

PREDIÇÃO DE SÉRIES TEMPORAIS PARA ANÁLISE DINÂMICA DE
ESTRUTURAS OFFSHORE

Nelson Antonio Fernandes de Matos

TESE SUBMETIDA AO CORPO DOCENTE DA COORDENAÇÃO DOS
PROGRAMAS DE PÓS-GRADUAÇÃO DE ENGENHARIA DA UNIVERSIDADE
FEDERAL DO RIO DE JANEIRO COMO PARTE DOS REQUISITOS
NECESSÁRIOS PARA A OBTENÇÃO DO GRAU DE MESTRE EM CIÊNCIAS EM
ENGENHARIA CIVIL.

Aprovada por:

Prof. Edison Castro Prates de Lima, D.Sc.

Prof. Luís Volnei Sudati Sagrilo, D.Sc.

Prof. Marcos Queija de Siqueira, D.Sc.

Prof. Gilberto Bruno Ellwanger, D.Sc.

Dr. Márcio Martins Mourelle, D.Sc.

RIO DE JANEIRO, RJ - BRASIL

MARÇO DE 2005

MATOS, NELSON ANTONIO
FERNANDES DE

Predição de Séries Temporais para
Análise Dinâmica de Estruturas Offshore
[Rio de Janeiro] 2005

XII, 83 p. 29,7 cm (COPPE/UFRJ,
M.Sc., Engenharia Civil, 2005)

Tese - Universidade Federal do Rio de
Janeiro, COPPE

1. Análise Dinâmica
2. Estruturas Offshore

I. COPPE/UFRJ II. Título (série)

Aos meus pais Nelson e Gorétti.
À minha namorada Patrícia.

AGRADECIMENTOS

Em primeiro lugar devo agradecer a Deus por tudo.

Agradeço aos meus pais e à minha namorada Patrícia pelo apoio, carinho e incentivo em todas as horas. Meus pais muitas vezes se sacrificaram em prol de minha educação, sem eles não seria o que sou hoje. À Patrícia devo agradecer também pela compreensão a respeito das horas de convívio que foram sacrificadas para a conclusão deste mestrado.

Agradeço aos Professores Gilberto e Magluta pelo incentivo que me deram desde os tempos de graduação para o ingresso no mestrado.

Agradeço aos Professores Gilberto, Edison, Queija e Sagrilo pela oportunidade de trabalho e experiência profissional adquirida no LCRAA (Laboratório de Confiabilidade, Risco e Análise Aleatória) da COPPE/UFRJ. Sem os conhecimentos adquiridos esta tese não seria a mesma.

Agradeço a todos do LCRAA e do LAMCE (Laboratório de Métodos Computacionais em Engenharia). Sempre havia alguém disposto a ajudar, seja com conhecimento técnico ou uma palavra de apoio. A estes não poderia deixar de agradecer também pelas horas de descontração que são indispensáveis para tornar nossos dias mais fáceis e agradáveis.

Agradeço aos orientadores. Ao Professor Luis Volnei Sudati Sagrilo agradeço pela sua dedicação, incentivo, paciência, disponibilidade e amizade. Ao Professor Edison Castro Prates de Lima agradeço pelas importantes observações (sempre de forma clara e objetiva) que deram rumo ao trabalho desde os primeiros passos.

Agradeço à Agência Nacional do Petróleo (ANP) que através do Programa de Recursos Humanos 35 (PRH-35) forneceu o suporte financeiro para a execução deste trabalho.

Finalmente agradeço a todas as pessoas que de alguma forma contribuíram para que chegasse onde estou, seria impossível citar todas.

Resumo da Tese apresentada à COPPE/UFRJ como parte dos requisitos necessários para a obtenção do grau de Mestre em Ciências (M.Sc.).

PREDIÇÃO DE SÉRIES TEMPORAIS PARA ANÁLISE DINÂMICA DE
ESTRUTURAS OFFSHORE

Nelson Antonio Fernandes de Matos

Março/2005

Orientadores: Edison Castro Prates de Lima

Luís Volnei Sudati Sagrilo

Programa: Engenharia Civil

Este trabalho apresenta o estudo e implementação de métodos de predição de séries temporais na análise dinâmica de estruturas offshore. Seu objetivo é identificar a aplicabilidade dos métodos e apresentar exemplos de aplicação. São investigados os modelos NARMAX (*Nonlinear AutoRegressive Moving Average with eXogenous inputs*) nas suas formas polinomial e racional. Também são consideradas as Redes Neurais Artificiais (RNAs). Um algoritmo híbrido para o cálculo dos coeficientes (pesos) associados aos termos dos modelos é desenvolvido. São apresentados exemplos incluindo um sistema dinâmico simples, um *riser* de produção e uma linha de ancoragem.

Abstract of Thesis presented to COPPE/UFRJ as a partial fulfillment of the requirements for the degree of Master of Science (M.Sc.).

TIME SERIES PREDICTION FOR DYNAMIC ANALYSIS OF OFFSHORE STRUCTURES

Nelson Antonio Fernandes de Matos

March/2005

Advisor: Edison Castro Prates de Lima

Luís Volnei Sudati Sagrilo

Department: Civil Engineering

This work presents the study and implementation of time series prediction methods for dynamic analysis of offshore structures. The objective is to identify the feasibility of the methods and present examples of applications. The NARMAX (Nonlinear **A**uto**R**egressive **M**oving **A**verage with **e**Xogenous inputs) models are investigated both in the polynomial and rational forms. The Artificial Neural Networks are also studied. A hybrid algorithm for the calculation of the coefficients (weights) associated to the terms of the models is developed. Examples are presented including a simple dynamic system, a production riser and a mooring line.

SUMÁRIO

CAPÍTULO I - INTRODUÇÃO	1
CAPÍTULO II - MODELO NARMAX.....	7
II.1 Modelo NARMAX polinomial	7
II.2 Modelo NARMAX racional	9
CAPÍTULO III - REDES NEURAIS ARTIFICIAIS.....	11
III.1 O neurônio biológico	11
III.2 O neurônio artificial	12
III.2.1 Funções de ativação.....	12
III.3 Redes multicamadas	15
CAPÍTULO IV - TREINAMENTO.....	18
IV.1 Vetor gradiente	19
IV.1.1 Modelo NARMAX polinomial	20
IV.1.2 Modelo NARMAX racional.....	20
IV.1.3 Redes Neurais Artificiais.....	21
IV.1.3.1 Derivadas dos pesos ligados ao neurônio de saída.....	21
IV.1.3.2 Derivadas dos pesos ligados a um neurônio oculto.....	22
IV.2 Caso particular: Modelo NARMAX polinomial.....	23
IV.3 Método do Gradiente Descendente	25
IV.4 Método do Gradiente Conjugado	26
IV.4.1 Forma linear	26
IV.4.2 Forma não-linear	28
IV.5 Estratégias Evolutivas	30
IV.6 Problemas no treinamento.....	31
IV.6.1 O problema do Mínimo local	32
IV.6.1.1 Metodologia de busca proposta.....	33
IV.6.2 O erro cometido na Extrapolação.....	35
IV.6.3 O problema do super ajuste ou <i>overfitting</i>	36
CAPÍTULO V - EXEMPLOS DE APLICAÇÃO.....	38

V.1 Sistema Dinâmico Massa-mola-amortecedor.....	40
V.1.1 Caso linear	43
V.1.2 Caso não-linear	48
V.1.3 Comentários.....	52
V.2 Resposta dinâmica de um <i>Riser</i> de aço com geometria <i>lazy wave</i>	53
V.2.1 Ponto na Corcova: Esforço MY.....	58
V.2.2 Ponto no topo: Esforço MY.....	60
V.2.3 Ponto na corcova: Esforço FX.....	63
V.2.4 Ponto no topo: Esforço FX.....	66
V.2.5 Comentários.....	69
V.3 Análise de Extremos via Amostra de Máximos Observados em Realizações Independentes da Resposta.....	70
V.3.1 Linha de ancoragem.....	72
V.4 Desempenho da metodologia de cálculo proposta.....	78
 CAPÍTULO VI - CONSIDERAÇÕES FINAIS.....	79
VI.1 Conclusões	79
VI.2 Sugestão para trabalhos futuros.....	80

INDICE DE FIGURAS

Figura I-1 – Esquema genérico de uma análise dinâmica	2
Figura I-2 – Esquema genérico do uso de Métodos de Predição de Séries Temporais....	5
Figura III-1 – Neurônio biológico	11
Figura III-2 – Neurônio artificial.....	12
Figura III-3 – Gráfico da função linear	13
Figura III-4 – Gráfico da tangente hiperbólica.....	14
Figura III-5 – Gráfico da sigmóide.....	14
Figura III-6 – Rede multicamadas	16
Figura III-7 – RNA com três entradas (dois atrasos) e uma camada oculta.....	17
Figura IV-1 – O problema do mínimo local.....	32
Figura IV-2 – Exemplo de erro na extrapolação	35
Figura V-1 – Sistema massa-mola-amortecedor	41
Figura V-2 – Espectro de Pierson-Moskovitz, $H_s = 7.8m$ e $T_z = 11.8s$	41
Figura V-3 – Função de transferência do sistema massa-mola-amortecedor.....	42
Figura V-4 – Treinamento de 200s, caso linear.....	44
Figura V-5 – Treinamento de 300s, caso linear.....	44
Figura V-6 – Treinamento de 400s, caso linear.....	45
Figura V-7 – Série temporal do deslocamento onde ocorre o maior erro, caso linear. ..	46
Figura V-8 – Série temporal da velocidade onde ocorre o maior erro, caso linear.	46
Figura V-9 – Série temporal da aceleração onde ocorre o maior erro, caso linear.	47
Figura V-10 – Treinamento de 200s, caso não-linear.	48
Figura V-11 – Treinamento de 300s, caso não-linear.	49
Figura V-12 – Treinamento de 400s, caso não-linear.	49
Figura V-13 – Série temporal do deslocamento onde ocorre o maior erro, caso não-linear.....	50
Figura V-14 – Série temporal da velocidade onde ocorre o maior erro, caso não-linear.	51
Figura V-15 – Série temporal da aceleração onde ocorre o maior erro, caso não-linear.	51
Figura V-16 – FPSO.....	53

Figura V-17 – Configuração geométrica em <i>lazy wave</i>	54
Figura V-18 – Pontos analisados	55
Figura V-19 – Direção do carregamento ambiental	56
Figura V-20 – deslocamento X (<i>surge</i>) imposto no topo	56
Figura V-21 – deslocamento Y (<i>sway</i>) imposto no topo	56
Figura V-22 – deslocamento Z (<i>heave</i>) imposto no topo	57
Figura V-23 – Treinamento de 300s, MY corcova.....	58
Figura V-24 – Treinamento de 400s, MY corcova.....	58
Figura V-25 – Treinamento de 500s, MY corcova.....	59
Figura V-26 – Série temporal, MY corcova.....	60
Figura V-27 – Treinamento de 300s, MY topo.	61
Figura V-28 – Treinamento de 400s, MY topo.	61
Figura V-29 – Treinamento de 500s, MY topo.	62
Figura V-30 – Série temporal, MY topo.	63
Figura V-31 – Treinamento de 300s, FX corcova.	64
Figura V-32 – Treinamento de 400s, FX corcova.	64
Figura V-33 – Treinamento de 500s, FX corcova.	65
Figura V-34 – Série temporal, FX corcova.	66
Figura V-35 – Treinamento de 300s, FX topo.....	67
Figura V-36 – Treinamento de 400s, FX topo.....	67
Figura V-37 – Treinamento de 500s, FX topo.....	68
Figura V-38 – Série temporal, FX topo.....	69
Figura V-39 – Amostra de treinamento de 200s.....	73
Figura V-40 – Amostra de treinamento de 300s.....	73
Figura V-41 – Amostra de treinamento de 400s.....	74
Figura V-42 – Série temporal da tração no topo. Realização #1.....	75
Figura V-43 – Ajuste de função de distribuição de probabilidade Tipo I.....	77

ÍNDICE DE TABELAS

Tabela II-1 – Polinômio NARMAX sob a forma vetorial.....	8
Tabela II-2 – Modelo racional NARMAX sob a forma de vetor	10
Tabela IV-1 – Algoritmo do Método do Gradiente Descendente	25
Tabela IV-2 – Algoritmo do Método do Gradiente Conjugado não linear	30
Tabela IV-3 – Algoritmo de Estratégias Evolutivas.....	31
Tabela IV-4 – Seqüência de cálculo proposta, modelo NARMAX	34
Tabela IV-5 – Seqüência de cálculo proposta, RNA.....	34
Tabela V-1 – Modelos NARMAX analisados.....	43
Tabela V-2 – Funções de ativação analisadas.	43
Tabela V-3 – Erro médio em função da amostra de treinamento, G1, 150 atrasos.....	45
Tabela V-4 – Tempo onde ocorre o maior erro em módulo, G1, 150 atrasos.....	46
Tabela V-5 – Coeficiente de correlação, G1, 150 atrasos.	47
Tabela V-6 – Média e Desvio padrão dos sinais original e simulado, G1, 150 atrasos.	47
Tabela V-7 – Erro médio em função da amostra de treinamento, G2/G1, 150 atrasos..	50
Tabela V-8 – Tempo onde ocorre o maior erro em módulo, G2/G1, 150 atrasos.....	50
Tabela V-9 – Coeficiente de correlação, G2/G1, 150 atrasos.	51
Tabela V-10 – Média e Desvio padrão dos sinais original e simulado, G2/G1, 150 atrasos.....	52
Tabela V-11 – Dados de carregamento - Corrente e Movimento de baixa freqüência ..	55
Tabela V-12 – Dados de carregamento de onda.....	55
Tabela V-13 – Dados de movimentos estáticos e de segunda ordem.....	55
Tabela V-14 – Modelos NARMAX analisados.....	57
Tabela V-15 – Redes Neurais, funções de ativação analisadas.....	57
Tabela V-16 – Erro médio em função da amostra de treinamento, G2, 60 atrasos.....	59
Tabela V-17 – Média, desvio padrão e coeficiente de correlação, G2, 60 atrasos.....	60
Tabela V-18 – Erro médio em função da amostra de treinamento, G2, 40 atrasos.....	62
Tabela V-19 – Média, Desvio Padrão e Coeficiente de correlação, G2, 40 atrasos.....	63
Tabela V-20 – Erro médio em função da amostra de treinamento, A1, 60 atrasos.....	65
Tabela V-21 – Média, Desvio Padrão e Coeficiente de correlação, A1, 60 atrasos.....	66
Tabela V-22 – Erro médio em função da amostra de treinamento, G2, 60 atrasos.....	68
Tabela V-23 – Média, Desvio Padrão e Coeficiente de correlação, G2, 60 atrasos.....	69

Tabela V-24 – Redução do custo computacional	69
Tabela V-25 – Erro médio em função da amostra de treinamento, G2, 10 atrasos.....	74
Tabela V-26 – Parâmetros do sinal original e simulado a partir de #1.	75
Tabela V-27 – Parâmetros do sinal original e simulado a partir de #15.	76
Tabela V-28 – Resumo Estatístico da Distribuição de Extremos do Tipo I.....	77
Tabela V-29 – Teste de desempenho da metodologia de cálculo proposta.....	78

CAPÍTULO I - INTRODUÇÃO

Com a exploração de petróleo e gás ocorrendo em águas cada vez mais profundas as estruturas *offshore*, tais como *risers* metálicos e flexíveis, sistemas de ancoragem, etc., tornaram-se desafios tecnológicos cada vez mais complexos. Devido às condições de segurança e aos altos custos envolvidos em função do aumento da lâmina d'água e da complexidade de instalação, torna-se necessária uma modelagem mais criteriosa do comportamento estrutural dinâmico de tais estruturas.

A análise dinâmica aleatória no domínio do tempo de estruturas discretizadas pelo método dos elementos finitos constitui-se na ferramenta mais adequada para análise de sistemas estruturais do tipo *risers* e linhas de ancoragem conectadas a unidades flutuantes de produção em águas profundas. Esta ferramenta tem a capacidade de permitir a modelagem de todas as não-linearidades físicas e geométricas presentes na análise. Porém, a grande capacidade deste tipo de análise tem um custo computacional elevado, principalmente nos dias atuais onde são utilizados modelos estruturais cada vez maiores, em função do aumento de profundidade da lâmina d'água de exploração.

O tipo de análise dinâmica aleatória usualmente empregada em projetos de *risers* e linhas (ou tendões) de ancoragem é a análise desacoplada, onde estas estruturas são analisadas separadamente da unidade flutuante. Neste tipo de análise os movimentos da unidade flutuante, obtidos através de uma análise dinâmica específica, são impostos à estrutura através de movimentos prescritos. No domínio do tempo estes movimentos prescritos (excitação) são constituídos por séries temporais dos movimentos de translação e rotação do corpo flutuante no ponto de conexão da estrutura analisada. Os resultados da análise são constituídos por séries temporais dos parâmetros de resposta de interesse, tais como: esforços, tensões, deslocamentos, etc. Na Figura I-1 ilustra-se genericamente este tipo de análise. É importante observar que para alcançar uma estabilidade estatística nos resultados obtidos, tanto na análise de extremos como na análise de fadiga, as séries temporais devem ser suficientemente longas, o que faz uma simples análise demandar um tempo computacional de algumas horas.

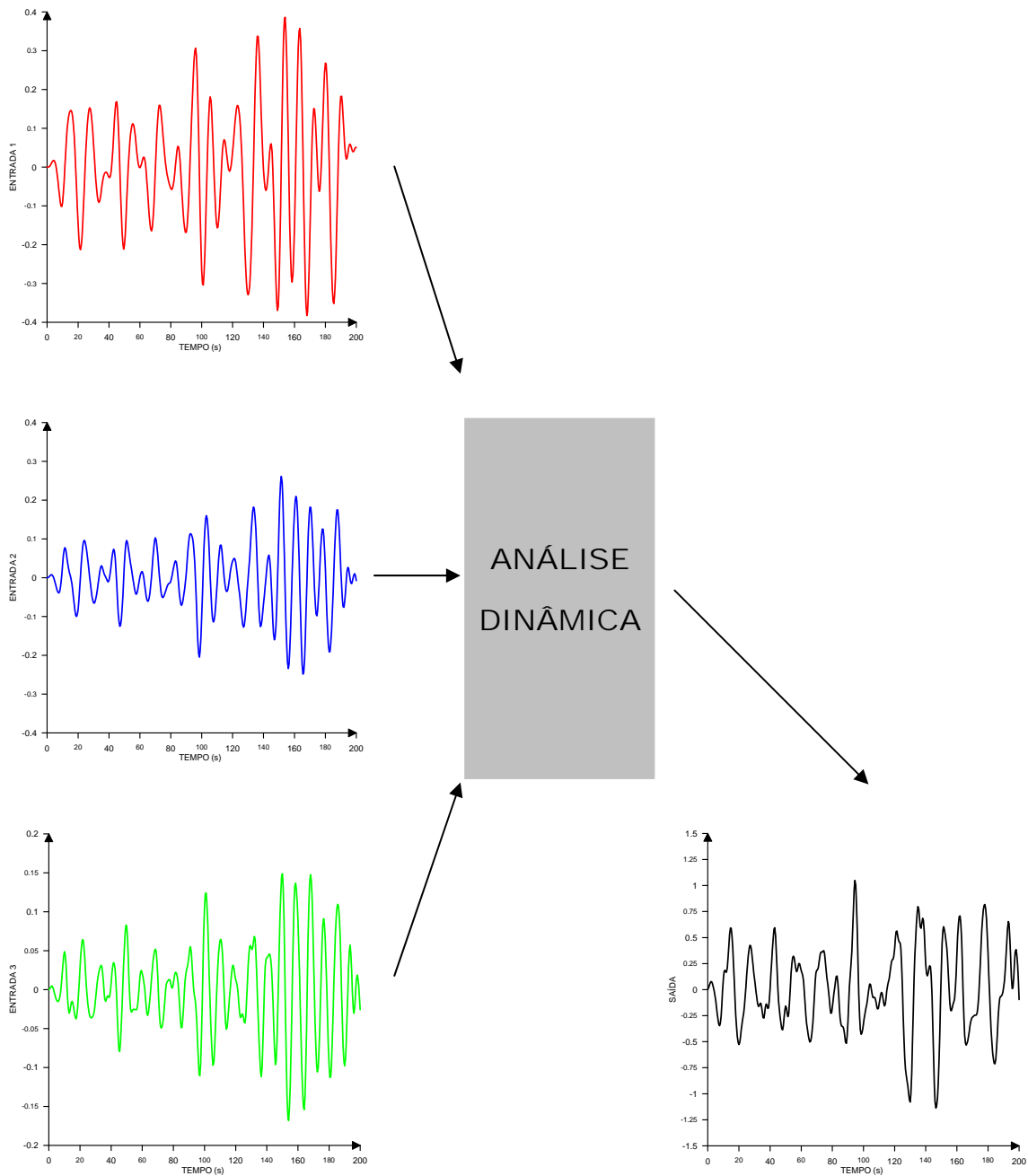


Figura I-1 – Esquema genérico de uma análise dinâmica

Por outro lado, em algumas áreas da engenharia, tais como controle e automação, devido à dificuldade de se obter um modelo matemático para o problema físico sob investigação, se recorre a modelos mais simples para descrever a resposta dinâmica de um sistema. Esta linha de análise é conhecida na literatura como Identificação de Sistemas [7].

A modelagem através de técnicas de identificação é uma abordagem semi-empírica. Consiste no desenvolvimento de modelos matemáticos simples que permitam a predição da resposta (saída) de sistemas dinâmicos em função somente dos dados de entrada (excitação), não exigindo conhecimento prévio dos parâmetros dinâmicos do sistema. Um parâmetro de resposta $r(t)$ do sistema num instante de tempo t é representado em um modelo matemático simples dado por:

$$r(t) = f[r(t-1 \cdot \Delta t), \dots, r(t-n_r \cdot \Delta t), x_1(t-0 \cdot \Delta t), \dots, x_1(t-n_{x1} \cdot \Delta t), x_m(t-0 \cdot \Delta t), \dots, x_m(t-n_{xm} \cdot \Delta t)] + \varepsilon(t) \quad (\text{I-1})$$

onde Δt é o intervalo de tempo discreto, $x_1(t), \dots, x_m(t)$ são as m fontes de excitação, $n_{x1}, \dots, n_{xm}, n_r$ são os números de atrasos das entradas e da saída, respectivamente, considerados na modelagem, $\varepsilon(t)$ é um resíduo (erro) de predição em função dos parâmetros de entrada, $f(\cdot)$ é uma função matemática que será comentada a seguir. Os atrasos estão relacionados ao intervalo de tempo (memória) em que um parâmetro de entrada continua a influenciar na resposta atual do sistema. É interessante observar que na identificação de sistemas é necessário se obter um conjunto de entradas com a respectiva resposta de interesse para que o modelo matemático possa ser resolvido.

Para a função de mapeamento $f(\cdot)$, apresentada na Equação (I-1), muitos modelos têm sido utilizados em problemas práticos de engenharia, dentre os quais as Redes Neurais Artificiais [2],[3] e o modelo NARMAX (*Nonlinear AutoRegressive Moving Average with eXogenous inputs*), na sua forma polinomial [5],[6],[7] ou racional [14]. Observa-se que no presente trabalho não foi levado em consideração os termos dependentes relacionados com a própria resposta, $r(t-1 \cdot \Delta t), \dots, r(t-n_r \cdot \Delta t)$, em nenhum dos modelos analisados. Desta forma, como na análise dinâmica de estruturas [8], a resposta em um instante de tempo é dependente, dentre outros parâmetros, dos valores da excitação no tempo presente e em instantes passados de tempo.

O objetivo deste trabalho é uma investigação sobre a possibilidade de se desenvolver um sistema híbrido para análise dinâmica aleatória de estruturas *offshore*, onde um período curto do parâmetro de resposta de interesse é obtido inicialmente através de uma análise dinâmica tradicional e posteriormente uma técnica de

identificação de sistemas é empregada na predição futura desta resposta, tal como mostra a Figura I-2, diminuindo assim o custo computacional total da análise. Para tal finalidade, como será visto mais adiante, são analisados e comparados três modelos distintos para representar o mapeamento simplificado entre as entradas e as saídas:

- 1) Redes Neurais Artificiais (RNA);
- 2) Modelo NARMAX polinomial e;
- 3) Modelo NARMAX racional

É importante mencionar que o estudo destes três métodos disponíveis na literatura tem a finalidade de verificar se a sua aplicação é possível na análise dinâmica de estruturas *offshore*. O objetivo do trabalho não é a comparação direta entre os mesmos.

No Capítulo II os modelos NARMAX polinomial e racional são apresentados matematicamente e sob a forma de um simples exemplo com dois atrasos. O Capítulo III apresenta as RNAs e seus componentes básicos (neurônio, camadas e etc.). Também é apresentada a arquitetura da rede para um exemplo simples. O Capítulo IV trata do cálculo dos pesos no modelo NARMAX e dos pesos sinápticos nas RNAs. Três métodos disponíveis (e seus algoritmos) são apresentados, bem como os problemas envolvidos em cada um. Também é proposta uma metodologia de cálculo a fim de se tirar partido das melhores características de cada método. O Capítulo V apresenta três exemplos de aplicação para o modelo NARMAX e as RNAs. O Capítulo VI apresenta as conclusões e sugestões para pesquisas futuras.

