

JHONNY LEONIDAS MERMATZ

**ANÁLISE DINÂMICA DE SISTEMAS DE PRODUÇÃO
FLUTUANTE DE PETRÓLEO ATRAVÉS DE REDES
NEURAIS ARTIFICIAIS**

Dissertação apresentada à Escola
Politécnica da Universidade de São Paulo
para a obtenção do título de Mestre em
Engenharia.

São Paulo
1999

São Paulo
1999

Orientador:
Prof. Dr. Marcelo Godoy Simões

Area de Concentração:
Engenharia Mecatrônica e Sistemas
Mecânicos.

Dissertação apresentada à Escola
Politécnica da Universidade de São Paulo
para obtenção do título de Mestre em
Engenharia.

ANÁLISE DINÂMICA DE SISTEMAS DE PRODUÇÃO FLUTUANTE DE PETRÓLEO ATRAVÉS DE REDES NEURAS ARTIFICIAIS

JHONNY LEONIDAS MERMA TIQUILLOCA

A meus pais Eva e Leonidas, e a minha
irmã Dania, pelo apoio e incentivo
incondicional. Vocês são a razão da minha
vida.

AGRADECIMENTOS

- A meu grande amigo e orientador Prof. Marcelo Godoy Simões por me dar a maior oportunidade de trabalho e superação pessoal, pela confiança depositada e o constante apoio na realização deste trabalho.
- Aos Professores Hélio M. Morishita e Newton Maruyama pelas valiosas orientações dadas no exame de qualificação que enriqueceram e melhoraram em muito a culminação desta pesquisa.
- A meu grande amigo e companheiro Petronio Vieira Jr. por seu total apoio na revisão deste trabalho.
- Aos meus amigos do Grupo de Ação de Acionamento e Controle Inteligente GACI (Nilson, Javier, Pitta, Silvio, Paulo, Fernando e Vanderci), e a Silvana pela revisão as sugestões e o encorajamento na culminação deste trabalho.
- Ao Conselho Nacional de Desenvolvimento Científico e Tecnológico CNPq pela concessão da bolsa de mestrado, processo N.º 133298/97-3.
- Ao Projeto "Simulação e Visualização em Engenharia" FINEP-RECOPE pelo financiamento da pesquisa.
- A todas as pessoas do departamento de Engenharia Mecatrônica, que direta e indiretamente colaboraram na execução deste trabalho.

ABSTRACT

This work develops a methodology for analyzing the dynamics of moored tanker ships FPSO (Floating Production Storage and Offloading System) using Neural Networks.

The approach for solving this problem is done using two methods: The first one estimates the mooring cables' efforts (mooring lines + hawser). It is based on a collection of mooring lines' tension historical data-base for anticipating prospective cables' behaviors given the present data. The second method identifies and models the oceanic system. The influence of every input variable (environmental forces) is understood by using a cascade shaped neural network.

Both methods are studied and analyzed comparing the advantages and disadvantages of the usage of any one of them.

ESCOLA POLITÉCNICA DA UNIVERSIDADE DE SÃO PAULO

Avenida Professor Luciano Gualberto, travessa 3 nº380 CEP05508-900 São Paulo SP
Telefone: (011) 818.5443 Fax (011) 818.5713

Pós-Graduação



MESTRADO

CANDIDATO: JHONNY LEONIDAS MERMA TIQUILLOCA

TÍTULO DO TRABALHO: "ANÁLISE DINÂMICA DE SISTEMAS DE PRODUÇÃO FLUTUANTE DE PETRÓLEO ATRAVÉS DE REDES NEURAIS ARTIFICIAIS".

ORIENTADOR: PROF.DR. MARCELO GODOY SIMÕES

DATA DA DEFESA: 13.12.1999

OBSERVAÇÕES:

Assinatura do Examinador

Marcelo Godoy Simões

(caso necessário faça cópias deste impresso)

ESCOLA POLITÉCNICA DA UNIVERSIDADE DE SÃO PAULO

Avenida Professor Luciano Gualberto, travessa 3 nº380 CEP05508-900 São Paulo SP
Telefone: (011) 818.5443 Fax (011) 818.5713



Pós-Graduação

MESTRADO

CANDIDATO: JHONNY LEONIDAS MERMATA TITULLOCA

TÍTULO DO TRABALHO: "ANÁLISE DINÂMICA DE SISTEMAS DE PRODUÇÃO FLUTUANTE DE PETRÓLEO ATRAVÉS DE REDES NEURAIS ARTIFICIAIS".

ORIENTADOR: PROF.DR. MARCELO GODOY SIMÕES
EXAMINADOR: PROF.DR. NEWTON MARYAMA

DATA DA DEFESA: 13.12.1999

OBSERVAÇÕES:

refazer fig. 2.4
refazer item 2.3.1
refazer item 2.3.2
refazer 2.3.3 (definição de parâmetros x não paramétrico)

Newton Maryama
Assinatura do Examinador

(caso necessário faça cópias deste impresso)

ESCOLA POLITÉCNICA DA UNIVERSIDADE DE SÃO PAULO

Avenida Professor Luciano Gualberto, travessa 3 nº380 CEP05508-900 São Paulo SP
Telefone: (011) 818.5443 Fax (011) 818.5713



Pós-Graduação

MESTRADO

CANDIDATO: JHONNY LEONIDAS MERMATA TIOULLOCA

TÍTULO DO TRABALHO: "ANÁLISE DINÂMICA DE SISTEMAS DE PRODUÇÃO FLUTUANTE DE PETRÓLEO ATRAVÉS DE REDES NEURAIS ARTIFICIAIS".

ORIENTADOR: PROF.DR. MÁRCELO GODOY SIMÕES
EXAMINADOR: PROF.DR. HÉLIO MITTO MORISHITA

DATA DA DEFESA: 13.12.1999

OBSERVAÇÕES:

REVISÃO 0 12/01 1.1.3
ACERTOS 0 12/01 2.3.2

Assinatura do Examinador
(caso necessário faça cópias deste impresso)

Helio Mitto Morishita

1.1.3 Conceitos de Ancoragem Tipo Turret

O sistema tipo *turret* é uma classe especial de um único ponto de ancoragem (SPM), composto por um grande sistema de orientação fixado diretamente ao navio.

O *turret* é uma estrutura conectada ao navio tanque ancorado através de uma articulação contendo um arranjo de orientação, é fixado ao leito marinho por linhas catenárias de ancoragem, conforme o navio tanque se movimenta há um alinhamento na direção resultante das forças ambientais (vento, ondas e correntezas).

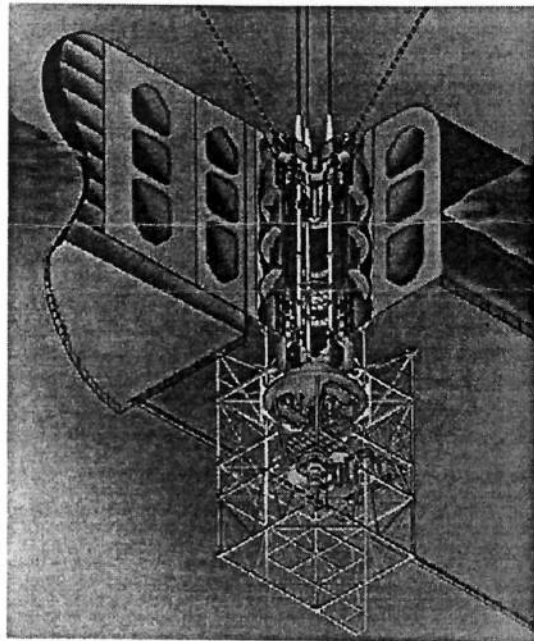


Fig.1.3: Detalhe da estrutura *turret*.

Este sistema utiliza uma estrutura em forma de torre localizada na proa navio (Fig. 1.3). Os *risers* flexíveis são unidos por bridas a rígidos condutos abaixo da torre. O número

de *risers* que pode ser acomodado neste sistema é principalmente determinado pela geometria da estrutura *turret* e o arranjo de orientação.

Os navios tanque com a habilidade para se movimentar ao redor de sistemas de ancoragem tipo *turret* são mais apropriados que as plataformas para a produção de hidrocarbonetos em muitas regiões *offshore*, devido a sua versatilidade para se ajustar a uma ampla variedade de parâmetros dos campos, junto com a facilidade para a remoção e adaptação a novos poços.

Vários tipos de ancoragens *turret* estão sendo desenvolvidos e alguns destes já estão instalados e em operação atualmente.

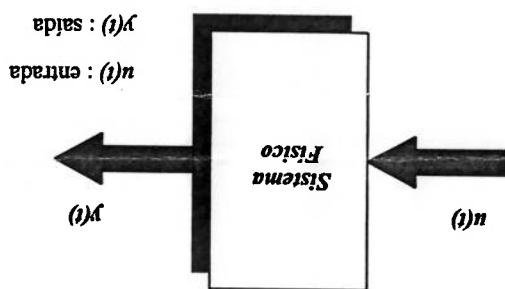


Fig. 2.4: Sistema físico simples com uma entrada e uma saída

2.3.1 Caracterização e Identificação de Sistemas

A caracterização e identificação são problemas fundamentais na teoria de sistemas. O problema de caracterização com a representação matemática de um sistema, e pode ser representada da forma seguinte:

Como um operador P de um espaço de entrada u em um espaço de saída y , o objetivo é caracterizar a classe \mathcal{P} ao qual P pertence. O problema da identificação é determinar a classe $\mathcal{P} \subset \mathcal{P}$ e um elemento de $\mathcal{P} \in \mathcal{P}$, de modo que \mathcal{P} se aproxime a P no sentido desejado.

