos. Essa variância está relacionada ao ruído da observação, já que o ruído do processo de medição está configurado para ser 1, de acordo com a idéia de que o processo gerado pelo sistema é um ruído do tipo *Poisson*.



Figura 34. Campo para a entrada da variância do ruído nas projeções.

Após a entrada do valor da variância do ruído, selecionado o botão *Aplicar*, serão aplicados o filtro de Kalman e suas variações. Após serem feitas todas as filtragens, a projeção filtrada pelo filtro de Kalman discreto e sua respectiva reconstrução serão mostradas no campo *Filtro 2*, como ilustra a Figura 35.

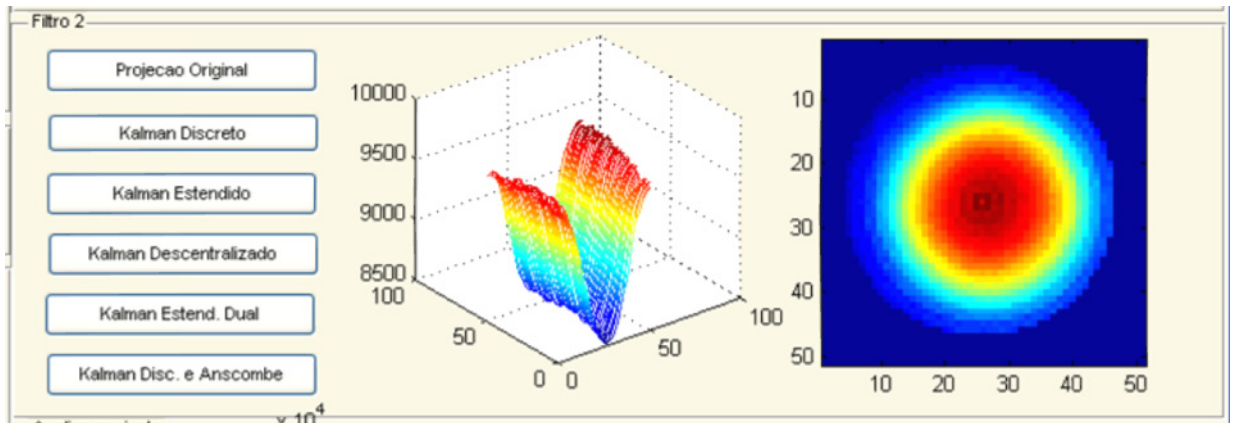


Figura 35. Campo *Filtro 2* onde são mostradas a projeção filtrada pelo Kalman discreto e sua reconstrução, após selecionar o botão *Aplicar*.

Há ainda a possibilidade de navegar pelos resultados selecionando os botões com os nomes dos filtros para analisar cada resultado.

Para uma análise mais profunda, pode-se também visualizar apenas o sinal intermediário, para uma comparação entre as filtragens, selecionando o botão *Exibir* apresentado na Figura 36.

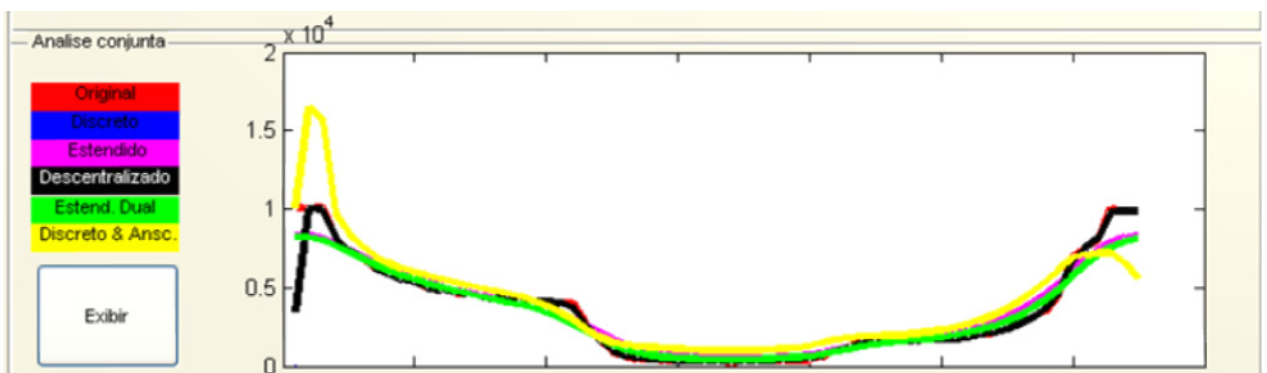


Figura 36. Análise dos sinais comparando as filtragens de Kalman e suas variações.

Por fim, pode-se salvar, de forma individual, cada resultado obtido selecionando os botões no campo *Salvar Projeção*, apresentados na Figura 37. Cada botão está relacionado ao filtro utilizado pela ferramenta. Quando selecionado o botão com o nome do filtro, uma

caixa de diálogo (Figura 38) irá se abrir, permitindo navegar entre os diretórios e salvar a projeção com um novo nome, mas seguindo o padrão do arquivo original.

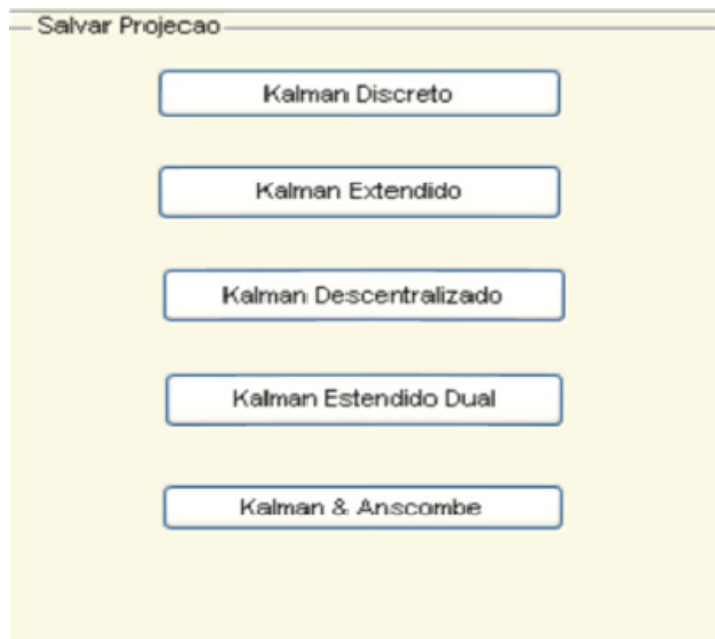


Figura 37. Botões para salvar de forma individual a projeção filtrada.

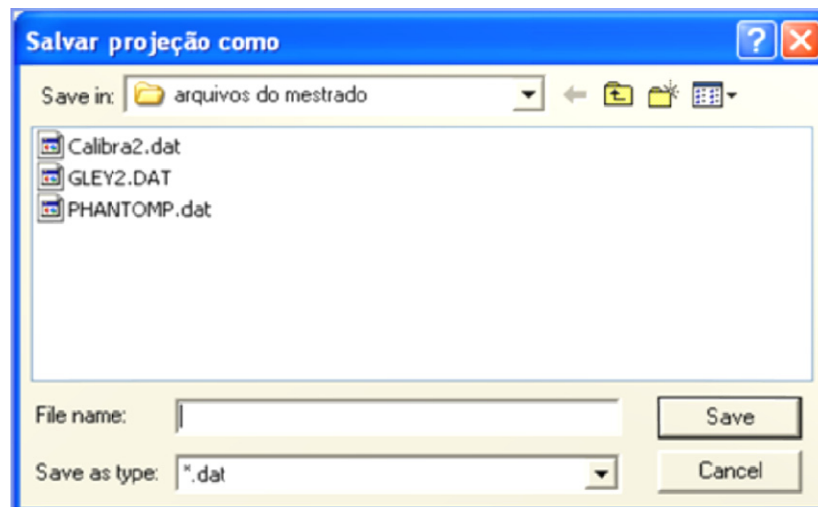


Figura 38. Caixa de diálogo para navegar pelos diretórios e salvar o conjunto de projeção filtrada.

Esta ferramenta é apenas para visualização. Outros parâmetros, como convergência da rede neural do filtro de Kalman estendido de estimação dual e espalhamento de pontos Sigma do filtro de Kalman descentralizado, estão pré-definidos, podendo ser atualizados diretamente nas funções.

#### 4.4 O Filtro de Kalman discreto para projeções tomográficas

O filtro de Kalman é um filtro linear como o filtro de médias, porém adaptativo, o que deve garantir um melhor resultado conforme o seu funcionamento, que consiste em

analisar os erros e modificá-los de acordo com o sinal observado. Para a filtragem do sinal tomográfico, foram utilizadas as equações para processo e sistema das equações 25 e 26 do capítulo 1.

A Figura 39 ilustra um diagrama para um sistema estocástico desenvolvido para a filtragem do ruído. A Figura 40 ilustra o sistema afetado por um ruído *Poisson* que possui média 0 e variância  $x$ , que corresponde ao valor de  $I$ . A variável  $\gamma$  representa a relação  $x/xp$ , onde  $xp$  corresponde ao sinal com ruído *Poisson*.

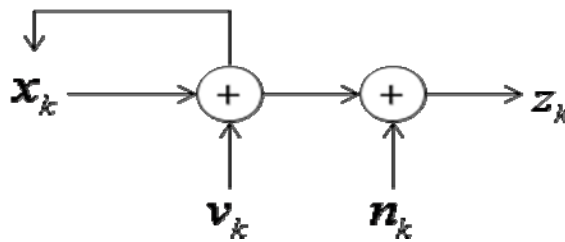


Figura 39. Diagramação do sinal para ruídos gaussianos brancos

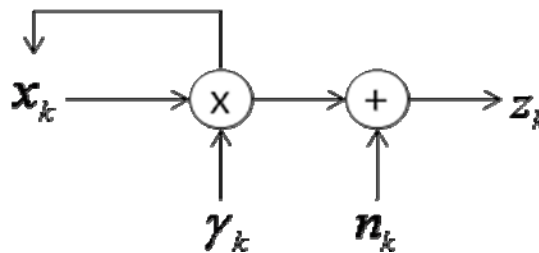


Figura 40. Diagramação do sinal para ruídos *Poisson* e gaussianos brancos. Nota-se que o ruído da função de transição do sistema depende diretamente do valor de  $x$ .

Nesta aplicação, consideraram-se apenas os ruídos gaussianos do sistema e a variância do ruído do processo  $Q$  com valor igual ao valor máximo do valor detectado no detector (feixe livre).

O diagrama de blocos dessa filtragem é apresentado na Figura 41, onde o suavizador RTS foi implementado conforme as equações apresentadas na seção 2.13.



Figura 41. Diagramas de blocos para o filtro de Kalman discreto.

Como o filtro de Kalman discreto não suporta funções não lineares ou tratamento de outros ruídos, adota-se um valor de variância  $Q$  que determina o grau de confiança no processo. Como o ruído do processo é um ruído *Poisson*, adota-se o valor do feixe livre como a variância do processo e para o ruído do sistema um desvio padrão que varia de 0 a 100 vezes o valor do feixe livre.

No quadro abaixo é apresentado o pseudocódigo da implementação do filtro em questão.

```

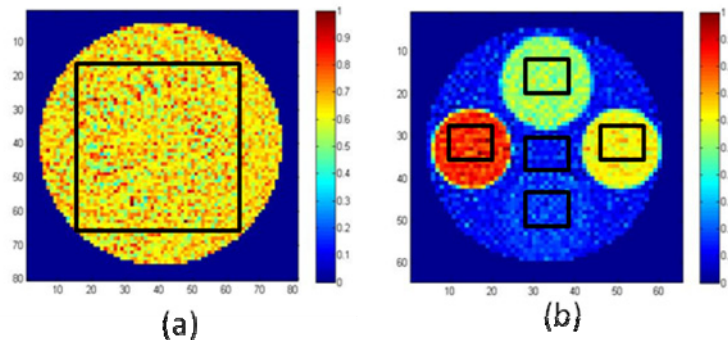
Início
  Lê projeções
  Define Projeção(1) % valor esperado da primeira projeção
  Define Covariância(1) % valor esperado da Covariância da primeira projeção
  k=número de projeções
  Valor para Q % Variância do ruído do processo.
  Valor para R % Variância do ruído do sistema.
  Para de i=2 até k
  Início
    [Projeção filtrada(i),Covariância(i)]=Kalman(Projeção(i), Projeção(i-1),
    Covariância(i-1),Q,R);
  Fim
  Para de i=k-1 até k
  Início
    [Projeção suavizada(i),Covariância da suavização(i)]=Filtro de
    Kalman(Projeção filtrada(i),
    Projeção filtrada(i-1),Covariância(i-1));
  Fim
  Reconstrução das projeções;
Fim

```

Para medir a qualidade da filtragem, foi utilizado o ISNR com base nas variâncias dos valores em janelas (ROIs) nas imagens reconstruídas. Um ISNR negativo indica a perda de detalhe ou deformação (presença de artefatos) na imagem final.

Para o *phantom* heterogêneo, utilizaram-se cinco janelas (ROIs), onde a presença de um valor negativo ou queda dos valores do ISNR indicam uma filtragem de pior qualidade.

Na Figura 42 são apresentadas as regiões de interesse usadas para quantificar os resultados obtidos com os filtros.



**Figura 42. Regiões de interesse para avaliar os resultados obtidos: Em (a) uma ROI de 42x28 pixels e em (b) cinco ROIs de 14x11 pixels.**

A Tabela 4 apresenta resultados de testes realizados para verificar a relação entre ISNR e a qualidade do sinal e da imagem reconstruída.

**Tabela 4 – Resultados obtidos com o filtro de Kalman discreto para *phantoms* para calibração.**

Variância do ruído do processo para $Q=10.000$	Phantom Homogêneo	Phantom Heterogêneo				
	ISNR (em dB)	ISNR (em dB)				
R=0	0,03	0	0	0	0	0
R=5.000	5,15	1,36	1,17	1,35	1,01	1,04
R=10.000	6,09	1,98	1,73	2,00	1,37	1,52
R=50.000	6,57	3,54	3,05	3,50	2,24	2,48
R=100.000	6,74	4,15	3,43	4,05	2,98	2,49
R=500.000	7,12	5,66	3,88	5,49	6,08	1,15
R=700.000	7,18	5,79	4,00	5,86	6,78	0,91
R=1.000.000	7,23	5,70	4,24	6,28	7,30	0,83

Um estudo com a variância do ruído do sistema foi realizado para obter o melhor valor de ISNR para as projeções filtradas. O melhor valor de ISNR para o *phantom* homogêneo foi de 7,23 dB para uma variância do ruído do sistema no valor de 1.000.000. A partir de 10.000, o mesmo valor da variância do ruído do processo, a melhoria foi aumentando em uma taxa menor. Para o *phantom* heterogêneo, a análise utilizou cinco janelas para os diferentes materiais do corpo. O melhor resultado foi obtido comparando os valores de ISNR para todos, sem haver regressão nos resultados. Com essa análise, verificou-se que os melhores resultados foram de 4,15 dB, 3,43 dB, 4,05 dB, 2,98 dB e 2,49 dB, com o valor da variância do ruído do processo 10 vezes maior do que o valor da variância ruído do sistema ( $Q=100.000$ ).

A Figura 43 apresenta o resultado da aplicação do filtro de Kalman discreto para projeções dos *phantoms* homogêneo e heterogêneo, respectivamente.



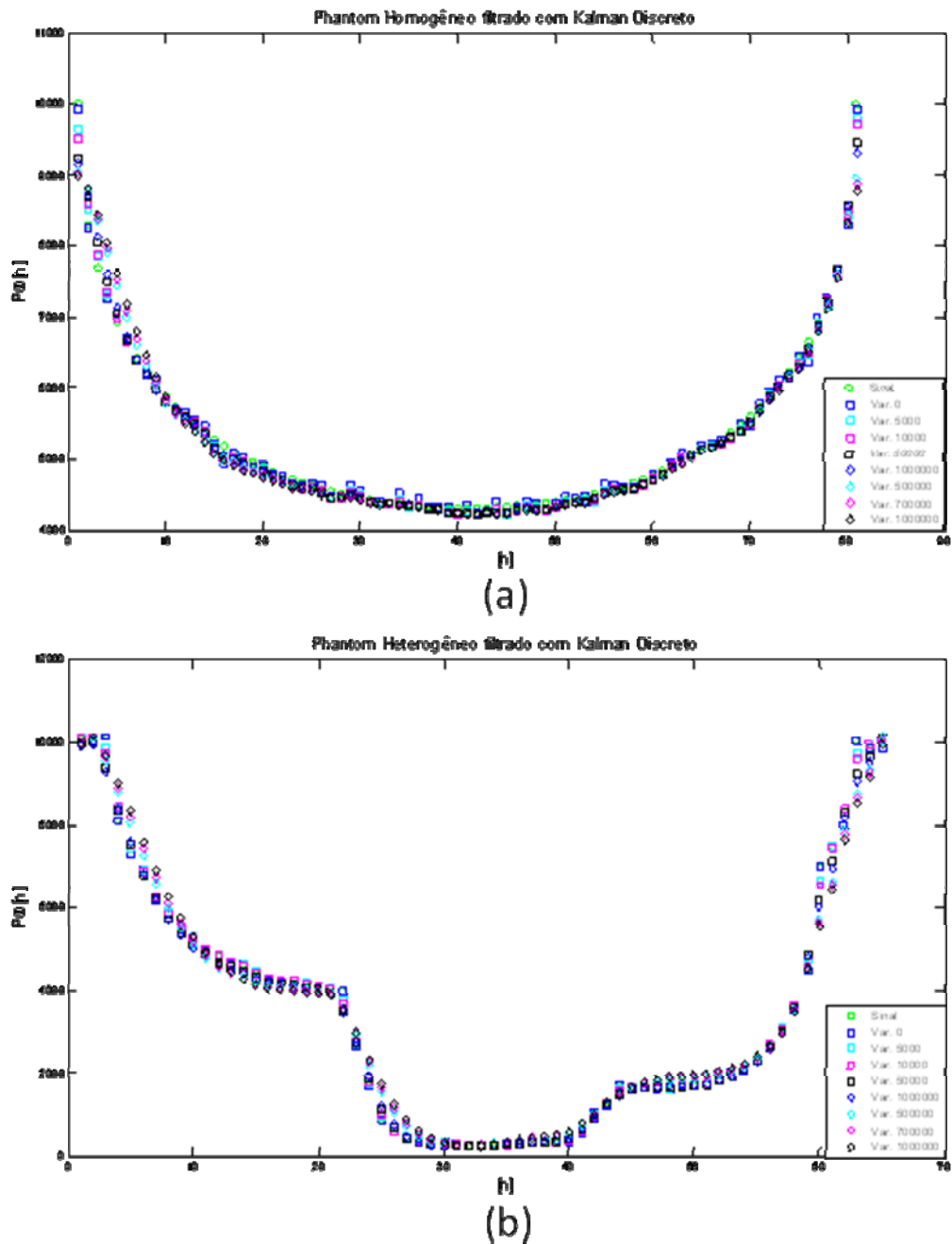


Figura 43. Resultados obtidos com o filtro de Kalman discreto com a variância do ruído do processo (Q) igual ao valor do feixe livre. Resultados obtidos pelo filtro com diferentes valores para o ruído do sistema dos *Phantoms* (a) homogêneo e (b) heterogêneo.

Os resultados obtidos com a aplicação do filtro para os *phantoms* podem ser vistos na Figura 44. Com base nas imagens reconstruídas, podem-se observar distorções, devido ao erro de filtragem. O algoritmo de suavização RTS adequa à variância do sinal para que esse erro seja minimizado, retornando, através de menores variâncias, melhores médias. Mesmo assim, é possível perceber uma definição melhor dos diferentes materiais na imagem do *phantom* heterogêneo e uma variação menor no *phantom* homogêneo.

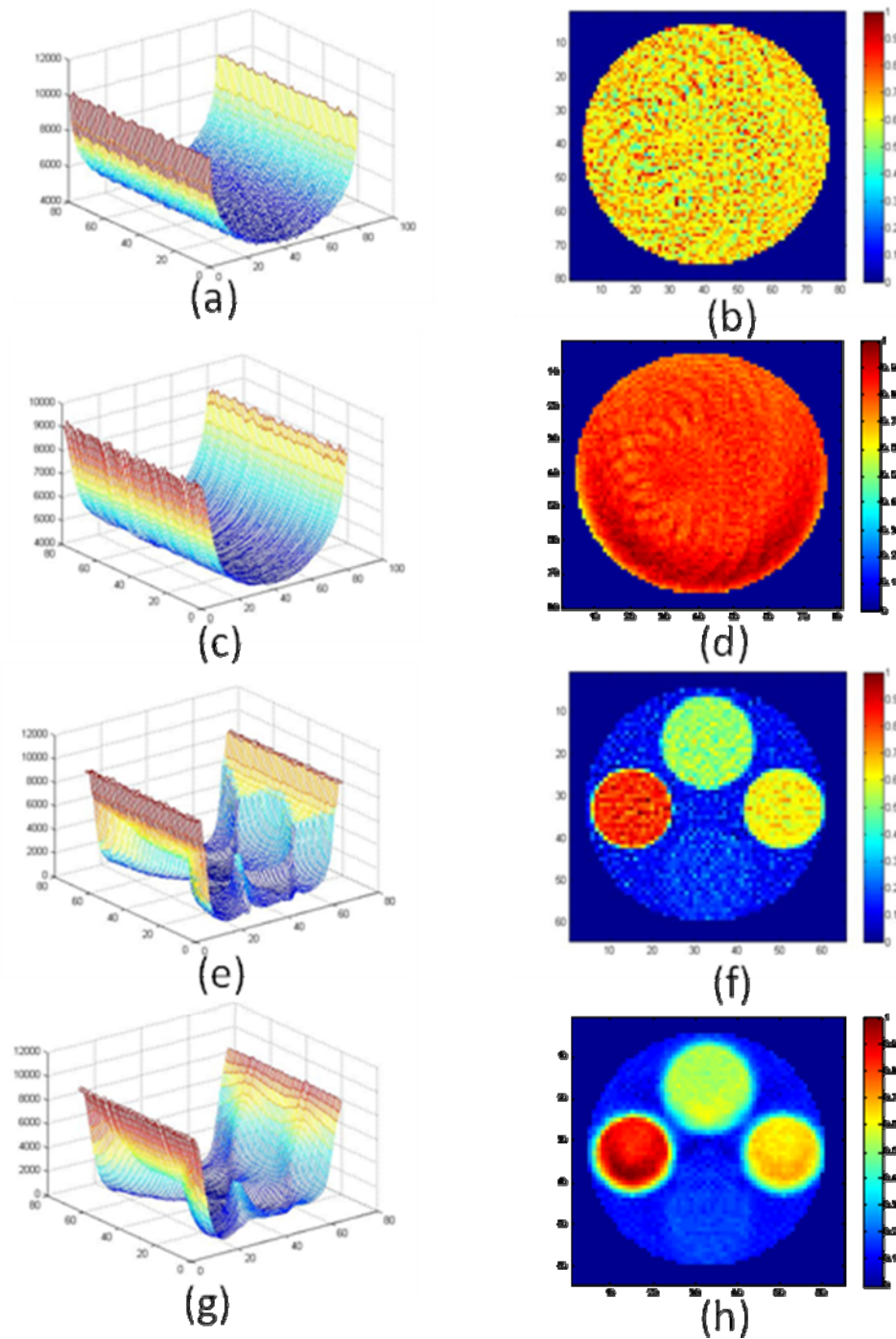


Figura 44. Projeções filtradas pelo filtro de Kalman discreto com variância igual ao valor do feixe livre (10000) para o ruído do processo e os melhores ISNR: (a) *phantom* homogêneo e (b) sua respectiva reconstrução; (c) *phantom* heterogêneo com sua respectiva reconstrução (d).