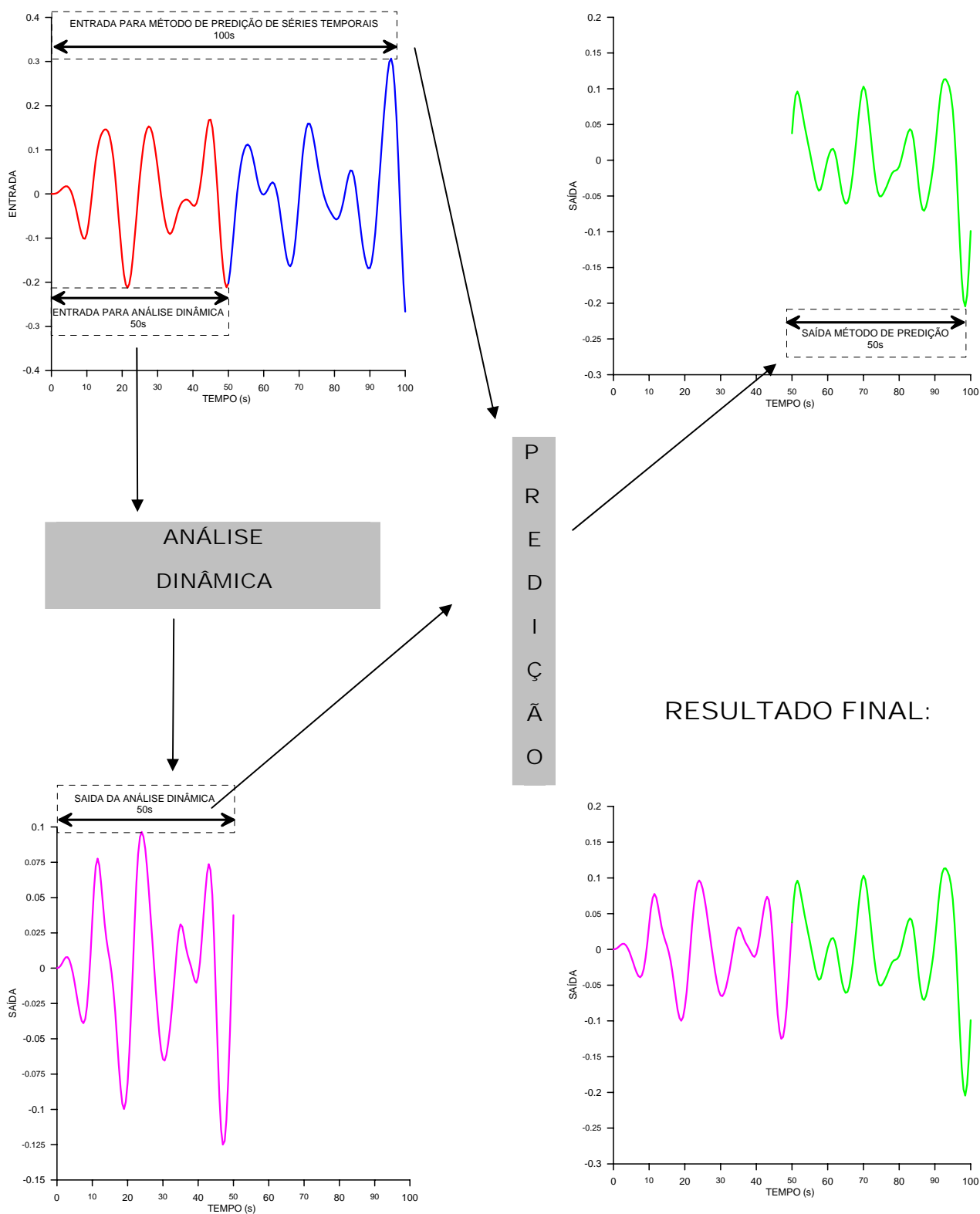


Figura I-2 – Esquema genérico do uso de Métodos de Predição de Séries Temporais

BREVE HISTÓRICO

O estudo de técnicas de Identificação de Sistemas aplicadas ao caso de análise dinâmica de estruturas teve início no Programa de Engenharia Civil (PEC) da COPPE/UFRJ com a tese de M.Sc. de GUARIZE [9], orientada pelos Professores Edison Castro Prates de Lima e Luís Volnei Sudati Sagrilo. O estudo contemplou as Redes Neurais Artificiais e constatou que redes com apenas uma camada oculta e saída linear (o significado destes termos é explicado mais adiante) são suficientes para representar de forma satisfatória a resposta dinâmica de estruturas.

CAPÍTULO II - MODELO NARMAX

Os modelos NARMAX (*Nonlinear Auto Regressive Moving Average with exogenous inputs*) foram desenvolvidos para modelagem de sistemas dinâmicos não-lineares discretos [5],[6],[7],[14]. Duas versões relacionadas a este modelo encontram-se disponíveis na literatura: modelo NARMAX polinomial e modelo NARMAX racional. A seguir estes dois modelos serão comentados. Sem perda de generalidade e para facilitar o entendimento os modelos serão exemplificados em função de um modelo dinâmico simples com uma entrada $x(t)$ e uma saída $y(t)$. A expansão deste modelo para mais de uma entrada envolve apenas alguns passos adicionais.

II.1 Modelo NARMAX polinomial

No modelo polinomial NARMAX a função $f(\cdot)$ apresentada na Equação (I-1) é representada por um polinômio de grau $p \leq 2$, i.e.,

$$y(t) = f[x(t), x(t - \Delta t), \dots, x(t - n_x \Delta t)] + \varepsilon(t)$$

$$y(t) \approx a_{0,0,0} + \sum_{j=1}^p \sum_{i=0}^{n_x} \left[a_{i,j,i} \cdot x(t - i \cdot \Delta t)^j + \left(\sum_{k=0}^{n_x} a_{i,j,k} \cdot x(t - i \cdot \Delta t)^{j-1} \cdot x(t - k \cdot \Delta t) \cdot H(i, j, k) \right) \right]$$

(II-1)

onde n_x é o número de atrasos considerados na entrada, os termos $a_{i,j,k}$ correspondem aos coeficientes (ou genericamente pesos) do polinômio (a obtenção destes coeficientes será comentada mais adiante) e Δt é o intervalo de tempo discreto utilizado. É importante mencionar que a Equação (II-1) é válida apenas para polinômios do primeiro e segundo graus. A função $H(i, j, k)$ é uma função indicadora que assume o valor zero quando um determinado termo não deve ser computado no polinômio e o valor 1 em caso contrário, i.e.,

$$H(i, j, k) = \left\{ \begin{array}{l} 1, (k \neq i) \\ 0, (k = i) \text{ ou } (j = 1) \text{ ou } [(j = 2) \text{ e } (k < i)] \end{array} \right\} \quad \text{(II-2)}$$

O resíduo $\varepsilon(t)$ será a diferença entre o valor verdadeiro de $y(t)$ e o valor calculado pela expansão polinomial.

É interessante observar que o número de coeficientes do polinômio é igual ao número de termos. Portanto basta expandir a Equação (II-1) e fazer a contagem para saber o número de coeficientes envolvidos no polinômio.

O modelo NARMAX polinomial expandido para o caso em que o número de atrasos é igual a dois ($n_x = 2$) e o polinômio é do segundo grau ($p = 2$) assume a seguinte forma:

$$y(t) = a_{0,0,0} + a_{0,1,0} \cdot x(t-0 \cdot \Delta t) + a_{1,1,1} \cdot x(t-1 \cdot \Delta t) + a_{2,1,1} \cdot x(t-2 \cdot \Delta t) + a_{0,2,0} \cdot x(t-0 \cdot \Delta t)^2 + a_{0,2,1} \cdot x(t-0 \cdot \Delta t) \cdot x(t-1 \cdot \Delta t) + a_{0,2,2} \cdot x(t-0 \cdot \Delta t) \cdot x(t-2 \cdot \Delta t) + a_{1,2,1} \cdot x(t-1 \cdot \Delta t)^2 + a_{1,2,2} \cdot x(t-1 \cdot \Delta t) \cdot x(t-2 \cdot \Delta t)$$

(II-3)

Do ponto de vista computacional é interessante armazenarmos os coeficientes $a_{i,j,k}$ e os termos referentes às entradas sob a forma de vetores \mathbf{A} e \mathbf{Z}_n , respectivamente. Vendo o polinômio sob esta forma a programação é mais simples, bem como o entendimento do cálculo dos coeficientes (a ser comentado mais adiante). A expressão (II-3) armazenada sob esta forma pode ser vista na Tabela II-1 e na equação (II-4), onde \mathbf{A} é o vetor dos coeficientes e \mathbf{Z}_n é o vetor que contém os termos referentes às entradas.

Índice do Vetor	Componentes	
	Vetor \mathbf{A}	Vetor \mathbf{Z}_n
0	a_0	1.00
1	a_1	$x(t-0 \cdot \Delta t)$
2	a_2	$x(t-1 \cdot \Delta t)$
3	a_3	$x(t-2 \cdot \Delta t)$
4	a_4	$x(t-0 \cdot \Delta t)^2$
5	a_5	$x(t-0 \cdot \Delta t) x(t-1 \cdot \Delta t)$
6	a_6	$x(t-0 \cdot \Delta t) x(t-2 \cdot \Delta t)$
7	a_7	$x(t-1 \cdot \Delta t)^2$
8	a_8	$x(t-1 \cdot \Delta t) x(t-2 \cdot \Delta t)$

Tabela II-1 – Polinômio NARMAX sob a forma vetorial

Utilizando a forma vetorial o modelo NARMAX polinomial pode ser re-escrito da seguinte forma:

$$y(t) = \mathbf{A} \cdot \mathbf{Z}_n(t) \quad (\text{II-4})$$

II.2 Modelo NARMAX racional

O modelo NARMAX racional [14] é definido matematicamente por:

$$y(t) = \frac{y_n(t)}{1 + y_d(t)} \quad (\text{II-5})$$

onde $y_n(t)$ e $y_d(t)$ são dois polinômios em função dos parâmetros de entrada de ordem p_n e p_d , respectivamente, definidos por:

$$y_n(t) = a_{0,0,0} + \sum_{j=1}^{p_n} \sum_{i=0}^{n_x} \left[a_{i,j,i} \cdot x(t-i \cdot \Delta t)^j + \left(\sum_{k=0}^{n_x} a_{i,j,k} \cdot x(t-i \cdot \Delta t)^{j-1} \cdot x(t-k \cdot \Delta t) \cdot H(i, j, k) \right) \right] \quad (\text{II-6})$$

e

$$y_d(t) = \sum_{j=1}^{p_d} \sum_{i=0}^{n_x} \left[b_{i,j,i} \cdot x(t-i \cdot \Delta t)^j + \left(\sum_{k=0}^{n_x} b_{i,j,k} \cdot x(t-i \cdot \Delta t)^{j-1} \cdot x(t-k \cdot \Delta t) \cdot H(i, j, k) \right) \right] \quad (\text{II-7})$$

onde os coeficientes $a_{i,j,k}$ e $b_{i,j,k}$ são coeficientes (pesos) dos polinômios empregados no numerador e denominador, respectivamente. As demais variáveis seguem a nomenclatura da expressão (II-1).

No modelo NARMAX racional os coeficientes do polinômio do numerador são diferentes dos coeficientes do denominador. O polinômio do denominador não tem o coeficiente constante, i.e., $b_{0,0,0}$ é igual a zero. Observa-se também que o grau do polinômio do numerador pode ser diferente ou igual ao grau do polinômio do denominador. Observa-se que fazendo $y_d(t)$ igual a zero na Equação (II-5) o modelo NARMAX polinomial constitui-se de um caso particular do modelo NARMAX racional.

Segundo ZHU [14], a vantagem no uso do modelo racional reside no fato de que na medida em que o polinômio $y_d(t)$ se aproxima de -1, a saída $y(t)$ cresce rapidamente, aumentando conseqüentemente a capacidade do modelo para capturar grandes saltos na resposta de problemas não-lineares.

Para exemplificar o modelo racional serão considerados dois atrasos na resposta e um polinômio de grau 2 no numerador e 1 no denominador. Desta forma o polinômio do numerador $y_n(t)$ tem o aspecto apresentado no item anterior. Fazendo o mesmo para o denominador, porém com o polinômio de grau 1 tem-se a seguinte expressão:

$$y_d(t) = b_{0,1,0} \cdot x(t-0 \cdot \Delta t) + b_{1,1,1} \cdot x(t-1 \cdot \Delta t) + b_{2,1,2} \cdot x(t-2 \cdot \Delta t) \quad (\text{II-8})$$

Para este mesmo desenvolvimento e usando o armazenamento sob a forma vetorial (já descrito no modelo polinomial) temos os termos expressos na Tabela II-2 para o polinômio do denominador, que matematicamente pode ser expresso por:

$$y_d(t) = \mathbf{B} \cdot \mathbf{Z}_d(t) \quad (\text{II-9})$$

Índice do Vetor	Componentes	
	Vetor \mathbf{B}	Vetor $\mathbf{Z}_d(t)$
0	b_0	$x(t-0 \cdot \Delta t)$
1	b_1	$x(t-1 \cdot \Delta t)$
2	b_2	$x(t-2 \cdot \Delta t)$

Tabela II-2 – Modelo racional NARMAX sob a forma de vetor

Utilizando as definições apresentadas anteriormente o modelo NARMAX racional pode ser re-escrito como:

$$y(t) = \frac{\mathbf{A} \cdot \mathbf{Z}_n(t)}{1 + \mathbf{B} \cdot \mathbf{Z}_d(t)} \quad (\text{II-10})$$

CAPÍTULO III - REDES NEURAIS ARTIFICIAIS

A origem da teoria de **Redes Neurais Artificiais (RNA)** remonta aos modelos matemáticos e aos modelos de Engenharia que tentam representar o funcionamento de neurônios biológicos [2]. As RNAs são uma representação computacional de um sistema de processamento de informação com desempenho e características semelhantes às redes neurais biológicas. A base da interconexão das RNAs são os elementos matemáticos denominados neurônios artificiais ou simplesmente neurônios [9].

III.1 O neurônio biológico

Podemos entender o neurônio biológico como sendo o dispositivo básico do sistema nervoso, que possui entradas (muitas), e uma única saída. As entradas ocorrem através das conexões sinápticas, que conectam os dendritos aos axônios (Figura III-1) de outras células nervosas. Os sinais que chegam por estes axônios são pulsos elétricos conhecidos como impulsos nervosos, e constituem a informação que o neurônio processará, de alguma forma, para produzir como saída um impulso nervoso no seu axônio [2].

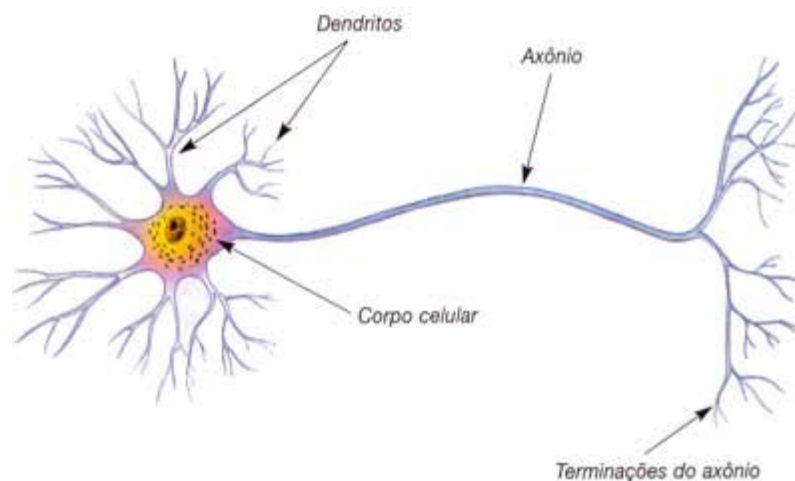


Figura III-1 – Neurônio biológico

III.2 O neurônio artificial

O neurônio artificial (Figura III-2) tem funcionamento análogo ao biológico. Ele recebe um vetor de entrada \mathbf{X} , ou estímulo, que em seguida é multiplicado pelo vetor de pesos sinápticos \mathbf{w} (o cálculo dos pesos sinápticos é descrito no capítulo seguinte). Em seguida o resultado $v = \mathbf{X} \mathbf{w}$ é processado pela função de ativação $g(v)$. O resultado deste processamento u é transmitido na saída do neurônio. O trabalho realizado pelo neurônio artificial é expresso matematicamente por:

$$u = g\left(\sum_{i=1}^N w_i \cdot x_i\right) \quad (\text{III-1})$$

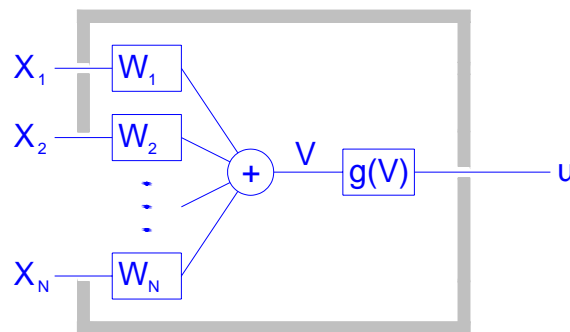


Figura III-2 – Neurônio artificial

Além disto, um componente de entrada adicional b [3], chamado de bias, é comumente inserido no modelo com o intuito de aumentar ou diminuir (se positivo ou negativo) o valor de entrada para a função de ativação [3]. Isto pode fazer com que o argumento da função de ativação caia dentro ou fora da faixa dinâmica (ver item seguinte). Desta forma, a Equação (III-1) é re-escrita como:

$$u = g\left(b + \sum_{i=1}^N w_i \cdot x_i\right) \quad (\text{III-2})$$

III.2.1 Funções de ativação

A função $g(v)$, referida como função de ativação do neurônio, possui algumas características. Entre estas podemos citar o comportamento monotônico sobre uma faixa

determinada de valores do argumento v , conhecida como faixa dinâmica, e a saturação fora desta faixa. Por saturação entende-se que para valores de v fora da faixa dinâmica, a função assume valor constante. As figuras seguintes esclarecem melhor. A literatura [2],[3] sobre redes neurais apresenta uma variedade de funções que procuram preservar estas características com maior ou menor fidelidade. A seguir são comentadas as mais comuns que são: a linear, a tangente hiperbólica e a função sigmóide.

A função de ativação mais simples é a linear, apresentada na Figura III-3. Matematicamente esta função pode ser expressa por:

$$g(v) = v \quad (\text{III-3})$$

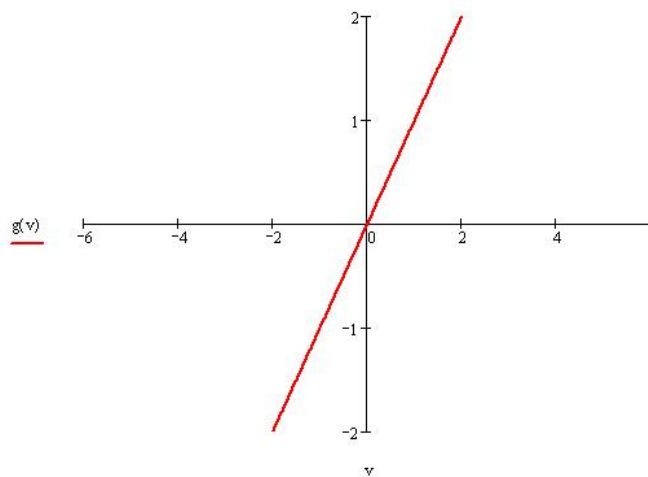


Figura III-3 – Gráfico da função linear

No caso de uma função de ativação linear somente o comportamento monotônico é preservado, desprezando-se a saturação.

A função de ativação do tipo tangente hiperbólica é mostrada na Figura III-4. Matematicamente esta função é representada por:

$$g(v) = \tanh(v) = \frac{1 - \exp(-v)}{1 + \exp(-v)} \quad (\text{III-4})$$

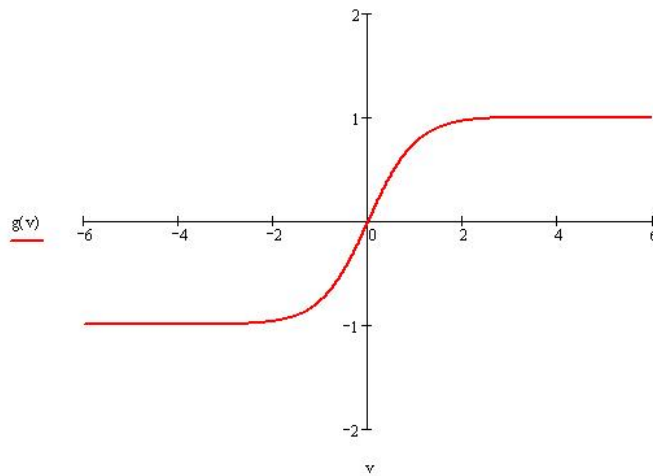


Figura III-4 – Gráfico da tangente hiperbólica

A função de ativação do tipo sigmóide é mostrada na Figura III-5 e matematicamente é expressa por:

$$g(v) = (1 + \exp(-v))^{-1} \quad (\text{III-5})$$

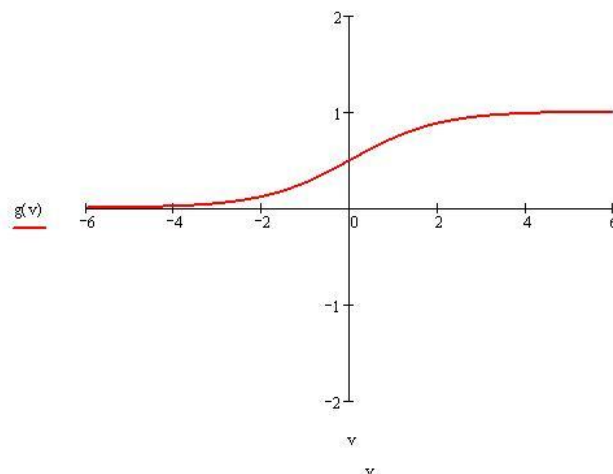


Figura III-5 – Gráfico da sigmóide

Tanto a tangente hiperbólica quanto a função sigmóide são funções de ativação que preservam as características de monotonicidade e saturação.

É interessante observar que o desenvolvimento em série [13] da tangente hiperbólica (III-6) e da sigmóide (III-7) apresenta, teoricamente, infinitos termos. Isto nos permite concluir que quando usamos estas funções de ativação, as RNAs na verdade são polinômios com infinitos termos. Já a função de ativação linear possui um número finito de termos, se assemelhando ao modelo polinomial NARMAX.

$$g(v) = \tanh(v) = v - \frac{v^3}{3} + \frac{2 \cdot v^5}{15} - \frac{17 \cdot v^7}{315} + \dots \quad (\text{III-6})$$

$$g(v) = (1 + \exp(-v))^{-1} = \frac{1}{2} + \frac{v}{4} - \frac{v^3}{48} + \frac{v^5}{480} - \frac{17 \cdot v^7}{80640} + \dots \quad (\text{III-7})$$

III.3 Redes multicamadas

As redes multicamadas ou redes perceptron foram criadas por ROSENBLATT [10] na Universidade de Cornell. O seu trabalho foi um prosseguimento do trabalho de MCCULLOCH [11].

O formato de uma rede multicamada perceptron é semelhante ao de um grafo direcionado, onde cada nó (quadrado cinza na Figura III-6) representa um neurônio. Os neurônios estão dispostos em uma ou mais camadas. Os neurônios que recebem como entradas as saídas daqueles da camada de entrada constituem a segunda camada e assim sucessivamente até a camada final que é a chamada camada de saída. As camadas internas que não são nem a entrada e nem a saída são geralmente referidas como camadas ocultas.

Ainda com referência à Figura III-6, uma rede neural multicamada de K camadas, terá como entrada um vetor X de dimensão J_1 . Estas se conectam às entradas dos J_2 neurônios numa primeira camada. As saídas destes, formando as componentes de um vetor U , conectam-se às entradas dos J_3 neurônios seguintes e assim sucessivamente até a camada de saída.