Em sistemas estáticos, os espaços u e y são subconjuntos de \mathcal{R}^n e \mathcal{R}^m respectivamente, enquanto que sistemas dinâmicos são assumidos geralmente para ser funções de fronteira integráveis de Lebesgue sobre o intervalo $[0, T]$ ou $[0, \infty]$.

Em ambos casos, o operador P é definido implicitamente por pares específicos de entrada-saída. A escolha da classe dos modelos de identificação assim como o método específico para determinar \mathcal{P} depende de uma variedade de fatores os quais estão relacionados à precisão desejada e do tratamento analítico. Eles incluem a adequação do modelo P para representar P .

2.3.2 Sistemas Lineares X Sistemas Não Lineares

Modelos são lineares se variações nas variáveis independentes produzem variações lineares nas dependentes ou em suas respectivas derivadas. Um sistema é linear se a regra da superposição é aplicável:

$$P(x_1 + x_2) = P(x_1) + P(x_2) \quad \text{e} \quad P(k \cdot x) = k \cdot P(x)$$

onde P é qualquer operador contido no modelo.

A teoria de sistemas dinâmicos usa técnicas firmemente estabelecidas baseadas em álgebra linear, teoria de variável complexa, e teoria de equações diferenciais ordinárias lineares. Como a maioria das técnicas para o projeto de sistemas dinâmicos são muito relacionadas com as propriedades de estabilidade de sistemas lineares invariantes no tempo, métodos bem conhecidos foram estabelecidos para tais sistemas (NARENDRA e PARTHASARTHY, 1990).

2.3.3 Identificação e Modelagem de Sistemas

Um dos processos mais comuns para os quais as redes neurais são usadas, é a identificação e modelagem de sistemas. Isto envolve a colocação da rede neural em paralelo com o sistema físico aplicando a entrada do sistema à entrada da rede, usando a saída do sistema como a saída desejada para a rede neural, e treinando a rede neural até que o erro entre a saída do sistema e a saída da rede atinjam um nível aceitável. Essa configuração é mostrada na (Fig. 2.5).

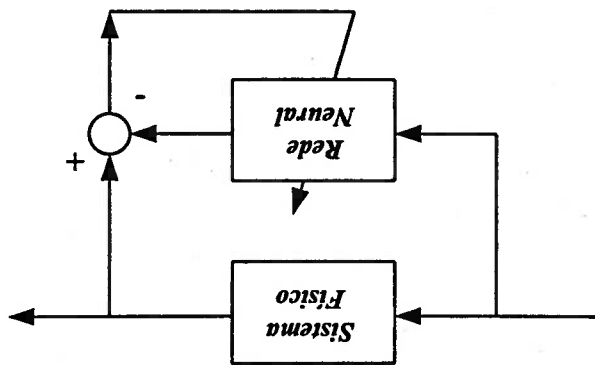


Fig. 2.5: Configuração adaptativa para modelagem por redes neurais

Depois da rede ter sido treinada, a relação entre as entradas e saídas reflete o sistema físico modelado, desde que as variáveis tenham sido treinadas sobre uma faixa dinâmica apropriada.

Normalmente torna-se necessário introduzir atrasos de tempo Δ dentro da configuração, no sentido de modelar o tempo que é requerido para que o sinal se propague através do processo físico (TSOUKALAS e UHRIG, 1997).

Segundo NARENDRA e PARATHASARATHY (1990), O problema de reconhecimento de padrões é um exemplo típico de identificação de sistemas estáticos, assim os conjuntos $U_i \subset \mathcal{R}^n$ são mapeados dentro de elementos $y_i \in \mathcal{R}^m$; ($i = 1, 2, \dots$) no espaço de saída por uma função de decisão P . Os elementos de U_i denotam os padrões de vetores de entrada. Em sistemas dinâmicos, o operador P define uma planta dada, que é

definida implicitamente por pares de entrada-saída de funções dependentes do tempo $u(t)$, $y(t)$, $t \in [0, T]$. Em ambos casos o objetivo é determinar P de modo que:

$$\|y - y\| = \|P(u) - P(u)\| \leq \varepsilon \quad (2.32)$$

para algum valor desejado $\varepsilon > 0$, $P(u) = \hat{y}$ denota a saída da identificação do modelo e daqui $\hat{y} - y \equiv \varepsilon$ é o erro na saída gerada por P e a saída observada y .

Em sistemas onde os princípios dos modelos não estão disponíveis, é necessária a utilização de identificação não paramétrica tipo *black box*, que será tratada com mais detalhe no capítulo 3.

2.3.4.1 Identificação não Paramétrica

A identificação não paramétrica permite a modelagem tipo *black-box* entre relações de entrada-saída com uma abordagem não linear. Por exemplo um filtro transversal adaptativo, quando empregado como uma rede neural torna-se uma rede de *finite impulse response* (FIR), que realiza a identificação não paramétrica do sistema. A rede é provida com um vetor de entrada de amostras ponderadas das variáveis, servindo como uma excitação dinâmica do fenômeno.

2.3.4.2 Identificação Paramétrica

A identificação paramétrica, identifica características estruturais e valores de parâmetros para modelos de sistemas do mundo real. Isto inclui identificação da estrutura do modelo na forma de equações diferenciais lineares e não lineares, assim como estimação de parâmetros onde o modelo da estrutura é conhecido. As redes neurais treinadas através de aprendizado supervisionado podem ser usadas para ambas as formas de identificação.

ÍNDICE

1	1.1 IDENTIFICAÇÃO DO PROBLEMA	1
	1.1.1 Histórico	1
	1.1.2 Exploração de Petróleo Offshore Através de FPSOs	3
	1.1.3 Conceitos de Ancoragem Tipo Turret	5
	1.1.4 Comportamento Dinâmico de Navios Amarrados	6
	1.2 OBJETIVOS	7
	1.3 ORGANIZAÇÃO DO TEXTO	8
2	MECANISMOS DE ANÁLISE	10
	2.1 INTRODUÇÃO	10
	2.2 REDES NEURAIS ARTIFICIAIS (RNAs)	10
	2.2.1 Neurônios Artificiais	11
	2.2.2 Definição das Redes Neurais Artificiais	12
	2.2.3 Funcionamento da Rede de Retropropagação	14
	2.2.4 Porque Utilizar Redes Neurais	15
	2.3 MODELAGEM DE SISTEMAS DINÂMICOS	16
	2.3.1 Caracterização e Identificação de Sistemas	18
	2.3.2 Sistemas Lineares X Sistemas Não Lineares	19
	2.3.3 Identificação e Modelagem de Sistemas	20
	2.3.3 Identificação do Sistema	21
	2.4 MODELOS DE SISTEMAS DINÂMICOS	23
	2.5 SOFTWARE PARA SIMULAÇÃO	25
	2.5.1 Objetivos para o Projeto do Simulador	25
	2.5.2 Estruturas de Dados Baseadas em Arrays	26
	2.5.3 Implementação do simulador de Backpropagation	29
3	MODELAGEM BLACK-BOX NÃO LINEAR EM IDENTIFICAÇÃO DE SISTEMAS	31
	3.1 INTRODUÇÃO	31
	3.1.1 Modelos Black Box	32
	3.1.2 Estruturas Não Lineares Black Box	33
	3.1.3 Modelos Lineares Black-box	35
	3.2 REGRESSORES PARA MODELOS DINÂMICOS NÃO LINEARES BLACK-BOX	36
	3.3 Mapeamento Não Linear	37
	3.4 QUESTÕES ACERCA DE REDES NEURAIS	38
	3.4.1 Redes Multicamadas	39
	3.4.2 Redes Recorrentes	40
	3.5 REDES NEURAIS COMO MODELOS ESTADÍSTICOS	41
4	ESTIMAÇÃO DE SÉRIES TEMPORAIS ATRAVÉS DE REDES NEURAIS ARTIFICIAIS	42
	4.1 ASPECTOS TEÓRICOS DAS SÉRIES TEMPORAIS	42
	4.1.1 Funções Estáticas e Funções Dependentes do Tempo	42
	4.1.2 Definição de Termos para Series Temporais	43