#### 4.5 Filtro de Kalman estendido com transformada de Anscombe para projeções tomográficas

Devido à natureza multiplicativa do ruído *Poisson*, cuja variância é dependente do valor do sinal, o desempenho do filtro de Kalman discreto é limitado. Utilizando a

modalidade estendida do filtro podemos aproximar o comportamento não-gaussiano do ruído com a transformada de Anscombe.

O mapeamento da função não-linear pode ser feito como:

$$\begin{bmatrix} P_{\theta_t}[m] \\ I_{m-1}\delta(n-m-1) \end{bmatrix} = \begin{bmatrix} f(I_m\delta(n-m), I_{m-1}\delta(n-m-1), w) \\ I_{m-1}\delta(n-m-1) \end{bmatrix} \quad (138)$$

$$P_{\theta_t}[n] = [1 \quad 0] \begin{bmatrix} P_{\theta_t}[n] \\ I_{m-1}\delta(n-m-1) \end{bmatrix} + v \quad (139)$$

onde  $w$  é um ruído branco com variância  $Q$  e  $v$  é um ruído branco com variância  $R$ . Um diagrama de blocos deste filtro é apresentado na Figura 45.

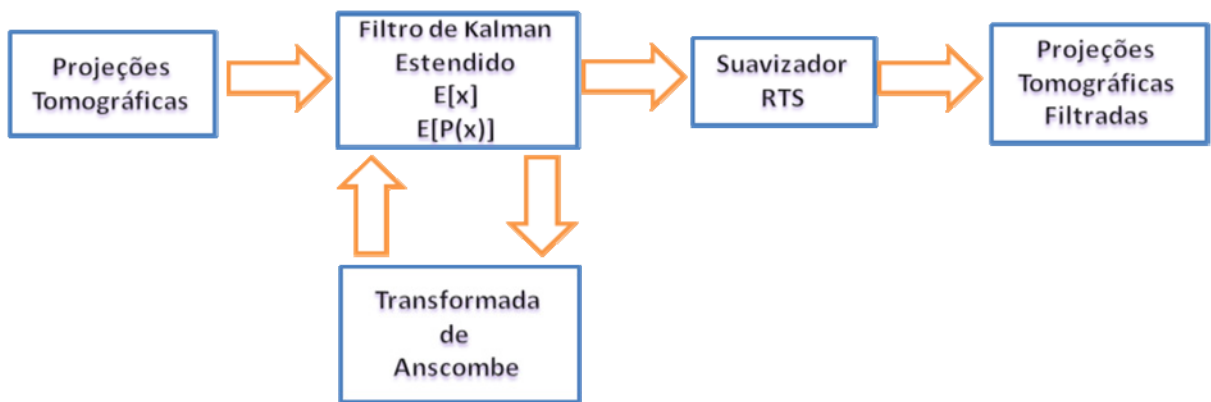


Figura 45. Diagrama de blocos para a filtragem com Kalman estendido e a função de transformada de Anscombe na equação do processo.

A Figura 46 ilustra o diagrama do sistema em que o ruído gaussiano branco é correlacionado com o valor dos estados da equação do processo e é transformado em ruído *Poisson*, através da transformada de Anscombe.

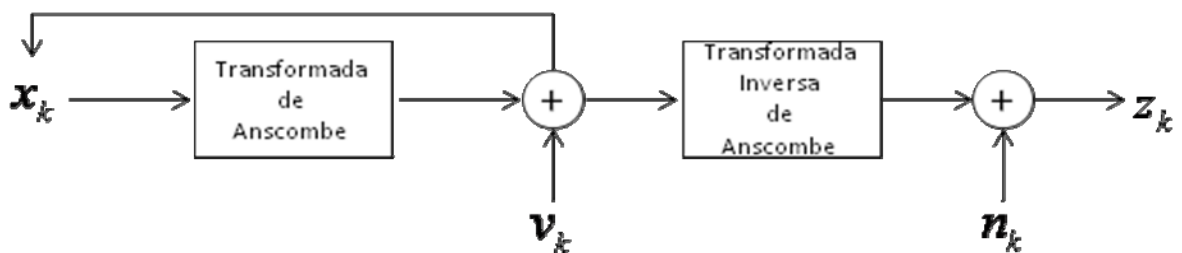


Figura 46. Diagramação do sinal utilizando as Transformadas de Anscombe.

A obtenção de projeções tomográficas pode ser definida pelo seguinte modelo dinâmico:

$$x_{k+1} = f(x_k, w_k) \quad (140)$$

$$z_k = Hx_k + n_k \quad (141)$$

onde  $v_k$  é um ruído gaussiano onde  $v_k \sim \mathcal{N}(0, Q)$ ,  $x_k$  um estado medido e  $f$  é uma função que aplica o ruído *Poisson* dependente do valor de  $v_k$ . A variável  $z_k$  é o estado observado perturbado por um ruído de  $n_k \sim \mathcal{N}(0, R)$ .

No quadro abaixo é apresentado o pseudocódigo da implementação do filtro em questão.

```

Início
  Lê projeções
  Define Projeção(1) % valor esperado da primeira projeção
  Define Covariância(1) % valor esperado da Covariância da primeira projeção
  Função Kalman % Algoritmo que possui duas funções internas: Transformada de
  Anscombe e % Transformada Inversa de Anscombe nas equações dos processos.
    k=número de projeções
    Valor para Q % Variância do ruído do processo.
    Valor para R % Variância do ruído do sistema.
    Para de i=2 até k
      Início
        [Projeção filtrada(i),Covariância(i)]=Kalman(Projeção(i),Projeção(i-1),Covariância(i-1,Q,R));
      Fim
    Para de i=k-1 até k
      Início
        [Projeção suavizada(i),Covariância da suavização(i)]=Filtro de Kalman(Projeção filtrada(i),
        Projeção filtrada(i-1),Covariância(i-1));
      Fim
    Reconstrução das projeções;
Fim

```

A Figura 47 apresenta o resultado da aplicação do filtro de Kalman estendido para projeções dos *phantoms* homogêneo e heterogêneo, respectivamente.

Conforme foram se ampliando os valores da variância do ruído do sistema, a confiança na equação do processo diminuiu, provocando distorções no sinal (excesso de filtragem). A Tabela 5 apresenta resultados de testes realizados para verificar a relação entre ISNR e a qualidade do sinal e da imagem reconstruída.

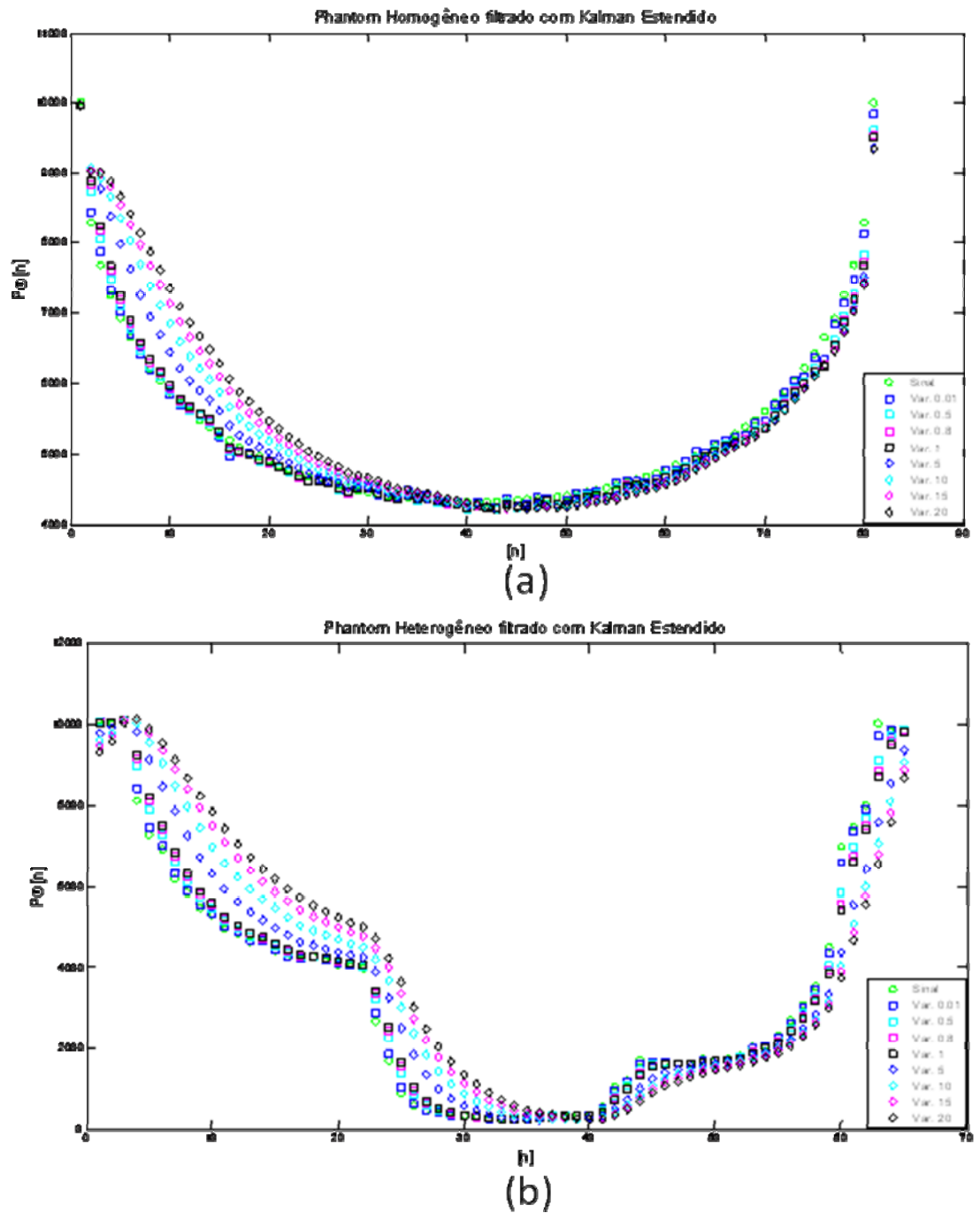


Figura 47. Resultados obtidos com o filtro de Kalman estendido com a transformada de Anscombe com a variância do ruído do processo ( $Q$ ) igual a 1. Resultados obtidos pelo filtro com diferentes valores para o ruído do sistema dos *Phantoms* (a) homogêneo e (b) heterogêneo.

No *phantom* homogêneo, percebe-se uma maior homogeneidade no corpo de prova, com uma menor filtragem do que o melhor resultado do filtro de Kalman discreto. Na imagem do *phantom* heterogêneo, o filtro de Kalman estendido houve um excesso de filtragem devido a erros no mapeamento da função do processo, com um atraso na convergência dos valores no sinal. Isto pode ser percebido tendo em vista a concentração das pseudocores na parte superior da imagem.

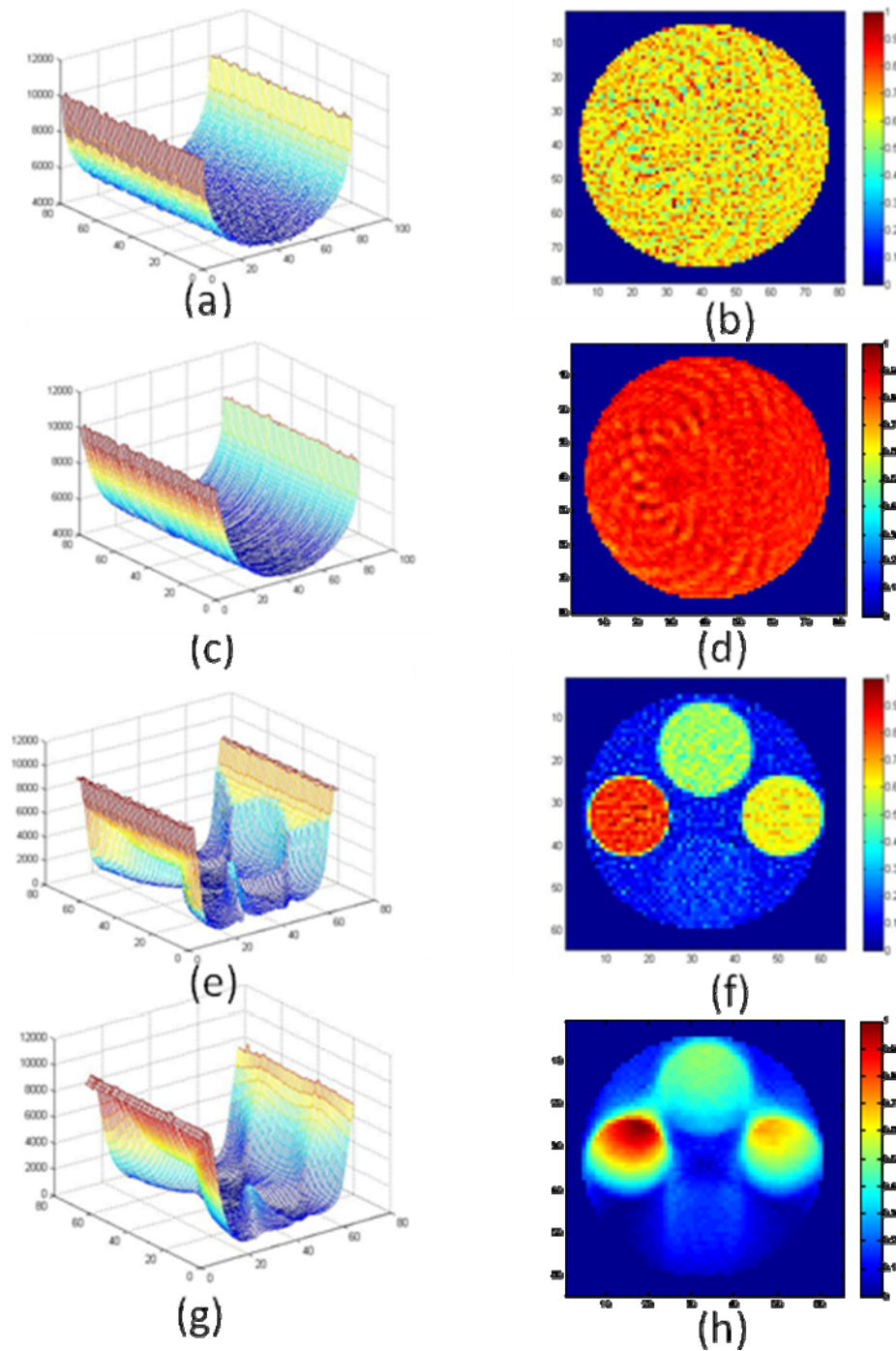
**Tabela 5 – Resultados obtidos com o filtro de Kalman estendido com transformada de Anscombe para *phantoms* para calibração.**

Variância do ruído do processo para Q=1	Phantom Homogêneo	Phantom Heterogêneo				
	ISNR (em dB)	ISNR (em dB)				
R=0,01	2,81	0,57	0,54	0,56	0,15	0,47
R=0,5	5,52	1,78	1,63	1,82	0,34	1,37
R=0,8	6,21	2,37	2,13	2,44	0,51	1,76
R=1	6,51	2,70	2,40	2,79	0,66	1,96
R=5	7,40	6,23	5,74	6,47	3,16	4,66
R=10	-39,83	8,27	7,48	8,60	4,00	6,19
R=15	-34,31	9,64	8,48	9,97	2,91	7,13
R=20	-33,84	10,82	9,47	11,13	1,56	8,30

Os maiores valores de ISNR foram obtidos pela adaptação ao ruído *Poisson*. Valores negativos mostram que houve um aumento na variância dos valores, o que pode ser provocado por aparecimento de distorções nas imagens. Logo, os melhores resultados obtidos foram 7,40 dB para o *phantom* homogêneo com variância do ruído do sistema igual a 5 e, para o *phantom* heterogêneo, a variância igual a 10. Apesar de melhores resultados para a variância igual a 20, houve uma queda nas medidas realizadas junto às janelas dispostas nos vários pixels dos materiais que compõem o *phantom* heterogêneo ISNR do material que envolve as amostras. Os resultados obtidos com a aplicação do filtro para os *phantoms* podem ser vistos na Figura 48.

#### 4.6 Filtro de Kalman estendido com redes neurais para projeções tomográficas

Como visto anteriormente, o filtro de Kalman pode receber uma função não-linear e treinar parâmetros (pesos). Há, então, a possibilidade de usar um mapeamento da função com redes neurais onde o próprio filtro treina os neurônios e avança para um sistema estável onde os pesos são estimados de forma ótima e a função mapeamento possui um menor erro possível. Este filtro permite trabalhar com ordens elevadas (com a precisão equivalente à terceira ordem da expansão das séries de Taylor), enquanto o filtro estendido trabalha apenas com funções de segunda ordem.



**Figura 48. Projeções filtradas pelo filtro de Kalman estendido com variância igual a 1 para o ruído do processo e os melhores ISNR obtidos: (a) *phantom* homogêneo e (b) sua respectiva reconstrução; (c) *phantom* heterogêneo com sua respectiva reconstrução (d).**

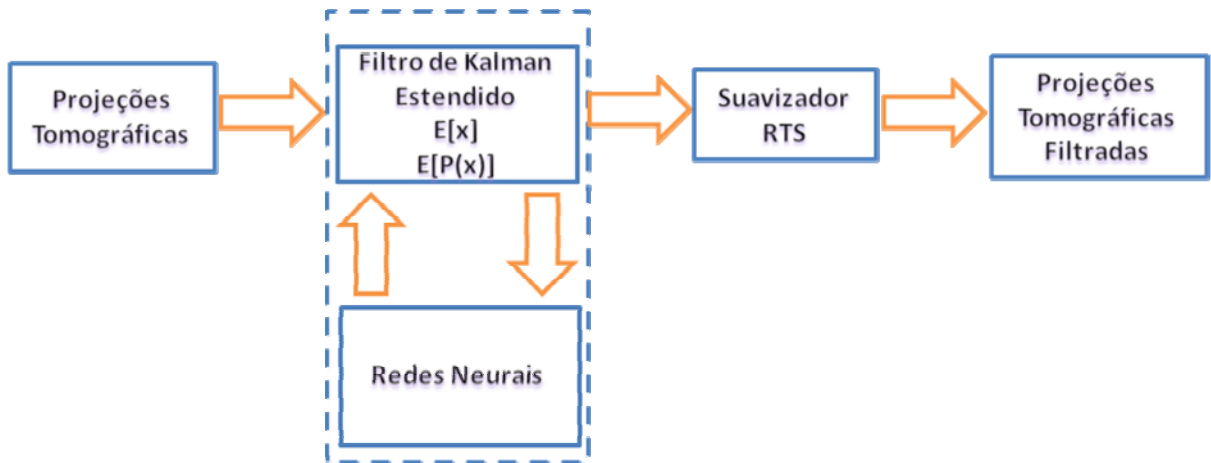
Com o conhecimento da função não-linear do processo e um filtro de Kalman que suporte funções lineares, se obtém uma melhora significativa no sinal. Como a transformada de Anscombe está presente na equação do processo, não há filtragem do ruído *Poisson*, além de o processo poder ser afetado por outros tipos de ruídos.

Uma solução seria utilizar uma rede neural para promover um melhor mapeamento da função de processo, diminuindo o ruído presente nas projeções. Para uma estimação dos pesos da rede neural em conjunto com as estimações dos estados, pode-se utilizar duas modalidades de filtragem: a estimação dual e a estimação em conjunto. Estas modalidades de filtragem determinam que os pesos iniciais sejam conhecidos, onde o próximo estado é obtido em um mapeamento linear com o estado anterior. Assim temos:

$$x_{k+1} = f(x_k, W_k, v_k) \quad (142)$$

Então, utiliza-se um filtro de Kalman para estimar os estados e um filtro de Kalman para estimar os pesos. Esta filtragem permite a aplicação em um sistema onde a dinâmica de estado é desconhecida ou caótica (não-determinística).