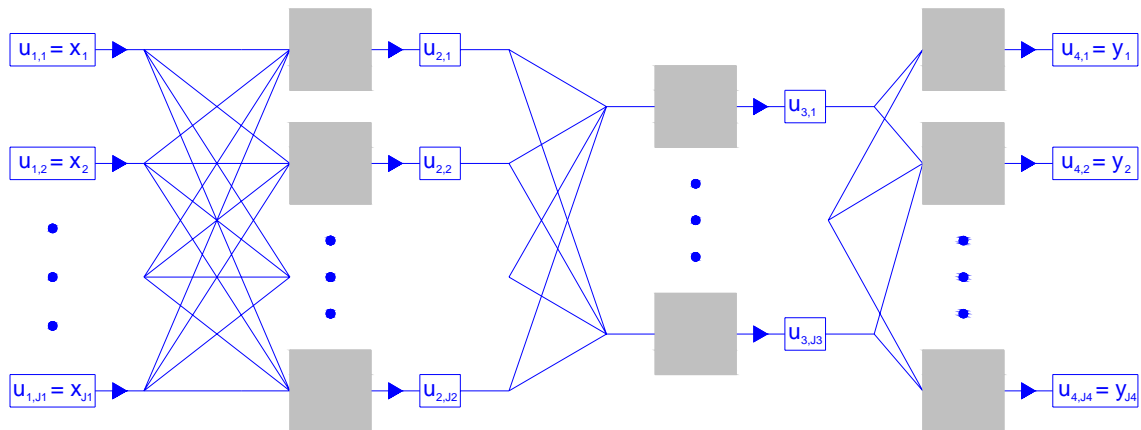


Figura III-6 – Rede multicamadas

A arquitetura de uma rede, que compreende principalmente o número de camadas ocultas e seus respectivos números de neurônios, deve ser definida de acordo com a aplicação em questão.

Especificamente no emprego de RNAs para predição de séries temporais, o vetor de entrada \mathbf{X} deve conter as entradas pertinentes ao fenômeno estudado junto com os respectivos atrasos. A Figura III-7 mostra um exemplo com dois atrasos (resultando em três parâmetros de entrada), uma camada oculta com três neurônios e uma camada de saída com apenas um neurônio. Nesta rede a saída $y(t)$ é escrita como:

$$y(t) = g^s \left(\begin{array}{l} w_{11} \cdot g^i(x(t) \cdot w_1 + x(t - \Delta t) \cdot w_2 + x(t - 2\Delta t) \cdot w_3) + \\ w_{12} \cdot g^i(x(t) \cdot w_4 + x(t - \Delta t) \cdot w_5 + x(t - 2\Delta t) \cdot w_6) + \\ w_{13} \cdot g^i(x(t) \cdot w_7 + x(t - \Delta t) \cdot w_8 + x(t - 2\Delta t) \cdot w_9) \end{array} \right) \quad (\text{III-8})$$

onde $g^s(\cdot)$ é a função de ativação da camada de saída e $g^i(\cdot)$ é a função de ativação utilizada nos neurônios da camada intermediária. Quando a função $g^s(\cdot)$ é a função linear costuma-se dizer que a rede possui saída linear. Observa-se que os únicos termos desconhecidos na Equação (III-8) são os pesos sinápticos \mathbf{w} , cuja obtenção é apresentada no capítulo seguinte.

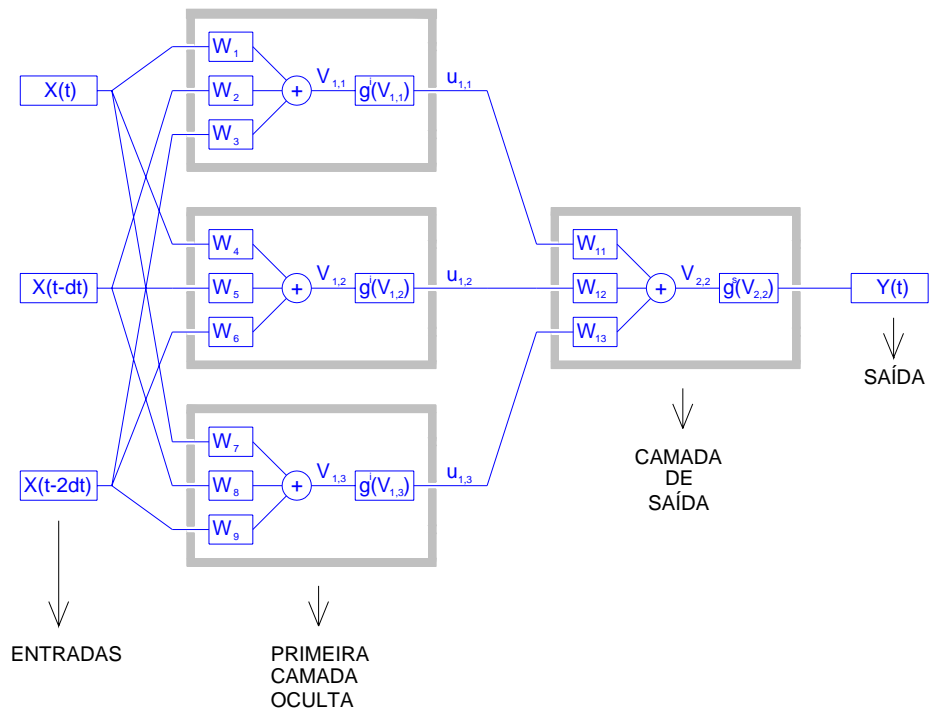


Figura III-7 – RNA com três entradas (dois atrasos) e uma camada oculta

Para uma rede com apenas uma camada oculta a Equação (III-9) fornece o número de pesos sinápticos (np) em função do número de entradas (ne) e do número de neurônios empregados na camada oculta (nn).

$$np = ne \cdot nn + nn \quad (III-9)$$

CAPÍTULO IV - TREINAMENTO

Genericamente, de acordo com o que foi apresentado nos capítulos anteriores, a resposta $y(t)$ no domínio do tempo de um sistema dinâmico pode ser aproximada por:

$$y(t) = f(\mathbf{C}, \mathbf{Z}(t)) \quad (\text{IV-1})$$

onde $f(\cdot)$ é uma função de mapeamento aproximada entre os parâmetros de entrada e a saída, $\mathbf{Z}(t)$ é um vetor conhecido representando as entradas (excitação) no sistema e \mathbf{C} é um vetor de coeficientes constantes e, a priori, desconhecido. No caso do modelo polinomial NARMAX \mathbf{C} representa os coeficientes do polinômio; em se tratando do modelo racional no vetor \mathbf{C} estão incluídos os coeficientes dos polinômios do numerador e do denominador; no caso das Redes Neurais Artificiais (RNAs) o vetor \mathbf{C} representa todos os pesos sinápticos da mesma.

O tamanho ou dimensão M do vetor \mathbf{C} depende de vários parâmetros e varia de modelo para modelo. Nos modelos NARMAX o grau do(s) polinômio(s) e o número de entradas, que no caso de problemas dinâmicos está relacionada aos atrasos considerados, são os fatores que determinam a dimensão de \mathbf{C} . Nas RNAs além do número de entradas, a dimensão de \mathbf{C} está diretamente ligada à arquitetura da rede, isto é, o número de camadas ocultas e o número de neurônios por camada.

O processo de cálculo do vetor \mathbf{C} é comumente conhecido na literatura como treinamento ou aprendizado. Para proceder com o processo de treinamento tornam-se necessários dados conhecidos de entradas e as respectivas saídas do sistema. Estes dados são conhecidos como amostra de treinamento. Parte da amostra de treinamento é separada para o cálculo do vetor \mathbf{C} propriamente dito e a outra parte restante é usada como um conjunto de teste, i.e., um conjunto onde se verifica o desempenho da aproximação.

O cálculo do vetor \mathbf{C} é feito com o objetivo de obter o menor erro quadrático médio possível no ajuste da Equação (IV-1) aos dados reais do problema. O erro médio quadrático é definido como:

$$E = \frac{1}{2 \cdot N} \cdot \sum_{i=1}^N (\bar{y}_i - y_i)^2 \quad (\text{IV-2})$$

onde N é a dimensão do conjunto de dados de treinamento, \bar{y}_i corresponde ao valor verdadeiro da resposta do sistema para a i -ésima amostra do conjunto de treinamento e y_i é o correspondente valor da resposta calculado através da Equação (IV-1). Esta Equação pode ser re-escrita como:

$$E = \frac{1}{2 \cdot N} \cdot \sum_{i=1}^N [\bar{y}_i - f(\mathbf{C}, \mathbf{Z}_i)]^2 \quad (\text{IV-3})$$

A partir do que foi exposto acima se observa que o vetor \mathbf{C} , solução para a Equação (IV-1), corresponde ao vetor que minimiza o erro médio quadrático definido na Equação (IV-3). Em outras palavras, a obtenção do vetor \mathbf{C} corresponde à solução de um problema de otimização sem restrições num espaço M -dimensional.

Matematicamente a solução para \mathbf{C} corresponde ao problema de encontrar o valor mínimo da superfície M -dimensional definida pelo erro quadrático médio. Qualquer algoritmo de otimização pode ser empregado para tal finalidade. Mais adiante serão comentados os métodos utilizados neste trabalho: Gradiente Descendente [2], Gradiente Conjugado [3],[4] e Estratégias Evolutivas [16].

IV.1 Vetor gradiente

Alguns métodos de otimização utilizados neste trabalho fazem uso das derivadas do erro quadrático médio, i.e., são métodos de otimização baseados no vetor gradiente da função objetivo. O vetor gradiente é definido por:

$$\mathbf{G} = \left[\begin{array}{cccc} \frac{\partial E}{\partial c_1} & \frac{\partial E}{\partial c_2} & \dots & \frac{\partial E}{\partial c_M} \end{array} \right]^T \quad (\text{IV-4})$$

onde c_1, c_2, \dots, c_M são os componentes do vetor \mathbf{C} .

Em função do que foi exposto acima, a seguir serão comentados alguns aspectos relacionados ao cálculo do vetor gradiente \mathbf{G} tanto para as RNAs bem como para os modelos NARMAX polinomial e racional.

IV.1.1 Modelo NARMAX polinomial

Observando o modelo polinomial NARMAX sob a forma da Equação (II-4) ou da Tabela II-1, e aplicando a regra da cadeia [13], a j -ésima componente do vetor gradiente é definida como:

$$G_j = \frac{\partial E}{\partial c_j} = \frac{\partial E}{\partial a_j} = \frac{1}{N} \cdot \sum_{i=1}^N (\bar{y}_i - \mathbf{AZ}_n) \cdot Z_n^j \quad (\text{IV-5})$$

onde Z_n^j corresponde ao j -ésimo termo do vetor \mathbf{Z}_n avaliado no instante de tempo discreto t_i .

IV.1.2 Modelo NARMAX racional

O modelo NARMAX racional apresenta dois polinômios, conforme a Equação (II-5). Usando a regra da cadeia e a derivada do quociente, os componentes do vetor gradiente referente aos pesos do numerador são dados por:

$$Gn_j = \frac{\partial E}{\partial c_j} = \frac{\partial E}{\partial a_j} = \frac{1}{N} \cdot \sum_{i=1}^N \frac{(\bar{y}_i - \mathbf{AZ}_n) \cdot Z_n^j}{(1 + \mathbf{BZ}_d)} \quad (\text{IV-6})$$

e as derivadas dos termos referentes ao denominador são dadas por:

$$Gd_j = \frac{\partial E}{\partial c_j} = \frac{\partial E}{\partial b_{j-n_n}} = \frac{1}{N} \cdot \sum_{i=1}^N - \frac{(\bar{y}_i - \mathbf{AZ}_n) \cdot \mathbf{AZ}_n \cdot \mathbf{Z}_d^{j-n_n}}{(1 + \mathbf{BZ}_d)^2} \quad (\text{IV-7})$$

onde $j=n_n+n_d$, n_n é o número de coeficientes do polinômio do numerador, n_d é o número de coeficientes do polinômio do denominador e \mathbf{Z}_d^k corresponde ao k -ésimo termo do vetor \mathbf{Z}_d avaliado no instante de tempo discreto t_i . Os demais termos já foram definidos anteriormente.

IV.1.3 Redes Neurais Artificiais

Para facilitar a obtenção do vetor gradiente associado às RNAs é interessante observar que as derivadas do erro quadrático médio em relação a um dado peso sináptico w_j , para toda a amostra de treinamento, pode também ser expressa pela soma das derivadas de cada amostra deste conjunto:

$$\frac{\partial E}{\partial c_j} = \frac{\partial E}{\partial w_j} = \frac{\partial}{\partial w_j} \left(\frac{1}{2 \cdot N} \sum_{j=1}^N (\bar{y}_j - y_j)^2 \right) = \frac{1}{N} \sum_{j=1}^N \frac{\partial E_j}{\partial w_j} \quad (\text{IV-8})$$

onde $E_j = \frac{1}{2} \cdot (\bar{y}_j - y_j)^2$ é a função dependente dos pesos sinápticos definida genericamente na Equação (IV-5). Os demais parâmetros estão definidos nas seções anteriores.

HAYKIN [3], apresenta o desenvolvimento matemático para se chegar nas expressões para o cálculo das derivadas que aqui são somente apresentadas.

IV.1.3.1 Derivadas dos pesos ligados ao neurônio de saída

Observando a Equação (III-1) e aplicando a regra da cadeia é possível concluir que a derivada do erro E_j em relação a um peso w_i pode ser escrita como:

$$\frac{\partial E_j}{\partial w_i} = \frac{\partial E_j}{\partial y_j} \cdot \frac{\partial y_j}{\partial v} \cdot \frac{\partial v}{\partial w_i} \quad (\text{IV-9})$$

onde

- Derivada do erro em relação a saída do neurônio:

$$\frac{\partial E_j}{\partial y_j} = \bar{y}_j - y_j \quad (\text{IV-10})$$

- Derivada da saída do neurônio em relação à soma ponderada (v):

$$\frac{\partial y_k}{\partial v} = \frac{\partial g(v)}{\partial v} \quad (\text{IV-11})$$

- Derivada da soma ponderada em relação ao peso:

$$\frac{\partial v}{\partial w_i} = y_j \quad (\text{IV-12})$$

Quando a rede apresentar saída linear o termo $\frac{\partial y_j}{\partial v}$ é igual a um.

IV.1.3.2 Derivadas dos pesos ligados a um neurônio oculto

Podemos observar na Figura III-7 que qualquer que seja o peso sináptico w_i ele interfere na saída y_j da rede. A derivada do erro E_j para uma amostra j qualquer do conjunto de treinamento em função de um peso sináptico w_i é dada por:

$$\frac{\partial E_j}{\partial w_i} = \frac{\partial E_j}{\partial y_j} \cdot \frac{\partial y_j}{\partial v} \cdot \frac{\partial v}{\partial u_i} \cdot \frac{\partial u_i}{\partial w_i} \quad (\text{IV-13})$$

onde:

$$\frac{\partial E}{\partial y_j} = \bar{y}_j - y_j \quad (\text{IV-14})$$

$$\frac{\partial y_j}{\partial v} = \frac{\partial g(v)}{\partial v} = \frac{\partial g(\sum w_i \cdot u_i)}{\partial v} \quad (\text{IV-15})$$

$$\frac{\partial v}{\partial u_j} = w_j \quad (\text{IV-16})$$

$$\frac{\partial u_i}{\partial w_i} = \frac{\partial (\sum w_i \cdot u_i)}{\partial w_i} = \frac{\partial g(\sum w \cdot u_i) \cdot u_i}{\partial v} \quad (\text{IV-17})$$

Nas expressões acima u_i é a saída do i -ésimo neurônio da rede. Se a rede possuir uma única camada oculta, u_i é a i -ésima variável de entrada.

IV.2 Caso particular: Modelo NARMAX polinomial

A solução do problema de otimização ou o ponto de mínimo associado ao vetor de constantes \mathbf{C} pode ser obtido analiticamente no caso do modelo polinomial NARMAX. No ponto de mínimo o vetor gradiente necessariamente deve ser igual a zero, i.e.

$$\mathbf{G} = \mathbf{0} \quad (\text{IV-18})$$

Como o modelo NARMAX polinomial constitui-se numa equação linear com relação aos coeficientes, a imposição da condição definida pela Equação (IV-18), após algumas manipulações algébricas, conduz ao seguinte sistema linear de equações [7]:

$$\mathbf{D}\mathbf{C} = \mathbf{E} \quad (\text{IV-19})$$

onde os termos das matrizes \mathbf{D} e \mathbf{E} são dados por:

$$D_{i,j} = \frac{\sum_{k=1}^N Z_i^k \cdot Z_j^k}{N} \quad (\text{IV-20})$$

$$E_j = \frac{\sum_{k=1}^N Z_j^k \cdot Y_k}{N} \quad (\text{IV-21})$$

onde Z_j^k corresponde ao j -ésimo termo do vetor \mathbf{Z}_n expresso na Equação (II-4), avaliado no tempo discreto correspondente ao k -ésimo ponto da amostra de treinamento, e Y_k é a correspondente saída verdadeira do sistema dinâmico. Desta forma os coeficientes \mathbf{C} podem ser calculados pela seguinte expressão:

$$\mathbf{C} = \mathbf{D}^{-1}\mathbf{E} \quad (\text{IV-22})$$

A matriz \mathbf{D} normalmente constitui-se de uma matriz mal-condicionada onde na maioria das vezes, principalmente com o aumento do grau do polinômio, os termos fora da diagonal são muito maiores que os da diagonal. Este aspecto faz com que a inversão desta matriz se torne difícil. Em função disto é necessário recorrer a algoritmos robustos para soluções de sistemas lineares, tais como o Single Value Decomposition Method (SVD) [15] e outros. No presente trabalho foi implementado o SVD.

Muitas vezes um grau elevado do polinômio e a quantidade de atrasos necessária para a representação adequada do comportamento dinâmico levam a um conjunto de coeficientes muito grande. Isto eleva muito o custo computacional para inverter a matriz \mathbf{D} . Como a solução apresentada na Equação (IV-22) constitui-se de um caso particular, os demais métodos apresentados a seguir também podem ser empregados para obter os coeficientes \mathbf{C} associados ao modelo NARMAX polinomial.

IV.3 Método do Gradiente Descendente

O método do Gradiente Descendente, também comumente chamado de *Steepest Descent*, é um método tradicional para solução de problemas de otimização sem restrição. Na literatura sobre Redes Neurais [2],[3] o mesmo é conhecido como algoritmo *backpropagation* ou método de retropropagação do erro.

Neste método parte-se de um valor inicialmente arbitrário para o vetor \mathbf{C} . A partir daí calcula-se o vetor gradiente e caminha-se iterativamente em direção contrária ao mesmo. Deste modo o deslocamento da solução vai à direção do ponto de mínimo mais próximo. Matematicamente o método do Gradiente Descendente é dado por:

$$\mathbf{C}_{j+1} = \mathbf{C}_j - \eta \cdot \mathbf{G}_j \quad (\text{IV-23})$$

onde η é a chamada taxa de aprendizagem, \mathbf{C}_j e \mathbf{C}_{j+1} são o vetor solução na iteração atual e na próxima, respectivamente, e \mathbf{G}_j corresponde ao vetor gradiente calculado na iteração atual de acordo com o que foi exposto no item IV.1 .

A Tabela IV-1 ilustra este método sob a forma de algoritmo. O algoritmo deve ser interrompido quando for atingido um número máximo de iterações e/ou um valor aceitável pré-estabelecido para o erro quadrático médio for alcançado.

ETAPA	PROCEDIMENTO
1	$j = 0$ Use um método de pré-condicionamento ou arbitre um valor inicial para o vetor de pesos \mathbf{C}_0 e arbitre um valor para η .
2	$j = j + 1$ Calcule o vetor gradiente \mathbf{G}_j
3	Faça $\mathbf{C}_{j+1} = \mathbf{C}_j - \eta \cdot \mathbf{G}_j$
4	Se o critério de parada não for atendido volte para etapa 2

Tabela IV-1 – Algoritmo do Método do Gradiente Descendente

Este método não apresenta garantia de convergência [2]. Pelo fato de adotar a direção de busca do mínimo como sendo a mesma do vetor gradiente, o custo computacional deste método também se torna elevado. Existem várias variações deste método na literatura que tentam melhorar a sua convergência bem como sua eficiência computacional [2],[4],[14]. Em linhas gerais, este método não é um procedimento robusto de solução.

IV.4 Método do Gradiente Conjugado

O método do gradiente conjugado [3],[4] pertence a uma classe de métodos de otimização de segunda ordem que coletivamente são conhecidos como métodos da direção conjugada. Este método existe sob a forma linear e não-linear [3].

IV.4.1 Forma linear

A seguir será apresentado um resumo do desenvolvimento em sua forma linear. Uma apresentação mais extensa foi feita por HAYKIN [3].

O método em sua forma linear tem um desenvolvimento teórico baseado na minimização da seguinte função:

$$f(\mathbf{X}) = \frac{1}{2} \cdot \mathbf{X}^T \cdot \mathbf{B} \cdot \mathbf{X}^T - \mathbf{F}^T \cdot \mathbf{X}^T + d \quad (\text{IV-24})$$

onde \mathbf{X} e \mathbf{F} são vetores, \mathbf{B} é uma matriz, e d é um escalar.

É possível demonstrar que se a matriz \mathbf{B} é simétrica e positiva definida a função (IV-24) é minimizada por:

$$\mathbf{B} \cdot \mathbf{X} = \mathbf{F} \quad (\text{IV-25})$$

Um conjunto de vetores não nulos $\mathbf{S}_0, \mathbf{S}_1, \dots, \mathbf{S}_n$ é um conjugado de \mathbf{B} se a seguinte condição for satisfeita:

$$\mathbf{S}_i \cdot \mathbf{B} \cdot \mathbf{S}_j^T = 0 \quad (\text{IV-26})$$

para todo i e j tal que $i \neq j$.

Uma importante propriedade dos vetores conjugados de \mathbf{B} é que eles são linearmente independentes. Logo eles formam uma base do espaço vetorial de \mathbf{B} .

Para um dado conjunto de vetores \mathbf{S} o método do gradiente conjugado para minimização da função (IV-24) é definido por:

$$\mathbf{X}_{j+1} = \mathbf{X}_j + \eta_j \cdot \mathbf{S}_j \quad (\text{IV-27})$$

onde \mathbf{X}_0 é um vetor inicial arbitrário e η_j é um escalar definido por (IV-28).

$$f(\mathbf{X}_j + \eta_j \cdot \mathbf{S}_j) = \min(f(\mathbf{X}_j) + \eta_j \cdot \mathbf{S}_j) \quad (\text{IV-28})$$

O procedimento para a escolha de η , de forma a minimizar a função $f(\mathbf{X}_j + \eta_j \cdot \mathbf{S}_j)$ para um dado j , é conhecido como busca em linha (*line search*). Para o método em sua forma linear a taxa de aprendizagem η é dada por:

$$\eta_j = \frac{\mathbf{S}_j^T \cdot \mathbf{B} \cdot \boldsymbol{\varepsilon}_j}{\mathbf{S}_j^T \cdot \mathbf{B} \cdot \mathbf{S}_j} \quad (\text{IV-29})$$

onde $\boldsymbol{\varepsilon}_j$ é o vetor erro definido por:

$$\boldsymbol{\varepsilon}_j = \mathbf{X}_j - \mathbf{X}^* \quad (\text{IV-30})$$

Começando de um ponto arbitrário \mathbf{X}_0 o método garante que converge para um ponto de mínimo \mathbf{X}^* da função (IV-24) em um número finito de iterações.

Como podemos ver o funcionamento deste método implica na disponibilidade de um conjunto de vetores $\mathbf{S}_0, \mathbf{S}_1, \dots, \mathbf{S}_n$ conjugados de \mathbf{B} . Esses vetores podem ser obtidos pelas expressões:

$$\mathbf{R}_j = \mathbf{F} - \mathbf{B} \cdot \mathbf{X}_j \quad (\text{IV-31})$$

$$\mathbf{S}_j = \mathbf{R}_j - \beta_j \cdot \mathbf{S}_{j-1} \quad (\text{IV-32})$$

onde o vetor \mathbf{S}_j é obtido através de uma combinação linear de \mathbf{R}_j e \mathbf{S}_{j-1} onde β_j é um escalar definido por:

$$\beta_j = \frac{\mathbf{S}_j^T \cdot \mathbf{B} \cdot \mathbf{R}_j}{\mathbf{S}_{j-1}^T \cdot \mathbf{B} \cdot \mathbf{S}_{j-1}} \quad (\text{IV-33})$$

O método em sua forma linear presta-se para a solução de sistemas lineares de equações (Equação (IV-25)) pois nestes casos conhecemos a matriz \mathbf{B} e o vetor independente \mathbf{F} . O seu emprego para o cálculo dos pesos dos polinômios NARMAX, e pesos sinápticos nas RNA, é possível através do método na forma não-linear, descrita a seguir.

IV.4.2 Forma não-linear

O método do gradiente conjugado não-linear apresenta o mesmo desenvolvimento que o linear. Entretanto como não conhecemos a matriz \mathbf{B} e o vetor \mathbf{F} , o vetor resíduo é dado por:

$$\mathbf{R}_j = -\mathbf{G}_j \quad (\text{IV-34})$$

O cálculo do escalar β_j deve ser feito a partir da fórmula de Polak-Ribière [3]:

$$\beta_j = \max\left\langle \frac{\mathbf{R}_j \cdot (\mathbf{R}_j - \mathbf{R}_{j-1})}{\mathbf{R}_j^T \cdot \mathbf{R}_j}, 0 \right\rangle \quad (\text{IV-35})$$

O método de busca em linha adotado no método não linear deve minimizar o erro quadrático médio. Basta atualizar o vetor

$$\mathbf{X}_{j+1} = \mathbf{X}_j + \eta \cdot \mathbf{S}_j \quad (\text{IV-36})$$

variando-se o valor de η até que se encontre o erro médio quadrático mínimo.