APÊNDICE B

APÊNDICE A

6.2	SUGESTÕES PARA TRABALHOS FUTUROS.....	93
6.1	CONCLUSÕES.....	91
6	CONCLUSÕES E SUGESTÕES PARA TRABALHOS FUTUROS.....	91
5.2.7	Validação dos Resultados.....	83
5.2.6	Configuração das Redes para a Simulação.....	79
5.2.5	Operação do Simulador.....	78
5.2.4	Modelagem e Identificação.....	75
5.2.3	Modelagem Baseada em Redes Neurais.....	75
5.2.2	Aquisição e Medição das Condições Ambientais.....	73
5.2.1	Introdução.....	71
5.2	MODELAGEM DA DINÂMICA DE NAVIOS ANCORADOS ATRAVÉS DE REDES NEURAIS ARTIFICIAIS.....	71
5.1.2	Equações de Movimento.....	66
5.1.1	Sistema de Coordenadas.....	65
5.1	MODELO MATEMÁTICO.....	65
5	MODELAGEM E IDENTIFICAÇÃO DO SISTEMA OCEÂNICO.....	65
4.3.7	Validação dos Resultados.....	54
4.3.6	Treinamento das Redes Neurais para a Estimação de Séries Temporais.....	53
4.3.5	Padrões de Qualidade.....	52
4.3.4	Coleta dos Dados.....	52
4.3.3	Janelamento dos Dados.....	51
4.3.2	Padronização.....	50
4.3.1	Fatores Determinantes no Desempenho das Redes Neurais.....	49
4.3	MODELAGEM DE REDES NEURAIS PARA PREVISÃO DE SÉRIES TEMPORAIS.....	49
4.2.3	Escolha da Topologia.....	48
4.2.2	Arquitetura.....	47
4.2.1	Histórico.....	46
4.2	APLICAÇÃO DE REDES NEURAIS ARTIFICIAIS EM PREVISÃO.....	46
4.1.7	Estimando Mais que Um Passo em Frente no Futuro.....	46
4.1.6	Modelagem de Espaço de Estado com Redes Neurais.....	46
4.1.5	Redes Neurais de Atrasos de Tempo (Time Delay Neural Network).....	45
4.1.4	O Espaço de Estado das Séries Temporais.....	44
4.1.3	Dimensionalidade Inerente da Série.....	44

APÊNDICE D
ATUALIZAÇÃO DOS PESOS DAS CAMADAS OCULTAS.....108

APÊNDICE C
ATUALIZAÇÃO DOS PESOS DA CAMADA DE SAÍDA.....105

LISTA DE FIGURAS

2	Fig. 1.1 : Navio <i>FPSO</i> tipo <i>turret</i>
3	Fig. 1.2: <i>FPSO</i> com <i>turret</i> , e navio <i>shuttle</i> amarrado em <i>tandem</i>
5	Fig. 1.3: Detalhe da estrutura <i>turret</i>
11	Fig. 2.1: Modelo do neurônio artificial.....
13	Fig. 2.2: Rede neural artificial.....
17	Fig. 2.4: Sistema físico simples com uma entrada e uma saída.....
20	Fig. 2.5: Configurações de modelado de sistemas adaptativos.....
24	Fig. 2.6: Atrasos na entrada para a implementação da dinâmica à rede.....
24	Fig. 2.7: Realimentação da saída com atrasos de tempo para outorgar dinâmica à rede.....
27	Fig. 2.8: Modelagem das estruturas das camadas.....
28	Fig. 2.9: Estrutura de dados para a implementação de redes neurais.....
40	Fig. 3.1: Rede neural com duas camadas ocultas.....
41	Fig. 3.2: Exemplo de rede recorrente, q^{-1} retardada o sinal uma amostra de tempo.....
45	Fig. 4.1: Rede neural de atrasos de tempo.....
49	Fig. 4.2: Topologia utilizada para a estimação.....
51	Fig. 4.3: Janelamento do sinal.....
54	Fig. 4.4: Configuração do arquivo de treinamento.....
58	Fig. 4.5: Saída real e saída produzida pela rede (estimação 1 passo = 25 seg.).....
58	Fig. 4.6: Saída real e diferença (<i>real-estimada</i>) na estimação 1 passo em frente.....
59	Fig. 4.7: Saída real e saída produzida pela rede (estimação 3 passos = 75 seg.).....
59	Fig. 4.8: Saída real e a diferença (<i>real-estimada</i>), estimação 3 passos em frente.....
60	Fig. 4.9: Saída real e saída produzida pela rede (estimação 6 passos = 150 seg.).....
60	Fig. 4.10: Saída real e diferença (<i>real-estimada</i>) na estimação 6 passos em frente.....
61	Fig. 4.11: Saída real e produzida pela rede no <i>hawser</i> (estimação 1 passo = 25seg.).....
61	Fig. 4.12: Saída real e diferença (<i>real-estimada</i>) no <i>hawser</i>
62	Fig. 4.13: Estimação um passo em frente (cabo 1).....
62	Fig. 4.14: Estimação três passos em frente (cabo 1).....
63	Fig. 4.15: Estimação seis passos em frente (cabo 1).....
63	Fig. 4.16: Estimação um passo em frente (<i>hawser</i>).....
64	Fig. 4.17: Precisão atingida pelas redes (a) um passo e (b) dois passos em frente.....
64	Fig. 4.18: Precisão atingida pelas redes (a) três passos e (b) quatro passos em frente.....
64	Fig. 4.19: Precisão atingida pelas redes (a) cinco passos e (b) seis passos em frente.....
65	Fig. 5.1: Sistema de coordenadas.....
66	Fig. 5.2 : Posição do navio em relação ao referencial inercial.....
73	Fig. 5.3: Movimentos do navio.....
74	Fig. 5.4 : Esquema de monitoração do <i>FPSO</i> P.P. Moraes.....
76	Fig. 5.5: Modelagem do simulador neural.....
77	Fig. 5.6: Forças atuantes no simulador neural para a configuração <i>tandem</i>
80	Fig. 5.7: Representação do navio mãe para a simulação.....

81	Fig. 5.8: Representação do navio filho para a simulação.
82	Fig. 5.9 : Disposição das redes para a simulação
84	Fig. 5.10: Esforço produzido na identificação do cabo 1
84	Fig. 5.11: Erro produzido na identificação do cabo 1
85	Fig. 5.13: Esforço produzido na identificação do cabo 6
85	Fig. 5.14: Erro produzido na identificação do cabo 6
86	Fig. 5.14: Esforço na identificação do cabo 2
86	Fig. 5.15: Esforço produzido na identificação do cabo 3
87	Fig. 5.17: Esforço produzido na identificação do cabo 4
87	Fig. 5.18: Esforço produzido na identificação do cabo 5
88	Fig. 5.19: Esforço produzido na identificação do <i>hawser</i>
89	Fig. 5.20: Erro produzido na identificação do <i>hawser</i>
89	Fig. 5.21: Precisão alcançada pela rede na identificação do (a) cabo 1 e (b) cabo2
90	Fig. 5.22: Precisão alcançada pela rede na identificação do (a) cabo 3 e (b) cabo4
90	Fig. 5.23: Precisão alcançada pela rede na identificação do (a) cabo 5 e (b) cabo 6
93	Fig. 6.1: Rede neural hierárquica

RESUMO

Este trabalho desenvolve uma metodologia para analisar a dinâmica do sistema de amarração de navios tanque FPSO, Sistemas de Produção Flutuante de Armazenamento e Descarga de Petróleo, por meio de redes neurais artificiais.

A abordagem para a solução deste problema é feita através de dois métodos. O primeiro método trata da estimação da série temporal dos esforços produzidos nas linhas de amarração (linhas de ancoragem + *hawser*). Este é baseado na coleção de dados históricos de medições das tensões nas linhas de amarração, de modo a conhecer futuros comportamentos das linhas a partir de dados de entrada atuais. No segundo método é feita a identificação e modelagem do sistema oceânico. Através de uma configuração cascata de redes neurais se obtém o conhecimento da influência que cada uma das variáveis de entrada (forças ambientais), tem sobre os esforços produzidos nas linhas de amarração.

Ambas metodologias são estudadas e os resultados são analisados comparando as vantagens e desvantagens na escolha do primeiro ou segundo método.

ABSTRACT

This work develops a methodology for analyzing the dynamics of moored tanker ships FPSO (Floating Production Storage and Offloading System) using Neural Networks.

The approach for solving this problem is done using two methods: The first one estimates the mooring cables' efforts (mooring lines + hawser). It is based on a collection of mooring lines' tension historical data-base for anticipating prospective cables' behaviors given the present data. The second method identifies and models the oceanic system. The influence of every input variable (environmental forces) is understood by using a cascade shaped neural network.

Both methods are studied and analyzed comparing the advantages and disadvantages of the usage of any one of them.

I INTRODUÇÃO

1.1 Identificação do problema

1.1.1 Histórico

A exploração mundial de petróleo no mar teve início no México em 1946, onde foi utilizada uma plataforma fixa de madeira, cuja lamina d'água tinha comprimento em torno de 5,0 m. Com a intensificação da busca, a exploração deixou a região costeira, caminhando para águas mais profundas, impulsionando o desenvolvimento das primeiras plataformas metálicas.

Pelo fato destas plataformas apresentarem custos em progressão geométrica com a lamina d'água, no final da década de 70, surgem as primeiras plataformas semi-submersíveis posicionadas através de linhas em catenária.