A Figura 49 mostra um diagrama de bloco com a implementação do filtro.



**Figura 49. Filtro de Kalman estendido com redes neurais.**

Podem-se utilizar as duas modalidades do filtro de Kalman para sistemas não-lineares para compor essa estimação dual. Apesar de ser uma forma mais completa de se estimar, ainda é passível de erros devido ao fato de a estimação do sinal observado ser aproximada.

Uma forma simplificada desse filtro é utilizar a estimação conjunta, em que há a possibilidade de estimar, de forma simultânea, os parâmetros necessários para a estimação dos estados do filtro, obtendo em um único filtro a função de estimação e os parâmetros, garantindo uma melhor eficiência no filtro, além de uma implementação simplificada do filtro:

$$\begin{bmatrix} P_{\theta_1}[n] \\ W \end{bmatrix} = \begin{bmatrix} f(P_{\theta_1}[n-1], W, v_k) \\ I * W \end{bmatrix} \quad (143)$$

$$P_{\theta_1}[n] = h(P_{\theta_1}[n], n_k) \quad (144)$$



onde  $W$  são os pesos para a função  $f$  que pode ser descrita como uma rede neural e  $I$  uma matriz identidade.

No quadro abaixo é apresentado o pseudocódigo da implementação do filtro em questão.

```

Início
  Lê projeções
  Define Projeção(1) % valor esperado da primeira projeção
  Define Covariância(1) % valor esperado da Covariância da primeira projeção
  Função Kalman % Algoritmo que possui uma função interna: que realiza os cálculos com os
  % pesos da rede neural e um valor de entrada.
    k=número de projeções
    Valor para Q % Variância do ruído do processo.
    Valor para R % Variância do ruído do sistema.
    Para de i=2 até k
      Início
        [Projeção filtrada(i),Covariância(i)]=Kalman(Projeção(i),Projeção(i-1),Covariância(i-1),Q,R);
      Fim
    Para de i=k-1 até k
      Início
        [Projeção suavizada(i),Covariância da suavização(i)]=Filtro de Kalman(Projeção filtrada(i),
        Projeção filtrada(i-1),Covariância(i-1));
      Fim
    Reconstrução das projeções;
Fim

```

Utilizou-se neste trabalho uma rede neural do tipo perceptron multicamadas com duas camadas, sendo a camada de entrada com apenas 2 neurônios na camada oculta com a função de transferência ou ativação sigmoideal e um neurônio na camada de saída com a função de transferência linear.

A Figura 50 apresenta o resultado da aplicação do filtro de Kalman estendido com redes neurais para projeções dos *phantoms* homogêneo e heterogêneo, respectivamente.

Como o filtro de redes neurais atua de maneira diferencial em relação ao filtro estendido, os valores para as variâncias do ruído do sistema foram modificados para

acompanhar melhor os resultados. São valores menores do que os dois filtros, pois a filtragem depende principalmente da variância dos valores dos pesos da rede neural.

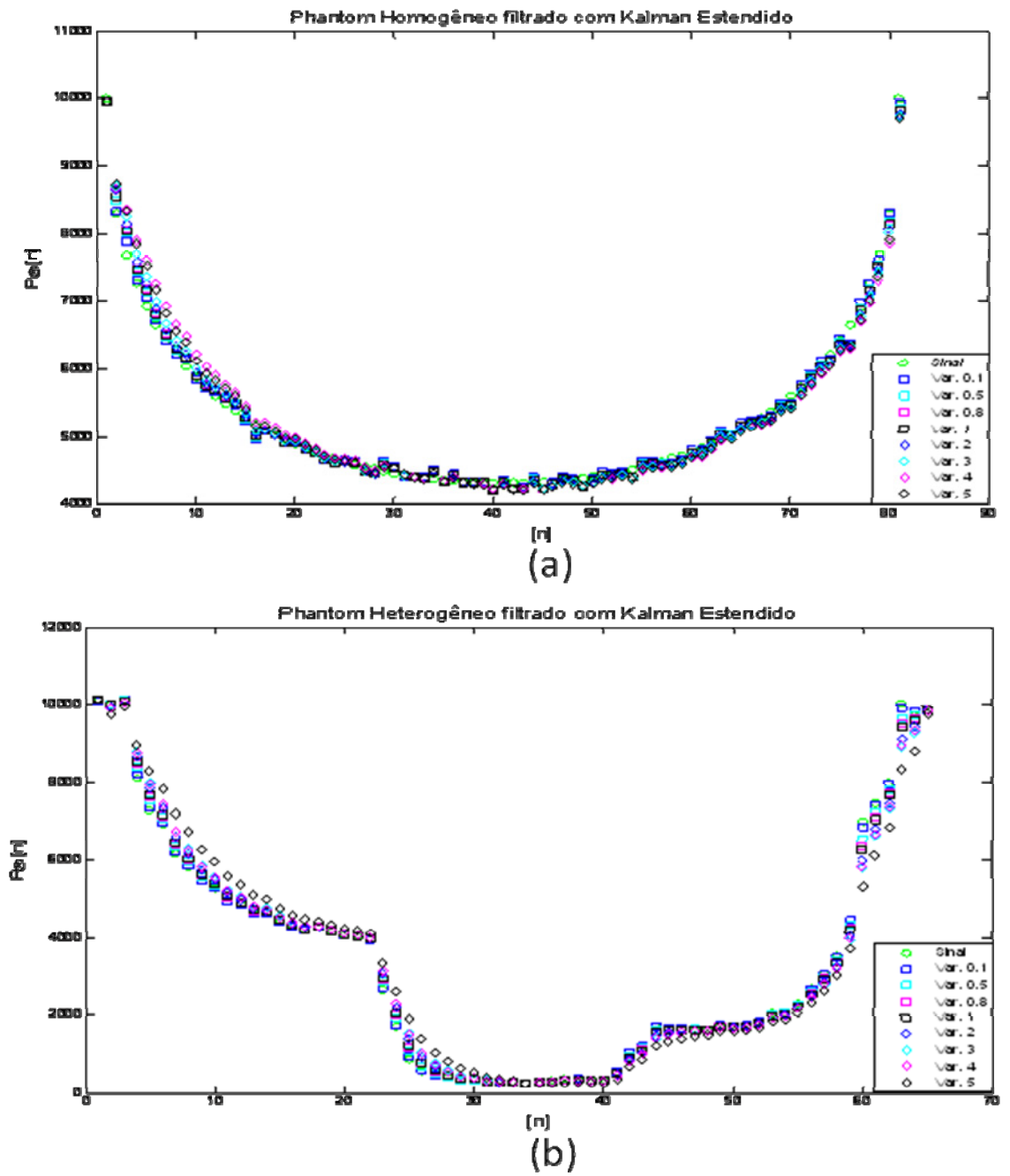


Figura 50. Resultados obtidos com o filtro de Kalman estendido com redes neurais com a variância do ruído do processo ( $Q$ ) igual a 1. Resultados obtidos pelo filtro com diferentes valores para o ruído do sistema dos *Phantoms* (a) homogêneo e (b) heterogêneo.

A

Tabela 6 apresenta resultados de testes realizados para verificar a relação entre ISNR e a qualidade do sinal e da imagem reconstruída.

**Tabela 6 – Resultados obtidos com o filtro de Kalman estendido com redes neurais para *phantoms* para calibração.**

Variância do ruído do processo para Q=1	Phantom Homogêneo	Phantom Heterogêneo				
	ISNR Máximo (em dB)	ISNR Máximo (em dB)				
R=0,1	0,84	0,31	0,23	0,30	0,15	0,24
R=0,5	2,64	1,19	0,98	1,20	0,50	0,69
R=0,8	3,42	1,55	1,20	1,52	0,91	1,48
R=1	3,87	2,01	2,10	1,80	1,18	0,99
R=2	4,70	3,20	2,50	2,89	1,55	1,90
R=3	5,38	3,55	2,54	3,77	1,76	2,63
R=4	5,23	3,97	3,51	4,23	2,13	2,19
R=5	5,02	4,14	3,13	4,92	3,01	3,98

Como se pode perceber nos resultados obtidos, o filtro de Kalman estendido com redes neurais gerou resultados com os valores de ISNR menores do que os do filtro de Kalman estendido com transformada de Anscombe.

Os valores do filtro permaneceram crescentes até o valor de R igual a 4 para o *phantom* homogêneo e R igual a 3 para o *phantom* heterogêneo.

Os resultados para os dois filtros se mantiveram na faixa da variância do ruído do sistema no valor de Q igual a 3. Apesar de ter valores de ISNR menores do que os obtidos com o filtro de Kalman estendido com transformada de Anscombe houve uma filtragem sem a perda de detalhamento na imagem. Pode-se comparar o resultado final do *phantom* heterogêneo do filtro em questão com os outros filtros obtidos anteriormente.

Os resultados obtidos com a aplicação do filtro para os *phantoms* podem ser vistos na Figura 51.

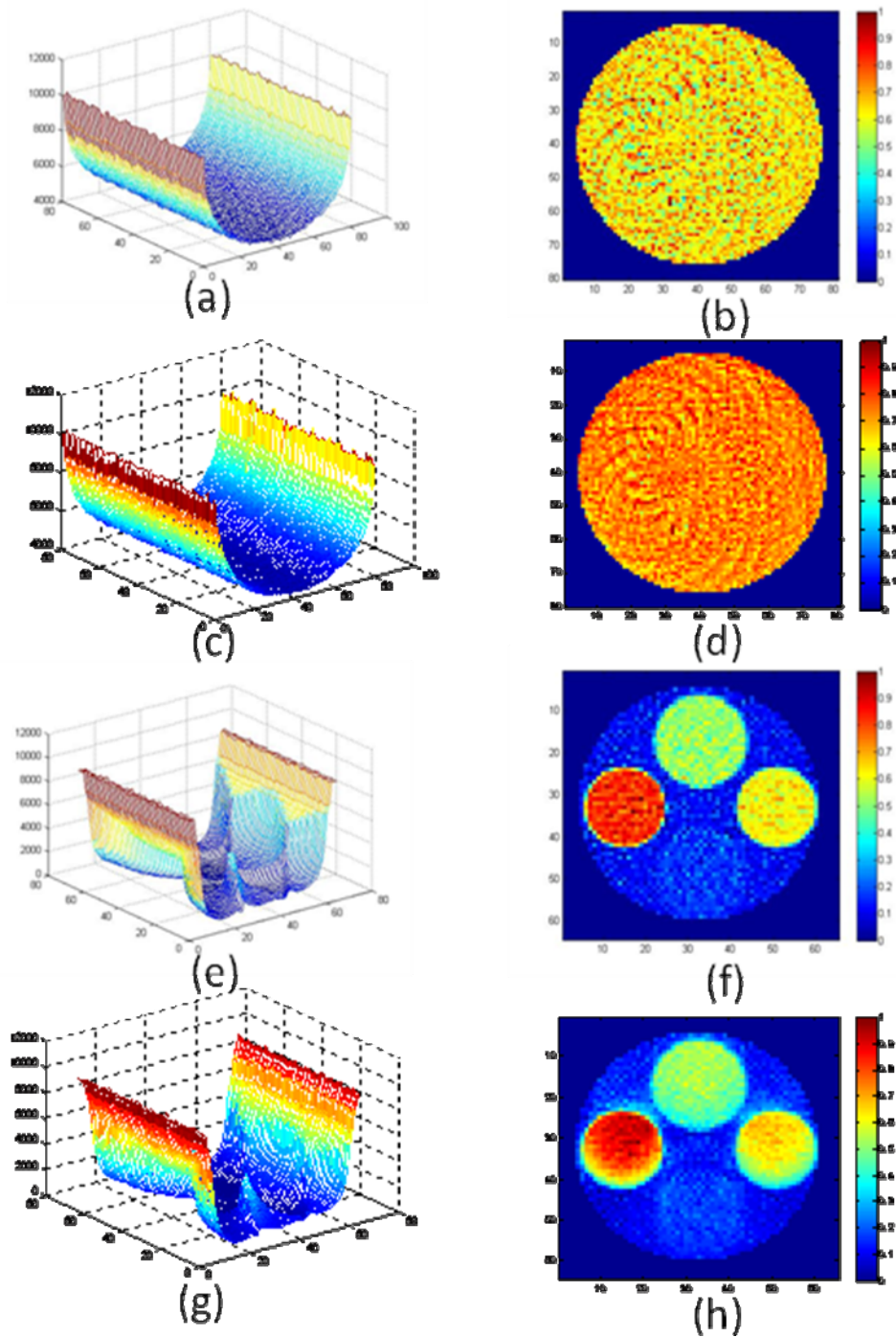


Figura 51. Melhores resultados obtidos com o filtro de Kalman estendido com redes neurais: (a) (b) *phantom* homogêneo e (c) (d) *phantom* heterogêneo.

#### 4.7 Filtro de Kalman descentralizado para projeções tomográficas

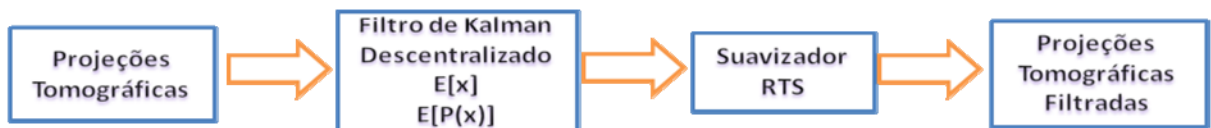
O filtro de Kalman estendido funciona com um processo de linearização da função não-linear dependendo de uma matriz jacobiana ou hessiana para a propagação das variâncias do sistema.

A linearização, em alguns casos, não compreende toda a função, levando aos erros de estimação. O filtro de Kalman descentralizado utiliza, com a transformação descentralizada, os recursos de criação de pontos sigmas na média e ao redor dela, realizando um mapeamento melhor do comportamento das variâncias em relação à função excluindo a necessidade de cálculo com matrizes de linearização.

O filtro, implementado neste trabalho, possui vários recursos agregados:

- Uso da covariância aumentada, que permite trabalhar com as variâncias do sinal, do ruído do processo e do ruído do sistema ao mesmo tempo, permitindo uma estimação do ruído, algo que não acontece com os outros filtros implementados.
- Uso da modalidade de filtro de raiz quadrada, que, utilizando a fatoração de Cholesky, permite uma estabilidade maior do filtro em relação aos valores do ruído, além de um ganho na ordem do filtro.
- Liberdade do uso de funções, sem necessidade de conhecimento *a priori* das respostas das funções.
- Precisão equivalente de funções de terceira ordem sem a necessidade de redes neurais.

Esses recursos foram realizados para o filtro para testar sua eficiência, substituindo os filtros na modalidade discreta, estendida com transformada de Anscombe e estendida com redes neurais. O diagrama de blocos para esse filtro é apresentado na Figura 52. Utilizando os mesmos valores para as variâncias do filtro de Kalman discreto, foram realizados testes para assegurar a estabilidade do filtro em sua modalidade descentralizada. O pseudocódigo do filtro em questão é semelhante ao pseudocódigo do filtro de Kalman discreto.



**Figura 52. Diagrama de blocos do filtro de Kalman descentralizado**

A Tabela 7 apresenta resultados de testes realizados para verificar a relação entre ISNR e a qualidade do sinal e da imagem reconstruída.

A Figura 53 apresenta o resultado da aplicação do filtro de Kalman descentralizado para projeções dos *phantoms* homogêneos e heterogêneos.

**Tabela 7 – Resultados obtidos com o filtro de Kalman descentralizado para *phantoms* para calibração.**

Variância do ruído do processo para Q=1	Phantom Homogêneo	Phantom Heterogêneo				
	ISNR Máximo (em dB)	ISNR Máximo (em dB)				
R=0,01	-0,57	4.52	4.43	4.63	1.82	3.96
R=0,5	-0,81	4.52	4.43	4.63	1.82	3.96
R=0,8	-0,59	4.52	4.43	4.63	1.82	3.96
R=1	-1,88	4.52	4.43	4.63	1.82	3.96
R=5	-2,76	4.52	4.43	4.63	1.82	3.96
R=10	-0,17	4.52	4.43	4.63	1.82	3.96
R=15	-2,40	4.52	4.43	4.63	1.82	3.96
R=20	-0,40	4.52	4.43	4.63	1.82	3.96

Analisando os valores da tabela, é possível perceber uma queda no ISNR do *phantom* homogêneo devido à presença de picos na imagem. Esses picos podem ser do próprio algoritmo de reconstrução ou até mesmo presença de outros ruídos mecânicos.

O filtro de Kalman descentralizado apresenta uma melhor convergência em relação ao filtro discreto, além de uma melhor estabilidade de filtragem, onde a variância dos valores para diferentes variâncias do ruído do sistema foram mínimas.

Os valores de ISNR do *phantom* heterogêneo demonstram a estabilidade do algoritmo em relação à sua versão básica discreta, ou seja, para quaisquer que sejam os valores do ruído do sistema, a filtragem é estável devido ao ruído *Poisson* ser mais influente do que o ruído branco em altas contagens de fótons. O melhor resultado com o *phantom* homogêneo foi com o valor da variância do sistema de 10 enquanto com o *phantom* heterogêneo foi estável para qualquer valor da variância. A filtragem do ruído branco do sistema foi eficaz para as duas amostras, sendo que a queda de valor do ISNR no *phantom* homogêneo é devido à estimação dos estados futuros do filtro condizer com os ruídos já apresentados nas amostras.

Nos gráficos de comparação entre os tipos de filtragem percebe-se que os sinais filtrados se apresentam de forma estável também com poucas diferenças visíveis, variando apenas os valores mais altos da contagem de fótons com o sinal ruidoso original.

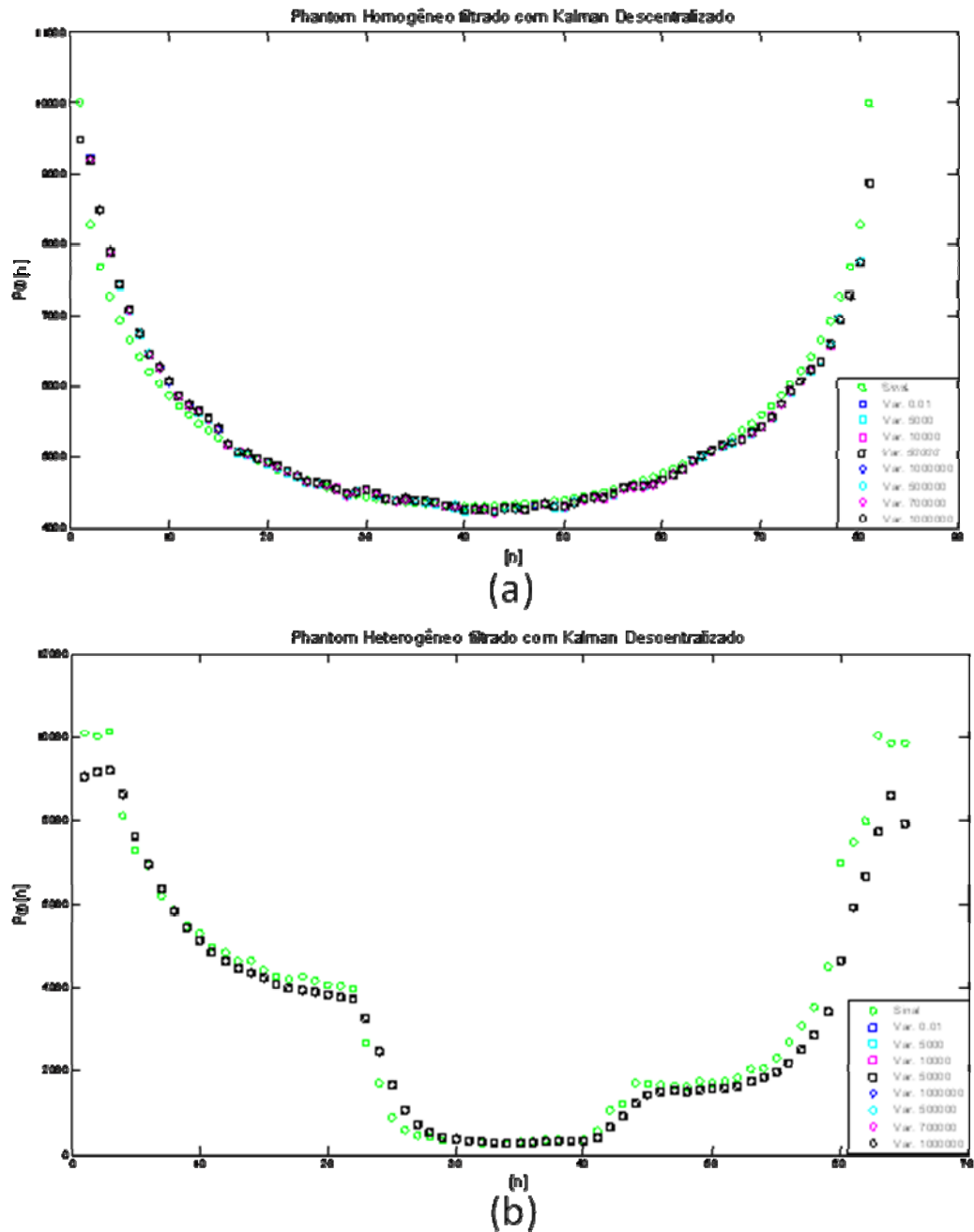
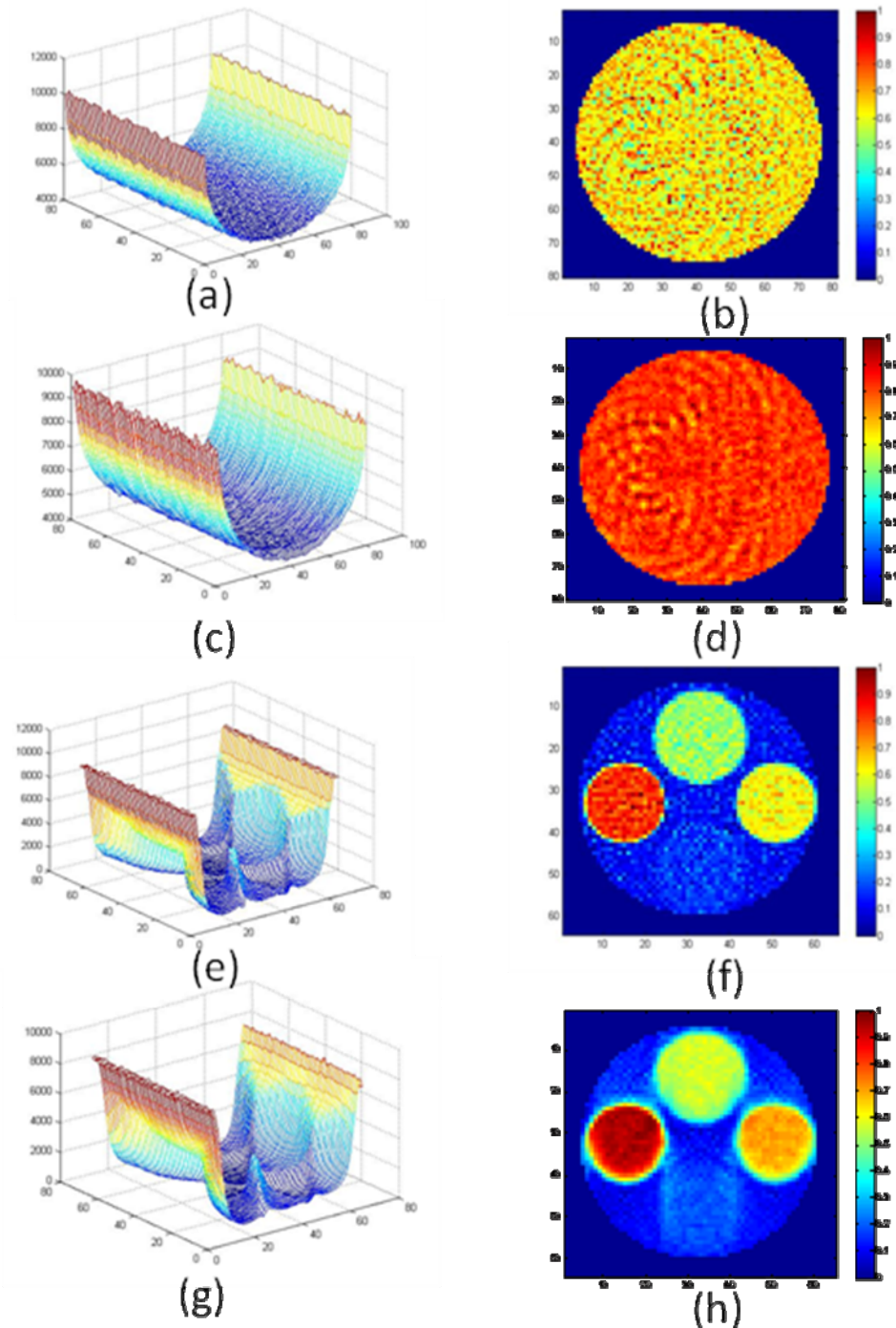