HAYKIN [3] descreve um mecanismo de busca em linha que se divide em duas etapas:

1. Faça η variar até que se descubra um intervalo no qual o erro quadrático médio seja o mínimo.
2. Quebre o intervalo em subintervalos de comprimentos progressivamente menores até que o mínimo seja encontrado.

Desta forma, o método do gradiente conjugado não linear aplicado ao cálculo do vetor \mathbf{C} , associado às RNAs ou aos modelos NARMAX polinomial e racional, dependendo do caso, pode ser escrito sob a forma de algoritmo na Tabela IV-2.

O algoritmo deve ser interrompido quando for atingido um número máximo de iterações e/ou quando um valor máximo aceitável pré-estabelecido para o erro médio quadrático for alcançado.

O método do gradiente conjugado faz uso de um vetor que dá a direção de busca. Isso faz com que o ponto de mínimo seja encontrado muito mais rápido que no método do gradiente descendente, além da convergência ser garantida [3].

ETAPA	PROCEDIMENTO
1	$j = 0$ Use um método de pré-condicionamento ou arbitre um valor inicial para o vetor de pesos \mathbf{C}_0
2	Calcule o vetor gradiente: \mathbf{G}_0
3	Faça $\mathbf{S}_0 = \mathbf{R}_0 = -\mathbf{G}_0$
4	Use um método de busca em linha para o cálculo de η
5	Atualize o vetor \mathbf{C} : $\mathbf{C}_{j+1} = \mathbf{C}_j + \eta \cdot \mathbf{S}_j$
6	Calcule o novo vetor gradiente \mathbf{G}_{j+1}
7	Faça: $\mathbf{R}_{j+1} = -\mathbf{G}_{j+1}$
8	Use a fórmula de Polak-Ribière para calcular o escalar β : $\beta_j = \max \left\langle \frac{\mathbf{R}_j \cdot (\mathbf{R}_j - \mathbf{R}_{j-1})}{\mathbf{R}_j^T \cdot \mathbf{R}_j}, 0 \right\rangle$
9	Atualize o vetor direção \mathbf{S} : $\mathbf{S}_{j+1} = \mathbf{R}_{j+1} + \beta_j \cdot \mathbf{S}_j$
10	$j = j + 1$ Se o critério de parada não for atendido volte para a etapa 4.

Tabela IV-2 – Algoritmo do Método do Gradiente Conjugado não linear

IV.5 Estratégias Evolutivas

As Estratégias Evolutivas [16] pertencem a uma classe de algoritmos denominados de Programas Evolutivos. Os programas evolutivos reúnem todos os algoritmos de otimização baseados na teoria da evolução natural das espécies desenvolvida por Darwin.

Os Programas Evolutivos são algoritmos probabilísticos que a cada iteração geram uma nova população de indivíduos. Cada indivíduo representa uma possível solução para o problema de otimização.

As Estratégias Evolutivas fazem uso de uma população de um indivíduo \mathbf{X}_i que é submetido à seguinte operação de mutação para a geração de um descendente \mathbf{X}_{i+1} :

$$\mathbf{X}_{i+1} = \mathbf{X}_i + \mathbf{N}(0, \sigma) \quad (\text{IV-37})$$

onde $N(0, \sigma)$ é um vetor de números aleatórios estatisticamente independentes, gerados a partir de uma distribuição Normal com média 0 e desvio padrão σ .

O novo indivíduo X_{i+1} somente é aceito se satisfaz às restrições do problema de otimização e oferece resultados melhores que seu progenitor X_i . O algoritmo de Estratégias Evolutivas empregado neste trabalho é resumido na Tabela IV-3.

ETAPA	PROCEDIMENTO
1	$j = 0$ Use um método de pré-condicionamento ou arbitre um valor inicial para o vetor C_0
2	Estabeleça um intervalo de busca do ponto de mínimo para cada componente do vetor C .
3	Estabeleça um desvio padrão para a geração de números aleatórios. Este desvio padrão deve ser aproximadamente 1/3 do intervalo estabelecido na etapa 2
4	Faça $C_{i+1} = C_i + N(0, \sigma)$
5	Se C_{i+1} estiver fora do intervalo estabelecido na etapa 2, volte para a etapa 4
6	Se o Erro quadrático médio calculado com C_{i+1} for menor que aquele calculado com C_i então C_{i+1} é aceito. Caso contrário C_i é mantido
7	Se um número pré-estabelecido de iterações for atingido pare. Caso contrário volte para a etapa 4.

Tabela IV-3 – Algoritmo de Estratégias Evolutivas

Fazendo uso das Estratégias Evolutivas para o cálculo dos pesos, observamos que para encontrar o ponto onde o erro é mínimo é necessário um número de iterações suficientemente grande. Isto nos leva a um custo computacional razoável. Entretanto, este algoritmo pode ser utilizado inicialmente para mapear aproximadamente a posição do ponto ótimo e então passar esta informação a um método baseado no vetor gradiente, que irá obter a posição correta do referido ponto.

IV.6 Problemas no treinamento

Como em qualquer problema de otimização, durante o processo de cálculo do vetor de coeficientes C podem surgir alguns problemas que serão discutidos a seguir.

IV.6.1 O problema do Mínimo local

Os métodos do Gradiente Conjugado e do Gradiente Descendente fazem uso de um ponto de partida arbitrado no espaço para que o algoritmo de busca seja iniciado. Acontece que o algoritmo converge (a convergência só é garantida no Gradiente Conjugado [3]) para o ponto de mínimo mais próximo do ponto arbitrado inicialmente. Isso faz com que exista a possibilidade de se convergir para um mínimo local ao invés de um mínimo global, que seria o desejado, como mostra a Figura IV-1. Sendo assim a escolha de um bom ponto de partida (pré-condicionamento) pode melhorar muito o desempenho do método.

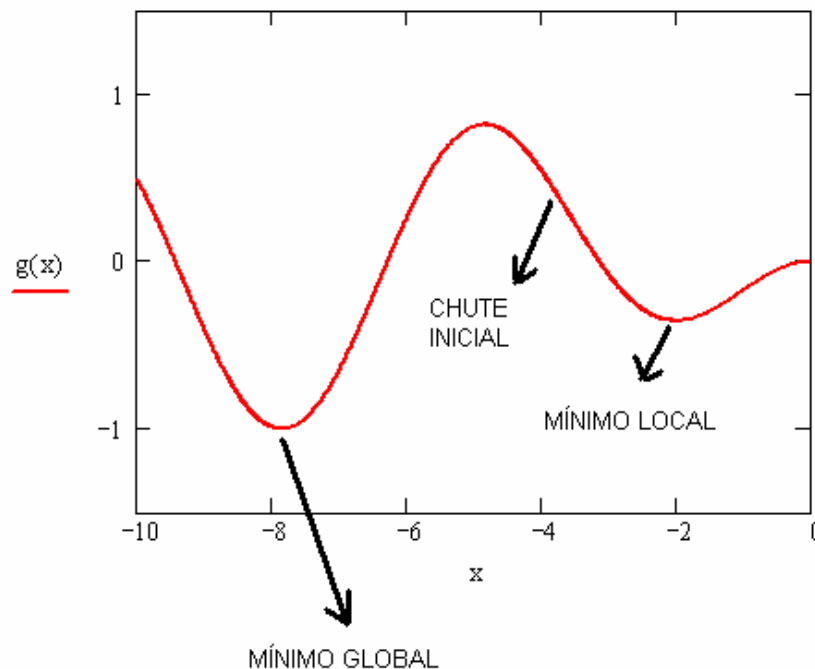


Figura IV-1 – O problema do mínimo local

O modelo NARMAX polinomial, em função da sua forma linear nos parâmetros, apresenta uma única solução analítica que corresponde ao mínimo global do caso, conforme comentado no item IV.2 . Entretanto este método apresenta outras limitações já discutidas.

O método de busca baseado nas Estratégias Evolutivas, devido ao seu caráter randômico, cobre vários pontos no espaço em que se está fazendo a busca (desde que seja empregado um valor adequado para o desvio padrão e um número suficiente de

simulações). Utilizando este método observamos que o ponto de mínimo global pode, em princípio, sempre ser encontrado. Caso haja dúvidas se o ponto encontrado é de fato o ponto de mínimo global, deve-se repetir a busca com valores maiores do desvio padrão e do número de simulações. O problema é que um número adequado de simulações pode muitas vezes nos levar a um custo computacional elevado.

IV.6.1.1 Metodologia de busca proposta

Cada um dos métodos de treinamento (otimização) apresentados possui um ou mais problemas envolvidos. Durante a execução deste trabalho constatamos que é possível montar um esquema de cálculo que tira partido das melhores características disponíveis em cada um dos métodos, de modo que o problema de convergência para um mínimo local tenha menor chance de vir a ocorrer.

É importante mencionar que chegamos nesse esquema de cálculo a partir da experiência e da sensibilidade acumuladas ao longo deste trabalho, portanto não existe uma apresentação formal para o esquema adotado. Também não existe a garantia de que um ponto de mínimo global seja atingido. Verifica-se que a sequência de cálculo sugerida, na maioria dos casos, chega a um erro menor do que se fosse empregado apenas um dos métodos isoladamente. O esquema de cálculo será apresentado a seguir.

MODELO NARMAX POLINOMIAL E RACIONAL

No modelo polinomial NARMAX foi constatado que mesmo usando um número considerável de atrasos, a solução analítica apresentada na seção IV.2 é computacionalmente eficiente somente quando é usado um polinômio do primeiro grau.

Sendo assim adotou-se o procedimento de inicialmente calcular os coeficientes como se o polinômio fosse do primeiro grau e zerar os coeficientes restantes. No modelo racional este cálculo é feito apenas para o polinômio do numerador e todos os coeficientes restantes, do denominador e numerador, são zerados. Desta maneira já se inicia o processo de busca com um bom ponto de partida. Em seguida faz-se uso das estratégias evolutivas, com um número reduzido de simulações (de 10 a 100 vezes o número de coeficientes do vetor \mathbf{C}) a fim de se obter uma aproximação do ponto que

forneça o menor erro médio quadrático. Isso faz com que nos aproximemos de um ponto de mínimo global (se já estivermos próximo dele, nada ocorre). Fazemos então uso do Método do Gradiente Conjugado para refinar a busca e finalmente determinar o ponto de mínimo global com a precisão desejada. Este procedimento é resumido na Tabela IV-4.

ETAPA	PROCEDIMENTO
1	Cálculo dos coeficientes para o polinômio de grau 1. Zerar todos os coeficientes restantes.
2	Estratégias Evolutivas com um número de simulações (10 a 100 vezes o número de pesos) reduzido.
3	Método do Gradiente Conjugado para refinar a busca pelo ponto de mínimo

Tabela IV-4 – Seqüência de cálculo proposta, modelo NARMAX

REDES NEURAIAS ARTIFICIAIS

Devido à não-linearidade da função de mapeamento nas RNA um algoritmo de otimização deve ser empregado. Neste procedimento toma-se como ponto de partida C_0 um ponto obtido gerando-se uma seqüência de números aleatórios com distribuição normal de média zero e desvio padrão um. O restante da seqüência de cálculo a ser seguida é a mesma do modelo polinomial NARMAX apresentado anteriormente: Estratégia Evolutiva seguida pelo Método do Gradiente Conjugado conforme mostra a Tabela IV-5.

ETAPA	PROCEDIMENTO
1	Gerar uma seqüência de números aleatórios com distribuição normal padrão e arbitrar como ponto de partida.
2	Estratégias Evolutivas com um número de simulações (10 a 100 vezes o número de pesos) reduzido.
3	Método do Gradiente Conjugado para refinar a busca pelo ponto de mínimo

Tabela IV-5 – Seqüência de cálculo proposta, RNA

IV.6.2 O erro cometido na Extrapolação

Quando calculamos os coeficientes C fazemos uso de um conjunto de pontos iniciais (amostra de treinamento). Desejamos que com esses coeficientes sejamos capazes de extrapolar os resultados para o restante do domínio. Pode acontecer que durante a fase de treinamento uma determinada curva apresente melhor ajuste (menor erro médio quadrático) que outra, porém, quando extrapolamos os resultados não é isso que se verifica.

A Figura IV-2 ilustra este problema. Para um determinado conjunto de pontos foram seleccionados os 4 primeiros pontos para a amostra de treinamento. Em seguida foram calculados os coeficientes para o ajuste de duas curvas: uma reta e uma parábola do segundo grau. Como podemos ver o ajuste da parábola, para a amostra de treinamento, é quase perfeito. Para a reta o erro cometido é maior. Mas quando extrapolamos o resultado não é isso que acontece, na verdade a aproximação por uma reta fornece resultados bem melhores.

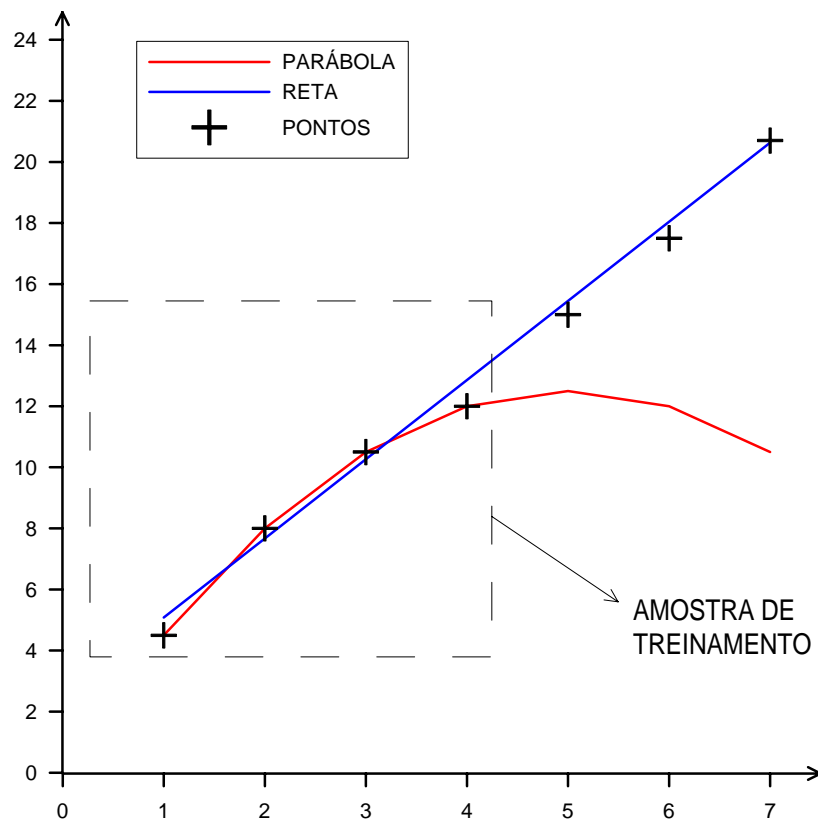


Figura IV-2 – Exemplo de erro na extrapolação

O emprego de um número de pontos razoável para a amostra de treinamento, capaz de representar o fenômeno em questão, faz com que o erro na extrapolação seja evitado. No exemplo da Figura IV-2 se fossem usados os 5 primeiros pontos na amostra de treinamento (em vez de 4) já seria possível verificar (com o erro da amostra de treinamento) que a parábola não era a melhor aproximação.

IV.6.3 O problema do super ajuste ou *overfitting*

O problema de *overfitting* acontece quando para a amostra de treinamento ajustamos um polinômio de grau superior ao que realmente representa o fenômeno físico em questão.

Nas RNA, em especial, o *overfitting* ocorre na medida em que os pesos sinápticos vão sendo calculados. De uma iteração para outra passamos de uma situação que fornece a melhor aproximação para o *overfitting*. Em RNAs isso se deve à complexidade da função de ativação. Tomemos como exemplo a tangente hiperbólica. O seu desenvolvimento em série, teoricamente, nos leva a um polinômio infinito (item III.2.1). Na medida em que os pesos vão sendo calculados isto pode ser amplificado e o *overfitting* ocorre.

Nos modelos NARMAX o *overfitting* ocorre se usarmos uma amostra de treinamento muito reduzida. Uma amostra de treinamento com menos pontos do que o número de coeficientes que devemos calcular poderia nos levar ao *overfitting* no modelo NARMAX.

A solução para o problema do *overfitting* é separar a amostra de treinamento em pontos de cálculo e de teste (ou validação). Os pontos de cálculo, conforme definidos anteriormente, são os que efetivamente são empregados no cálculo do vetor **C**. Já os pontos de teste ou de validação são usados apenas para que se calcule o erro cometido. Quando o erro nos pontos de validação começar a crescer significa que o *overfitting* está ocorrendo e o treinamento deve ser interrompido.

Nos problemas analisados durante o desenvolvimento deste trabalho, verificamos que para dados referentes à análise dinâmica de estruturas, onde se emprega

um intervalo de tempo de análise razoável ($\Delta t \cong 0.5s$), o *overfitting* não ocorre se usarmos uma amostra de treinamento de aproximadamente 300 a 600 pontos. Empregando um intervalo de tempo de análise grande ($\Delta t \cong 10.0s$), e reduzindo o número de pontos na amostra de treinamento o *overfitting* pode ser observado, mas estes dados são incompatíveis com a prática em análise dinâmica de estruturas.

CAPÍTULO V - EXEMPLOS DE APLICAÇÃO

Neste trabalho foram implementados em linguagem *FORTRAN*[®] dois programas, um para os modelos NARMAX e outro para Redes Neurais Artificiais (RNAs). Os programas usam o mesmo número de atrasos (fornecido pelo usuário) para todas as entradas. Os pontos da amostra de treinamento de índice par são tomados como pontos de teste e os de índice ímpar como pontos de cálculo. A metodologia adotada no treinamento é aquela descrita no item IV.6.1.1 .

A seguir são apresentados exemplos de aplicação para validação dos programas desenvolvidos e verificação do desempenho dos métodos de predição de séries temporais na reprodução da resposta dinâmica de estruturas.

O primeiro exemplo é um sistema massa-mola-amortecedor com apenas um grau de liberdade. Os parâmetros de rigidez do sistema são alterados para que se verifique o desempenho dos vários modelos na predição da resposta dinâmica do sistema. Dois casos são analisados, um com rigidez linear e outro com rigidez não-linear.

Os demais exemplos envolvem a avaliação dos modelos de predição aplicados à análise de *risers* e linhas de ancoragem. Desta forma, o segundo exemplo contempla a análise da resposta de um *riser* de aço, instalado em um FPSO na configuração conhecida como *lazy wave*. O terceiro exemplo trata da análise de extremos da resposta numa condição ambiental extrema usando uma amostra de valores extremos. Cada ponto da amostra de extremos representa o valor máximo da resposta observado numa realização independente dos movimentos da unidade flutuante. Os modelos de predição são utilizados para obter as séries da resposta a partir unicamente dos movimentos do flutuante. Assim, o terceiro exemplo considera a tração no topo de uma linha de ancoragem conectada a um FPSO instalado em águas profundas.

O quarto exemplo faz um teste do desempenho da metodologia de cálculo proposta no item IV.6.1.1 a partir do esforço FX no topo do *riser* de aço apresentado no exemplo 2.

Nos exemplos que se seguem a resposta obtida a partir dos métodos de predição de séries temporais (resposta simulada) é comparada com aquela obtida a partir de algum outro método disponível na literatura (resposta original). Em cada exemplo será citado o método empregado para obtenção da resposta original.

GUARIZE [9] verificou que usando RNAs com apenas uma camada oculta e saída linear, o número de neurônios tem pequena influência sobre o resultado final da rede. Sendo assim o presente trabalho, em todos os exemplos, irá adotar uma rede com saída linear e dez neurônios em apenas uma camada oculta. A investigação será feita variando a função de ativação. Para os polinômios NARMAX a pesquisa do melhor modelo será feita com os modelos polinomial e racional fazendo variar o grau dos polinômios envolvidos.

É possível observar que o desempenho dos métodos para predição de séries temporais é diretamente influenciado pelo tamanho da amostra de treinamento e número de atrasos, como ficará mais claro adiante. Sendo assim nos exemplos que se seguem foram feitos testes com algumas configurações de número de atrasos e tamanho da amostra de treinamento, para cada configuração de modelo investigada. A comparação entre os modelos empregados é feita tomando-se como parâmetro o erro médio (Equação (V-1)). É importante observar que esta comparação é feita com o intuito de mostrar qual modelo, dentre os analisados, é capaz de reproduzir o fenômeno em análise com o menor erro.

$$E_m = \frac{1}{N} \cdot \sum_{i=1}^N |ro_i - rs_i| \quad (V-1)$$

A Equação (V-1) define o erro médio (quanto se erra em média em cada ponto do sinal), onde N é o número de valores disponíveis no sinal, ro_i é a resposta calculada

a partir de um método disponível (resposta original) e rs_i é a resposta calculada a partir dos métodos de predição de séries temporais (resposta simulada).

Uma vez identificado (através do erro médio) o modelo que oferece a melhor predição para a série temporal analisada (menor erro médio) devemos verificar se a série temporal foi fielmente reproduzida. Para tanto são apresentados os trechos da série temporal onde ocorre a maior diferença entre o valor original e o simulado. Em seguida é apresentado o coeficiente de correlação [1], Equação (V-2), entre o sinal original e o simulado. Um coeficiente de correlação próximo de +1 ou -1 mostra que existe uma forte correlação linear entre os sinais. Observando a Equação (V-2) podemos ver que o coeficiente de correlação depende da média e do desvio padrão dos sinais, portanto esta informação também será mostrada.

$$C_c = \frac{\sum_{i=1}^N (ro_i - \mu_o) \cdot (rs_i - \mu_s)}{N \cdot \sigma_o \cdot \sigma_s} \quad (V-2)$$

A Equação (V-2) define o coeficiente de correlação, onde N é o número de pontos no sinal, μ_o e μ_s são as médias do sinal original e simulado e σ_o e σ_s são os desvios padrões do sinal original e simulado.

V.1 Sistema Dinâmico Massa-mola-amortecedor

Como primeiro exemplo deste trabalho, considera-se o sistema massa-mola-amortecedor com apenas um grau de liberdade apresentado na Figura V-1. São considerados dois casos: (a) um onde a rigidez é linear e outro (b) onde a rigidez do sistema é quadrática. O sistema é submetido genericamente a uma força aleatória gaussiana $f(t)$ diretamente proporcional às elevações da superfície $\eta(t)$ de um estado de mar com altura significativa de onda (H_s) igual a 7.8m e período de cruzamento zero (T_z) igual a 11.8s.

$$f(t) = a \cdot \eta(t) \quad (V-3)$$

Na Equação (V-3), a é o parâmetro de proporcionalidade que será definido mais adiante. O espectro considerado é o de Pierson-Moskovitz, conforme ilustra a Figura V-2.

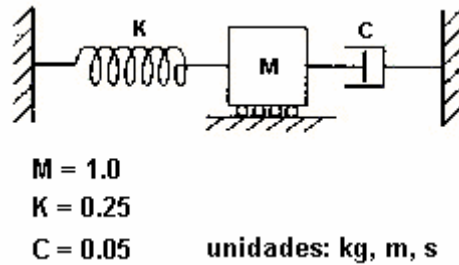


Figura V-1 – Sistema massa-mola-amortecedor

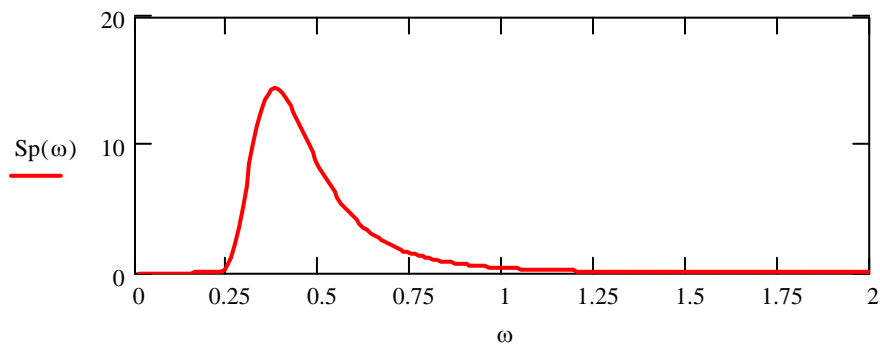


Figura V-2 – Espectro de Pierson-Moskovitz, $H_s = 7.8\text{m}$ e $T_z = 11.8\text{s}$

No caso de rigidez linear a equação de equilíbrio do sistema pode ser dada por:

$$\ddot{x}(t) + 2\xi\omega_n\dot{x}(t) + \omega_n^2x(t) = \frac{a}{m}\eta(t) = a'\eta(t) \quad (\text{V-4})$$

onde $x(t)$, $\dot{x}(t)$ e $\ddot{x}(t)$ são, respectivamente, o deslocamento, a velocidade e aceleração do sistema, $\omega_n = \sqrt{\frac{k}{m}}$ é a frequência natural do sistema, sendo k a rigidez linear da mola e m a massa do sistema, ξ é a proporção do amortecimento c em relação ao amortecimento crítico do sistema $c_c = 2\sqrt{m \cdot k}$.