Uma vez que as plataformas semi-submersíveis possuem baixa capacidade de armazenamento, faz-se necessária presença de um navio tanque por perto para armazenar a produção, sendo que este é invariavelmente posicionado através de uma monobóia (SPM - *Single Point Mooring*). Uma vez produzido, o petróleo *offshore* é normalmente escoado para o continente através de navios tanque. O descarregamento é feito em terminais junto à costa, onde o navio também é posicionado através de uma monobóia.

Atualmente, a tecnologia FPSO (*Floating Production Storage and Offloading System*) vem se afirmando como uma nova tendência em unidades de produção flutuantes em detrimento das atuais plataformas semi-submersíveis. Tal fato vem ocorrendo devido

ao alto custo desta última e relativamente baixo da primeira. A conversão de antigos petroleiros em FPSOs, com sistemas de amarração do tipo *Turret* (Fig. 1.1), tem se tomado uma solução interessante, não só economicamente, como ecologicamente correta, pelo fato de se estar criando uma destinação para os antigos petroleiros, resultado do contínuo sucateamento da atual frota mundial esperado para os próximos anos.

Com isso, os sistemas de amarração destes petroleiros começaram a desempenhar um papel cada vez mais importante na viabilização de exploração de petróleo em águas profundas.

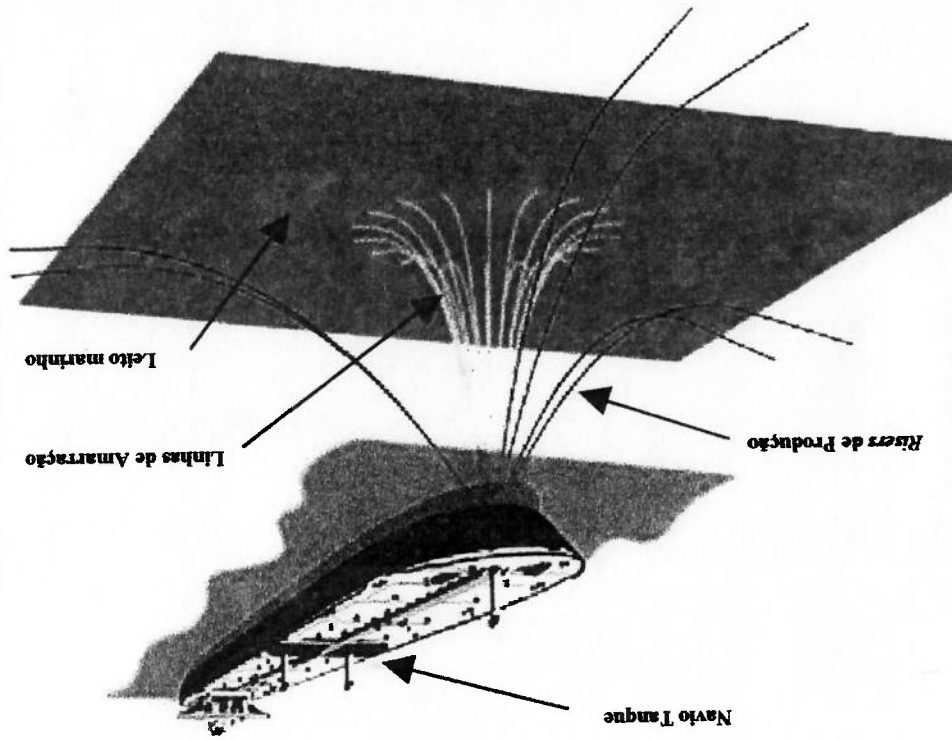


Fig. 1.1 : Navio FPSO tipo *turret*.

1.1.2 Exploração de Petróleo Offshore Através de FPSOs

É um moderno conceito para as unidades flutuantes de exploração de petróleo *offshore* ancorados em águas profundas. Um navio tanque é ancorado *offshore*, e o petróleo é processado e armazenado antes de ser transportado por navios tanque "shuttle" que periodicamente são conectados ao navio mãe em formação "tandem" (Fig. 1.2). Os navios estão sujeitos às cargas ambientais, devido à ação conjunta das correntes oceânicas, ondas e vento.

Os navios tanque são muito susceptíveis à direção das cargas ambientais e possuem, por isto, uma tendência latente à instabilidade, associada às significativas variações das forças para pequenas variações na direção de incidência. Neste caso o projeto do sistema de ancoragem exige um estudo criterioso de estabilidade dinâmica, dada a natureza essencialmente não linear do sistema (FUCATU, 1998).

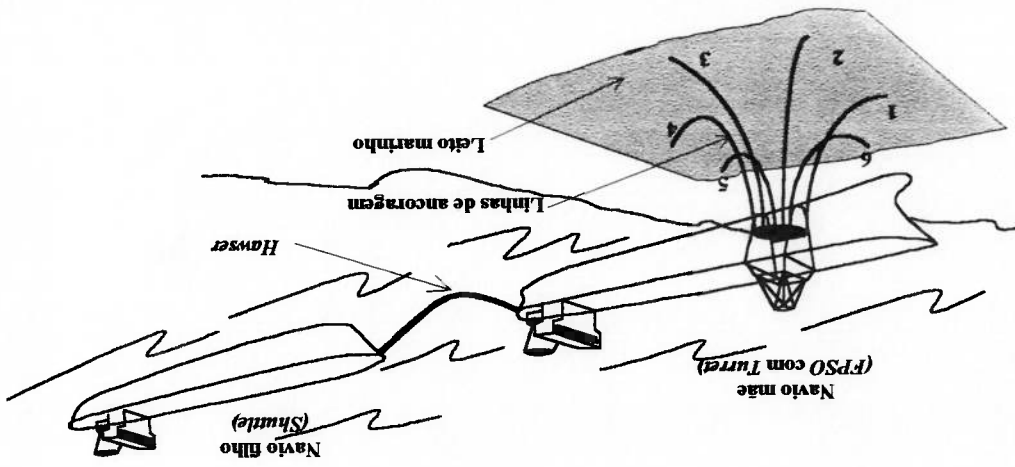


Fig. 1.2: FPSO com *turret*, e navio *shuttle* amarrado em *tandem*.

Sistemas de unico ponto de ancoragem SPM (*Single Point Mooring*) são alternativos

aos sistemas amplamente convencionais, que enfrentam a tarefa de manter fixa a estação.

A principal motivação de tais sistemas é permitir que o navio fique alinhado com a

“resultante” das forças ambientais, diminuindo o movimento e as cargas estruturais sobre

as linhas de ancoragem, *Hawsers* (cabos pelo qual o navio é conectado à bôia) e *Risers*

(conduto pelo qual o petróleo é transportado).

O sistema convencional SPM é um sistema composto por uma monobóia ancorada

ao leito marinho por meio de cabos e correntes, para o qual o FPSO é um tanque

especialmente convertido, é conectado por um “*cabo Hawser*”. Não só o comprimento do

cabo, mas também as características elásticas do cabo e o ponto de ligação são

parâmetros importantes no projeto.

A engenharia *offshore*, constitui um tópico de importância fundamental no

desenvolvimento de novos conceitos e técnicas para a exploração de hidrocarbonetos do

solo submarino em águas profundas. Um sistema FPSO (Sistema de Produção Flutuante

de Armazenamento e Descarga) completo se constitui em:

- Navio convertido (tanque);
- Linhas de amarração;
- Cabos umbilicais, e
- *Risers* de Produção.

Tais sistemas têm, em geral natureza dinâmica não linear, tornando-se bastante complexas as tarefas de modelagem e de determinação de sua resposta as ações ambientais associadas à incidência das ondas, correntezas e ventos.

1.1.3 Conceitos de Ancoragem Tipo Turret

O sistema tipo "turret" é uma classe especial de um único ponto de ancoragem (SPM), composto por um grande sistema de orientação diretamente fixado ao navio.

Uma estrutura *turret* é uma construção diretamente conectado ao navio tanque ancorado através de um arranjo de orientação por vela e fixada ao leito marinho por linhas catenárias de ancoragem. A parte de ancoragem do *turret* tem uma orientação fixa como um cata-vento do navio tanque, podendo-se movimentar de acordo com as condições ambientais (vento, ondas e correntezas).

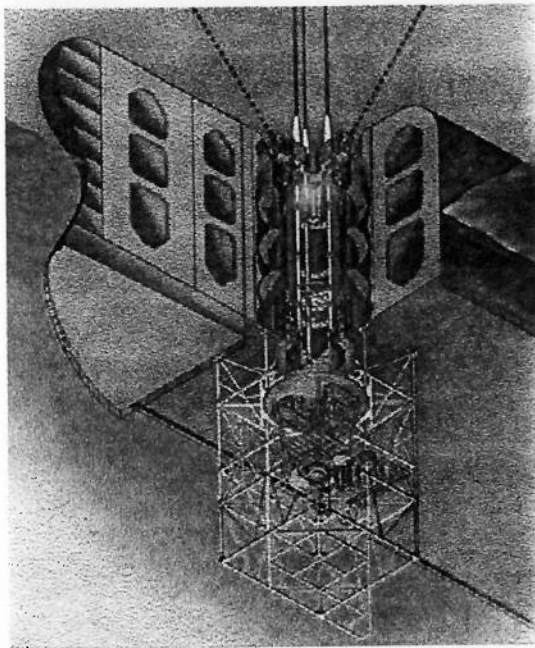


Fig.1.3: Detalhe da estrutura *turret*.