Figura 53 – Resultados obtidos com o filtro de Kalman descentralizado na sua forma pura, equivalente ao filtro de Kalman discreto.

Os resultados obtidos com a aplicação do filtro para os *phantoms* podem ser vistos na Figura 54. Verificando as projeções filtradas, é possível notar uma filtragem mais consistente e estável. Observando a variância de valores nos pixels da imagem reconstruída do *phantom* homogêneo, percebe-se uma concentração dos valores numa faixa menor, mesmo com a presença de pixel de diferentes valores (mais escuros). O *phantom* heterogêneo apresentou uma melhor estabilidade, com as amostras bem definidas dentro do

corpo estudado. Como o filtro consegue estimar de forma precisa os ruídos que afetam os estados do processo e da observação, os resultados apresentam melhores detalhes.



**Figura 54. Resultados obtidos com o filtro de Kalman descentralizado para (a)(b) *phantom* homogêneo e (c)(d) *phantom* heterogêneo.**

O filtro de Kalman descentralizado também pode substituir as equações da transformada de Anscombe, com a vantagem de utilizar uma estimativa do ruído gaussiano (que pode ser transformado em ruído *Poisson* nas equações de atualização das medidas) e,



com uso da transformação descentralizada, mapear a variância de forma exata sem o uso de linearização.

A linearização da transformada de Anscombe não demonstrou ser perfeita pela necessidade de se variar o ruído estimado de acordo com um resultado não conhecido ou aproximado.

A Figura 55 apresenta o diagrama de blocos para a filtragem com Kalman descentralizado e transformada de Anscombe.

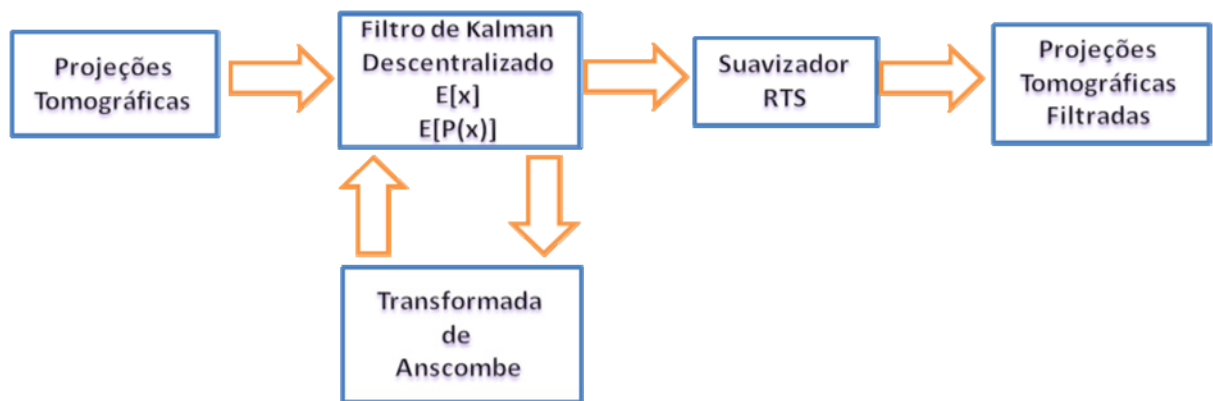


Figura 55. Filtro de Kalman descentralizado com transformada de Anscombe.

O filtro de Kalman descentralizado com a transformada de Anscombe apresenta a mesma estabilidade do filtro de Kalman descentralizado, mas respeitando o ruído *Poisson* da equação do processo, ou seja, não realiza a filtragem deste ruído, se adaptando ao comportamento do ruído gerado pela fonte de emissão de fótons, filtrando apenas o ruído branco nas projeções tomográficas, pois em baixas contagens de fótons, a variância do ruído *Poisson* é baixa, sendo uma tarefa árdua tentar separar um ruído aditivo, com influência maior do que o ruído multiplicativo, do sistema.

Essa característica do ruído aditivo de ter sempre uma variância constante independente das projeções permite uma filtragem mais estável, só que os estados obtidos ainda são ruidosos.

A Figura 56 apresenta o resultado da aplicação do filtro de Kalman descentralizado com transformada de Anscombe para projeções dos *phantoms* homogêneo e heterogêneo, respectivamente.

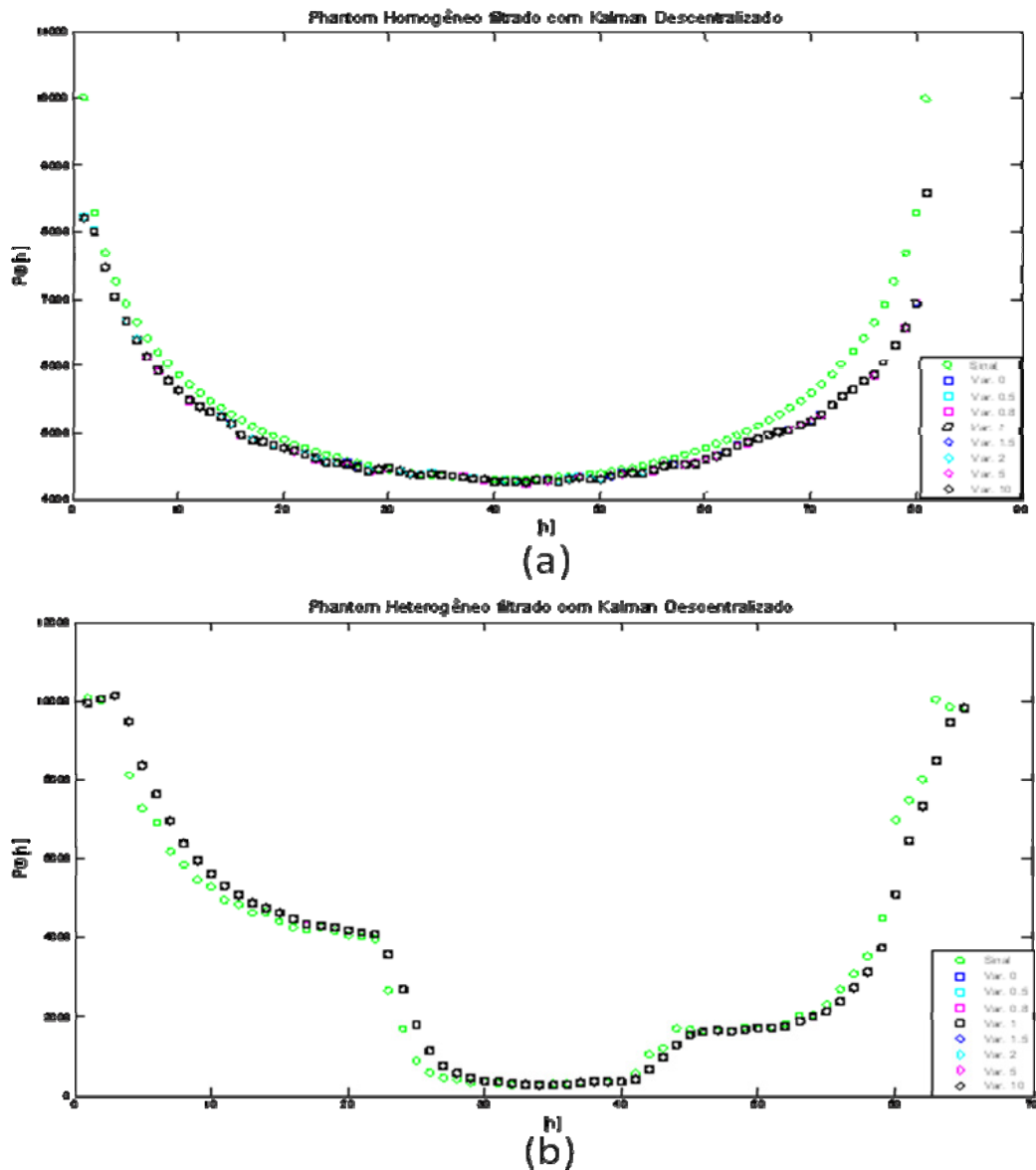


Figura 56. Comparação entre os sinais para a filtragem com Kalman descentralizado usando a transformada de Anscombe na equação do processo.

Devido ao erro de estimação na transição dos estados (equação de processo), houve uma distorção no sinal filtrado no *phantom* homogêneo. Esses erros de estimação são caracterizados devido à natureza da medida do ruído *Poisson* que, apesar de poder ser mapeado por uma função não linear, ainda está sob o domínio de uma função linear (translação) para a estimação de estados futuros. O mesmo pode ser visto no *phantom* heterogêneo, agora com uma estimação melhor, onde o filtro convergiu de maneira mais eficiente, comparando as projeções iniciais com as projeções iniciais.

A Tabela 8 apresenta resultados de testes realizados para verificar a relação entre ISNR e a qualidade do sinal e da imagem reconstruída.

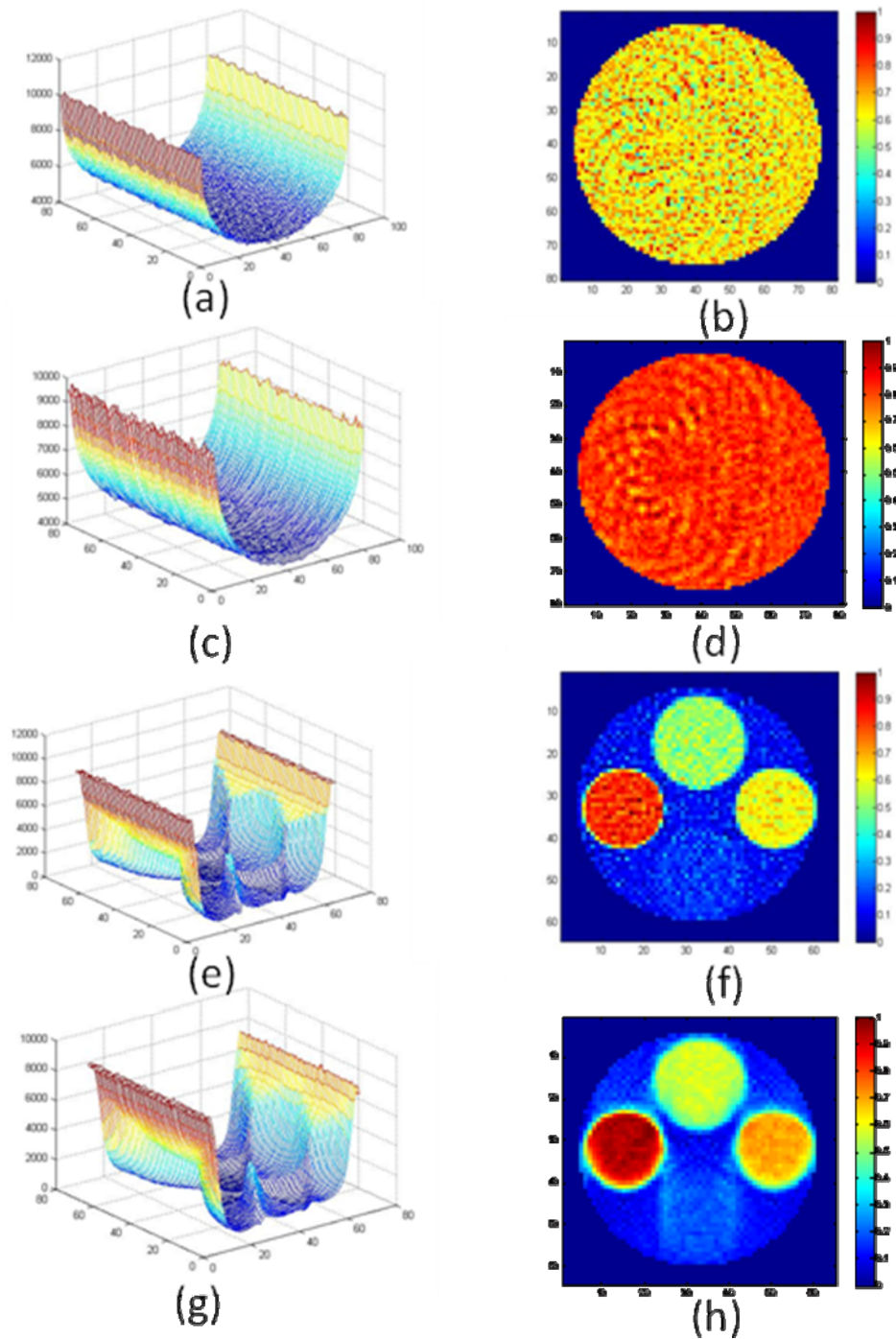
**Tabela 8 – Resultados obtidos com o filtro de Kalman descentralizado com a transformada de Anscombe para *phantoms* para calibração.**

Variância do ruído do processo para Q=1	Phantom Homogêneo	Phantom Heterogêneo				
	ISNR Máximo (em dB)	ISNR Máximo (em dB)				
R=0,01	-8,78	2,84	2,65	2,94	0,57	2,05
R=0,5	-8,87	2,84	2,65	2,94	0,57	2,05
R=0,8	-8,97	2,84	2,65	2,94	0,57	2,05
R=1	-8,74	2,84	2,65	2,94	0,57	2,05
R=5	-9,23	2,84	2,65	2,94	0,57	2,05
R=10	-8,74	2,84	2,65	2,94	0,57	2,05
R=15	-8,80	2,84	2,65	2,94	0,57	2,05
R=20	-8,85	2,84	2,65	2,94	0,57	2,05

O filtro se mostrou instável com os demais valores, mas com valores mais baixos de ISNR devido ao mapeamento do ruído *Poisson*. A partir desse resultado, pode-se verificar que ambas projeções são afetadas por ruídos não gaussianos, pois não há nenhuma melhoria quando se aumenta o valor da variância do ruído do sistema. O melhor resultado com o *phantom* homogêneo foi com o valor da variância do sistema com o valor de 15 enquanto com o *phantom* heterogêneo foi estável para qualquer valor da variância. Os resultados obtidos com a aplicação do filtro para os *phantoms* podem ser vistos na Figura 57.

Os resultados foram semelhantes para qualquer valor da variância do sistema, provando uma estabilidade do filtro. No caso do *phantom* homogêneo houve uma queda na qualidade da imagem (aumento na variância dos valores) que, possivelmente é devido ao algoritmo de reconstrução e até mesmo outros ruídos presentes na equação do processo ou adicionados na equação do sistema (como ruídos impulsivos ou coloridos).

Para eliminar o ruído *Poisson* das projeções, passou-se a adotar como variância do ruído do processo o valor próximo a zero (a sensibilidade do filtro não permite adotar valores menores do que o valor do parâmetro do filtro para espalhamento dos pontos sigmas).



**Figura 57. Conjunto de projeções e suas respectivas imagens reconstruídas filtradas pelo filtro de Kalman descentralizado com transformada de Anscombe: (a)(b) *phantom* homogêneo e (c)(d) *phantom* heterogêneo.**

A Figura 58 apresenta o resultado da aplicação do filtro de Kalman descentralizado com transformada de Anscombe para projeções dos *phantoms* homogêneo e heterogêneo, respectivamente.

Com a diminuição do valor da variância do processo com o objetivo de eliminar o ruído *Poisson*, o filtro já não apresenta a mesma estabilidade, mas aproxima-se no valor ótimo, como se pode ver nos valores finais do sinal.

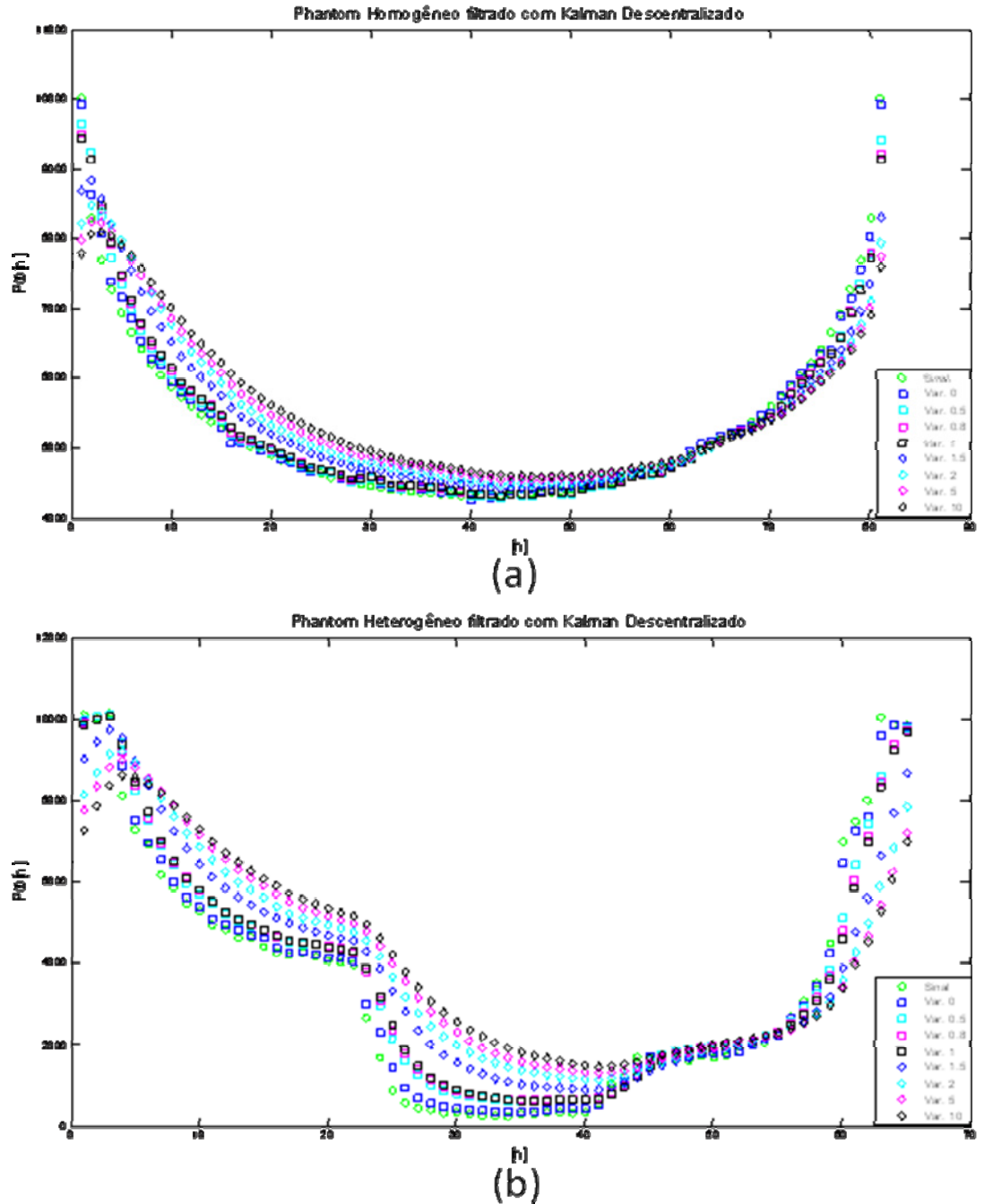


Figura 58. Resultados obtidos com o filtro de Kalman descentralizado com transformada de Anscombe com variância do ruído do processo próximo de zero para a filtragem do ruído *Poisson*.