Neste trabalho para se amplificar a resposta dinâmica do sistema foram utilizados os seguintes parâmetros $\omega_n = 0.5$ e $\xi = 0.05$ de tal forma que o pico da função de transferência do sistema [8], mostrada na Figura V-3, fique dentro da região de maior energia das ondas (vide Figura V-2). O parâmetro a' foi calculado de forma que o desvio padrão de $x(t)$ ficasse unitário, resultando em $a' = 0.031317$.

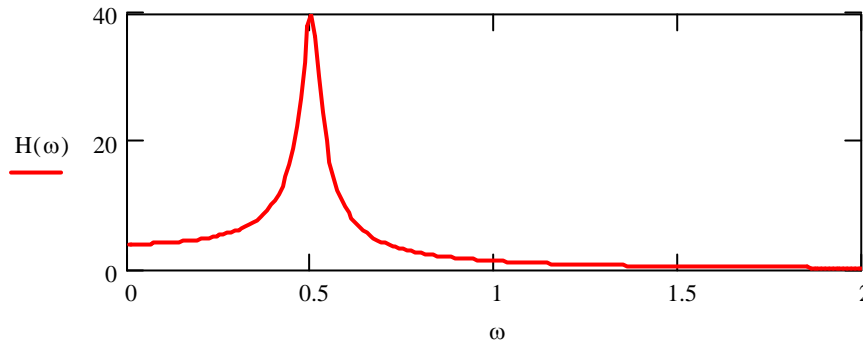


Figura V-3 – Função de transferência do sistema massa-mola-amortecedor

Para obter um modelo não-linear tornando a resposta $x(t)$ do sistema não simétrica e conseqüentemente não gaussiana foi adicionado um termo quadrático à Equação (V-4), i.e.

$$\ddot{x}(t) + 2\xi\omega_n\dot{x}(t) + \omega_n^2x(t) + \kappa x(t)^2 = a'\eta(t) \quad (V-5)$$

sendo o parâmetro κ arbitrado igual a 0.025.

A resposta correta do sistema para cada um dos casos considerados foi calculada com o algoritmo Runge-Kuta de quarta ordem com um intervalo de tempo discreto de 0.5s ($\Delta t = 0.5s$) para um tempo total de simulação de 10800s. É importante mencionar que para o caso linear a resposta poderia ser obtida analiticamente [8]. Para o cálculo de cada caso, a partir dos modelos NARMAX e RNA, foi considerada como entrada do sistema a força de excitação $f(t)$ e como saída o deslocamento $x(t)$.

Uma vez que os parâmetros variáveis dos modelos foram estabelecidos para $x(t)$ eles foram mantidos para avaliar os demais parâmetros de resposta do sistema:

velocidade e aceleração. Foram testados modelos com 130, 140 e 150 atrasos (o que equivale a atrasos de 65s, 70s e 75s) e amostras de treinamento de 200s, 300s e 400s.

Nos modelos NARMAX a predição das séries temporais contemplou o modelo polinomial e o racional sendo avaliados os modelos descritos na Tabela V-1.

Modelo	Descrição
G1	Modelo polinomial do primeiro grau
G2	Modelo polinomial do segundo grau
G1/G2	Modelo racional grau 1 no numerador e grau 2 no denominador
G2/G1	Modelo racional grau 2 no numerador e grau 1 no denominador

Tabela V-1 – Modelos NARMAX analisados

Para as redes neurais foi analisado o desempenho das funções de ativação apresentadas, de acordo com a Tabela V-2.

Modelo	Função
A1	Linear
A2	Tangente Hiperbólica
A3	Sigmóide

Tabela V-2 – Funções de ativação analisadas.

V.1.1 Caso linear

As Figuras V-4 à V-7 descrevem o comportamento do erro médio (vide Equação (V-1)) para todos os modelos investigados, no caso linear, em função do tamanho da janela inicial de treinamento e do número de atrasos (cada atraso corresponde a um intervalo de tempo $\Delta t = 0.5s$).

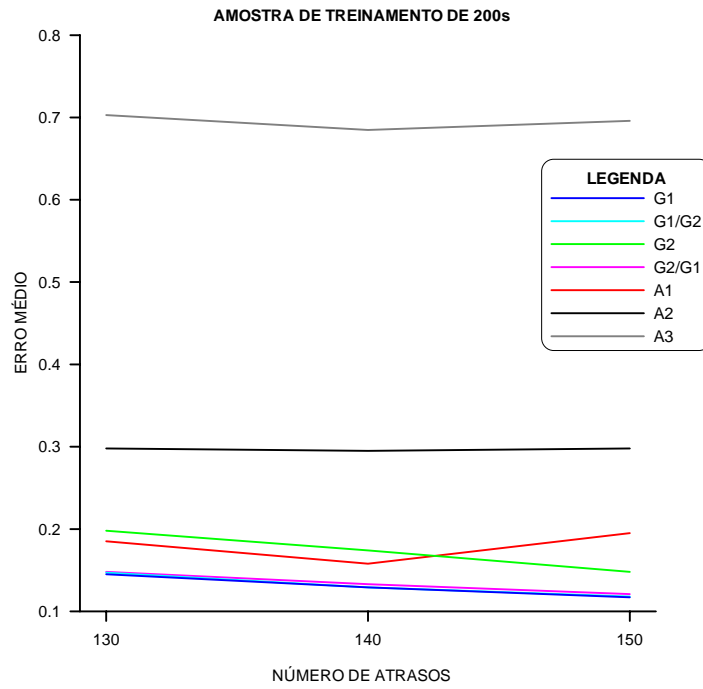


Figura V-4 – Treinamento de 200s, caso linear

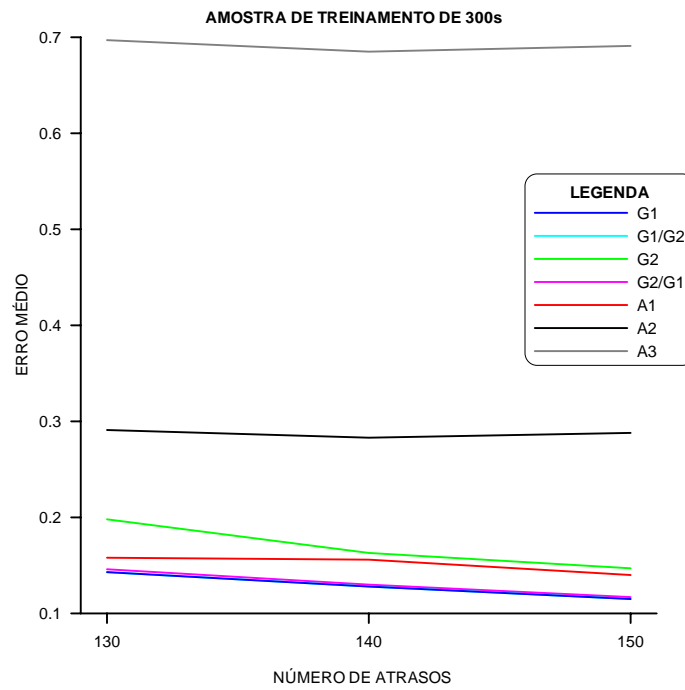


Figura V-5 – Treinamento de 300s, caso linear

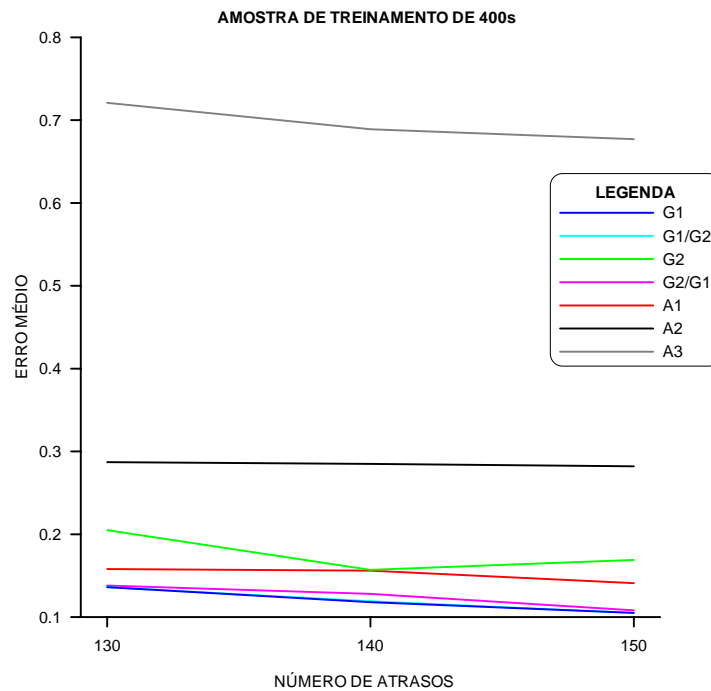


Figura V-6 – Treinamento de 400s, caso linear

É possível observar que o modelo polinomial NARMAX (G1), com amostra de treinamento de 400s e 150 atrasos é o que fornece o menor erro médio (Tabela V-3). Esta configuração de modelo nos leva ao cálculo de 152 coeficientes (ou pesos) na fase de treinamento. É importante observar que as RNA com função de ativação linear também ofereceram um erro médio próximo daquele computado com o modelo NARMAX.

Amostra de treinamento (s)	Erro Médio (m)
200	0.117
300	0.115
400	0.105

Tabela V-3 – Erro médio em função da amostra de treinamento, G1, 150 atrasos

Uma vez estabelecido o modelo que fornece o menor erro médio, i.e., o modelo NARMAX G1, vamos agora investigar se a resposta dinâmica foi reproduzida com fidelidade para o deslocamento e suas derivadas (velocidade e aceleração).

A Tabela V-4 mostra o tempo discreto onde ocorre o maior erro em módulo (diferença entre o valor original e o simulado) para o deslocamento e suas derivadas. As

figuras seguintes mostram o trecho das séries temporais onde esta diferença é verificada.

Resposta	Tempo onde ocorre o maior erro (em módulo)
Deslocamento	1464.0s
Velocidade	1464.0s
Aceleração	1466.5s

Tabela V-4 – Tempo onde ocorre o maior erro em módulo, G1, 150 atrasos.

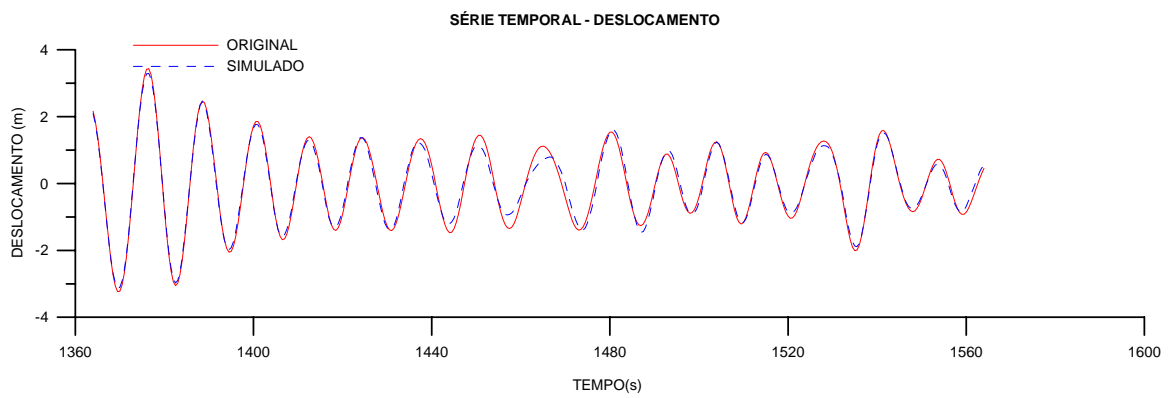


Figura V-7 – Série temporal do deslocamento onde ocorre o maior erro, caso linear.

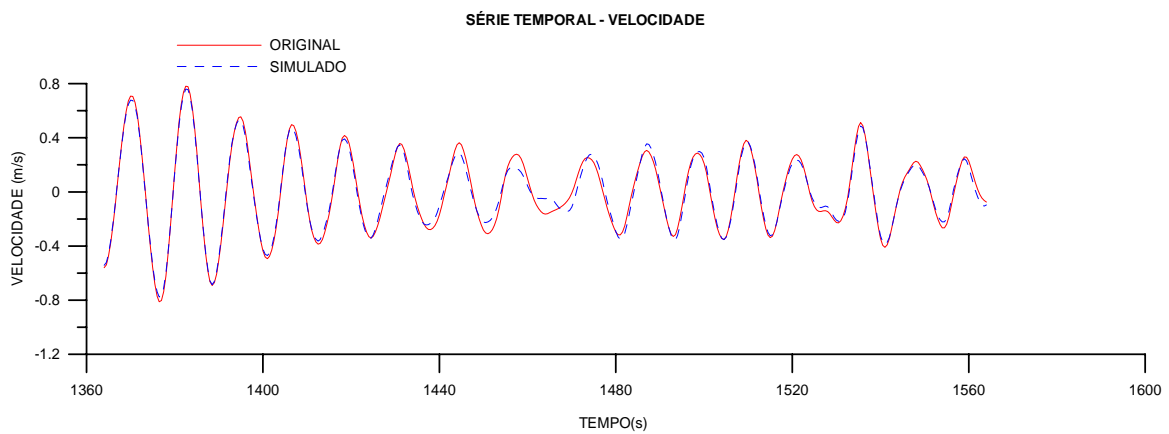


Figura V-8 – Série temporal da velocidade onde ocorre o maior erro, caso linear.

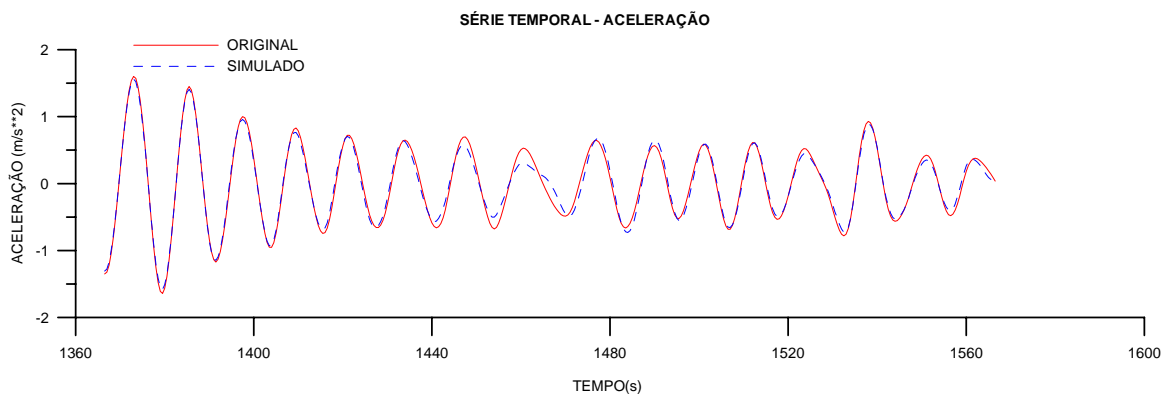


Figura V-9 – Série temporal da aceleração onde ocorre o maior erro, caso linear.

A Tabela V-5 mostra o coeficiente de correlação entre os sinais originais (Runge-Kutta) e os simulados pelo modelo. A Tabela V-6 mostra as médias e os desvios padrões destes sinais.

Resposta	Coeficiente de correlação
Deslocamento	0.992
Velocidade	0.991
Aceleração	0.991

Tabela V-5 – Coeficiente de correlação, G1, 150 atrasos.

Resposta	Média		Desvio Padrão	
	Original	Simulado	Original	Simulado
Deslocamento (m)	0.000	0.000	0.993	0.950
Velocidade (m/s)	0.000	0.000	0.237	0.225
Aceleração (m/s ²)	0.000	0.001	0.479	0.459

Tabela V-6 – Média e Desvio padrão dos sinais original e simulado, G1, 150 atrasos.

Observando as Figuras V-7 à V-9 e a Tabela V-5 é possível constatar que as séries temporais são fielmente reproduzidas.

V.1.2 Caso não-linear

No caso não-linear empregou-se a mesma metodologia de pesquisa adotada no caso linear, como pode ser observado nas figuras e tabelas seguintes. Na Figura V-10 os modelos G1, G1/G2 e G2/G1 apresentaram resultados muito próximos, por isso ocorre sobreposição nos gráficos. O mesmo se verifica na Figura V-11 e Figura V-12 com os modelos G1 e G1/G2.

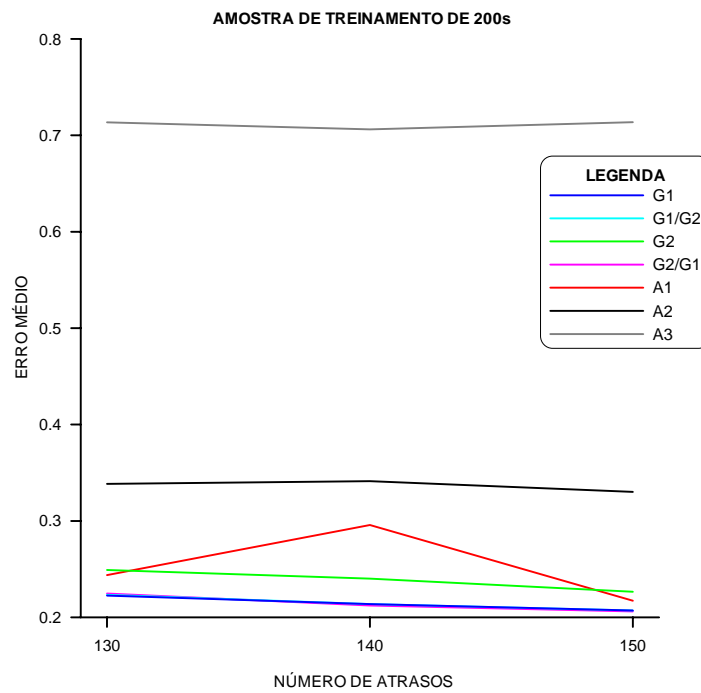


Figura V-10 – Treinamento de 200s, caso não-linear.

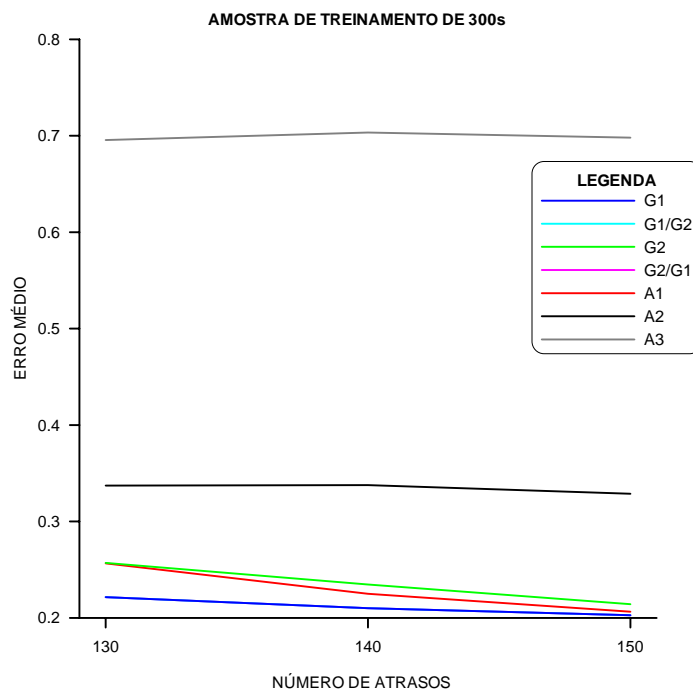


Figura V-11 – Treinamento de 300s, caso não-linear.

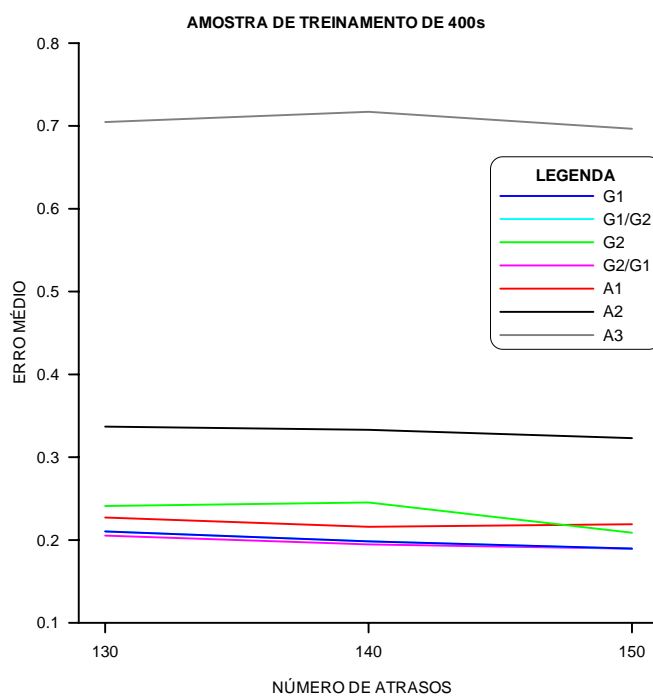


Figura V-12 – Treinamento de 400s, caso não-linear.

É possível observar na Tabela V-7 que o modelo polinomial NARMAX (G2/G1), com amostra de treinamento de 400s e 150 atrasos, é o que fornece o menor erro médio. Esta configuração de modelo nos leva ao cálculo de 11780 coeficientes (ou pesos) na fase de treinamento.

Amostra de treinamento (s)	Erro Médio (m)
200	0.2234
300	0.2216
400	0.1895

Tabela V-7 – Erro médio em função da amostra de treinamento, G2/G1, 150 atrasos.

A Tabela V-8 mostra o tempo discreto onde ocorrem os maiores erros em módulo. As Figuras V-13 à V-15 mostram o trecho das séries temporais onde isso se observa, para o deslocamento e suas derivadas.

Resposta	Tempo onde ocorre o maior erro (em módulo)
Deslocamento	2122.0s
Velocidade	2135.0s
Aceleração	2124.5s

Tabela V-8 – Tempo onde ocorre o maior erro em módulo, G2/G1, 150 atrasos.

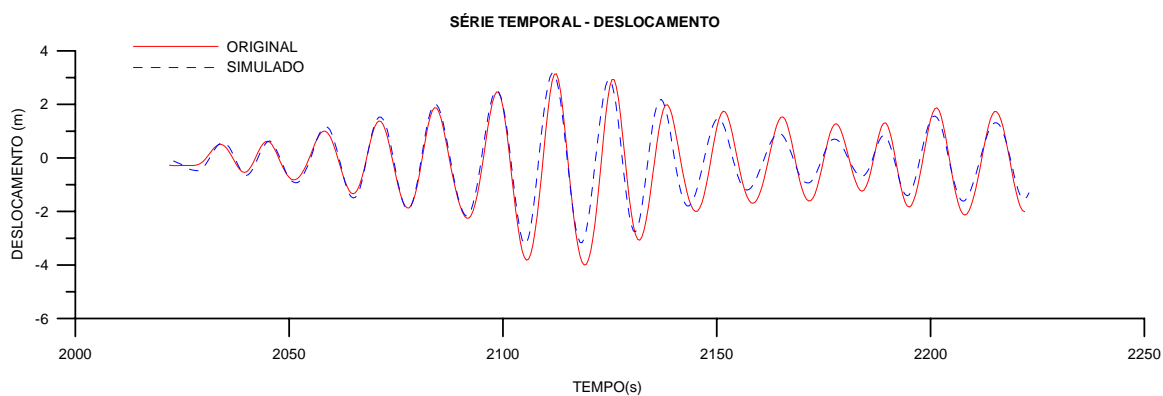


Figura V-13 – Série temporal do deslocamento onde ocorre o maior erro, caso não-linear.

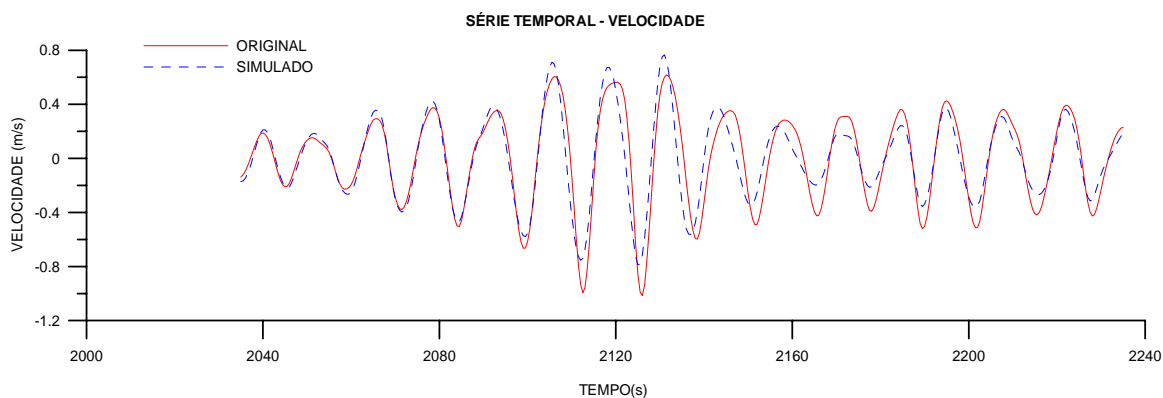


Figura V-14 – Série temporal da velocidade onde ocorre o maior erro, caso não-linear.