Em função de suas características construtivas, as plataformas semi-submersíveis não apresentam problemas de aprornamento, em relação ao vento, e são praticamente transparentes à ação das ondas. No caso de plataformas com *Pontoons* (flutuadores), estas devem estar aprovadas na direção da corrente predominante. De uma maneira geral

1.1.4 Comportamento Dinâmico de Navios Amarrados

instalados e operativos atualmente.

Vários tipos de ancoragens *turret* estão sendo desenvolvidos e alguns destes já estão

adaptação a novos poços.

Os navios tanque com a habilidade para se movimentar ao redor de sistemas de ancoragem tipo *turret* são mais apropriados que as plataformas para a produção de hidrocarbonetos em muitas regiões *offshore*, devido a sua versatilidade para se ajustar a uma ampla variedade de parâmetros dos campos, junto com a facilidade para a remoção e

A estrutura *turret* consiste primariamente de um acoplamento de correntes, às quais as linhas catenárias de ancoragem são conectadas. Os *risers* flexíveis são enlaçados por uniões a rígidos condutos abaixo da torre e estão desprotegidos na zona de respingado. O número de *risers* que pode ser acomodado neste sistema é principalmente determinado pela geometria da estrutura da torre e o arranjo de orientação.

sobre o nível do mar.

Este sistema utiliza uma estrutura relativamente pequena em forma de torre suspensa de um cata-vento incorporada na extensão rígida da estrutura enlaçada na proa ou na popa do navio (Fig. 1.3), a extensão é normalmente montada acima do poço

O objetivo principal deste trabalho é de realizar a estimação das forças de tensão nas linhas de amarração dos FFSO em configuração *Tandem*, no caso do navio mãe (FFSO

1.2 Objetivos

A grande vantagem da identificação de sistemas dinâmicos através de redes neurais artificiais está na excelente resposta a problemas não lineares além de não precisar de um conhecimento profundo do sistema a ser modelado.

- Ensaio de modelos reduzidos em tanque de provas;
- Estudo da estabilidade do sistema, através das equações características;
- Análise do comportamento no domínio da frequência;
- Simulação dinâmica no domínio do tempo, e
- Identificação de sistemas dinâmicos através de redes neurais artificiais.

Existem várias formas de se fazer a predição do comportamento dinâmico:

Já no caso de navios, qualquer que seja o sistema de amarração, dependendo da combinação entre as inúmeras variáveis envolvidas, tais como: porte do navio, condição de calado, direção e intensidade de corrente, vento e ondas, posição do *Turret*, pré-tensão das linhas, o navio pode apresentar um comportamento instável que têm como consequência o aparecimento de altos picos de tensão no sistema de amarração. Assim, a predição do comportamento dinâmico de navios amarrados é de extrema importância para o projeto do sistema de amarração (NISHIMOTO et al., 1996).

pode-se dizer que as plataformas são estruturas relativamente estáveis quando sujeitas às condições ambientais.

com *turret*), a determininação dos esforços nas 6 linhas de ancoragem e para caso do navio Filho (*Shuttle*), a obtenção do esforço no cabo de ligação (*Hawser*) entre os dois navios.

A abordagem para a solução do problema focaliza-se na utilização de técnicas de inteligência artificial, mais especificamente redes neurais artificiais, ocorrendo em duas etapas:

1. Estimacão das Sries Temporais dos Esforços nas Linhas de Amarração, e

2. Modelagem Dinâmica do Navio Ancorado através de Redes Neurais Artificiais.

A primeira parte do trabalho, *Estimacão das series temporais*, está baseada na coleta de dados históricos de medidas da tensoes nas linhas de amarração, realizando-se previsões "single step" e "multi step".

Na parte do trabalho relativo à *Modelagem dinamica através de redes neurais*, as condições das forcas ambientais são reproduzidas, neste procedimento é feita a identificacão do sistema oceânico, cujos dados são gerados através de equações diferenciais lineares e não lineares, que descrevem o comportamento dinâmico do navio ancorado.

1.3 Organizacão do Texto

Esta dissertacão está dividida em 6 capitulos explanando o desenvolvimento de temas especificos.

O capítulo 1 trata da identificação do problema, descrevendo o sistema oceânico e mostrando os marcos teóricos da exploração de petróleo *offshore* através da utilização dos navios FPSO. Em seguida é descrito o sistema de ancoragem em que está baseada essa nova tecnologia, mostrando as vantagens desta forma de extração de petróleo em alto mar.

No capítulo 2 descrevem-se os mecanismos de análise utilizados para a solução do problema, começando com as redes neurais (funcionamento e processo de aprendizado), métodos para a identificação e modelagem de sistemas dinâmicos, assim como o desenvolvimento do software para a implementação das redes neurais.

O capítulo 3 apresenta a fundamentação matemática da modelagem não linear "black box" na identificação de sistemas dinâmicos.

O capítulo 4 está dedicado exclusivamente à estimação de séries temporais por meio de redes neurais, como uma forma de estimar os esforços produzidos nas linhas de ancoragem do navio partindo de dados históricos, além da estimação um passo em frente são realizadas estimações de múltiplos passos em frente.

O capítulo 5 se ocupa da identificação e modelagem do navio FPSO através de redes neurais utilizando dados gerados por meio de equações diferenciais para o aprendizado das redes.

O capítulo 6 apresenta as conclusões e recomendações observadas no desenvolvimento do trabalho.

2 MECANISMOS DE ANÁLISE

2.1 Introdução

Basicamente, uma rede neural artificial consiste de um certo número de unidades ligadas por conexões direcionadas. Cada unidade recebe sinais, via conexões de entrada emitindo sinais para todas as outras unidades ligadas, via conexões de saída, formando redes. Estas redes possuem capacidade computacionais e de aprendizado.

A teoria de redes neurais possui uma natureza interdisciplinar, englobando: matemática, física, eletrônica, ciência da computação, pesquisa operacional, psicologia, biologia, filosofia, linguística e ciências cognitivas. De todas estas ciências há contribuições visando alcançar um objetivo comum: o de construir sistemas inteligentes (MELLO, 1991).

2.2 Redes Neurais Artificiais (RNAs)

O cérebro humano é um sistema muito complexo, capaz de pensar, lembrar e solucionar problemas. Existem muitos modelos computacionais que visam imitar as funções do cérebro. O neurônio é a unidade celular fundamental do sistema nervoso do cérebro ele é, um elemento de processamento que recebe e mistura sinais de outros neurônios através de caminhos chamados *dendritos*. Se o sinal de entrada é de uma certa intensidade, o neurônio dispara, produzindo um sinal de saída através do *axônio* o qual liga-se aos outros *dendritos* de muitos outros neurônios.

2.2.1 Neurônios Artificiais

Um neurônio artificial é um modelo matemático dos componentes de um neurônio natural; a seguir será mostrada a representação analógica de um neurônio artificial, conforme pode-se observar na (Fig. 2.1), os sinais de entrada são representados por $x_0, x_1, x_2, \dots, x_n$. Esses sinais são variáveis contínuas, cada uma das quais são modificadas por pesos (ou sinapses), analogamente à da unidade sináptica do neurônio biológico. Este elemento de processamento consiste de duas partes: (1) a primeira simplesmente soma os pesos das entradas resultando em uma quantidade I_j , (2) a segunda efetivamente não linear, é chamada por função de ativação y_j através da qual o sinal flui (FREEMAN, 1993).

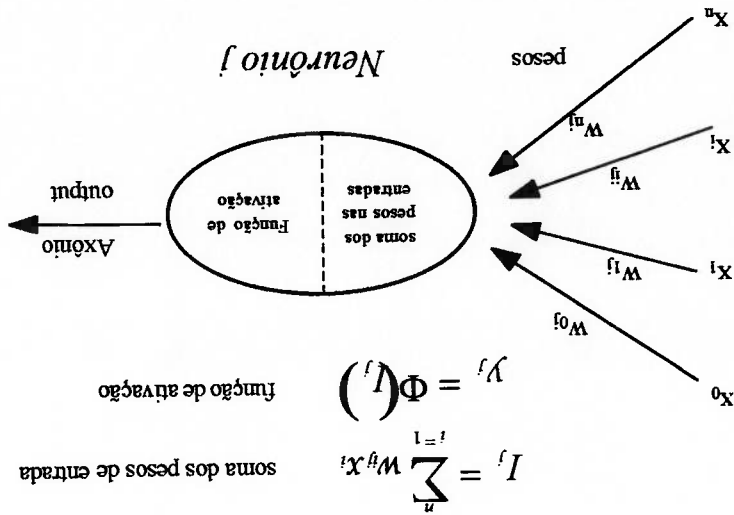


Fig. 2.1: Modelo do neurônio artificial.