Nas imagens, pode-se observar uma diferença em relação à homogeneidade das amostras, devido à filtragem do ruído *Poisson*.

Os resultados foram os melhores que os dos outros filtros. Seguindo uma tendência de estabilidade do filtro, na próxima seção serão apresentados os resultados com redes neurais e o filtro de Kalman descentralizado.

A Tabela 9 apresenta resultados de testes realizados para verificar a relação entre ISNR e a qualidade do sinal e da imagem reconstruída.

**Tabela 9 – Resultados obtidos com o filtro de Kalman descentralizado com a transformada de Anscombe para *phantoms* para calibração.**

Variância do ruído do processo para $Q=0,0001$	Phantom Homogêneo	Phantom Heterogêneo				
	ISNR Máximo (em dB)	ISNR Máximo (em dB)				
R=0,01	3,93	1,25	1,88	1,75	0,78	1,26
R=0,5	6,53	4,37	3,28	4,42	5,48	2,78
R=0,8	6,75	5,81	4,38	5,39	5,30	4,52
R=1	6,79	6,70	5,30	6,09	5,27	5,38
R=5	5,57	12,18	10,86	13,18	5,18	13,63
R=10	-2,25	15,74	14,55	16,16	4,99	16,99
R=15	-20,86	18,07	15,71	18,61	5,31	17,98
R=20	-23,64	15,74	14,55	16,16	4,99	16,99

Os melhores valores para a filtragem tanto do *phantom* homogêneo quanto do *phantom* heterogêneo foram obtidos com a variância do sistema igual a 1, tendo uma melhoria considerável comparando com a filtragem anterior. Os resultados obtidos com a aplicação do filtro para os *phantoms* podem ser vistos na Figura 59.

Os valores se apresentaram de forma desestabilizada, mas permitiu uma melhora considerável em relação à definição das amostras no *phantom* heterogêneo, observando a uniformidade das pseudocores.

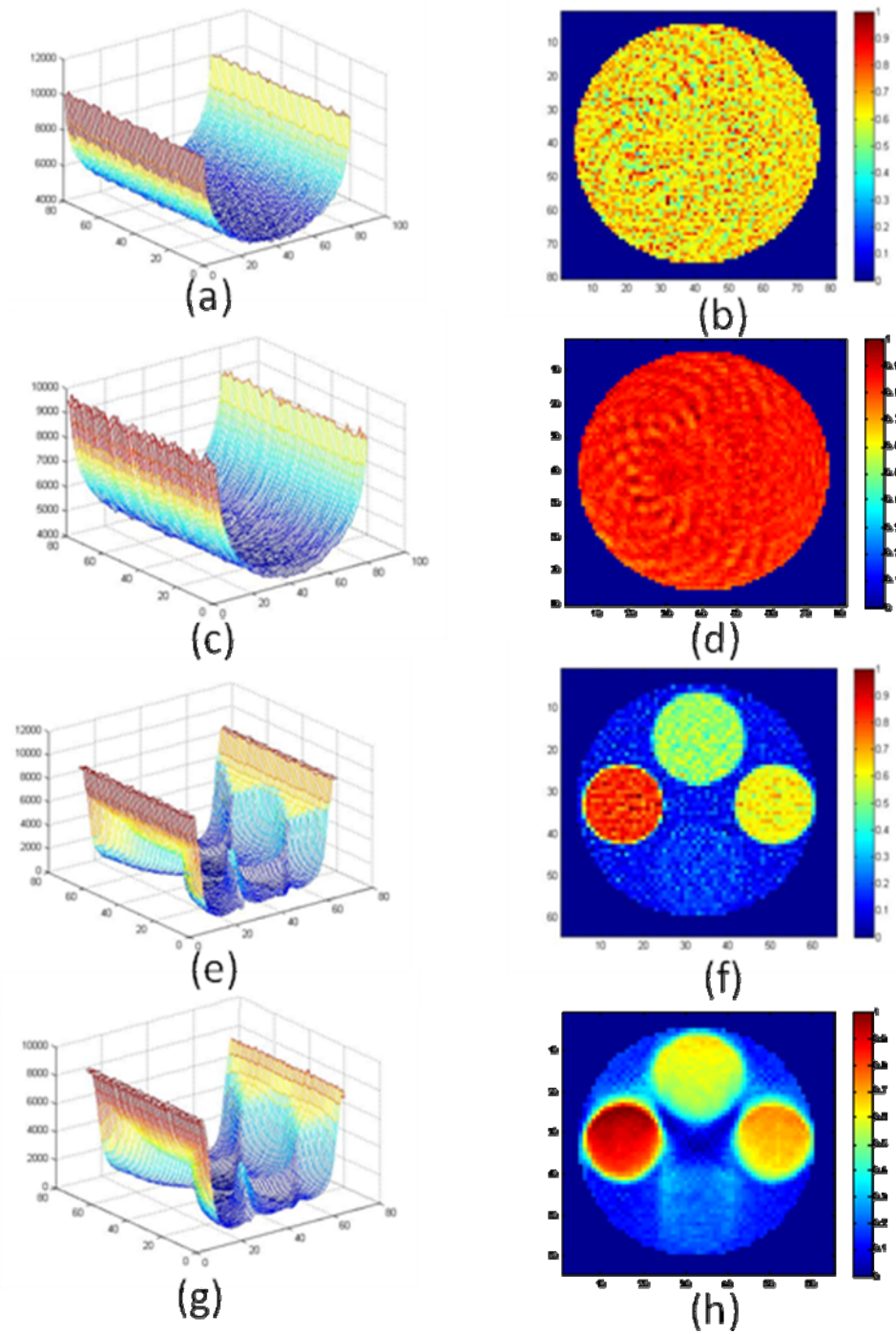
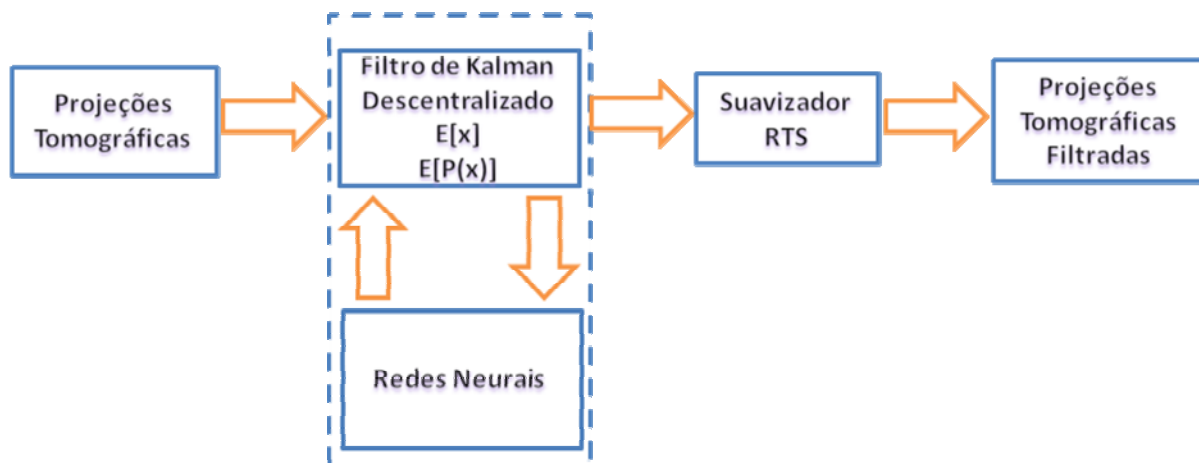


Figura 59. Conjunto de projeções e suas respectivas reconstruções, obtidos com o filtro de Kalman descentralizado com transformada de Anscombe e ruído do processo próximo a zero.

A Figura 60 mostra um diagrama de bloco com a implementação do filtro.



**Figura 60 – Filtro de Kalman descentralizado dual estimation com redes neurais.**

No quadro abaixo é apresentado o pseudocódigo da implementação do filtro em questão.

```

Início
  Lê projeções
  Define Projeção(1) % valor esperado da primeira projeção
  Define Covariância(1) % valor esperado da Covariância da primeira projeção
  Função Kalman % Algoritmo que possui uma função interna: que realiza os cálculos com os
  %pesos da rede neural e um valor de entrada.
    k=número de projeções
    Valor para Q % Variância do ruído do processo.
    Valor para R % Variância do ruído do sistema.
    Para de i=2 até k
      Início
        [Projeção filtrada(i),Covariância(i)]=Kalman(Pesos estimados, Projeção(i),Projeção(i-1),
        Covariância(i-1),Q,R);
        [Pesos estimados]=Correção(Projeção filtrada(i),Pesos);
      Fim
    Para de i=k-1 até k
      Início
        [Projeção suavizada(i),Covariância da suavização(i)]=Filtro de Kalman(Projeção filtrada(i),
        Projeção filtrada(i-1),Covariância(i-1));
      Fim
    Reconstrução das projeções;
  Fim
  
```



A rede neural utilizada nesse filtro apresenta a mesma configuração da utilizada anteriormente no filtro de Kalman estendido, ou seja, uma rede neural *perceptron* multicamadas com 2 neurônios na camada oculta e 1 neurônio na camada de saída. A Figura 61 apresenta o resultado da aplicação do filtro de Kalman descentralizado com redes neurais para projeções dos *phantoms* homogêneo e heterogêneo, respectivamente.

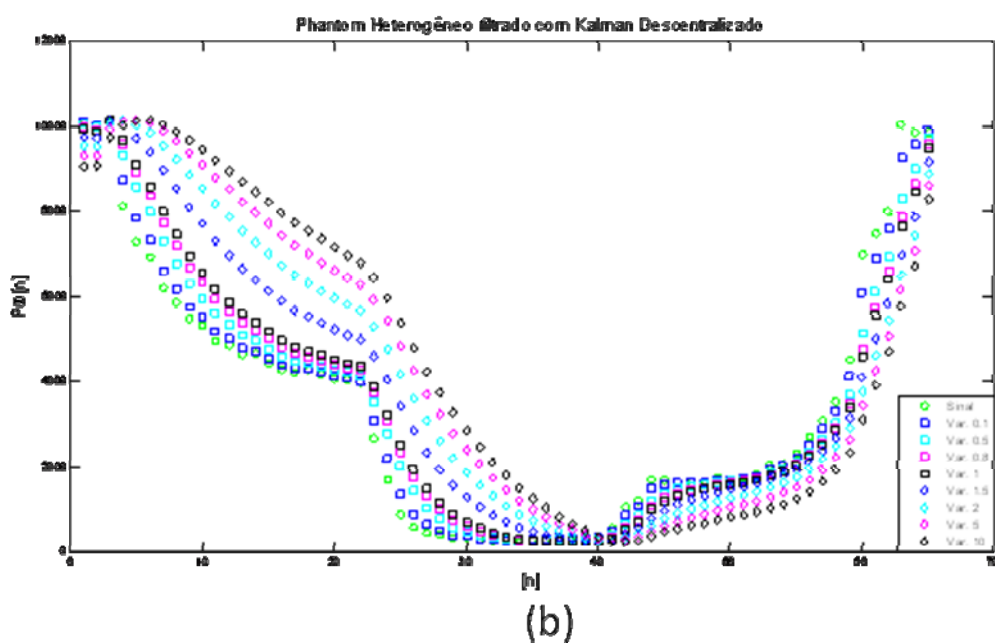
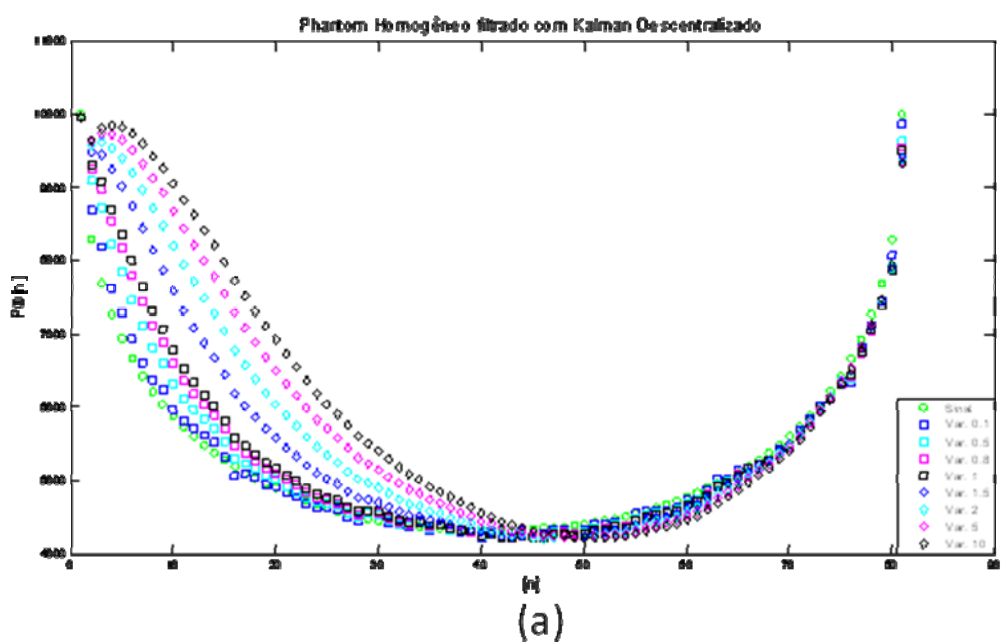


Figura 61. Resultados obtidos com o filtro de Kalman descentralizado com redes neurais: (a) *phantom* homogêneo e (b) *phantom* heterogêneo.

O filtro descentralizado com estimação dupla provou ser estável com o *phantom* homogêneo com valores pouco diferenciados. O mesmo resultado não ocorreu com o *phantom* heterogêneo em que é possível reparar a distorção e perda de detalhamento, devido a simplicidade da rede neural aplicada. Os resultados obtidos com a aplicação do filtro para os *phantoms* podem ser vistos na Figura 62. A Tabela 10 apresenta os valores de ISNR dos resultados de testes realizados.

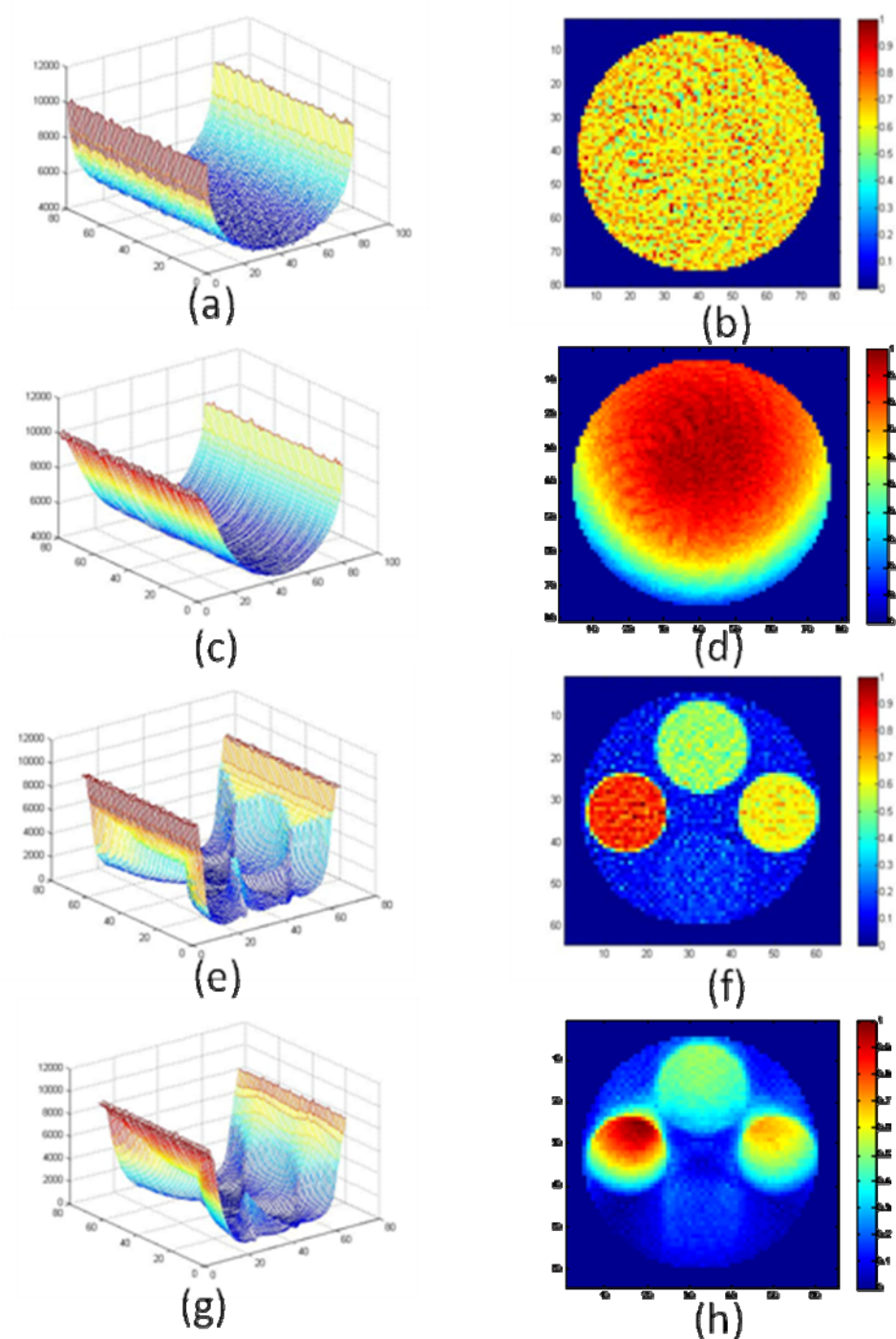


Figura 62. Conjunto de projeções e suas respectivas reconstruções, obtidos com o filtro de Kalman descentralizado e redes neurais para os *phantoms* (a) (b) homogêneo e (c)(d) heterogêneo.

**Tabela 10 - Resultados obtidos com o filtro de Kalman descentralizado com redes neurais para *phantoms* para calibração.**

Variância do ruído do processo para Q=1	Phantom Homogêneo	Phantom Heterogêneo				
	ISNR Máximo (em dB)	ISNR Máximo (em dB)				
R=0,1	4.75	2,31	1,91	2,35	1,24	1,67
R=0,5	6.63	5,11	4,31	5,34	2,81	3,86
R=0,8	6.77	6,53	5,64	6,86	3,85	5,03
R=1	6.73	7,37	6,31	7,74	4,34	5,68
R=5	7.62	10,86	8,69	11,13	2,30	8,38
R=10	7.91	13,92	10,88	13,76	-0,73	12,52
R=15	7.65	16,73	12,66	15,84	-1,85	16,14
R=20	7.05	19,01	14,13	17,38	-2,47	14,95

Nota-se uma melhoria superior nos valores obtidos pelo filtro devido à eficiência do algoritmo descentralizado. Os valores mais altos representam uma queda maior do valor da variância nas regiões de interesse, que demonstram uma melhor filtragem. Os testes com o filtro descentralizado com redes neurais proporcionaram um resultado melhor do que o filtro estendido, apesar da distribuição dos valores das ROIs serem diferenciadas. Isto se deve à demora da convergência da rede ao sinal. Para a melhoria dos resultados, foram realizados testes aumentando o número de camadas e de neurônios nestas camadas com a configuração 1-2-2-1.

A Tabela 11 apresenta resultados de testes realizados para verificar a relação entre ISNR e a qualidade do sinal e da imagem reconstruída. A Figura 63 apresenta o resultado da aplicação do filtro de Kalman descentralizado com redes neurais para projeções dos *phantoms* homogêneo e heterogêneo com uma rede de 3 camadas, respectivamente. Os resultados obtidos com a aplicação do filtro para os *phantoms* podem ser vistos na Figura 64.

**Tabela 11 - Resultados obtidos com o filtro de Kalman descentralizado com redes neurais para *phantoms* para calibração.**

Variância do ruído do processo para Q=1	Phantom Homogêneo	Phantom Heterogêneo				
	ISNR Máximo (em dB)	ISNR Máximo (em dB)				
R=3	6,29	2,13	2,26	2,16	-0,43	1,54

É possível perceber a filtragem do sinal sem distorção ou excessos de suavização quando se tem a mesma variância para o ruído. O filtro converge de forma mais rápida devido ao mapeamento preciso dos pontos sigmas. Para obter o mesmo resultado com os outros filtros é necessário fazer uma cópia dos sinais várias vezes para que os filtros converjam numa resposta ótima.