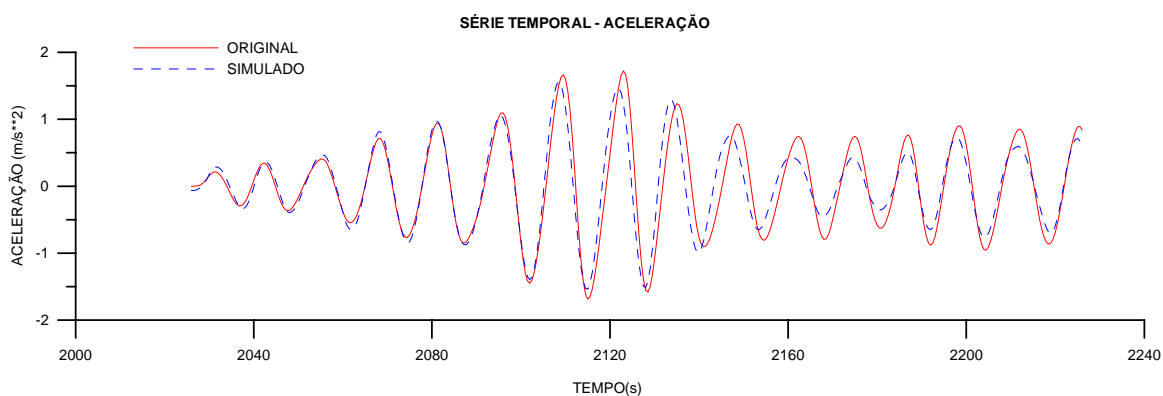


Figura V-15 – Série temporal da aceleração onde ocorre o maior erro, caso não-linear.

A Tabela V-9 mostra o coeficiente de correlação entre os sinais originais (Runge-Kutta) e os simulados pelo modelo. A Tabela V-10 mostra as médias e os desvios padrões destes sinais.

Resposta	Coefficiente de correlação
Deslocamento	0.964
Velocidade	0.960
Aceleração	0.967

Tabela V-9 – Coeficiente de correlação, G2/G1, 150 atrasos.

Resposta	Média		Desvio Padrão	
	Original	Simulado	Original	Simulado
Deslocamento (m)	-0.108	-0.054	1.036	0.959
Velocidade (m/s)	0.000	0.000	0.240	0.224
Aceleração (m/s ²)	0.000	0.003	0.490	0.461

Tabela V-10 – Média e Desvio padrão dos sinais original e simulado, G2/G1, 150 atrasos.

É possível observar que no modelo não-linear as séries temporais são razoavelmente bem reproduzidas.

V.1.3 Comentários

A seguir são apresentados alguns comentários a respeito das análises que foram realizadas.

É interessante observar que na medida em que aumentamos o número de atrasos, para um dado modelo de uma amostra de treinamento, o erro médio pode aumentar (Figura V-10).

Para as RNAs é interessante observar que a função de ativação adotada apresenta grande influência no resultado final. Foi possível constatar que a função de ativação linear apresenta resultados melhores que a tangente hiperbólica e a sigmóide.

Foi possível observar que os métodos de predição foram fiéis na reprodução das séries temporais para o caso linear. No caso não-linear as séries foram reproduzidas, mas a fidelidade não foi a mesma obtida para o caso linear. Essa observação é importante sob o ponto de vista de pesquisa, pois revela que os métodos de predição de séries temporais possuem maior dificuldade para representar o comportamento estrutural na medida em que a não-linearidade vai se tornando mais importante.

V.2 Resposta dinâmica de um *Riser* de aço com geometria *lazy wave*

Este exemplo investiga a aplicabilidade dos métodos para predição de séries temporais na modelagem dinâmica de um caso real. A estrutura em estudo é um *riser* de aço com diâmetro de 18” instalado em FPSO (Floating Production Storage Offloading), conforme mostra a Figura V-16, com um sistema de ancoragem do tipo complacência diferenciada (DICAS).

Este *riser* está instalado em uma lâmina d’água de 1800m com ângulo de topo na saída do conector de 10 graus. A configuração de instalação é do tipo *lazy wave*, apresentada na Figura V-17. Esta configuração geométrica é obtida instalando-se flutuadores em um determinado trecho do *riser*. A configuração em *lazy wave* apresenta a vantagem de suavizar os efeitos dinâmicos na região em que o *riser* toca o solo (TDP – *Touch Down Point*).



Figura V-16 – FPSO

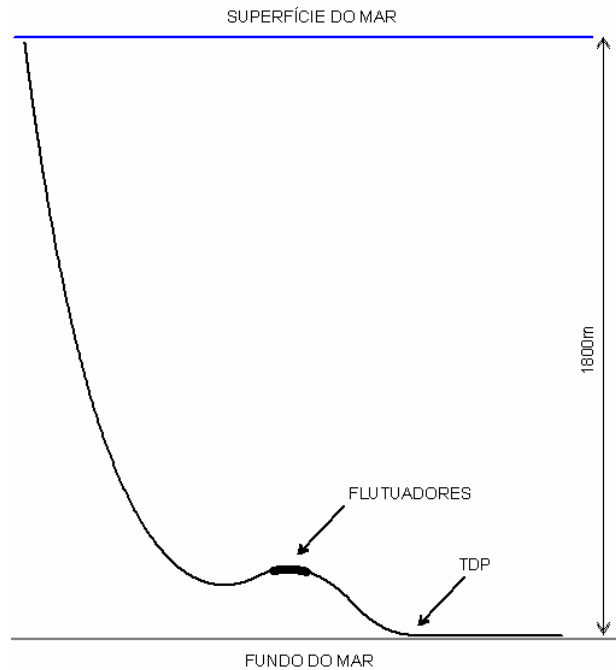


Figura V-17 – Configuração geométrica em *lazy wave*

Consideraram-se como parâmetros de entrada (ou excitação) do sistema os movimentos do FPSO no topo do *riser* nas direções x (surge), y (sway) e z (heave). Os parâmetros de resposta dinâmica considerados são a tração (FX) e o momento y (MY) por serem parâmetros de projeto adotados na prática. A pesquisa é feita em dois pontos distintos do *riser*: um no topo e outro na corcova (vide Figura V-18).

Os modelos de predição foram utilizados para uma condição ambiental com as características ilustradas nas Tabelas V-11 e V-12. A direção de incidência do carregamento ambiental, considerando onda, vento e corrente alinhados, é mostrado na Figura V-19. Inicialmente foram realizadas análises dinâmicas aleatórias de 2500s de duração usando o programa ANFLEX [17] onde os movimentos estáticos e de segunda ordem equivalentes (harmônico equivalente) foram obtidos, para cada condição ambiental, a partir de simulações feitas com o programa DYNASIM [18]. Um resumo destes movimentos é mostrado na Tabela V-13. Os movimentos de primeira ordem foram obtidos pelo cruzamento do RAO da embarcação com o espectro de mar.

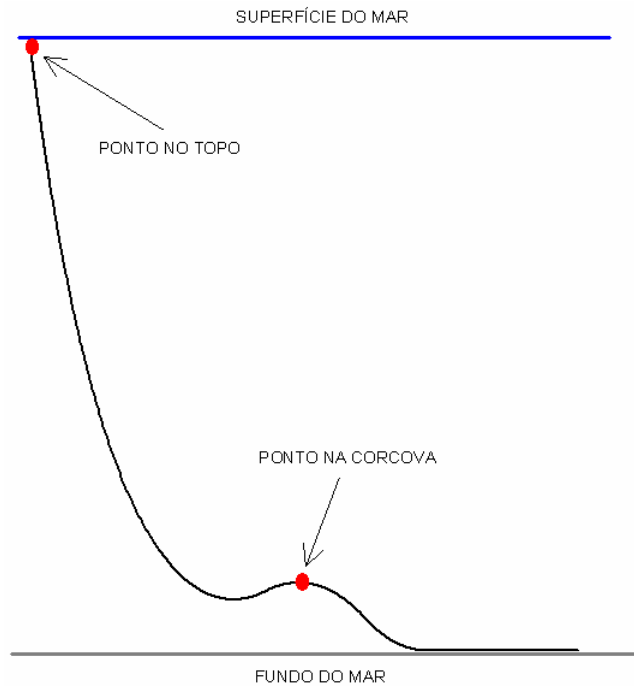


Figura V-18 – Pontos analisados

Corrente (m/s)		Vento	
Vel. Fundo	Vel. Superfície	Velocidade média horária (m/s)	Espectro
0.345	1.617	38.80	Harris

Tabela V-11 – Dados de carregamento - Corrente e Movimento de baixa frequência

Onda (Jonswap)			
α	γ	Hs	Tp
0.0065151	2.569	8.100	13.885

Tabela V-12 – Dados de carregamento de onda

Movimento de baixa frequência (amplitude/período)		Movimento Estático(m)		
(m)/(s)	(m)/(s)	X	Y	Z
4.3877/281.99	9.4373 / 284.4	-87.90	31.342	-8.57

Tabela V-13 – Dados de movimentos estáticos e de segunda ordem

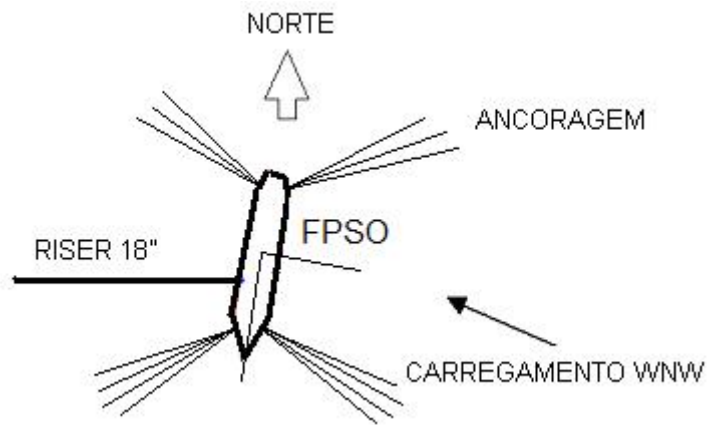


Figura V-19 – Direção do carregamento ambiental

Foram utilizadas séries temporais de 2500s (com intervalo de tempo de 0.5s) de duração onde os primeiros 250s foram desprezados devido ao regime transiente da resposta. As Figuras V-20 à V-22 mostram as séries temporais dos movimentos *surge*, *sway* e *heave* impostos no topo do *riser* para o caso de carregamento analisado.

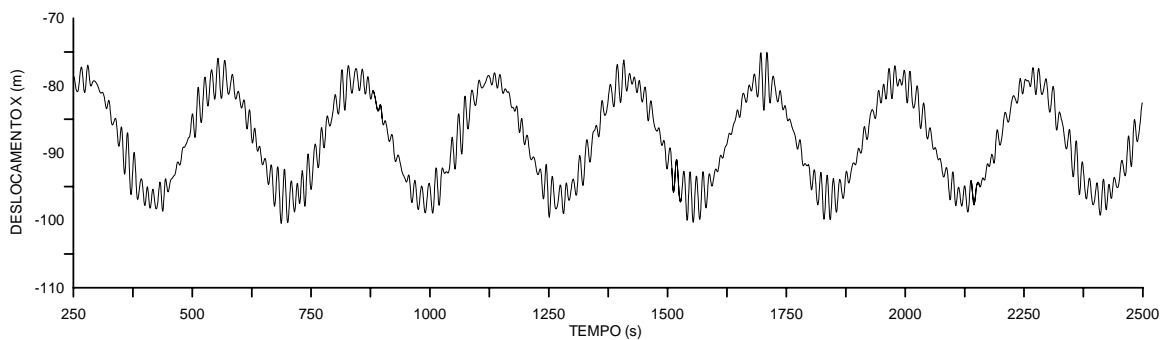


Figura V-20 – deslocamento X (*surge*) imposto no topo

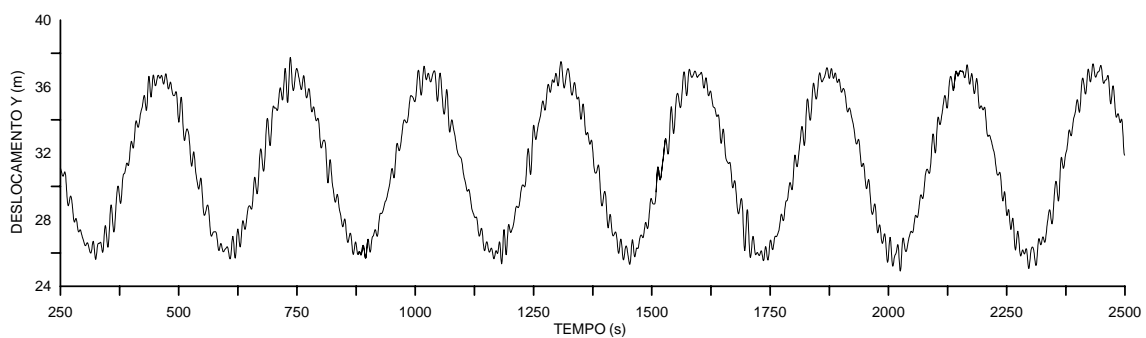


Figura V-21 – deslocamento Y (*sway*) imposto no topo

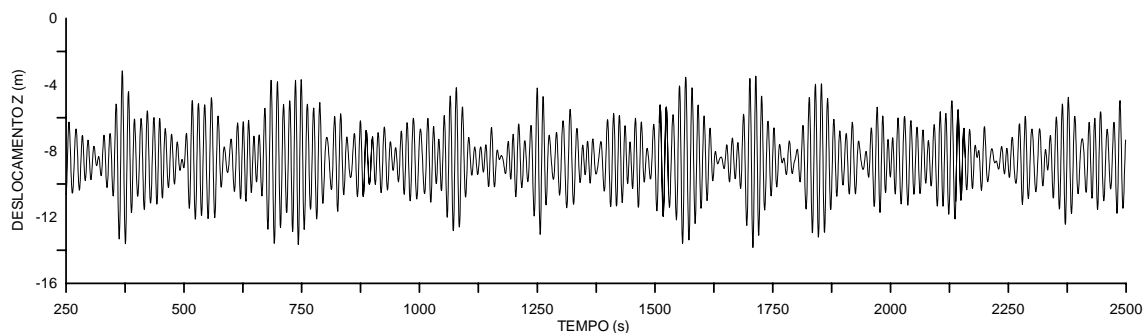


Figura V-22 – deslocamento Z (*heave*) imposto no topo

A seguir são apresentados os resultados obtidos com os métodos de predição de séries temporais investigados neste trabalho onde se variou o tamanho da amostra de treinamento (300s, 400s e 500s) e o número de atrasos (40,50 e 60; o que equivale a 20s, 25s e 30s de atraso). A investigação com os modelos NARMAX contemplou somente o modelo polinomial do primeiro e segundo grau. O modelo racional foi descartado, pois neste caso temos três variáveis de entrada (*surge*, *sway*, *heave*), o que aumenta muito o número de coeficientes a se processar no treinamento, acarretando em maior custo computacional. A Tabela V-14 e a Tabela V-15 mostram de forma resumida os modelos analisados. Para as redes neurais o estudo contempla a função de ativação linear e a tangente hiperbólica.

Modelo	Descrição
G1	Modelo polinomial do primeiro grau
G2	Modelo polinomial do segundo grau

Tabela V-14 – Modelos NARMAX analisados

Modelo	Função
A1	Linear
A2	Tangente Hiperbólica

Tabela V-15 – Redes Neurais, funções de ativação analisadas.

V.2.1 Ponto na Corcova: Esforço MY

As Figuras V-23 a V-25 mostram como o erro médio varia para o esforço MY no ponto da corcova em função do número de atrasos para cada amostra de treinamento estudada.

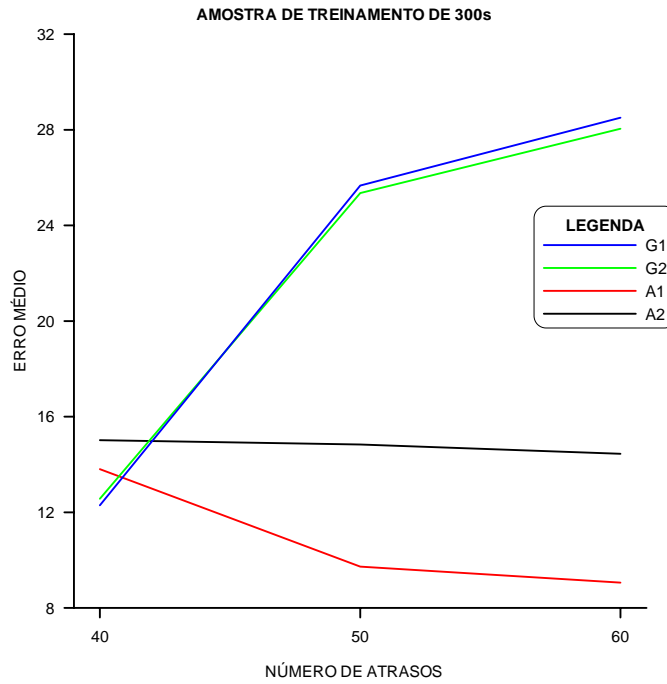


Figura V-23 – Treinamento de 300s, MY corcova.

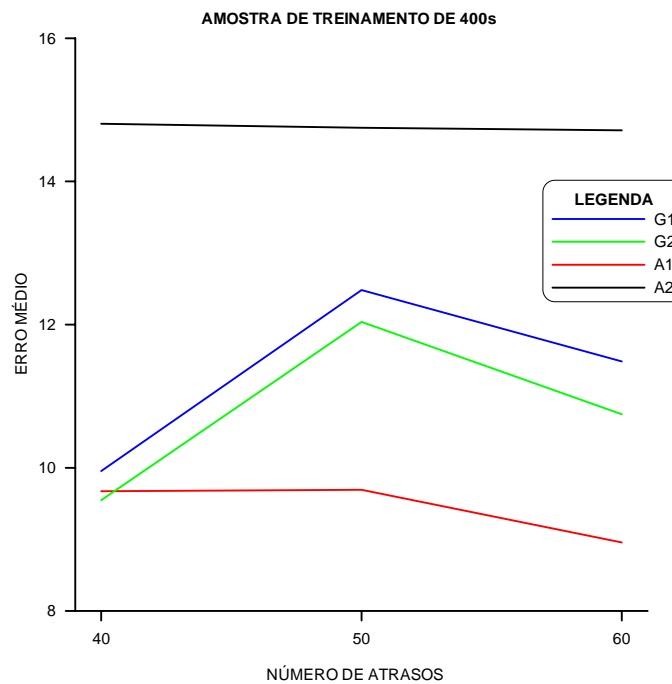


Figura V-24 – Treinamento de 400s, MY corcova.

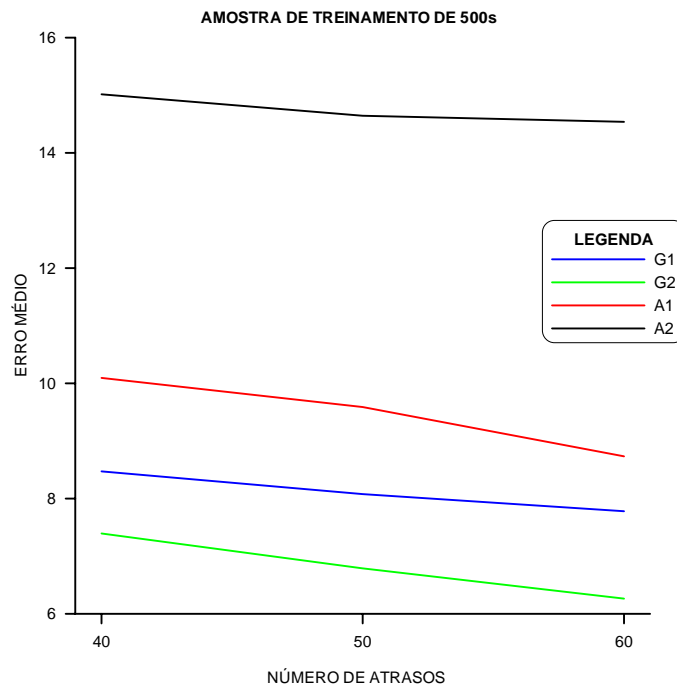


Figura V-25 – Treinamento de 500s, MY corcova.

É possível observar que para as amostras de treinamento de 300s e 400s as RNAs com função de ativação linear forneceram o menor erro médio. Entretanto na amostra de treinamento de 500s foi observado o menor erro médio (dentre as três amostras analisadas) com o modelo polinomial NARMAX (G2) e 60 atrasos (vide Tabela V-16). Esta configuração de modelo nos leva ao cálculo de 17020 coeficientes (ou pesos) na fase de treinamento. Neste modelo, a maior diferença entre o sinal simulado e o original foi registrada no intervalo de tempo de 1466.5s. A Figura V-26 apresenta um trecho da série temporal incluindo este ponto. A Tabela V-17 apresenta um resumo estatístico das séries temporais analisadas.

Amostra de treinamento (s)	Erro Médio (kN·m)
300	28.048
400	10.750
500	6.263

Tabela V-16 – Erro médio em função da amostra de treinamento, G2, 60 atrasos.

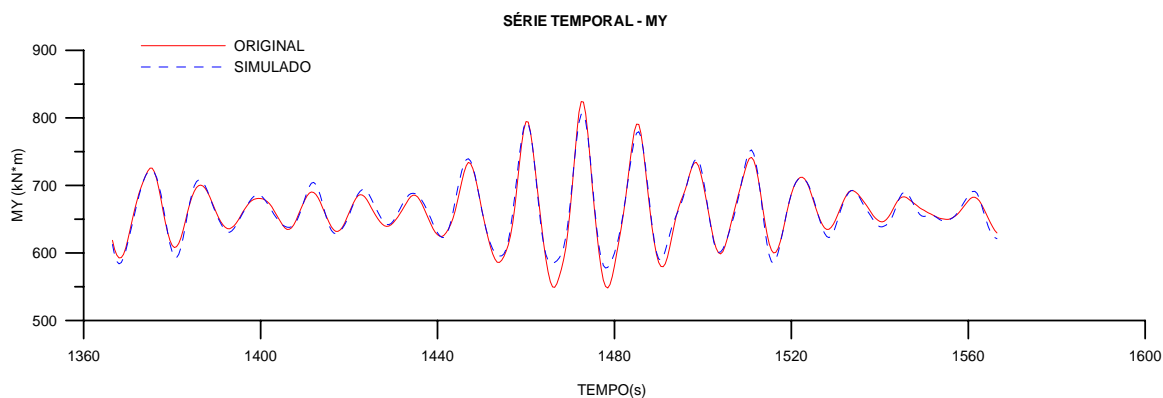


Figura V-26 – Série temporal, MY corcova.

Parâmetro	Sinal	
	Original	Simulado
Média (kN·m)	663.42	664.00
Desvio Padrão (kN·m)	43.06	44.28
Coeficiente de correlação	0.983	

Tabela V-17 – Média, desvio padrão e coeficiente de correlação, G2, 60 atrasos.

Observando a Tabela V-17 e a Figura V-26 podemos dizer que a série temporal foi bem reproduzida. A maior diferença está nas amplitudes dos picos em torno do ponto onde a diferença é máxima.

V.2.2 Ponto no topo: Esforço MY

As Figuras V-27 a V-29 seguintes mostram como o erro médio do esforço MY no topo do *riser* varia em função do número de atrasos para cada amostra de treinamento estudada.

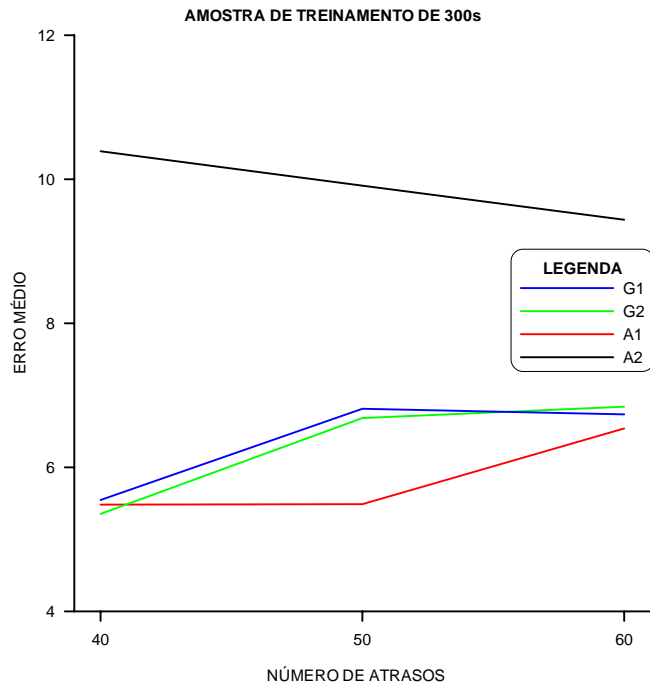


Figura V-27 – Treinamento de 300s, MY topo.