A função de ativação mais comum é uma função contínua que varia entre dois valores assintóticos, tipicamente entre 0 e 1 ou entre -1 e +1, chamada por função sigmoide, representada pela equação (2.1):

Uma rede neural típica é completamente conectada, o que significa que há conexão completa entre cada um dos neurônios de uma camada com os demais neurônios da camada seguinte, conforme pode ser observado na figura (2.2).

Esses elementos de processamento são usualmente organizados em uma sequência de camadas totalmente conectadas de forma aleatória. Na (Fig. 2.2) descreve-se que a camada de entrada é um *buffer* que apresenta dados à rede. Esta camada de entrada não é uma camada de computação neural, pois os nós nas entradas não têm pesos e funções de ativação. A camada superior é a camada que apresenta a resposta de saída para uma entrada dada. A outra camada (ou camadas) é chamada intermediária ou escondida, porque usualmente não tem conexões com o mundo exterior. Tipicamente a camada de entrada, escondida, e a de saída, são designadas como camadas *ih*, *jh*, e *kh*, respectivamente (MELO, 1991).

Uma rede neural artificial pode ser definida como um sistema de processamento de dados consistindo em um grande número de elementos simples de processamento (neurônios artificiais) altamente interconectados, em uma arquitetura inspirada pela estrutura cerebral (WASSERMANN, 1989).

2.2.2 Definição das Redes Neurais Artificiais

onde α é um coeficiente que ajusta a derivada desta função.

$$\Phi(I) = \frac{1}{1 + e^{-\alpha I}} \quad (2.1)$$

RUMMELHART e McCLELLAND (1988).

Uma rede neural artificial útil para resolver problemas que requerem o reconhecimento de padrões complexos e a realização de mapeamentos não lineares é a Rede Multicamada de Retropropagação (*backpropagation*), que foi descrita formalmente em primeiro lugar por WERBOS (1974), posteriormente por PARKER (1982), e por

conexões.

Cada uma das conexões entre neurônios têm um peso ajustável. A rede mostrada na (Fig. 2.2) é denominada por rede *feedforward* com três neurônios na camada de entrada, quatro na camada escondida e dois na camada de saída. O símbolo w_{ij} indica o peso da conexão entre os neurônios i e j , assim como os demais pesos com w associados às

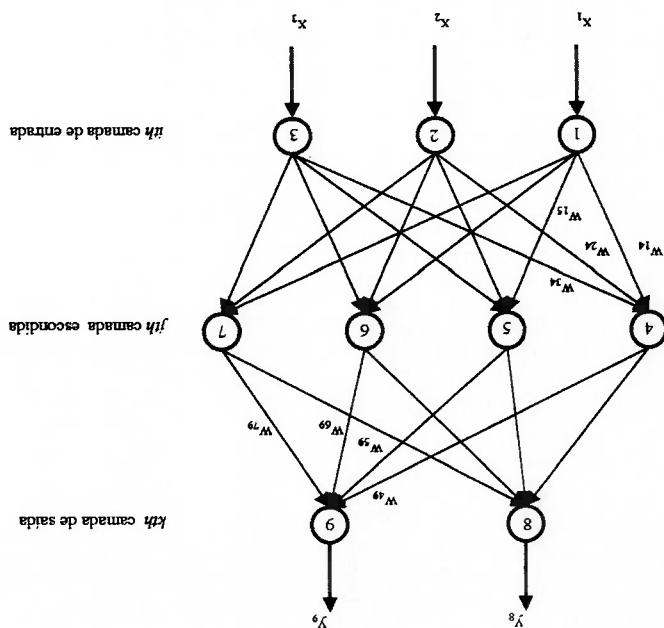


Fig. 2.2: Rede neural artificial.

A importância deste processo consiste em que, a medida que se treina a rede, os nós das camadas intermediárias se auto-organizam, de modo que aprendam a reconhecer distintas características do espaço total das entradas. Depois do treinamento ter ocorrido, quando apresenta-lhes um padrão arbitrário de entrada, que contenha ruído ou que seja incompleto, as unidades das camadas ocultas da rede respondem com uma saída ativa, se a nova entrada contém um padrão semelhante àquela característica que as unidades individuais tenham aprendido a reconhecer durante o seu treinamento. No caso

estado que permita codificar todos os padrões de treinamento.

recebido, atualizam-se os pesos de cada unidade para fazer convergir a rede até um erro que descreva sua contribuição relativa ao erro total. Baseando-se no sinal do erro repete-se, camada por camada, até que todos os nós da rede tenham recebido um sinal de aproximadamente na contribuição relativa de cada unidade à saída original. Este processo da camada intermediária recebem só uma fração do sinal total do erro, baseando-se (neurônios) da camada intermediária que contribuem diretamente na saída. As unidades Os sinais de erro propagam-se de volta a partir da camada da saída, a todos os nós

desejada, e calcula-se o erro para cada unidade da saída.

camadas superiores até gerar uma saída. O sinal da saída é comparado com a saída a primeira camada de unidades da rede, este vai se propagando através de todas as (FREEMAN, 1993). Uma vez que foi aplicado um padrão de entrada como estímulo para fornecidos como exemplos, empregando um ciclo propagação-adaptação de duas fases Este tipo de rede aprende de um conjunto pré-definido de pares de entradas e saídas

2.2.3 Funcionamento da Rede de Retropropagação

Sabe-se que as redes neurais artificiais são utilizadas para solucionar problemas onde as regras não precisam ser conhecidas, mas sim um conjunto de exemplos da solução a serem aprendidos estejam disponíveis. Então é claro que uma considerável quantidade de dados de treinamento é requerido. Os modelos de redes neurais possuem um número de qualidades que as fazem ser idealmente apropriadas para certos tipos de problemas listados a seguir (SWINGLER, 1996).

2.2.4 Porque Utilizar Redes Neurais

HINTON e SEJNOWSKI (1987) demonstraram que durante o processo de treinamento, as redes de retropropagação tendem a desenvolver relações internas entre os nós com o objetivo de organizar os dados de treinamento nestes tipos de padrões.

Na medida em que os sinais se propagam através das diferentes camadas da rede, o padrão de atividade que está presente em todas as camadas superiores, pode-se ver como um padrão com características que são reconhecidas pelas unidades da camada subsequente. O padrão da saída que foi produzido, pode ser visto como um mapa de características que oferece uma indicação da presença ou ausência de muitas combinações de características distintas dentro da entrada. O resultado global deste comportamento, faz que a rede de retropropagação constitua um meio eficaz para permitir a um sistema computadorizado examinar padrões de dados que possam ser ruidosos, ou estar incompletos e reconhecer padrões sutis de entradas parciais.

contrário, as unidades das camadas ocultas têm a tendência de inibir suas saídas, se o padrão de entrada não contém a característica para a qual a rede foi treinada.

As redes neurais operam sobre representações numéricas. Uma tarefa deve ser expressa em termos de uma função de uma codificação numérica a outra função. Se o problema não pode ser expresso nesse sentido, não poderá ser resolvida por uma rede neural.

As redes neurais são funções não lineares. Uma tarefa é de natureza não linear, se o efeito de uma mudança em qualquer variável sobre a saída é dependente do próprio valor, ou de qualquer outra variável de entrada.

As redes neurais são diferenciáveis. É possível diferenciar a função pela qual a rede aprende. Muitos sistemas industriais ou físicos necessitam ser expressos em termos do efeito que uma pequena mudança teria nas entradas, melhor que o efeito do ajuste completo de uma nova entrada.

As redes neurais são capazes de generalizar. Uma rede neural é um modelo generalizável de um conjunto de dados. A habilidade para generalizar significa que a rede é capaz de aprender com ruído ou com dados incompletos. A generalização deve ser vista como a realização duma média não linear sobre um conjunto de exemplos, de modo que ruído e dados corrompidos não tenham efeito.

2.3 Modelagem de Sistemas Dinâmicos

Neste ponto se descreve a utilização de redes neurais artificiais para a modelagem de sistemas dinâmicos lineares e não lineares, fazendo uso de sinais temporais de entrada e saída do sistema que será modelado (Fig. 2.4).

Nos últimos anos ocorreram muitos desenvolvimentos de arquiteturas baseadas em redes neurais com importantes resultados. Contudo, nem todas as arquiteturas neurais são apropriadas para propósitos de modelagem dinâmica (PATAKI, 1993). Apenas algumas redes são capazes de reconhecer padrões de séries (classificação espaço temporal), muitas delas são desenvolvidas para o mapeamento estático de padrões de entradas e saída, ou pela seleção de características de padrões de entrada.