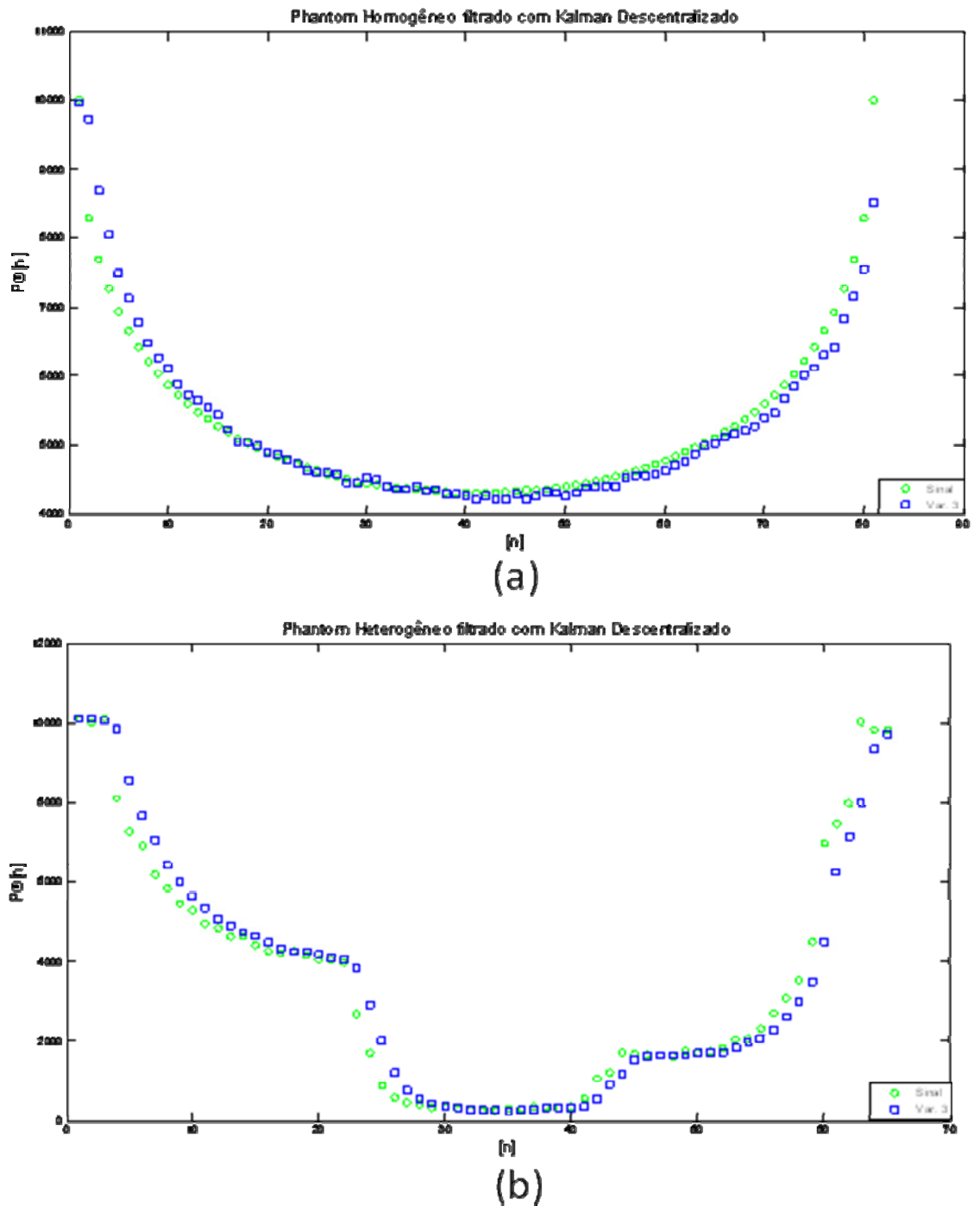


Figura 63 - Resultados obtidos com o filtro de Kalman descentralizado com redes neurais com duas camadas ocultas: (a) *phantom* homogêneo e (b) *phantom* heterogêneo.

. Os resultados obtidos pelo filtro em questão apresentam um ISNR mais baixo do que os resultados dos filtros anteriores devido à ausência suavização excessiva e maior detalhamento. Mas percebe-se ainda uma alta nos valores com exceção ao ROI do centro da imagem reconstruída do *phantom* heterogêneo, que teve valores adicionados (artefatos) devido ao algoritmo de reconstrução.

Os resultados foram os esperados e apresentam de certa forma, os mesmos resultados visuais da aplicação dos filtros descentralizados básicos e com transformada de Anscombe, provando que as características do ruído *Poisson*, que são definidas por uma função de comportamento não linear, podem ser mapeadas por uma rede neural, onde toda a complexidade de se entender de forma aprofundada as equações de processo podem ser substituídas por um sistema iterativo inteligente capaz de atender até novas características dos sensores com o tempo (diferença de temperatura, envelhecimento de equipamentos e novas estruturas mecânicas).

O mapeamento com a rede neural com duas camadas ocultas permitiu que a perda de detalhes e o excesso de suavização fossem contidos, já que o excesso de suavização decorre de problemas com o mapeamento dos estados de transição.

Apesar do uso de uma função de translação ser algo mais simples e talvez indicado para uma gama de problemas devido ao tempo de processamento e memória necessária, o filtro de Kalman com redes neurais substitui os antigos algoritmos com estas técnicas devido ao uso mínimo de camadas e a quantidade de neurônios nestas camadas, além do uso de estimação dual ou conjunta, substituir um treinamento intensivo prévio que nem sempre garante a adaptabilidade e a convergência para um melhor resultado, algo que prova que uma função de translação aplicada no modelo de cadeias ocultas de Markov não serem tão eficientes quanto o uso deste modelo para identificar os pesos.

O uso do modelo de Markov para estimar pesos de uma rede neural surgiu diretamente da necessidade de obter um estimador para o filtro de Kalman que pudesse mapear as funções não lineares de forma eficiente e melhor do que o processo de linearização (aplicado em rotações) e fusão de outras funções, sem haver a necessidade de criar vários estados para mapear um comportamento único.

Para validar a filtragem aplicada, um *phantom* real obtido de um solo arenoso foi submetido a este filtro.

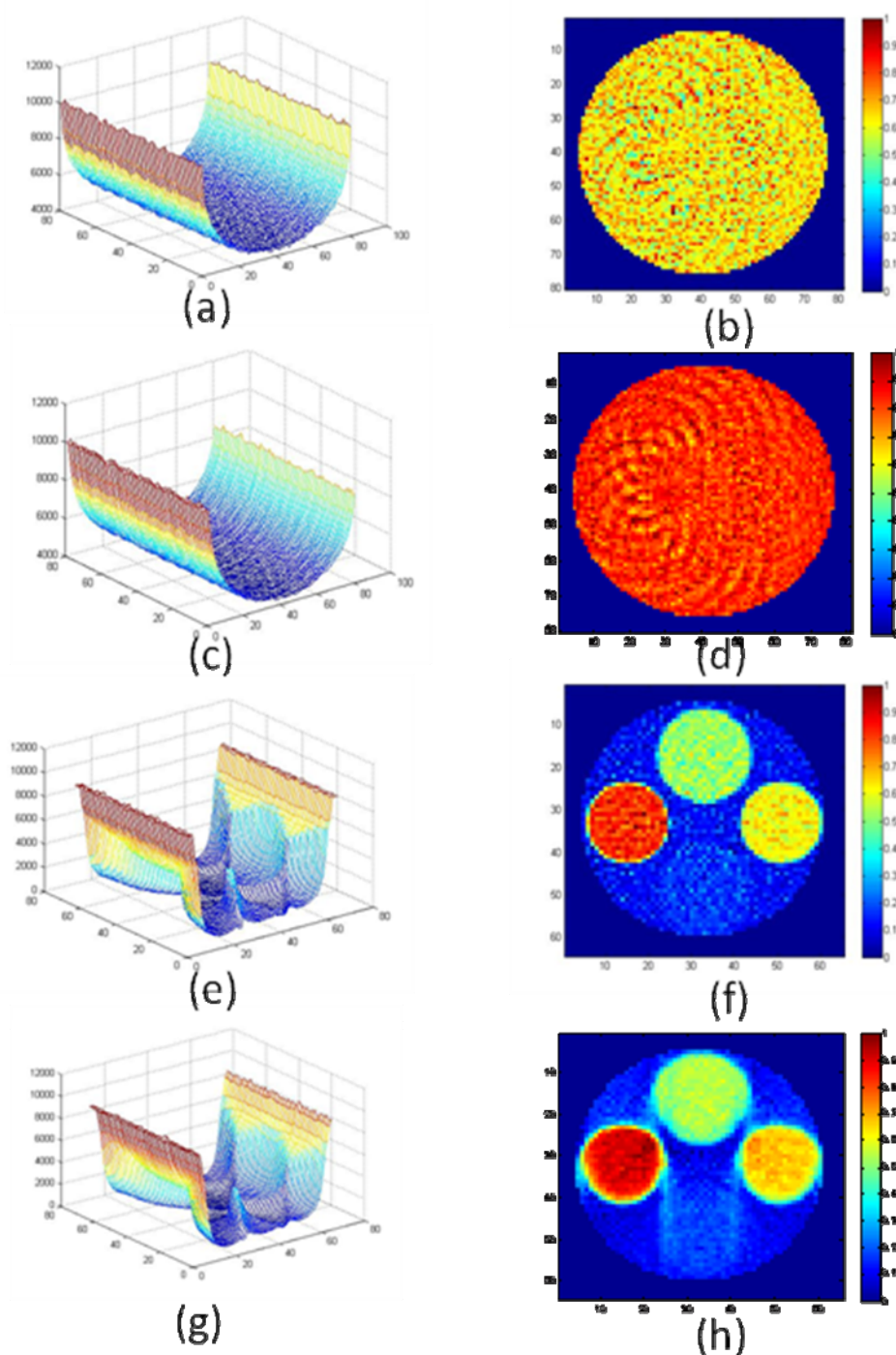


Figura 64. Conjunto de projeções e suas respectivas reconstruções, obtidos com o filtro de Kalman descentralizado e redes neurais para os *phantoms* (a) (b) homogêneo e (c)(d) heterogêneo.

#### 4.8 Aplicação do filtro de Kalman descentralizado com redes neurais em uma amostra de solo

As filtrações convencionais não adaptativas provocam perda de detalhes e deformação dos sinais e imagens geradas. O uso do filtro de Kalman apresenta uma solução

para prever os próximos valores e estabelecer correções internas adaptativas, bem como possibilita separar o que seria indesejado no sinal. A Figura 65 apresenta o resultado da aplicação do filtro de Kalman descentralizado com redes neurais para projeções de uma amostra de solo com dois tipos de correção da rede neural.

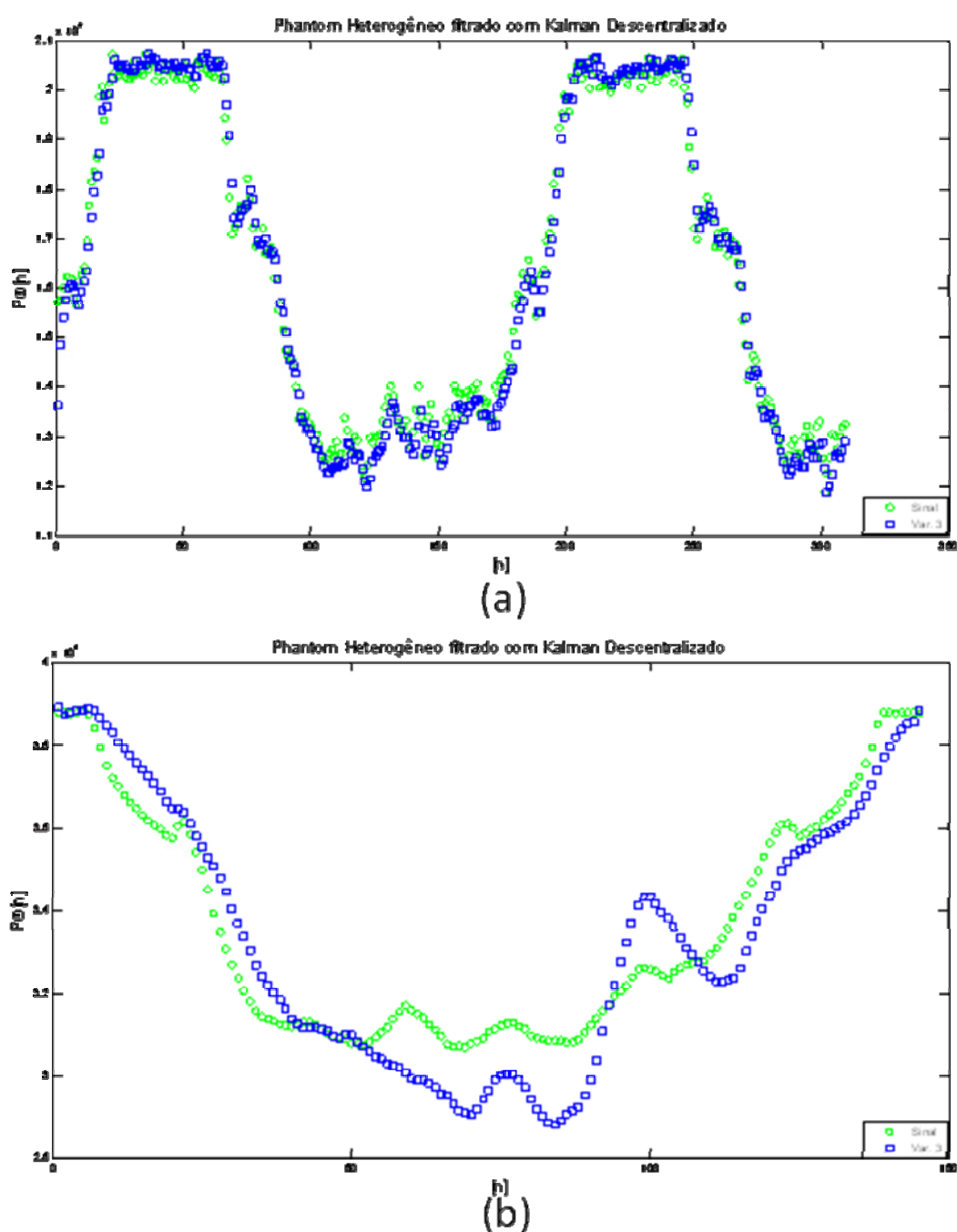


Figura 65. Resultado obtido com um *phantom* real de solo arenoso com o filtro de Kalman descentralizado com redes neurais com três camadas (a) correção da rede neural feita através dos valores observados e (b) correção feita pelos valores estimados.

Os resultados obtidos com a aplicação do filtro para uma amostra de solo podem ser vistos na Figura 66.

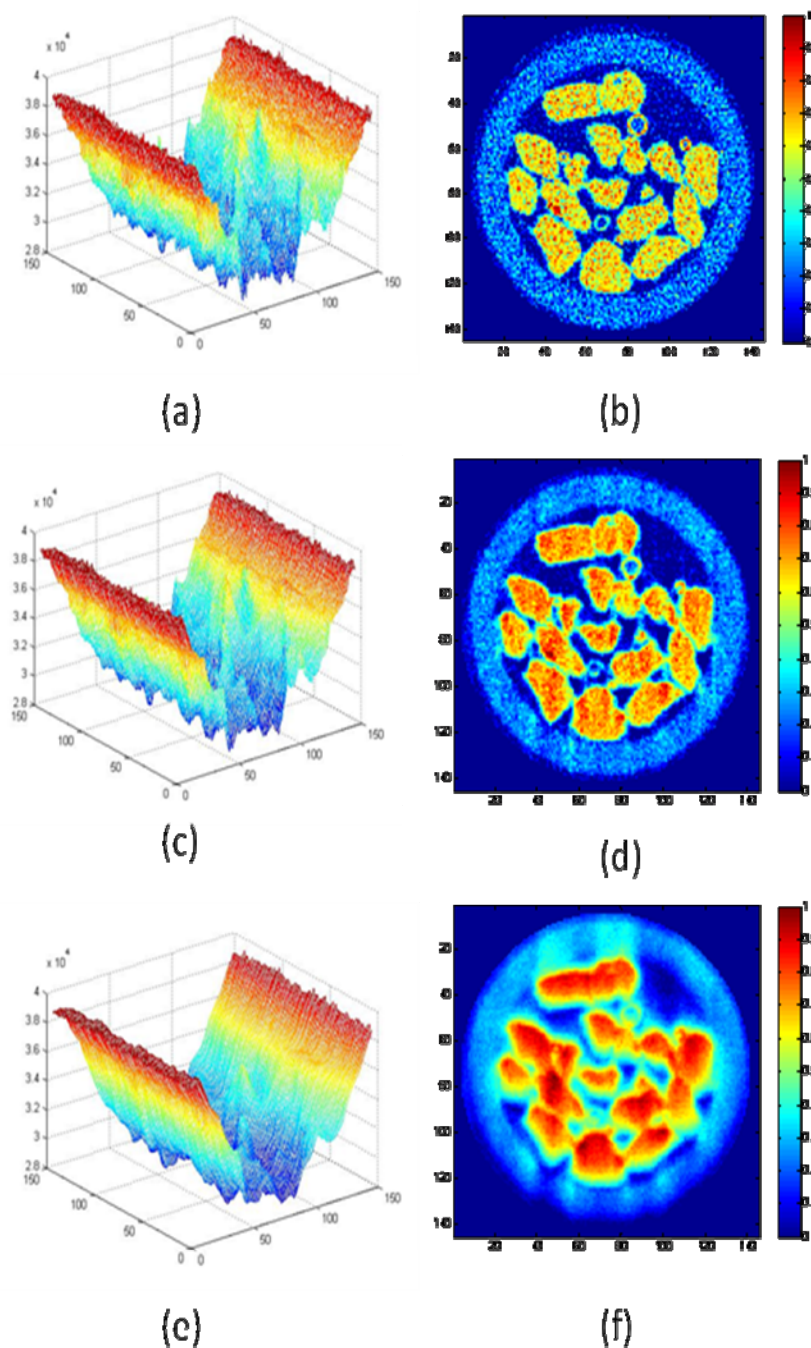


Figura 66. Amostra de solo: (a) (b) sem filtragem e com redes neurais: (c)(d) correções da rede baseadas nas observações e (e)(f) correções da rede baseada nos valores estimados.

A correção por estado estimado retorna os mesmos resultados do filtro descentralizado com duas camadas. Apesar da fidelidade dos valores do filtro de Kalman com correção da rede pelos valores observados (entre observação dada e observação estimada), a filtragem com correção da rede pelos valores medidos (entre valores estimado



*a priori* e valor estimado corrigido), apresenta um melhor ISNR (9,52 dB) quando comparado ao uso de um filtro de Kalman com as correções baseadas nos estados observados (1,92 dB).

## 4.9 Conclusões

Quanto ao filtro de Kalman discreto, o mesmo foi considerado um estimador ótimo para problemas lineares e suas extensões podem ser utilizadas em tratamentos não lineares. Os resultados apresentaram uma deformação do sinal e um baixo desempenho na filtragem em alguns casos devido ao fato da equação do processo ser direcionada por uma função não linear onde é tratado o ruído *Poisson*, que possui um comportamento dependente do sinal, algo que não acontece com o ruído branco.

Quanto ao filtro de Kalman estendido, o mesmo permite um mapeamento melhor das médias dos valores do sistema através do uso de função não lineares. A utilização das matrizes para a propagação das variâncias ainda se faz necessária, mas a linearização das funções permite ainda uma solução para sistemas não lineares. O uso da transformada de Anscombe permitiu ao filtro o mapeamento da equação do processo, que é afetada não por um ruído branco, mas sim por um ruído *Poisson*.

A aplicação de redes neurais com o filtro de Kalman estendido permitiu ao filtro alcançar funções não lineares superiores. Como o filtro de Kalman pode ser utilizado para treinar redes neurais, o uso das extensões com estimação conjunta ou dual permitiu que o filtro seja completo e independente no mapeamento de qualquer função não linear.

Quanto ao filtro de Kalman descentralizado, o mesmo permitiu o mapeamento de funções de ordem elevada pelo uso de pontos sigmas para mapear a média e a variância sem recorrer ao uso da linearização por matrizes Jacobianas e Hessianas. O uso de redes neurais com esse tipo de filtro permitiu o mapeamento de qualquer função, com estimação precisa dos resultados. Adicionalmente, verificando os resultados obtidos com o filtro de Kalman descentralizado com redes neurais no *phantom* real, foi possível observar a eficiência do filtro em se adaptar a funções caóticas, tais como as heterogeneidades geralmente presentes em amostras reais.

Para o ensaio com amostra do solo foi possível perceber a diminuição da variância nos valores da amostra ensaiada. São duas operações com o filtro de Kalman descentralizado que devem ser consideradas, visto que houve um excesso de filtragem. Logo, uma solução

viável vem a ser o aumento do sinal através da duplicação de trechos, mesmo não sendo garantida a convergência da rede.

Por ser uma pré-filtragem, manter detalhes na imagem vem a ser o mais importante.

## Referências Bibliográficas

---

- ANSCOMBE, F. J. *The Transformation of Poisson, Binomial, Negative-Binomial Data*. Vol. 15, in *Biometrika*. 1948.
- BRIGHAM, E. O. *The Fast Fourier Transform and Its Applications*. New Jersey: Prentice Hall, 1988.
- BUCHSBAUM, R. S. "Linear and nonlinear filtering (Invited Paper)." *Proceedings of the IEEE*, 1970: 10.
- CORMACK, A. M. "Representation of a function by its line integrals, with some radiological applications." *Journal of Applied Physics*, 1963.
- CRESTANA, S. *A Tomografia Computadorizada como um novo método para estudos da física da água no solo*. São Carlos: Universidade de São Paulo, 1985.
- CRUVINEL, P. E.; CESAREO, R.; CRESTANA, S. e MASCARENHAS, S. "X and gamma-ray computerized minitomograph scanner for soil science." *IEEE Transactions on Instrumentation and Measurement*, 1990, 39 ed.: 745-750.
- CRUVINEL, P. E.; CRESTANA, S.; ARTAXO, P. E.; MARTINS, J. V. e ARMELIN, M.J., "A study of the spatial variability of Cr in agricultural field using both particle induced X-ray emission (PIXE) and instrumental neutron activation analysis (INAA) technique." *Nuclear Instruments and Methods in Physics Research*, 1996, 109-10 ed.
- CRUVINEL, P.E. "Minitomógrafo de raios X e gama computadorizado para aplicações multidisciplinares." *Tese de Doutorado*, 1987.
- DEREMACK, E. e CROWE, D.G. , *Optical Radiation Detectors*. John Wiley & Sons, Inc., 1984.
- DUARTE, M. C. G. *Análise de Redes Neurais para Aplicação em Visão de Manipulador*. Vol. Tese de Doutorado. São Carlos: UFSCar, 1996.
- DUERINCKX, A. J., e MACOCSKI, A., "Polychromatic Streak Artifacts in Computed Tomography Images." *J. Comput. Assist. Tomogr.*, 1978.
- FARLEY, B. G. e CLARK, W. A., "Proceedings of the Western." *Joint Computer Conference*, 1954.
- FAUSETT, L. *Fundamentals of Neural Network: Architectures, Algorithms and Applications*. Prentice Hall International Editions, 1994.
- GONZALES, R.C. *Digital Image Processing*. Tennesse: Addison Wesley, 1987.