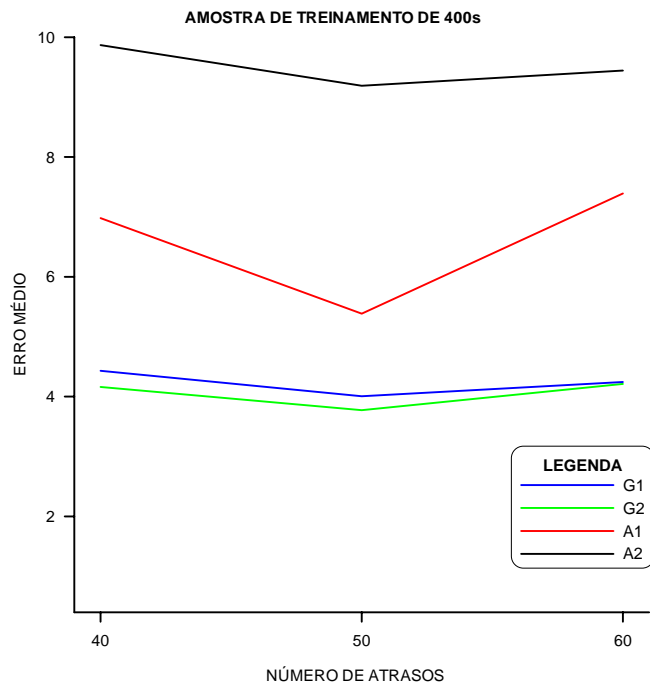


Figura V-28 – Treinamento de 400s, MY topo.

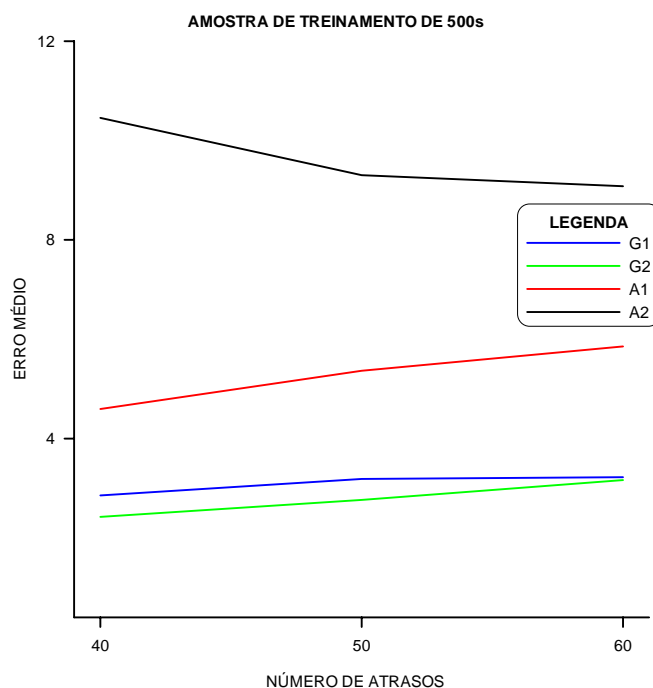


Figura V-29 – Treinamento de 500s, MY topo.

É possível observar que o modelo polinomial NARMAX (G2), com amostra de treinamento de 500s e 40 atrasos é o que fornece o menor erro médio (Tabela V-18). Esta configuração de modelo nos leva ao cálculo de 7750 coeficientes (ou pesos) na fase de treinamento. Neste modelo analisado, a maior diferença entre o sinal simulado e o original foi registrada no intervalo de tempo de 2001.0s. A Figura V-30 apresenta um trecho da série temporal que inclui este ponto. A Tabela V-19 apresenta um resumo estatístico dos resultados.

Amostra de treinamento (s)	Erro Médio (kN·m)
300	5.354
400	4.161
500	2.422

Tabela V-18 – Erro médio em função da amostra de treinamento, G2, 40 atrasos.

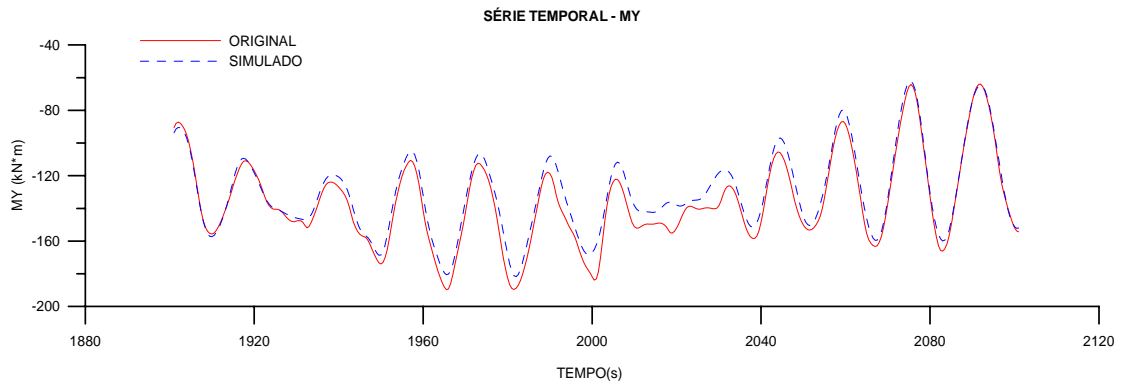


Figura V-30 – Série temporal, MY topo.

Parâmetro	Sinal	
	Original	Simulado
Média (kN·m)	-126.51	-124.75
Desvio Padrão (kN·m)	28.03	27.35
Coefficiente de correlação	0.995	

Tabela V-19 – Média, Desvio Padrão e Coeficiente de correlação, G2, 40 atrasos.

Observando a Tabela V-19 e a Figura V-30 podemos dizer que a série temporal foi bem reproduzida.

V.2.3 Ponto na corcova: Esforço FX

As Figuras V-31 a V-32 seguintes mostram como o erro médio do esforço FX no ponto da corcova do *riser* varia em função do número de atrasos para cada amostra de treinamento estudada.

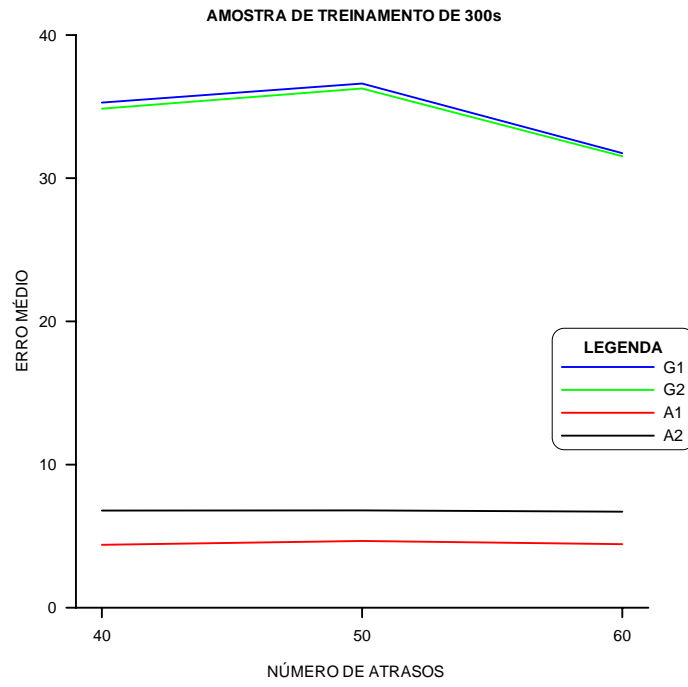


Figura V-31 – Treinamento de 300s, FX corcova.

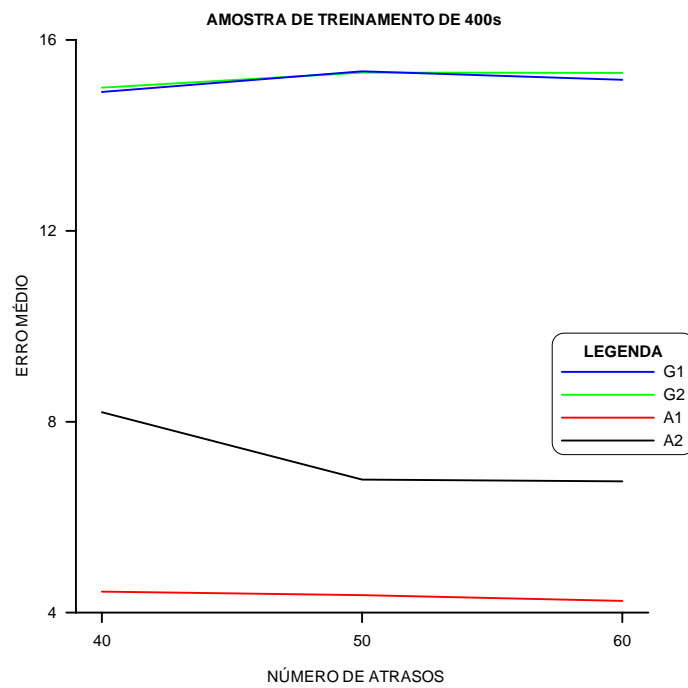


Figura V-32 – Treinamento de 400s, FX corcova.

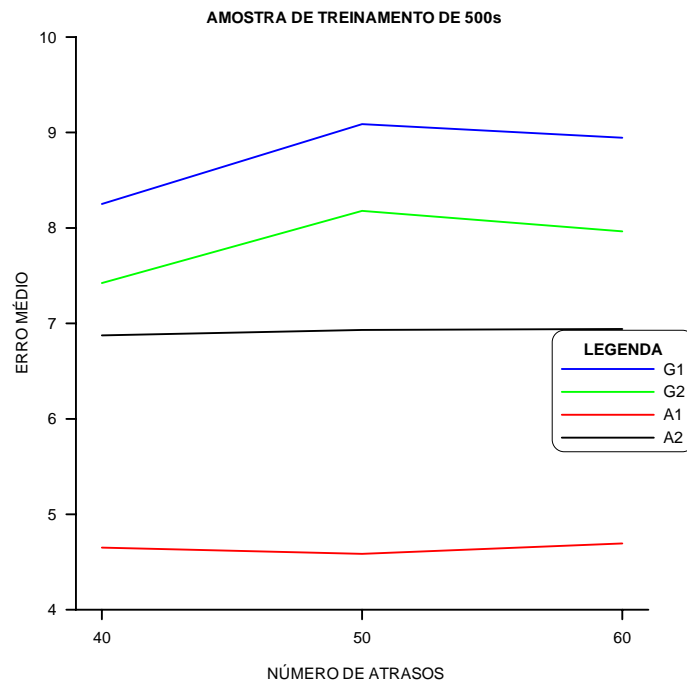


Figura V-33 – Treinamento de 500s, FX corcova.

É possível observar que o modelo Redes Neurais (A1), com amostra de treinamento de 400s e 60 atrasos é o que fornece o menor erro médio (vide Tabela V-20). Esta configuração de modelo nos leva ao cálculo de 1840 coeficientes (ou pesos sinápticos) na fase de treinamento. Neste modelo analisado, a maior diferença entre o sinal simulado e o original foi registrada no intervalo de tempo de 122.0s. A Figura V-34 ilustra uma parte da série temporal que inclui este ponto. A Tabela V-21 apresenta um resumo estatístico dos resultados.

Amostra de treinamento (s)	Erro Médio (kN)
300	4.440
400	4.247
500	4.695

Tabela V-20 – Erro médio em função da amostra de treinamento, A1, 60 atrasos.

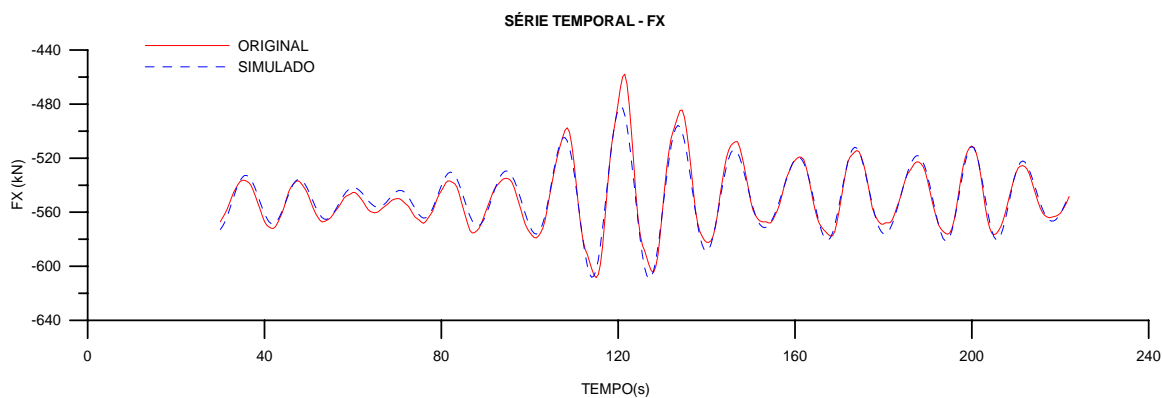


Figura V-34 – Série temporal, FX corcova.

Parâmetro	Sinal	
	Original	Simulado
Média (kN)	-549.20	-549.25
Desvio Padrão (kN)	23.31	22.76
Coefficiente de correlação	0.971	

Tabela V-21 – Média, Desvio Padrão e Coeficiente de correlação, A1, 60 atrasos.

Observando a Tabela V-21 e a Figura V-34 podemos dizer que a série temporal foi bem reproduzida.

V.2.4 Ponto no topo: Esforço FX

As Figuras V-35 a V-37 mostram como o erro médio do esforço FX no topo do *riser* varia em função do número de atrasos para cada amostra de treinamento estudada.

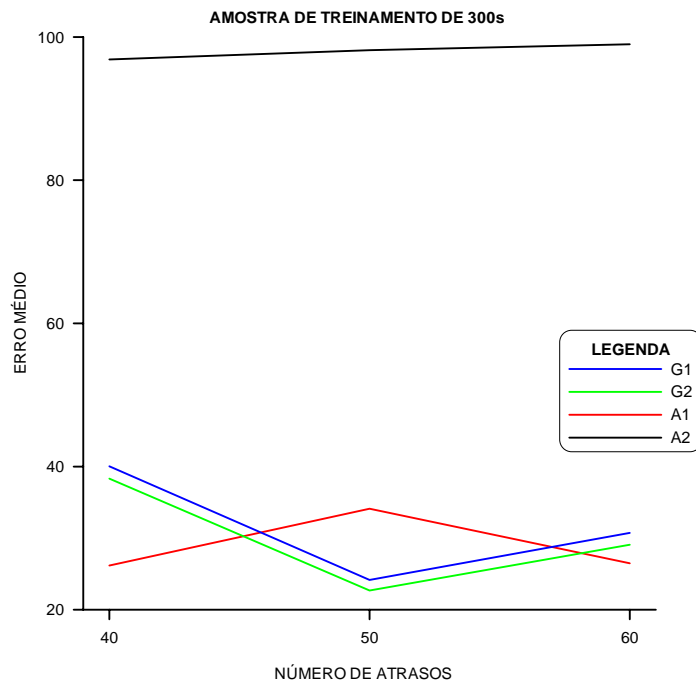


Figura V-35 – Treinamento de 300s, FX topo.

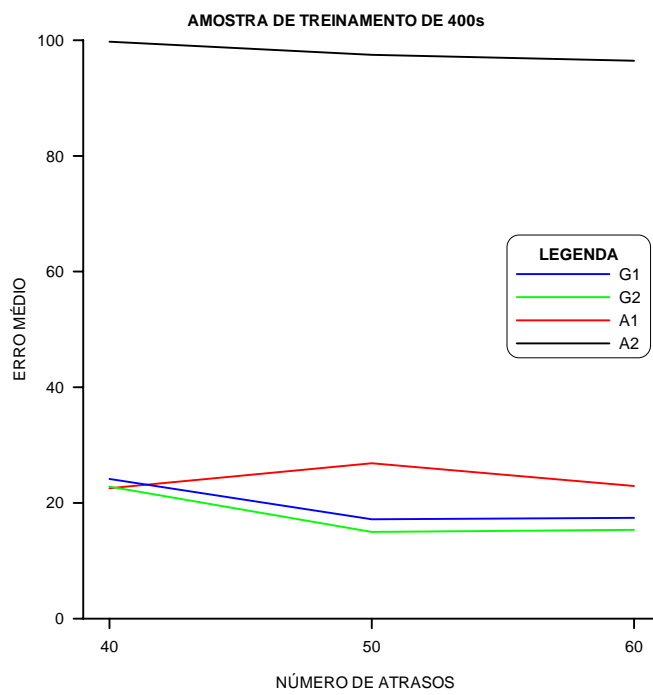


Figura V-36 – Treinamento de 400s, FX topo.

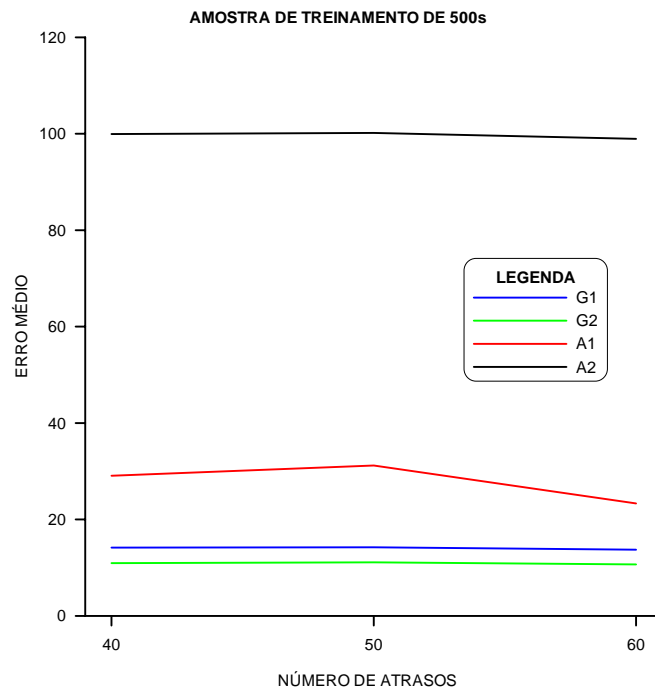


Figura V-37 – Treinamento de 500s, FX topo.

É possível observar que o modelo polinomial NARMAX (G2), com amostra de treinamento de 500s e 60 atrasos é o que fornece o menor erro médio (vide Tabela V-22). Esta configuração de modelo nos leva ao cálculo de 17020 coeficientes (ou pesos) na fase de treinamento. Neste modelo analisado, a maior diferença entre o sinal simulado e o original foi registrada no intervalo de tempo de 1446.0s. A Figura V-38 apresenta uma parte da série temporal incluindo este ponto. A Tabela V-23 apresenta um resumo estatístico dos sinais original e simulado.

Amostra de treinamento (s)	Erro Médio (kN)
300	29.074
400	15.374
500	10.689

Tabela V-22 – Erro médio em função da amostra de treinamento, G2, 60 atrasos.

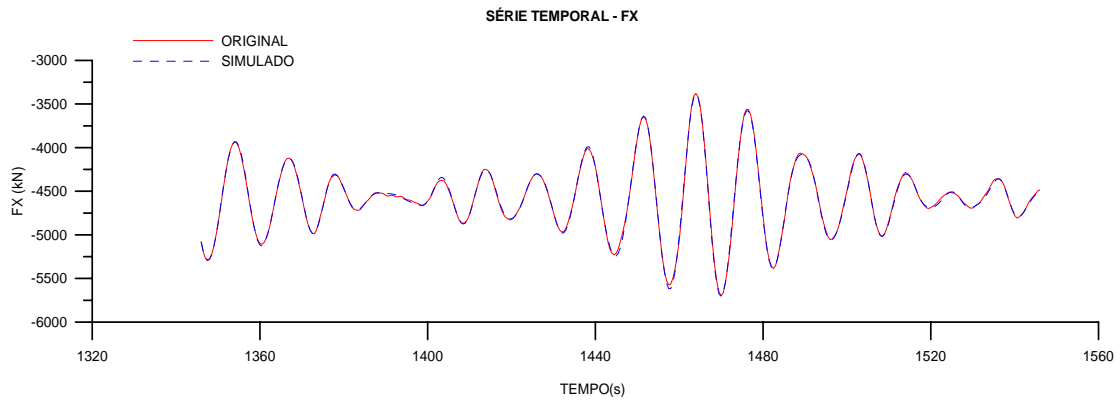


Figura V-38 – Série temporal, FX topo.

Parâmetro	Sinal	
	Original	Simulado
Média (kN)	-4572.55	-4572.32
Desvio Padrão (kN)	397.31	398.11
Coeficiente de correlação	0.999	

Tabela V-23 – Média, Desvio Padrão e Coeficiente de correlação, G2, 60 atrasos.

Observando a Tabela V-23 e a Figura V-38 podemos dizer que a série temporal foi muito bem reproduzida.

V.2.5 Comentários

Podemos observar que de um modo geral as séries temporais foram bem reproduzidas. Outro comentário interessante é sobre a redução no custo computacional. Para a obtenção da resposta dinâmica a partir do MEF foram necessários 120 minutos. Escolhendo um modelo de predição, que demande um tempo computacional pequeno, seriam necessários aproximadamente 20 minutos de cálculo a partir do MEF (para obter a amostra de treinamento) e mais 5 minutos (se não menos) para o cálculo a partir do método de predição, conforme ilustra a Tabela V-24.

Método	Tempo gasto (min)
MEF	120
Predição de Séries Temporais	25

Tabela V-24 – Redução do custo computacional

Neste exemplo também foi possível observar que para alguns modelos testados, para uma dada amostra de treinamento, o aumento do número de atrasos pode resultar em aumento do erro médio.

V.3 Análise de Extremos via Amostra de Máximos Observados em Realizações Independentes da Resposta

Uma maneira de se obter diretamente a estatística do valor extremo de um parâmetro de resposta de uma estrutura *offshore* numa dada condição ambiental é através de um determinado número de realizações independentes. Cada realização independente corresponde, por exemplo, a uma análise dinâmica aleatória considerando um conjunto distinto de fases aleatórias para a geração da série temporal das elevações no mar. Para cada realização independente, com duração correspondente a um dado tempo de interesse (normalmente 3.h.), identifica-se o máximo observado para a resposta. Supondo N realizações independentes, uma amostra \mathbf{X}_m de N valores máximos x_{m_i} é obtida, i.e.,

$$\mathbf{X}_m = \{x_{m_1}, \dots, x_{m_N}\} \quad (\text{V-6})$$

A média e o desvio padrão da amostra \mathbf{X}_m são dados respectivamente por:

$$\bar{x}_m = \sum_{i=1}^N \frac{x_{m_i}}{N} \quad (\text{V-7})$$

e

$$s_{x_m} = \sqrt{\sum_{i=1}^N \frac{(x_{m_i} - \bar{x}_m)^2}{N-1}} \quad (\text{V-8})$$

A partir da amostra \mathbf{X}_m uma distribuição de probabilidades pode ser ajustada para representar a estatística do valor extremo da resposta considerada. Com base na Estatística de Ordem [1], observa-se que a distribuição Tipo I (ou de Gumbel) é uma

distribuição que pode representar uma vasta gama de populações de extremos. A função densidade de probabilidades associada à distribuição Tipo I é definida por:

$$f_x(x) = \alpha \exp(-\alpha(x-u) - \exp(-\alpha(x-u))) \quad (\text{V-9})$$

onde α e u são os parâmetros da distribuição. A média e o desvio padrão de uma população representada pela distribuição de Gumbel são dados, respectivamente, por:

$$\mu_x = u + \frac{0.5722}{\alpha} \quad (\text{V-10})$$

e

$$\sigma_x = \frac{\pi}{\sqrt{6}\alpha} \quad (\text{V-11})$$

Utilizando-se o método dos momentos [1] os parâmetros u e α da distribuição Tipo I usada para representar a amostra \mathbf{X}_m podem ser obtidos utilizando as equações acima e igualando-se os valores da média e desvio padrão da amostra \bar{x}_m e s_{x_m} aos respectivos valores da população μ_x e σ_x .

Neste tipo de análise o custo computacional é excessivamente grande, pois envolve um número N (20 a 40) de análises dinâmicas aleatórias no domínio do tempo com duração mínima igual ao tempo de interesse para a análise de extremos (usualmente 3-h).

Neste exemplo investiga-se a possibilidade de se empregar os métodos de predição de séries temporais para facilitar este tipo de análise. Assumindo-se que a obtenção dos movimentos prescritos do flutuante é relativamente de baixo custo computacional, a idéia básica é treinar um modelo NARMAX ou RNA utilizando uma simulação (análise aleatória da estrutura considerada, por exemplo, um *riser* ou linha de ancoragem) de curta duração para dada realização específica e depois utilizá-lo nas demais realizações para fazer a predição da resposta de interesse sem realizar outras análises dinâmicas aleatórias.

A seguir será considerado um caso de análise de extremos correspondente à tração no topo de uma linha de ancoragem conectada a um FPSO instalado em 780m de lâmina d'água.