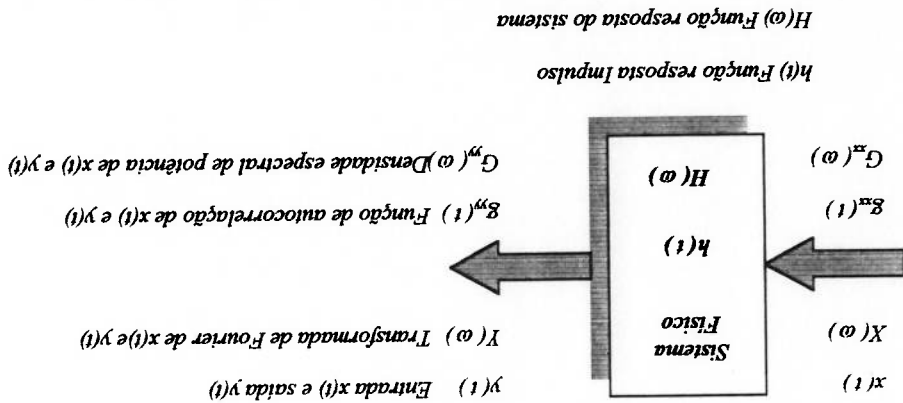


Fig. 2.4: Sistema físico simples com uma entrada e uma saída

Para a solução do problema de modelagem dinâmica, um sistema complexo de redes neurais e subsistemas lineares, é o mais apropriado

A capacidade de uma rede neural para modelar um sistema ou fenômeno permite que seja usada em diversas aplicações. De fato, muito das chamadas "teorias de sistemas adaptativos lineares" são diretamente aplicáveis às redes neurais artificiais. Além disto, a capacidade das redes neurais para desenvolver modelos não lineares de sistemas oferece uma vantagem adicional que pode ser usada em muitos casos.

A teoria de sistemas lineares é um campo muito bem desenvolvido sendo de extrema importância para o processamento de dados e a aplicação de tecnologias tais como as redes neurais aplicadas a problemas práticos e quando combinada com teoria de ruído aleatório, a tecnologia resultante se faz uma ferramenta poderosa para a investigação de sistemas complexos (TSOUKALAS e URRHIG, 1997).

2.3.1 Caracterização e Identificação de Sistemas

A caracterização e identificação são problemas fundamentais na teoria de sistemas. O problema de caracterização concerne com a representação matemática de um sistema, e é expressa como um operador P de um espaço de entrada u em um espaço de saída y , e o objetivo é caracterizar a classe \mathcal{C} ao qual P pertence. Dada a classe \mathcal{C} e o fato que $P \in \mathcal{C}$, o problema da identificação é para determinar a classe $\mathcal{C} \subset \mathcal{P}$ e um elemento de $\mathcal{P} \in \mathcal{C}$, de modo que P se aproxime a P no sentido desejado (SJOBERG et al., 1995).

Em sistemas estáticos, os espaços u e y são subconjuntos de \mathcal{R}^n e \mathcal{R}^m respectivamente, enquanto que sistemas dinâmicos são assumidos geralmente para ser funções de fronteira integráveis de Lebesgue sobre o intervalo $[0, T]$ ou $[0, \infty[$.

Em ambos casos, o operador P é definido implicitamente por pares específicos de entrada-saída. A escolha da classe dos modelos de identificação \mathcal{C} assim como o método específico para determinar P depende de uma variedade de fatores os quais estão relacionados à precisão desejada e do tratamento analítico. Eles incluem a adequação do modelo P para representar P .

2.3.2 Sistemas Lineares X Sistemas Não Lineares

O melhor aspecto teórico desenvolvido em sistemas dinâmicos, trata a definição de sistemas por operadores lineares usando técnicas firmemente estabelecidas baseadas em álgebra linear, teoria de variável complexa, e teoria de equações diferenciais ordinárias lineares. Como as técnicas para o projeto de sistemas dinâmicos são muito relacionadas com as propriedades de estabilidade de sistemas lineares invariantes no tempo, e como as condições de necessária e suficiente foram geradas nas décadas passadas, métodos bem conhecidos foram estabelecidos para tais sistemas (NARENDRA e PARTHASARTHY, 1990).

Em contraste a isto, a estabilidade de sistemas não lineares, pode ser estabelecido para a maioria de sistemas tão só sobre a base de sistema por sistema e daqui que não é surpresa que os procedimentos do projeto que se ajustem simultaneamente aos requerimentos de estabilidade, robustez e boa resposta dinâmica, não sejam muito disponíveis para grandes classes de tais sistemas.

O termo "redes neurais adaptativas" refere-se a que uma rede neural adapta o seus pesos para acompanhar o mapeamento das entradas à saída desejada. Muitas das redes neurais que utilizam o aprendizado supervisionado pertencem a esta categoria. Os algoritmos de adaptação dos mínimos quadrados são sistemas básicos de aprendizado para sistemas adaptativos de processamento de sinais e de redes neurais adaptativas (PAL et al., 1993).

O ajuste dos sistemas adaptativos são feitos com o objetivo de otimizar o desenvolvimento específico de sistemas complexos variantes no tempo e não lineares.

Assim a vantagem da habilidade das redes neurais, é a de utilizar funções de ativação não lineares na modelagem de processos.

2.3.3 Identificação e Modelagem de Sistemas

Um dos processos mais comuns para os quais as redes neurais são usadas, é a identificação e modelagem de sistemas. Isto envolve a colocação da rede neural em paralelo com o sistema físico aplicando a entrada do sistema à entrada da rede, usando a saída do sistema como a saída desejada para a rede neural, e treinando a rede neural até que o erro entre a saída do sistema e a saída da rede atinjam um nível aceitável. Essa configuração é mostrada na (Fig. 2.5).

Para sistemas com múltiplas entradas e saídas, entradas individuais e saídas desejadas podem ser destinadas a certas partes das camadas de entrada e saída, respectivamente. Geralmente, isto é necessário para modelagem com múltiplas entradas e saídas.

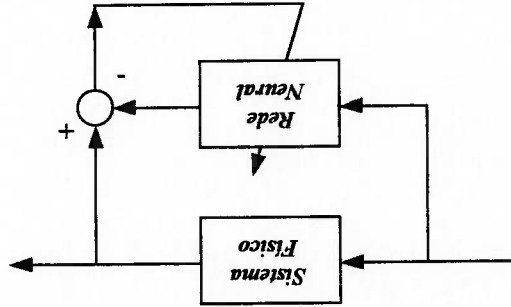


Fig. 2.5: Configurações de modelado de sistemas adaptativos

O conhecimento correto da dinâmica do processo a ser controlado, é o primeiro passo necessário em controle, normalmente um conhecimento *a priori* acerca do processo esta disponível na forma de um modelo parametrizado, onde os parâmetros podem ser estimados através dos dados de entrada e saída do processo. Os modelos dinâmicos são tipicamente não lineares e quando linearizados eles são usualmente válidos só numa pequena faixa de operação. Tais modelos de processos são relativamente simples, por exemplo alguns sistemas de segunda ordem com atrasos no tempo são frequentemente adequados para representar um processo químico.

2.3.3 Identificação do Sistema

Para sistemas lineares ou que possam ser linearizados sobre uma faixa de operação, a teoria convencional de controle linear é adequada para a maioria das aplicações. Porém, muitos sistemas complexos são não lineares e requerem um tratamento matemático muito sofisticado. Como são necessários parâmetros para modelagem matemática e simulação, muitos sistemas complexos não estão usualmente disponíveis, assim uma determinação experimental das características do sistema frequentemente se faz necessária.

As vezes é necessário introduzir atrasos de tempo Δ dentro da configuração, no sentido de modelar o tempo finito que é requerido para que o sinal se propague através do processo físico (TSOUKALAS e UHRIG, 1997).

Depois que a rede foi treinada, a relação entre as entradas e saídas reflete o sistema físico modelado, desde que as variáveis tenham sido treinadas sobre uma faixa dinâmica apropriada.

Segundo NARENDRA e PARATHASARATHY (1990), O problema de reconhecimento de padrões é um exemplo típico de identificação de sistemas estáticos.

Conjuntos compactos $U_i \subset \mathbb{R}^n$ são mapeados dentro de elementos $y_i \in \mathbb{R}^m$; ($i = 1, 2, \dots$) no espaço de saída por uma função de decisão P . Os elementos de U_i denotam os padrões de vetores correspondentes à classe y_i . Em sistemas dinâmicos, o operador P define uma planta dada, que é definida implicitamente por pares de entrada-saída de funções dependentes do tempo $u(t), y(t), t \in [0, T]$. Em ambos casos o objetivo é determinar P de modo que:

$$\| \hat{y} - y \| = \| P(u) - P(n) \| \leq \varepsilon \quad (2.32)$$

para algum valor desejado $\varepsilon > 0$, $P(n) = \hat{y}$ denota a saída da identificação do modelo e daqui $\hat{y} - y \equiv \varepsilon$ é o erro na saída gerada por P e a saída observada y .

Em sistemas onde os princípios dos modelos não estão disponíveis, é necessária a utilização de identificação não paramétrica tipo *black box*, que será tratada com mais detalhe no capítulo 3.

2.3.3.1 Identificação não Paramétrica

A identificação não paramétrica permite a modelagem tipo *black-box* entre relações de entrada-saída. A modelagem não paramétrica com redes neurais é uma extensão não linear do problema da identificação do sistema. Para tal instância um filtro transversal adaptativo, quando empregado como uma rede neural torna-se uma rede de *finite impulse response* (FIR), o que é uma identificação não paramétrica do sistema. A rede é provida

com um vetor de entrada de amostras ponderadas passadas das variáveis, servindo como uma excitação dinâmica do fenômeno. O número de amostras deve ser o suficiente para prover um intervalo de tempo através do qual o sinal anterior x_{k-n} não tenha qualquer efeito significativo sobre a resposta no tempo k .