- GRANATO, L. F. *Algoritmo adaptativo para a melhoria em imagens tomográficas obtidas em múltiplas energias*. São Carlos: UFSCAR, 1998.
- HAGAN, M. T., e MENHAJ, M., "Training feedforward networks with the Marquardt algorithm." *IEEE Transactions on Neural Networks*, 1994, 6 ed.
- HAINSWORTH, J. M. e AYLMORE, L. A. G., "The use of the computed-assisted tomography to determine spatial distribution of soil water conten." *Aust. Journal Soil Res.*, 1983.
- HEBB, D. O. *The Organization of Behavior*. New York: Wiley, 1949.
- HENDER, W. R. "The physical principals of computed tomography." 1983.
- HINTON, G.E. e SEJNOWSKI, J. T., "Learning and Relearning in Boltzmann Machines." In *Parallel Distributed Processing*, by D. E. RUMELHART and J. L. MCCLELLAND. Cambridge: MIT Press, 1986.
- HOPFIELD, J. J. "Neural Network and Physical System with Emergent Colletive Computational Abilities." *Proc. National Academy of Science Biophysics*, 1982.
- HOUSNFIELD, G. N. "Computerized transverse axial scanning tomography." 1973.
- IBBOTT, G. S. "Radiation therapy treatmen planning and the distortion of CT images." *Med. Phys*, 1980.
- INOUYE, T. "Square Root Transform for the Analysis of Quantum Flutuaton in Spectrum Data." *Nuclear Instrument and Methods*, 1971.
- JAIN, A. K. e MOHIUDDIN, K. M. "Artificial Neural Networks: A Tutorial." *IEEE Computer*, 1996: 31-44.
- JOSEPH, P. M. e SPITAL, R. D., "A method for correction bone-induced artifacts of CT scanners." 1978: 100–108.
- JULIER, S. J. e UHLMANN, J. K., "A new extension of Kalman filter to nonlinear systems." *Symp. Aerospace/Defense Sensing, Simul. and Controls*, 1997.
- KALMAN, R. E. "A new approach to Linear Filtering and Prediction Problems." *Transaction of the ASME - Journal of basic Engineering*, 1960.
- LE CUN, Y. "Learning Process in Asymmetric Threshold Network." In *Disordered System and Biological Organization*, by E. BIENENSTOCK, F. FOGEMAN-SOULI and G. WEISBUCH. Berlin: NATO ASI Series, 1986.
- MAREN, A.; HARSTON, C.T. e PAP, R. M., *Handbook of Nueral Computing Applications*. San Diego: Academic Press, Inc., 1990.

- MASCARENHAS, N. D. A.; SANTOS, C. A. N. e CRUVINEL, P. E., "Transmission Tomography Under Poisson Noise Using the Anscombe Transformation and Wiener Filtering of Projections." *Nuclear Instruments and Methods in Physics Research Section A*, 1999.
- MCCLELLAN, J. L e RUMELHART, D. E.. "Explorations in Parallel Distributed Processing." 1988.
- MCCULLOCH, W. S. e PITTS, W., "A logical calculus of the ideas immanent in Nervous Activity." *Bulletin of mathematical biophysics*, 1947.
- . "How we know universals: the perception of auditory and visual forms." *Bulletin of mathematical biophysics*, 1943.
- MINATEL, E. R. *Desenvolvimento de Algoritmo para Reconstrução e Visualização Tridimensional de Imagens Tomográficas com uso de Técnicas Freqüenciais e Wavelets*. São Carlos: UFScar, 1997.
- MINSKY, M. L. e PAPERT, S., *Perceptrons*. Cambridge: MIT Press, 1968.
- OPPENHEIM, A. V. e SCHAFER, R. W., *Digital Signal Processing*. Prentice-Hall, 1975.
- PEREIRA, G. A. S. "Filtro de Kalman: Teoria e Aplicações." *Centro de Pesquisa e Desenvolvimento em Engenharia Elétrica.*, 2000.
- PETROVIC, A. M.; SIEBERT, J. E. e RIEKE, P. E. "Soil bulk analysis in three-dimensions by computed tomographic scanning." *Soil Sci. Soc. Am. J.*, 1982.
- PIZZOLATO, E. B. *Implementação de Redes Neurais no CPER*. São Carlos : UFSCar, 1992.
- RADON, J. "On the determination of functions from their integrals along certain manifolds." *Ber Saechs Akad wiss. Leipzig Math. Phys*, 1917.
- ROCHESTER, N. J. H.; HAIBT, L. H.; DUDA, W. L. e HOLLAND, J. H., "Tests on a Cell Assembly Theory of the Action of the Brain, Using a Large." *IRE Transactions on Information Theory*, 1956.
- ROSENBLATT, F. "The Perceptron: a Probabilistic Model for Information Storage and Organization in the Brain." *Psychological Review*, 1958, 65 ed.
- ROSENBLATT, F. *Principles of Neurodynamics: Perceptrons and Theory of Brain Mechanisms*. Washington: Spartan Books, 1962.
- RUMELHART, D. E.; HINTON, G. E. e WILLIAMS, R. J., "Learning Internal Representations by Error Propagation." In *Parallel Distributed Processing*, by D. E. RUMELHART and J. L. MCCLELLAN. Cambridge: MIT Press, 1986.

- SEJNOWSKI, T. J., e ROSENBERG, C. R.. "NETtalk: A Parallel NetWork that Learns to Read Aloud." *Tech. Re. JHU*, 1986, ECC-86/01 ed.
- TAKAHASHI, H.; SHIMODA, K. e TOWNES, C.H., "Fluctuation in amplification of quanta with application to master amplifiers." *J. Phys. Soc. Japan*, 1957.
- TREALEVEN, P.; PACHECO, M. e VELLASCO, M., "VLSI Architetures for Neural Networks." *IEEE MICRO*, 1988, dezembro ed.
- VAN DER MERVE, R. e WAN, E. A.. "The square-root unscented Kalman Filter for state and parameter-estimation." *Acoustics, Speech, and Signal Processing, 2001. Proceedings. (ICASSP '01). 2001 IEEE International Conference on*, 2001: 3461-3464.
- VENTURINI, Y. R. *Análise quantitativa da qualidade de imagens digitais com o uso de espectro de Wiener*. São Carlos: UFSCar, 1995.
- VON NEUMANN, J. *The Computer and the Brain*. New Have: Tale University Press, 1958.
- WAN, E. A. e MERWE, R.. "The Unscented Kalman Filter for Signal Processing." *Proceedings of Symposium 2000 on Adaptive Systems for Signal Processing, Communication and Control*, 2000, outubro ed.
- WELCH, G., e BISHOP, G.. *An Introduction to the Kalman Filter*. Chapel Hill: Departament of Computer Science, University of North Carolina, 2004.
- WERBOS, P. "Maximizing Long-term Gas Industry Profits in Two Minutes in Lotus Using Neural Network Methods." *IEEE Trans on System, Man and Cybernetics*, 1989.
- WIDROW, B., e HOFF, M. E. Jr.. "Adaptative Swicthing Circuits." *IRE WESCON Convection Record*, 1960.
- ZIEL, A. D. *Noise in measurements*. John Wiley & Sons, inc, 1976.

## Apêndice A

---

### Probabilidade e Variáveis Aleatórias

Para processamento de sinais e imagens, o entendimento sobre estes tópicos é importante, pois são utilizados desde a obtenção dos sinais e imagens até reconhecimento de padrões.

#### a. Probabilidade

Baseia-se na noção de ocorrência aleatória ou a probabilidade de algum evento em um espaço de amostragem que poderá ocorrer. A probabilidade de, por exemplo, lançar uma moeda e dar cara (ou coroa), que é a saída de um evento discreto, favorecerá um evento particular que é definido como

$$p(A) = \frac{\text{Possíveis saídas favorecendo o evento A}}{\text{Número total de saídas possíveis}}$$

A probabilidade de uma saída favorecer tanto A ou B é dada por

$$p(A \cup B) = p(A) + p(B)$$

Se a probabilidade de duas saídas forem independentes (uma não afetar a outra) então a probabilidade de ambas ocorrerem é o produto de suas probabilidades individuais, como se fossem jogadas duas moedas e saber a probabilidade de uma saída for A e outra for B em cada moeda, é dada por

$$p(A \cap B) = p(A) p(B)$$

Por exemplo, se a probabilidade de ter cara for 50%, então a probabilidade de ter cara em ambas moedas lançadas ao mesmo tempo é 25%.

E a probabilidade de saída A dada uma ocorrência de saída B é chamada de probabilidade condicional de A dada por B, e é definida como

$$p(A | B) = \frac{p(A \cap B)}{p(B)}$$

#### b. Variáveis aleatórias

Como oposto para eventos discretos, na captura de imagens e sinais interessa-se associar a aleatoriedade a uma voltagem elétrica contínua. Em cada caso o item de interesse é uma variável aleatória contínua. A variável aleatória é essencialmente uma função que mapeia todos os pontos no espaço de amostragem para números reais. Por

exemplo, a variável aleatória contínua  $X(t)$  pode mapear posição em determinado tempo. A qualquer ponto no tempo,  $X(t)$  pode dizer a posição esperada.

No caso de variáveis aleatórias contínuas, a probabilidade de qualquer evento discreto simples A é no de fato 0. isto é,  $p(A) = 0$ . Somente pode-se validar a probabilidade de eventos com algum intervalo. Uma função comum representando a probabilidade de variáveis aleatórias é definida como a *função de distribuição* ou *função acumulada de probabilidade* (WELCH,2004)]:

$$f_x(x) = p(-\infty, x].$$

Esta função mostra que a propriedade acumulada da variável aleatória continua  $X$  para todos os (incontáveis) eventos incluindo  $x$ . Propriedades importantes para a função de densidade acumulada são:

1.  $F_x(x) \rightarrow 0$  se  $x \rightarrow -\infty$
2.  $F_x(x) \rightarrow 1$  se  $x \rightarrow +\infty$
3.  $F_x(x)$  é a função não decrementada de  $x$

Contudo, a equação mais usada é a sua derivada, conhecida como a função de densidade de probabilidade:

$$f_x(x) = \frac{df_x(x)}{dx}$$

Seguindo as propriedades da função acumulada de probabilidade, a função de densidade também tem as seguintes propriedades:

1.  $f_x(x)$  é uma função não negativa
2.  $\int_{-\infty}^{\infty} f_x(x) dx = 1$ .

Finalmente note que a probabilidade sobre qualquer intervalo  $[a,b]$  é definido como

$$p[a,b] = \int_a^b f_x(x) dx .$$

Então se assume probabilidade de eventos discretos, como na equação  $p(A \cup B) = p(A) + p(B)$ , para variáveis aleatórias contínuas que integra a função de densidade probabilidade sobre o intervalo de interesse.

### c. Média e Variância

Baseando-se na noção da média de uma seqüência de números. Para algum N exemplos de uma variável aleatória discreta de X, a média é dada por



$$\bar{x} = \frac{x_1 + x_2 + x_3 + \dots + x_N}{N}.$$

Em gravações ou rastreamento (tracking) trabalha-se com sinais contínuos (com um incontável espaço de amostragem) e é útil pensar em termos de números infinitos de tentativas, e a correspondência da saída que se pode esperar para ver a possibilidade de mostrar a variável infinitamente, cada tempo vendo uma das  $n$  possíveis saídas  $x_1 \dots x_n$ . Neste caso, o valor esperado da variável aleatória discreta pode ser aproximado pela média de eventos prováveis pesados:

$$\bar{x} = \frac{(p_1 N)x_1 + (p_2 N)x_2 + (p_3 N)x_3 + \dots + (p_n N)x_n}{N}.$$

Esperamos ver  $(p_1 N)$  ocorrências do evento  $x_1$ , etc. Esta noção de tentativas infinitas (amostras) levam para a definição convencional de valor esperado para variáveis aleatórias discretas

$$\text{Valor esperado de } X = E(X) = \sum_{i=1}^n p_i x_i$$

para  $n$  saídas possíveis  $x_1 \dots x_n$  e probabilidades correspondentes  $p_1 \dots p_n$ . Similarmente para variáveis aleatórias contínuas o valor esperado é definido como

$$\text{Valor esperado de } X = E(X) = \int_{-\infty}^{\infty} x f_X(x) dx.$$

Finalmente, note que as equações  $E(X) = \sum_{i=1}^n p_i x_i$  e  $E(X) = \int_{-\infty}^{\infty} x f_X(x) dx$  podem ser aplicadas para funções de variáveis aleatórias  $X$  como a seguir:

$$E(g(X)) = \sum_{i=1}^n p_i g(x_i)$$

$$E(g(X)) = \int_{-\infty}^{\infty} g(x) f_X(x) dx.$$

O valor esperado da variável aleatória é também conhecido como primeiro momento estatístico. Pode-se aplicar a noção das equações  $E(g(X)) = \sum_{i=1}^n p_i g(x_i)$  e  $E(g(X)) = \int_{-\infty}^{\infty} g(x) f_X(x) dx$ , deixando  $g(X) = X^k$ , para obter o  $k$ -ésimo momento estatístico momento. O  $k$ -ésimo momento é uma variável aleatória contínua  $X$  e é dada por

$$E(X^k) = \int_{-\infty}^{\infty} x^k f_X(x) dx .$$

De particular interesse em geral, e para este estudo em particular, é o segundo momento da variável aleatória. O segundo momento é dado por

$$E(X^2) = \int_{-\infty}^{\infty} x^2 f_X(x) dx .$$

Quando se deixa  $g(X) = X - E(X)$  e aplica a equação acima, encontra-se a variância do sinal sobre a média. Em outras palavras,

$$\begin{aligned} \text{Variância } X &= E[(X - E(X))^2] \\ &= E(X^2) - E(X)^2 . \end{aligned}$$

Variância é uma propriedade estatística muito útil para sinais aleatórios, porque quando se conhece a variância de um sinal que era suposto ser constante em volta de algum valor – a média, a magnitude da variância nos dá um senso de quanto ruído está no sinal.

A raiz quadrada da variância, conhecida como o desvio padrão, é também uma unidade estatística de medida muito útil porque enquanto for sempre positivo, tem (como oposto da variância) as mesmas unidades do sinal original. O desvio padrão é dado por

$$\text{Desvio padrão de } X = \sigma_x = \sqrt{\text{Variância de } x}$$

#### d. Distribuição Normal ou Gaussiana

Uma probabilidade especial conhecida como a distribuição Normal ou Gaussiana (Figura 67) tem, historicamente, sido popular em modelamento de sistemas aleatórios de razões variadas. Alguns processos aleatórios que ocorrem na natureza parecem ser normalmente distribuídos, ou aproximáveis. De fato, sobre algumas condições moderadas, pode ser provado que um soma de variáveis aleatórias com qualquer distribuição tende a ser uma distribuição normal. O teorema que convencionou estados desta propriedade é chamado de teorema do limite central. Finalmente, a distribuição normal tem algumas boas propriedades que o faz matematicamente tratável e atraente.

Dado um processo aleatório  $X \sim N(\mu, \sigma^2)$ , como, por exemplo, um processo aleatório contínuo  $X$  que é distribuído normalmente com média  $\mu$  e variância  $\sigma^2$  (desvio padrão  $\sigma$ ), a densidade de probabilidade da função é dada por

$$f_X(x) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{1}{2} \frac{(x-\mu)^2}{\sigma^2}}$$

para  $-\infty < x < \infty$ .

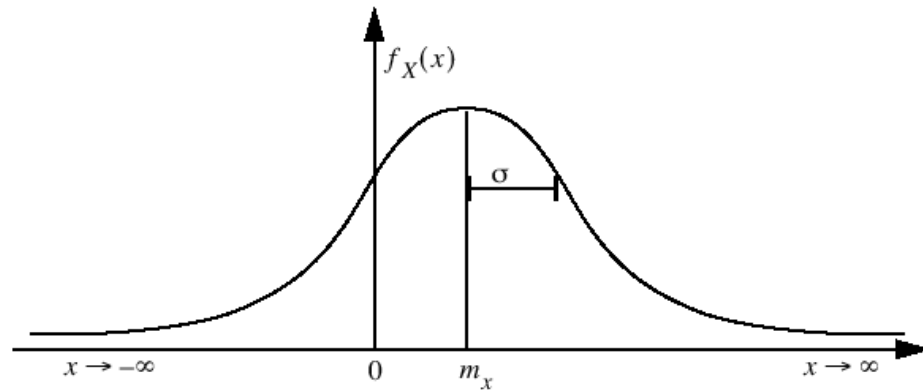


Figura 67. A função de distribuição de probabilidade Normal ou Gaussiana.

Qualquer função linear de um processo aleatório distribuído normalmente (variável) é também um processo aleatório distribuído normalmente. Em particular se  $X \sim N(\mu, \sigma^2)$  e  $Y = aX + b$ , então

$$Y \sim N(a\mu + b, a^2\sigma^2).$$

A densidade da função de probabilidade para Y é dada por

$$f_X(y) = \frac{1}{\sqrt{2\pi a^2 \sigma^2}} e^{-\frac{1}{2} \frac{(y - (a\mu + b))^2}{a^2 \sigma^2}}$$

Finalmente, se  $X_1$  e  $X_2$  são independentes,

$$X_1 \sim N(\mu_1, \sigma_1^2)$$

e

$$X_2 \sim N(\mu_2, \sigma_2^2)$$

então

$$X_1 + X_2 \sim N(\mu_1 + \mu_2, \sigma_1^2 + \sigma_2^2),$$

e a função de densidade vem a ser

$$f_X(x_1 + x_2) = \frac{1}{\sqrt{2\pi(\sigma_1^2 + \sigma_2^2)}} e^{-\frac{1}{2} \frac{(x_1 + x_2 - (\mu_1 + \mu_2))^2}{\sigma_1^2 + \sigma_2^2}}$$

Graficamente, a distribuição normal parece ser uma curva em forma de sino, como mostra a Figura 67.

## Independência contínua e probabilidade condicional

Nota-se que como no caso discreto e as equações

$$p(A|B) = \frac{p(A \cap B)}{p(B)}$$

$$F_x(x) = p(-\infty, x]$$

a probabilidade condicional e independente são definidas por variáveis aleatórias contínuas. Duas variáveis aleatórias contínuas  $X$  e  $Y$  são ditas a serem estatisticamente independentes se sua probabilidade de junção  $f_{XY}(x, y)$  é igual ao produto de suas probabilidades individuais, em outras palavras, elas são consideradas independentes se  $f_{XY}(x, y) = f_X(x)f_Y(y)$ .

### a. Teorema de Bayes

Em adição, o teorema de Bayes segue de  $p(A|B) = \frac{p(A \cap B)}{p(B)}$ , oferecendo um caminho para especificar a densidade de probabilidade de uma variável aleatória  $X$  dada (na presença de) uma variável aleatória  $Y$ . O teorema de Bayes é dado como

$$f_{X|Y}(x) = \frac{f_{Y|X}(y)f_X(x)}{f_Y(y)}$$

### b. Contínuo-Discreto

Dado um processo discreto  $X$  e um contínuo processo  $Y$ , a função massiva de probabilidade discreta para  $X$  condicionada em  $Y = y$  é dada por

$$p_X(x|Y=y) = \frac{f_Y(y|X=x)p_X(x)}{\sum_z f_Y(y|X=z)p_X(z)}$$

Nota-se que esta formula provem de uma probabilidade discreta baseada na densidade condicionante, sem qualquer integração.

### c. Características do Sinal Espacial versus Espectral

Nas seções anteriores viam-se somente características espaciais de sinais aleatórios. Como estado anterior, a magnitude da variância de um sinal pode dar um senso de como o ruído está no sinal. Contudo, uma variância do sinal não diz nada sobre o espaçamento ou a taxa de ruído sobre o tempo. Aqui brevemente discutem-se as características dos espectros temporais e adiante as de um sinal aleatório.

Uma característica útil do tempo relacionado de um sinal aleatório é sua autocorrelação – sua correlação consigo mesma sobre o tempo. Formalmente a autocorrelação de um sinal aleatório  $X(t)$  é definido como

$$R_X(t_1, t_2) = E[X(t_1)X(t_2)]$$

para tempos de amostragem  $t_1$  e  $t_2$ .

Se o processo é estacionário (a densidade é invariante com o tempo) então a equação acima depende somente da diferença  $\tau = t_1 - t_2$ . Neste caso comum a autocorrelação pode ser reescrita como

$$R_X(\tau) = E[X(t)X(t + \tau)].$$

Dois hipóteses para funções de autocorrelação são mostradas abaixo na Figura 68. Note que comparado ao sinal aleatório  $X_2$ , o sinal aleatório  $X_1$  é relativamente menor e mais comprido. Como  $|\tau|$  aumenta (como se movesse fora de  $\tau = 0$  no centro da curva) o sinal de autocorrelação para  $X_2$  cai relativamente rápido. Isto indica que  $X_2$  é menos correlacionado consigo mesmo do que  $X_1$ .