V.3.1 Linha de ancoragem

Neste item foi desenvolvida uma análise de extremos da tração no topo de uma linha de ancoragem conectada a um FPSO instalado numa lâmina d'água de 780m, considerando uma condição ambiental extrema da Bacia de Campos. A linha é composta de trecho de amarra no fundo e no topo. O trecho intermediário é de poliéster.

As séries temporais dos movimentos prescritos no topo da linha assim como as correspondentes séries da tração no topo aqui empregadas foram fornecidas pelo LCRAA. As séries foram fornecidas com 10800s já livres do transiente (150s). Neste exemplo as entradas (excitação) dos modelos de predição são os deslocamentos x (*surge*), y (*sway*) e z (*heave*) impostos no topo da linha. A saída, ou resposta, considerada é a tração no topo. Foram fornecidas 20 séries temporais correspondentes a realizações distintas.

Inicialmente foi feito um treinamento com a realização #1. A investigação sobre qual a configuração de modelo de predição que melhor representa o fenômeno foi feita seguindo a mesma metodologia do item V.2 . O erro médio em função dos atrasos (10, 20 e 30) e em função do tamanho da amostra de treinamento (200s, 300s e 400s) pode ser visto nas Figuras V-39 a V-40.

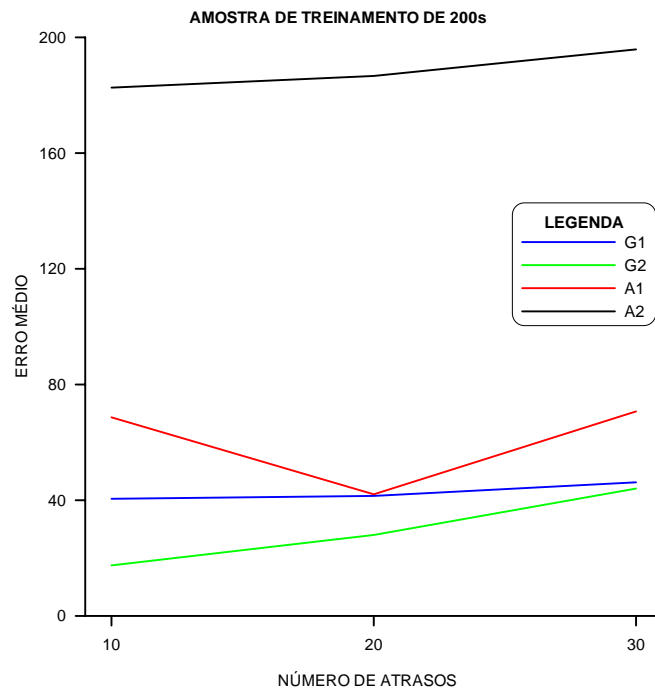


Figura V-39 – Amostra de treinamento de 200s

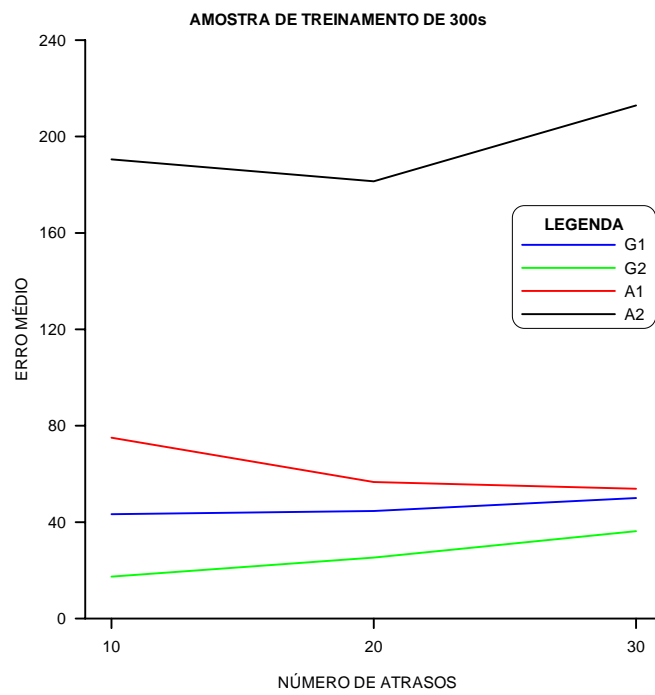


Figura V-40 – Amostra de treinamento de 300s

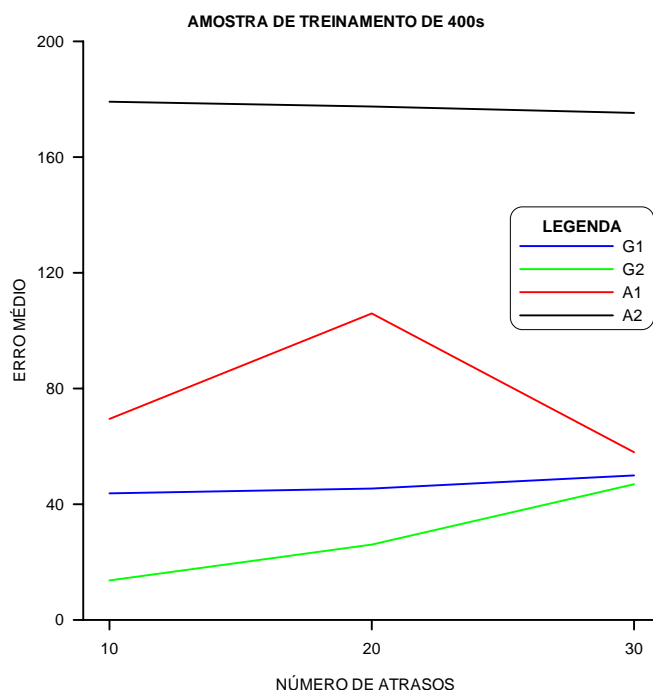


Figura V-41 – Amostra de treinamento de 400s

Amostra de treinamento (s)	Erro Médio (kN)
200	17.461
300	17.355
400	13.660

Tabela V-25 – Erro médio em função da amostra de treinamento, G2, 10 atrasos.

Observando as Figuras V-39 à V-41 e a Tabela V-25 é possível observar que o modelo polinomial NARMAX do segundo grau com 10 atrasos e amostra de treinamento de 400s é o que oferece a melhor predição da série temporal. O modelo com essa configuração nos leva ao cálculo de 595 coeficientes (ou pesos) na fase de treinamento. O maior erro foi observado no instante de 10948.5s. A Figura V-42 mostra parte da série temporal incluindo este instante de tempo. Sendo assim, os coeficientes calculados com este modelo para a realização #1 foram usados para a predição das 19 realizações restantes. O processo foi repetido tomando-se como base o cálculo dos coeficientes utilizando a realização #15. A Tabela V-26 e a Tabela V-27 mostram alguns parâmetros importantes dos sinais original e simulado (a partir da realização #1 e #15 respectivamente). Observamos que o coeficiente de correlação indica que as séries temporais foram reproduzidas com fidelidade.

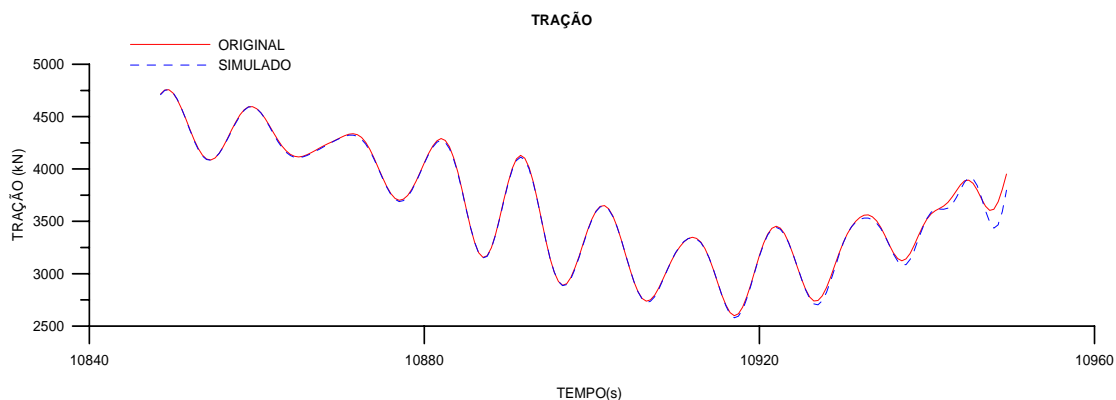


Figura V-42 – Série temporal da tração no topo. Realização #1.

#	Valor máximo (kN)		Média (kN)		Desvio Padrão (kN)		Coeficiente de Correlação
	Original	Simulado	Original	Simulado	Original	Simulado	
1	7067.22	7048.65	4195.16	4189.24	707.03	706.91	1.000
2	7007.78	6998.33	4218.40	4212.65	712.91	710.20	1.000
3	6616.88	6580.03	4197.89	4192.05	692.45	692.00	0.999
4	7041.75	7059.49	4171.95	4166.11	698.05	701.90	0.999
5	6887.85	6892.93	4203.88	4198.10	711.11	711.97	1.000
6	6933.98	6967.94	4210.55	4204.38	719.07	724.05	1.000
7	7358.02	7468.54	4177.69	4171.69	708.45	712.73	0.999
8	7000.57	7004.05	4173.96	4167.96	696.18	698.06	0.999
9	6873.00	6829.27	4176.95	4171.21	695.80	698.49	0.999
10	6748.38	6741.42	4195.59	4189.73	706.61	705.18	1.000
11	7251.64	7221.75	4186.28	4181.05	687.20	684.88	0.999
12	6542.31	6499.58	4218.16	4212.64	710.53	704.54	0.999
13	6461.33	6461.82	4166.24	4160.23	680.66	678.52	0.999
14	6812.62	6808.83	4214.16	4208.41	729.01	726.66	0.999
15	6684.23	6651.99	4182.61	4177.09	707.85	711.32	0.999
16	7034.11	7054.47	4191.52	4185.62	711.82	713.06	1.000
17	6583.80	6545.96	4173.16	4167.45	717.68	719.13	1.000
18	6242.94	6222.17	4182.16	4176.28	699.16	700.12	1.000
19	6883.27	6874.62	4199.33	4193.61	701.13	697.57	0.999
20	7116.71	7164.18	4166.36	4160.51	702.98	712.23	0.999

Tabela V-26 – Parâmetros do sinal original e simulado a partir de #1.

#	Valor máximo (kN)		Média (kN)		Desvio Padrão (kN)		Coeficiente de Correlação
	Original	Simulado	Original	Simulado	Original	Simulado	
1	7067.22	7063.19	4195.16	4200.67	707.03	702.64	0.999
2	7007.78	6983.02	4218.40	4224.52	712.91	705.82	1.000
3	6616.88	6601.91	4197.89	4203.84	692.45	689.01	1.000
4	7041.75	7049.64	4171.95	4177.96	698.05	697.91	0.999
5	6887.85	6871.26	4203.88	4209.74	711.11	707.74	1.000
6	6933.98	6908.23	4210.55	4216.19	719.07	720.38	1.000
7	7358.02	7468.45	4177.69	4183.62	708.45	709.28	0.999
8	7000.57	6969.99	4173.96	4179.55	696.18	693.81	0.999
9	6873.00	6868.71	4176.95	4182.71	695.80	696.11	0.999
10	6748.38	6727.24	4195.59	4201.39	706.61	701.70	1.000
11	7251.64	7211.00	4186.28	4192.46	687.20	680.30	0.999
12	6542.31	6512.42	4218.16	4224.44	710.53	700.18	0.999
13	6461.33	6436.68	4166.24	4171.61	680.66	674.95	0.999
14	6812.62	6767.87	4214.16	4220.43	729.01	722.54	0.999
15	6684.23	6668.44	4182.61	4188.82	707.85	707.02	1.000
16	7034.11	6979.38	4191.52	4197.43	711.82	708.68	1.000
17	6583.80	6560.99	4173.16	4179.30	717.68	714.93	0.999
18	6242.94	6238.07	4182.16	4188.11	699.16	695.59	1.000
19	6883.27	6860.07	4199.33	4205.03	701.13	693.62	0.999
20	7116.71	7184.02	4166.36	4172.35	702.98	708.87	1.000

Tabela V-27 – Parâmetros do sinal original e simulado a partir de #15.

Para a amostra de extremos original e para a simulada foi feito o ajuste de uma função densidade de probabilidade do Tipo I conforme descrito no item anterior. Na Tabela V-28 são apresentados os parâmetros estatísticos desta distribuição e na Figura V-43 as distribuições são comparadas.

Parâmetro	Série Original	Séries simuladas	
		A partir da realização # 1	A partir da realização # 15
Média (kN)	6857.42	6854.801	6846.529
Desvio Padrão (kN)	268.667	290.894	284.751
Coef. de Variação	0.039	0.042	0.042
u (kN)	6736.508	6723.887	6718.379
α	0.005	0.004	0.005
V95*	7358.70	7397.56	7377.82

* V95 é o valor que tem probabilidade 0.05 de ser excedido

Tabela V-28 – Resumo Estatístico da Distribuição de Extremos do Tipo I.

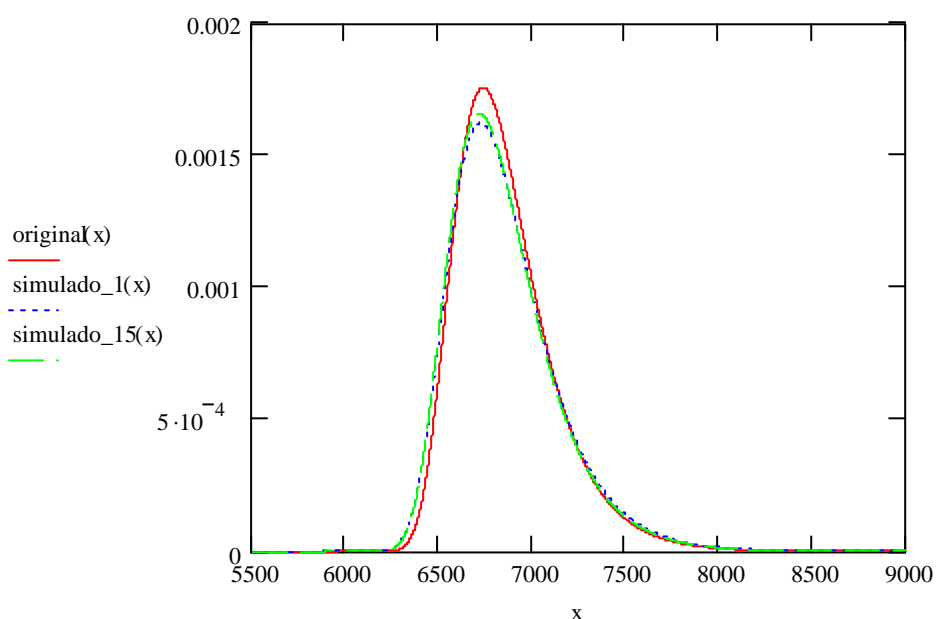


Figura V-43 – Ajuste de função de distribuição de probabilidade Tipo I

A Figura V-43 nos permite concluir que os métodos de predição de séries temporais são capazes de reproduzir a função de distribuição de probabilidade de uma amostra de extremos da tração no topo da linha analisada, independentemente de qual realização é empregada para o cálculo dos coeficientes. É importante mencionar que o uso de métodos de predição, neste caso, consome um tempo computacional muito inferior àquele necessário se fosse empregado o MEF. Na verdade o tempo só é gasto para obtenção de uma única amostra de treinamento de 400s, pois a partir destes coeficientes podemos calcular todas as outras quase instantaneamente.

V.4 Desempenho da metodologia de cálculo proposta

Neste item é feito um teste do desempenho da metodologia de cálculo proposta no item IV.6.1.1 A comparação toma como base o segundo exemplo (*riser lazy wave*). Para a investigação pegaremos o esforço FX no ponto do topo. O treinamento será feito empregando a metodologia de cálculo proposta (com metodologia) e usando somente o método do Gradiente Conjugado (sem metodologia), a partir de um ponto arbitrado inicialmente. A comparação foi feita através do erro médio e pode ser vista na Tabela V-29.

MÉTODO	ERRO MÉDIO (kN)	
	COM METODOLOGIA	SEM METODOLOGIA
NARMAX (G2)	10.689	26.612
RNA (A1)	23.315	29.965

Tabela V-29 – Teste de desempenho da metodologia de cálculo proposta

Os testes foram feitos com 60 atrasos (30s) e amostra de treinamento de 500s. Para o modelo NARMAX (G2) isso nos leva ao cálculo de 184 pesos na fase de treinamento. Para as Redes Neurais (A1) foram 1840 pesos sinápticos calculados no treinamento.

Na Tabela V-29 é interessante observar que em todos os casos o cálculo a partir da metodologia proposta no item IV.6.1.1 fornece resultados melhores do que se a mesma não fosse empregada.

CAPÍTULO VI - CONSIDERAÇÕES FINAIS

VI.1 Conclusões

Neste trabalho foram investigados os modelos polinomial e racional NARMAX (do primeiro e segundo graus), além das RNAs. Foram desenvolvidos programas de computador em linguagem *FORTRAN*[®] para o estudo destes dois métodos de predição de séries temporais.

No estudo desenvolvido foi constatada a dificuldade existente no cálculo dos pesos (treinamento) no que diz respeito ao problema do mínimo local. Sendo assim procedemos ao estudo de três métodos de otimização disponíveis na literatura, que são: Gradiente Descendente, Gradiente Conjugado e Estratégias Evolutivas, além da solução analítica disponível para o caso particular do modelo polinomial NARMAX. No estudo destes três métodos foi possível constatar as dificuldades presentes no uso de cada um. Isto culminou na metodologia de cálculo proposta no item IV.6.1.1 de modo que tiramos partido das melhores características disponíveis em cada um deles, evitando ou minimizando o problema do mínimo local. A metodologia gera um bom ponto de partida com as Estratégias Evolutivas e depois refina a busca com o Método do Gradiente Conjugado. Esta metodologia foi aplicada na solução dos exemplos, o que forneceu resultados satisfatórios.

Durante o desenvolvimento deste trabalho, também foi possível constatar que usando séries temporais com intervalo de tempo discreto compatível com a prática em análise dinâmica de estruturas o problema de *overfitting* não se verifica. Também podemos afirmar que o uso de uma amostra de treinamento que tenha um tamanho suficiente, e que seja representativa do fenômeno físico em questão, é necessário para que não sejam verificados problemas durante a extrapolação dos resultados.

Tendo em vista os resultados obtidos nos exemplos podemos concluir que o uso de modelos NARMAX e Redes Neurais Artificiais para predição de séries temporais na análise dinâmica de estruturas fornecem bons resultados, desde que observemos alguns parâmetros importantes, tais como o número de atrasos

considerados, tamanho da amostra de treinamento e o modelo empregado. Com os exemplos estudados não foi possível achar relação entre número de atrasos, tamanho da amostra de treinamento e erro médio. Também não é possível concluir qual o grau do polinômio a ser adotado no modelo NARMAX.

Nas Redes Neurais foi possível observar que a função de ativação linear fornece resultados melhores que a tangente hiperbólica, que por sua vez fornece resultados melhores que a sigmóide. É importante ressaltar que os resultados obtidos nos exemplos não podem ser generalizados, pois a função de ativação fornece características importantes, devendo ser investigada em cada caso analisado.

A resposta dinâmica obtida com estes modelos de predição de séries temporais, na maioria dos casos reproduz fielmente a resposta dinâmica original da estrutura. Quando isto não ocorre podemos observar que pelo menos as aproximações obtidas são bastante razoáveis.

A principal vantagem dos Modelos NARMAX e das RNAs é que eles são bastante eficientes e precisam de séries temporais curtas da resposta verdadeira para fazer a predição futura da mesma. Comparando com uma análise dinâmica aleatória não-linear no domínio do tempo, baseada no Método dos Elementos Finitos, o ganho computacional destes modelos é bastante vantajoso para obter séries temporais de um parâmetro de resposta de interesse. O maior ganho computacional é verificado na análise de extremos, pois a partir do treinamento com uma única amostra somos capazes de prever todas as outras realizações e conseqüentemente a estatística de extremos.

VI.2 Sugestão para trabalhos futuros

Seria desejável que fosse desenvolvida uma pesquisa a fim de verificar a possibilidade de construir um algoritmo que fosse capaz de calcular automaticamente o número de atrasos, o tamanho da amostra de treinamento, e qual a arquitetura de modelo ideal. Para os modelos NARMAX seria desejável que o algoritmo fosse capaz de identificar qual o grau do(s) polinômio(s) adotado(s). Para as RNA seria desejável que a função de ativação ideal pudesse ser identificada.

Tendo métodos que fossem capazes de determinar a arquitetura de modelo ideal, o número de atrasos e tamanho da amostra de treinamento ótimos, poderíamos finalmente comparar o desempenho dos modelos NARMAX e das RNAs (ambos com a configuração ótima), a fim de obter a melhor aproximação para a análise estrutural em questão.

Seria muito útil o estudo de processamento de sinais, de modo que fosse pesquisado (ou desenvolvido) algum parâmetro que desse noção absoluta do erro cometido. Isso é dito pois durante a montagem dos exemplos exibidos no Capítulo V foi preciso recorrer ao erro médio (para fazer a comparação relativa entre modelos) e em seguida ao coeficiente de correlação para verificar se a predição da série temporal foi satisfatória. Tudo isso ainda foi acompanhado do trecho da série temporal onde se verifica a diferença máxima entre o sinal original e o simulado.

Uma outra linha de pesquisa seria o estudo do intervalo de confiança do sinal simulado a partir dos erros verificados na amostra de treinamento. Isso permitiria ao Engenheiro Estrutural decidir se o uso do método de predição é compatível com as incertezas envolvidas no projeto em questão.

É importante observar que a modelagem a partir de técnicas de identificação de sistemas não requer conhecimento do fenômeno físico em questão. Entretanto para o caso de análise dinâmica de estruturas este conhecimento está disponível. Seria interessante investigar se é possível o uso de um esquema de cálculo, a partir de técnicas de identificação de sistemas, que tire partido do conhecimento do comportamento estrutural.

REFERÊNCIAS BIBLIOGRÁFICAS

- [1] ANG, A.H-S., TANG, W. H., *Probability Concepts in Engineering Planning and Design, Volume 1 – Basic Principles*. New York, John Wiley & Sons Inc., 1975
- [2] KÓVACS, Z.L., *Redes Neurais Artificiais*. São Paulo, Editora Livraria da Física, 2002
- [3] HAYKIN, S., *Neural Networks – A comprehensive foundation*. New Jersey, Prentice Hall, 2001
- [4] SHEWCHUK, J.R., *An Introduction to the Conjugate Gradient Method Without te Agonizing Pain*. Pittsburgh, School of Computer Science Carnegie Mellon University, 1994
- [5] AGUIRRE, L.A.R., GIOVANI G., JÁCOME, C.R.F., “Identificação de Sistemas Não-Lineares Utilizando Modelos Narmax Polinomiais – Uma Revisão e Novos Resultados”. *SBA Controle & Automação* v.9, n. 2, pp 90-106, Jun, Jul e Agosto de 1998
- [6] CINAR, A., *Nonlinear Time Series Models For Multivariable Dynamic Processes*. Chicago, Department of Chemical Engineering Illinois Institute of Technology., 2001
- [7] CASSINI, C.C.S., AGUIRRE, L.A., *Uma Introdução à Identificação de Sistemas Não Lineares* , Relatório #PPGEE/MAC SIN-03/99, Macsin, Belo Horizonte, 1999.
- [8] CLOUGH, R.W. and PENZIEN, J. *Dynamic of Structures*. California, McGraw-Hill, 1975
- [9] GUARIZE, R. *Uso de Redes Neurais Na Análise de Resposta Dinâmica de Estruturas*. Teses de M.Sc., COPPE/UFRJ, Rio de Janeiro, RJ, Brasil, 2004.
- [10] ROSENBLATT, F. Perceptron Simulation Experiments. *In Proceedings of the Institute of Radio Engineers*, 301-309, 1960
- [11] MCCULLOCH, W. S. e PITTS W. A logical calculus of the ideas immanent in nervous activities. *Bulletin of Mathematical Biophysics*, 115-133, 1969
- [12] PINTO, D. e MORGADO, M.C.F. *Cálculo Diferencial e Integral de Funções de Várias Variáveis*. Rio de Janeiro, Editora UFRJ, 1997

- [13] LEITHOLD, L. *O Cálculo Com Geometria Analítica*. Califórnia, Editora HARBRA, 1986
- [14] ZHU, Q. M. “A back propagation algorithm to estimate the parameters of non-linear dynamic rational models”. *Applied Mathematical Modelling* v.27 pp 169-187, 2003
- [15] CAMBRIDGE UNIVERSITY, *Numerical Recipes in Fortran 77*. Cambridge, Cambridge University, 1986
- [16] MICHALEWICZ, Z. *Genetic Algorithms + Data Structures = Evolution Programs* Springer-Verlag Berlin Heidelberg New york, 1992
- [17] ANFLEX. Manual do Usuário, PETROBRAS/COPPE, 1998.
- [18] DYNASIM. Manual Teórico e do Usuário, PETROBRAS/USP, 1999.