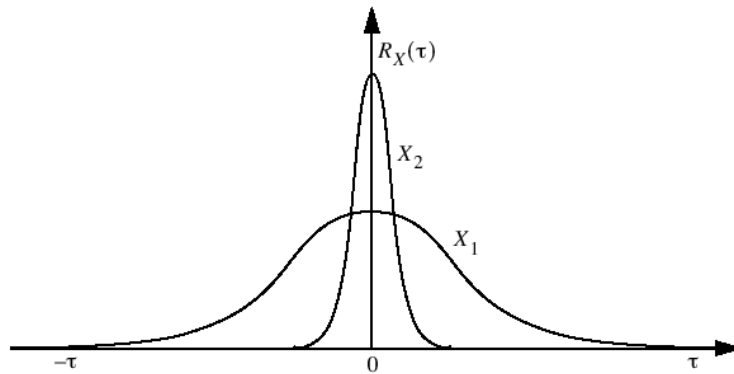


Figura 68. Dois exemplos (hipotéticos) de funções de autocorrelação  $X_1$  e  $X_2$ .

Claramente a autocorrelação é a função do tempo, que significa que é ela tem uma interpretação espectral no domínio da frequência também. De novo com o processo estacionário, existe um importante relacionamento espectro-temporal conhecido como a relação Wiener-Khinchine (WELCH e BISHOP 2004):

$$S_X(j\omega) = \mathfrak{F}[R_X(\tau)] = \int_{-\infty}^{\infty} R_X(\tau) e^{-j\omega\tau} d\tau$$

onde  $\mathfrak{F}[\bullet]$  indica a transformada de Fourier, e  $\omega$  indica o número de  $(2\pi)$  ciclos por segundo. A função  $S_X(j\omega)$  é chamada de densidade de potência espectral do sinal aleatório. Como se pode ver, este relacionamento importante amarra juntas as representações do tempo e frequência dos espectros do mesmo sinal.

#### d. Ruído Branco

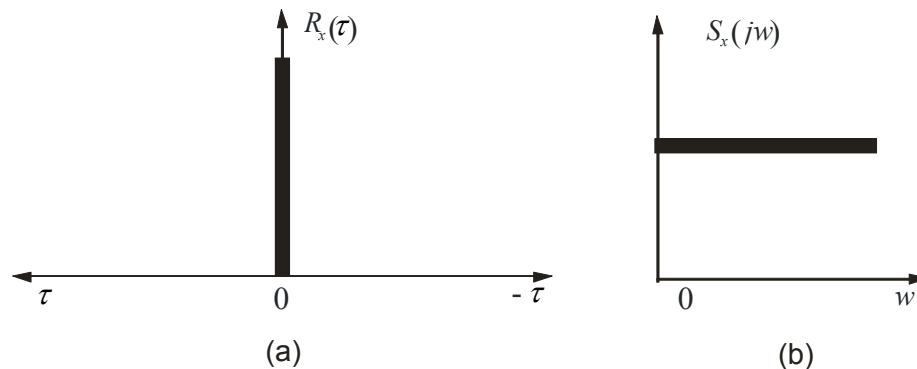
Uma importante ocorrência de sinal aleatório é o caso onde a função de autocorrelação é a função delta  $\delta(\tau)$  que tem valor zero em todo lugar, exceto quando  $\tau = 0$ , sendo  $\tau = \text{tempo final} - \text{tempo inicial}$ . Em outras palavras, o caso onde

$$R_x(\tau) = \begin{cases} \text{se } \tau = 0 \text{ então} & A \\ \text{senão} & 0 \end{cases}$$

para alguma magnitude de constante A.

Neste caso especial onde a auto-correlação tem a forma de um “espinho”, a transformada de Fourier resulta em um espectro de frequência constante (Figura 69). Esta é de fato uma descrição de ruído branco, que pode ser tratado por ambos como tendo a potência de todas as frequências no espectro, e ser completamente não-correlatado consigo mesmo em qualquer tempo, exceto no presente ( $\tau = 0$ ). Esta última interpretação é que leva o sinal de ruído branco a ser chamado de independente, ou seja, qualquer amostra de sinal que uma vez é completamente independente (não-correlacionado) de uma amostra em qualquer outro tempo.

Enquanto é impossível obter ou ver na prática (nenhum sistema pode exibir uma energia infinita através de espectro infinito), o ruído branco é um importante bloco de construção para projetar e análises. Frequentemente sinais aleatórios podem ser modelados como ruídos brancos filtrados ou modelados.



**Figura 69. Ruído branco mostrado em ambos o domínio de tempo (a) e frequência (b).**

Literalmente isto significa que se pode filtrar a saída de uma (hipotética) fonte de ruído branco para obter uma fonte de ruído não-branco ou colorido que são ambos com banda limitada no domínio de frequência e mais correlacionado no domínio do tempo.

## Apêndice B

---

### Treinamento de uma rede neural

O treinamento de redes neurais pode ser mais eficiente se certos passos de pré-processamento são feitos nas entradas e nos alvos. Antes de treinar, devem-se normalizar as entradas e os alvos para que eles sempre fiquem dentro de um campo específico, por exemplo,  $[-1,1]$ . Como a entrada e os alvos são normalizados, a saída também será. As saídas deverão ser convertidas às mesmas unidades dos alvos originais.

A técnica de *backpropagation* foi criada pela generalização da regra de aprendizado Widrow-Hoff para redes multicamadas e funções de transferências não-lineares.

Os vetores de entradas e os de alvos são usados para treinar a rede até que possa aproximar uma função, vetor de entrada associado com vetores de saída específico. Redes com bias, uma camada sigmoideal e uma camada linear são capazes de aproximar qualquer função com um número finito de descontinuidade.

O *backpropagation* padrão é um algoritmo gradiente, como a regra de aprendizado de Widrow-Hoff, em que os pesos da rede são movidos para o negativo do gradiente da função de desempenho. O termo *backpropagation* refere-se para a maneira em que o gradiente é computado nas camadas não lineares das redes multicamadas. Existe um número de variações do algoritmo básico que são baseados nas técnicas padrões de otimização, como gradientes conjugados e métodos de Newton.

As redes *backpropagation* treinadas apropriadamente retornam uma resposta razoável quando apresentadas às novas entradas que resultam em uma saída similar à saída correta pelos valores das entradas usadas no treinamento, que são similares às novas entradas que estão sendo apresentadas. Isso faz com que treinar uma rede com um conjunto representativo de pares entrada/alvo dê um bom resultado sem treinar a rede em todos os possíveis pares input/output.

O *backpropagation* ainda permite treinar redes multicamadas com diferentes funções de transferência para fazer uma aproximação de função, associação e classificação de padrões. Outros tipos de redes podem ser treinados para isso também, mas a rede multicamadas é a mais usada. O termo *backpropagation* também se refere ao processo pelo quais as derivadas dos erros da rede, a respeito de pesos e bias, podem ser computadas. O processo pode ser usado com várias estratégias de otimização.

A arquitetura multicamadas não é completamente restrita ao problema a ser resolvido. O número de entradas da rede é restrito ao problema e o número de neurônios na camada de saída é restrito ao número de saídas requeridas pelo problema. Contudo, o número de camadas entre as entradas da rede e a camada de saída e o tamanho das camadas são feitas através da melhor adequação ao problema pelo projetista.

Uma rede de duas camadas sigmoideal/linear pode representar qualquer relacionamento funcional entre entradas e saídas se a camada sigmoideal tiver o número suficiente de neurônios.

### Princípios de treinamento de uma rede neural multi-camadas utilizando o algoritmo *backpropagation*

A seguir é feita uma descrição do uso de um processo de aprendizagem de uma rede neural multicamadas utilizando o algoritmo *backpropagation*. Para demonstrar este processo, uma rede com três camadas com duas entradas e uma saída é mostrada na Figura 70.

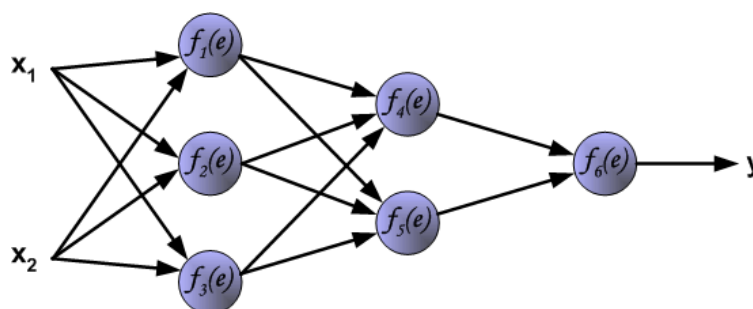
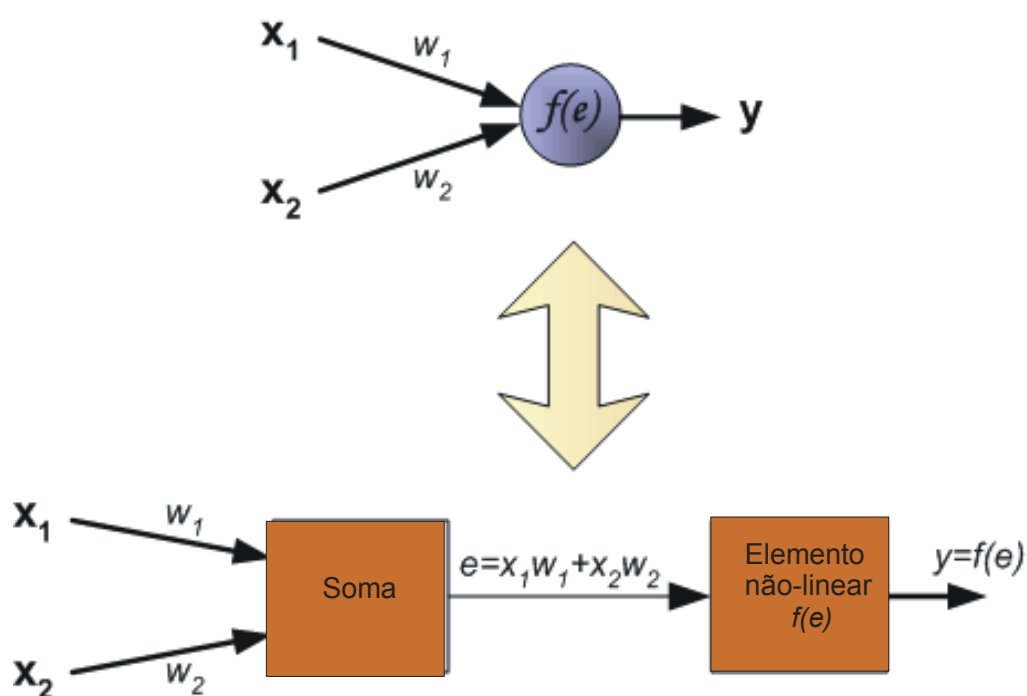


Figura 70. Rede neural com três camadas, com duas entradas, uma saída e três neurônios na primeira camada, dois na segunda e um na terceira.

Cada neurônio é composto de duas unidades como mostra a Figura 71. A primeira unidade adiciona o coeficiente dos produtos dos pesos e dos sinais de entrada.

A segunda unidade realiza uma função não-linear, chamada função de ativação do neurônio. O sinal  $e$  é adicionado ao sinal de saída, e  $y=f(e)$  é o sinal de saída do elemento não-linear. O sinal  $y$  é também o sinal de saída do neurônio.





**Figura 71. Detalhe da função soma do neurônio.**

Para ensinar a rede neural, deve-se treina-la com um conjunto de dados. O conjunto de dados para treinamento consiste em sinais de entrada ( $x_1$  e  $x_2$ ) assinalados com o alvo correspondente, ou seja, a saída desejada  $y$ .

O treinamento da rede é um processo iterativo. Em cada iteração os pesos dos neurônios são modificados usando um novo valor dos dados de treinamento. A modificação é calculada usando o algoritmo descrito a seguir:

Cada passo de aprendizagem começa com ambos os sinais de entradas do treinamento. Depois deste estágio, são determinados os valores de saída de todos os neurônios em cada camada da rede.

A Figura 72 mostra como o sinal é propagado através da rede a partir das entradas, até a primeira camada, sendo demarcado pelas setas azuis.

A Figura 73 mostra o próximo passo, da primeira camada com os dados sendo repassados para a camada oculta, e a Figura 74 mostra a saída da camada oculta para a camada de saída.

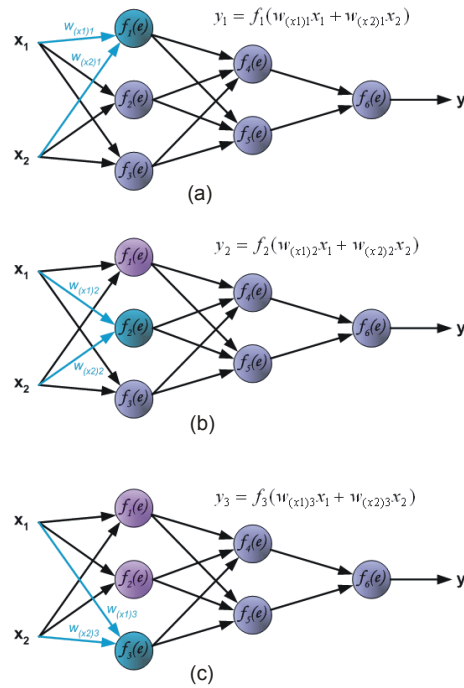


Figura 72. Em (a), os valores da entrada  $x_1$  e  $x_2$  para o primeiro neurônio da primeira camada, seguido, em (b), para o segundo neurônio e em (c), para o terceiro.

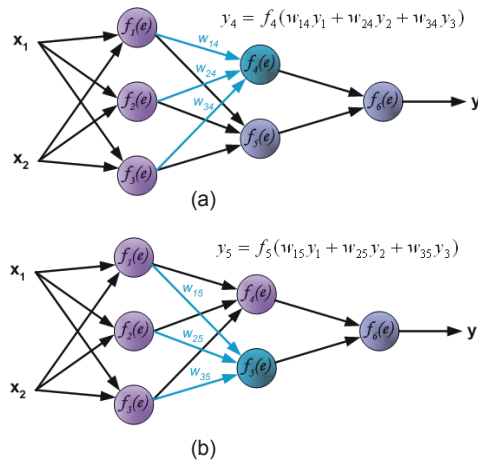


Figura 73. Em (a), a saída dos neurônios da primeira camada são repassados ao primeiro neurônio da segunda camada (ou camada oculta), enquanto em (b), ao segundo neurônio.

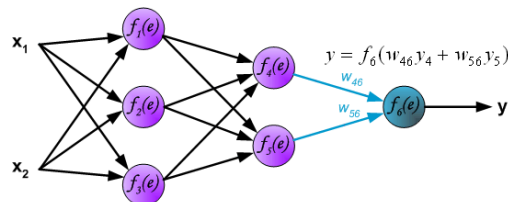
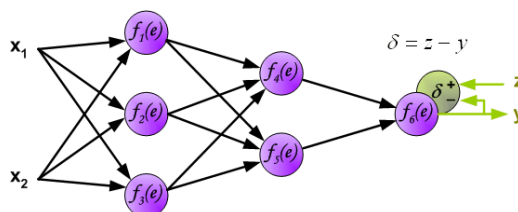


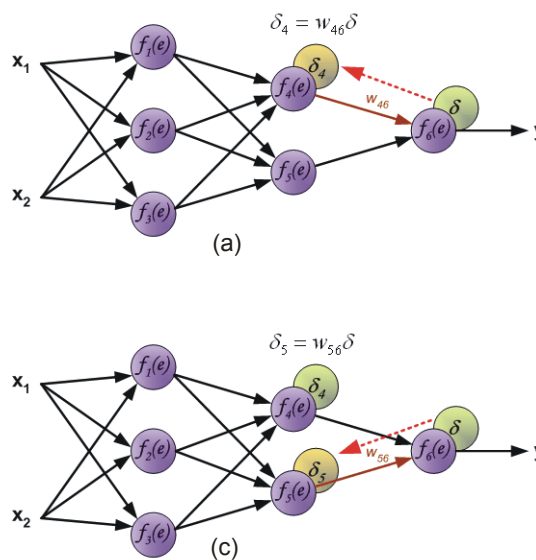
Figura 74. As saídas da camada oculta são repassadas ao neurônio da última camada (camada de saída).

No próximo passo do algoritmo (Figura 75), o sinal de saída  $y$  é comparado com o valor de saída desejada (o alvo), que é encontrada nos dados de tratamento. A diferença é chamada de sinal de erro  $\delta$  do neurônio da camada de saída.



**Figura 75. Erro sendo propagado de volta ao neurônio de saída.**

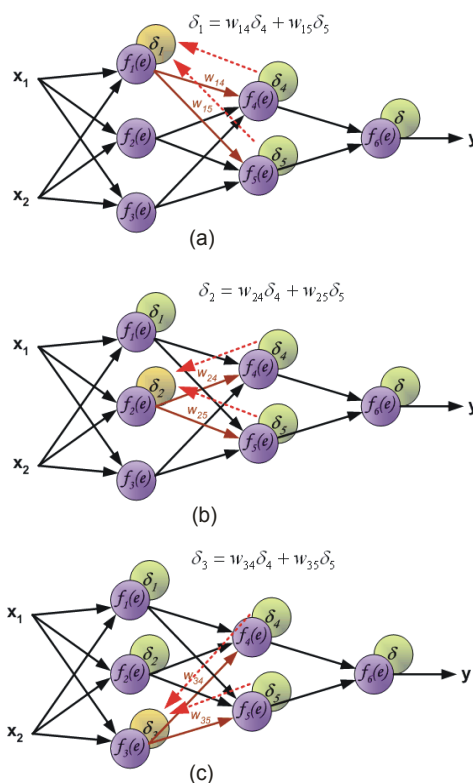
É impossível computar o sinal de erro para os neurônios internos de forma direta, porque os valores de saída destes neurônios são desconhecidos. Como já foi analisado anteriormente, por alguns anos o método efetivo para treinar redes multicamadas era desconhecido. Somente no meio dos anos oitenta que o algoritmo *backpropagation* foi trabalhado. A idéia é propagar o sinal de erro (computado num simples passo de aprendizado) de volta a todos os neurônios, os quais os sinais de saída foram entradas para o neurônio em questão, como visto na Figura 76.



**Figura 76. Em (a), o erro do neurônio da camada de saída é propagado para o primeiro neurônio da camada oculta. Em (b), o erro é propagado para o segundo neurônio.**

Os pesos utilizados para propagar os erros de volta são iguais àqueles utilizados para computar o valor de saída. Somente a direção do fluxo de dados é modificada (sinais são propagados das saídas para as entradas depois de cada um). Esta técnica é usada para

todas as camadas de redes. Se os erros propagados vierem de outros neurônios, eles são somados (Figura 77).



**Figura 77. Retropropagação dos erros da camada oculta para a camada de entrada (setas tracejadas em vermelho) e erros são somados aos erros anteriores (setas vermelhas).**

Quando o sinal de erro de cada neurônio é computado, os pesos de cada neurônio de entrada são modificados. Nas equações abaixo,  $df(e)/de$  representa a derivada da função de ativação do neurônio (que tem os pesos modificados), como vistos nas Figura 78 e Figura 79 .

O coeficiente  $\eta$  afeta a velocidade de aprendizagem da rede. Existem algumas técnicas para selecionar esse parâmetro. O primeiro método é começar o processo de aprendizagem com um alto valor do parâmetro. Enquanto os pesos são estabelecidos para diminuir gradualmente. O segundo método, que é o mais complexo, começa o aprendizado com um valor pequeno. Durante o processo de aprendizado, o valor é aumentado quando o aprendizado é avançado e então é decrementado de novo no estágio final. Começando o processo de aprendizado com um valor baixo permite determinar os pesos.

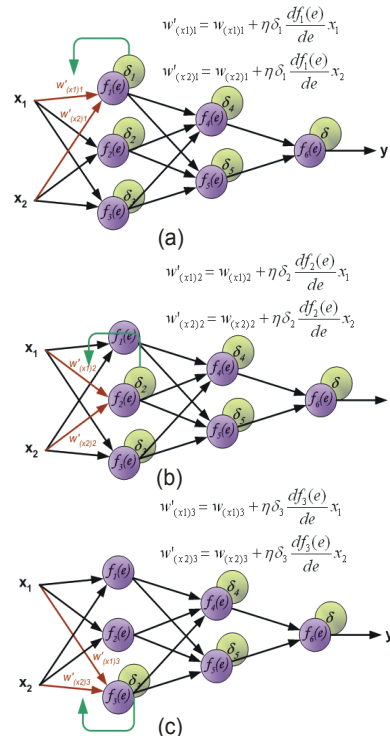


Figura 78. Em (a), os pesos de entrada são modificados de acordo com o erro do primeiro neurônio da primeira camada. Em (b) e (c) segue da mesma forma para seus respectivos pesos e neurônios.

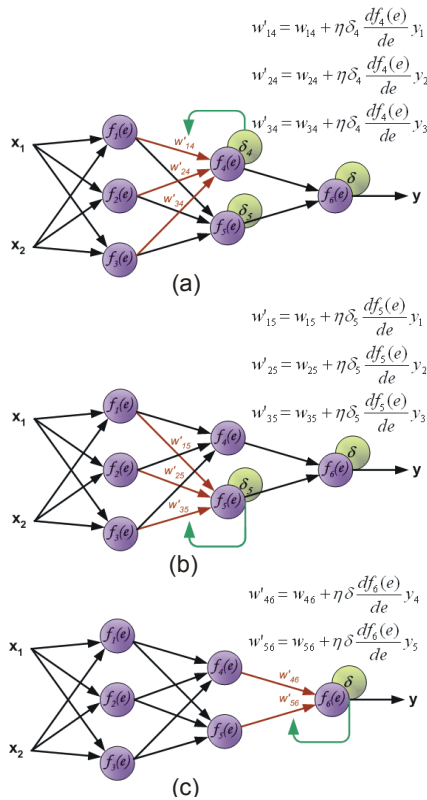


Figura 79. Em (a) e (b), a atualização dos pesos da segunda camada são feitas de acordo com os erros dos neurônios da segunda camada. Em (c), os pesos são atualizados pelo erro do neurônio de saída.