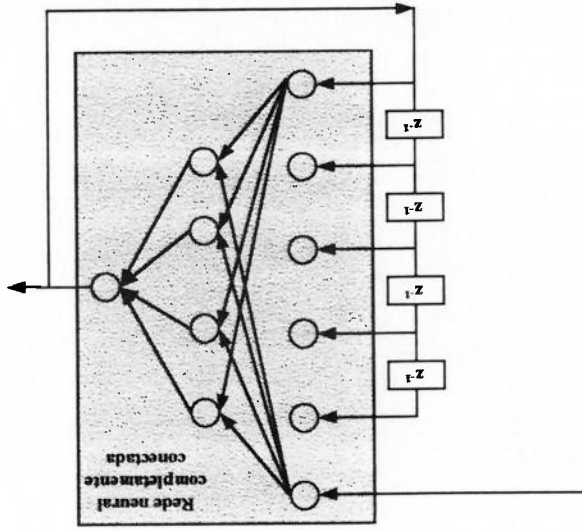
2.3.3.2 Identificação Paramétrica

A identificação paramétrica, identifica características estruturais e valores de parâmetros para modelos de sistemas do mundo real. Isto inclui identificação da estrutura do modelo (na forma de equações diferenciais lineares e não lineares, assim como estimação de parâmetros onde o modelo da estrutura é conhecido). Redes neurais treinadas através de aprendizado supervisionado podem ser usadas para as duas formas de identificação da estrutura e estimação de parâmetros, assim os identificadores estruturais podem ser treinados para selecionar elementos de um modelo da estrutura de um conjunto predeterminado. Este tipo de identificação estrutural com redes neurais requer que um espaço parecido de estruturas seja conhecido antecipadamente.

2.4 Modelos de Sistemas Dinâmicos

Muitos sistemas têm um comportamento temporal dinâmico e versões de redes neurais com atrasos de tempo das sinais de entrada e/ou saída são necessárias para modelar apropriadamente o sistema (TSOUKALAS e URHIG, 1997).

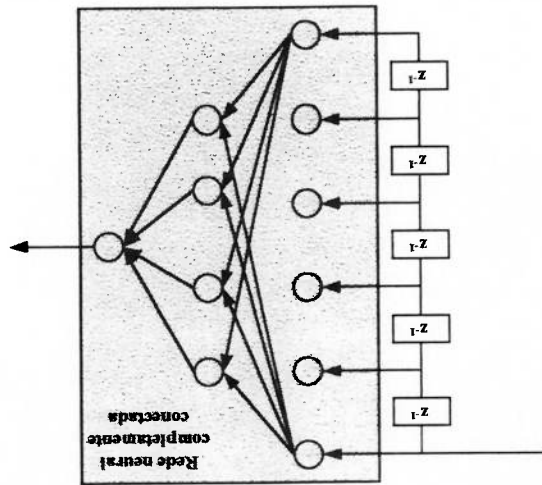
Fig. 2.7: Realimentação da saída com atrasos de tempo para outorgar dinâmica à rede



treinamento.

Quando a saída atual do sistema dinâmico depende das saídas prévias, a conexão recorrente das versões atrasadas são realimentadas às entradas como mostra a (Fig. 2.7). Esta topologia é conectada em paralelo com o sistema a ser modelado, possibilitando o

Fig. 2.6: Atrasos na entrada para a implementação da dinâmica à rede



2.5 Software para Simulação

A seguir mostra-se as técnicas utilizadas para simular as redes neurais artificiais empregando metodologias convencionais de programação. A rede utilizada compartilha os conceitos básicos de elementos de processamento (PE) distribuídos e fortemente interconectados.

Como pretende-se utilizar o simulador em sistemas embarcados, é importante considerar os aspectos de transportabilidade e reutilização do código desde o primeiro momento do projeto do simulador.

2.5.1 Objetivos para o Projeto do Simulador

Para simular redes neurais é necessário projetar o software de simulação de modo que o tamanho da rede possa ser determinado dinamicamente, assim como a capacidade de se especificar o número de neurônios necessários, o que é muito importante. A justificação desta observação é baseada na ideia de não se reprogramar e recompilar uma aplicação simplesmente porque se deseja mudar o tamanho da rede. Para isto se fará uso da alocação dinâmica de memória para realizar as estruturas de dados da rede (DACONTA, 1995).

É preciso escrever um programa para que o microprocessador compartilhe seu tempo com todos os neurônios da rede durante um certo período e o cálculo das somas de produtos faz-se usando números de ponto flutuante. Portanto, o microprocessador terá duas operações de ponto flutuante (uma multiplicação e uma adição) para cada entrada das unidades da rede. Devido ao grande número de nós da rede é fácil ver que o

computador deve ser capaz de realizar vários milhões de operações em ponto flutuante por segundo (MFLOPS).

2.5.2 Estruturas de Dados Baseados em Arrays

Uma vez que os dados serão processados como somas de produtos (ou como o produto escalar de dois vetores), estes estarão dispostos em grupos de matrizes (*arrays*) sequenciais lineares, cada um dos quais contendo dados homogêneos. Os dados destes *arrays* são dispostos sequencialmente para que correspondam-se analogamente, ou seja a saída da primeira unidade de entrada estará armazenada na primeira posição do *array* saídas, a segunda estará na segunda, e assim sucessivamente. De maneira similar, o peso associado à conexão entre a primeira unidade de entrada e a unidade w_{11} , estará situada no primeiro valor da *i-ésima* matriz de pesos.

Logo após ter definido as estruturas implementação dos cálculos de entrada em todos os nós da rede, pode-se estender estas estruturas básicas para realizar toda uma rede. Devido às estruturas de dados baseadas em *arrays* tendem a ser mais eficientes realiza-se a construção dos pesos de conexão e as saídas dos nós na forma de *arrays* reservados dinamicamente.

Similarmente, qualquer parâmetro adicional que poderia ser necessário para as redes, e que esteja associado a conexões individuais também será modelada empregando *arrays* que coexistam com os *arrays* de pesos de conexão.

É fácil criar um modelo adequado para a estrutura integrada da rede se adotamos as seguintes regras:

- As unidades da rede são sempre vistas como camadas de unidades que possuem características similares, inclusive no caso em que exista só uma unidade em algumas camadas;
- Todas as unidades de qualquer camada devem ter sido processadas completamente antes que o microprocessador possa começar a simular unidades em outra camada;
- O número de camadas que admitirá o simulador é definido, e estará limitado unicamente pela memória disponível, e
- O processamento que se fará em todas as camadas implicará normalmente às conexões de entrada de um nó, e só excepcionalmente implicará às conexões de saída de um nó.

Baseados nestas regras supõe-se que será uma camada, a estrutura da rede que ligará as unidades entre si. Então, uma única camada estará formada por um registro que contém os ponteiros dos distintos *arrays* que contém a informação acerca dos nós desta camada. Esse modelo de camadas é mostrado na (Fig. 2.8).

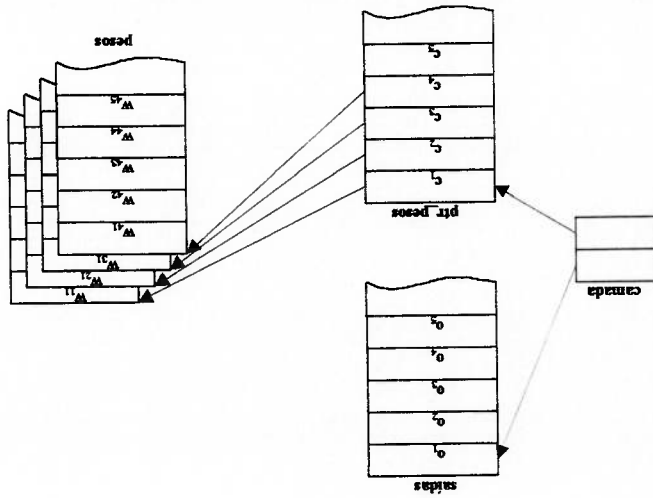


Fig. 2.8: Modelagem das estruturas das camadas.

Observe-se que o registro da camada vai localizar diretamente a saída de cada nó, através de um *array* intermediário de ponteiros. A razão pela qual se utiliza esta estrutura intermediária está relacionada com a otimização das estruturas de dados para um cálculo eficiente do valor total de saída de cada nó. Como todos os nós de qualquer camada produziram exatamente um valor de saída, as saídas de todos os nós de qualquer camada podem ser armazenados num único *array*. No entanto, cada nó também terá muitas conexões de entrada, onde cada um possuirá pesos únicos. Logo, deve-se construir as estruturas de dados para permitir que os *arrays* de pesos de entradas sejam identificados de maneira única com certos nós da camada. O *array* intermediário *ptr_pesos* satisfaz a necessidade de associar os pesos das entradas com o nó apropriado (através da posição do ponteiro dentro do *array* intermediário), e ao mesmo tempo permite modelar os pesos de entrada de cada nó em forma de *arrays* sequenciais, mantendo desse modo a eficiência desejada nas estruturas de dados da rede.

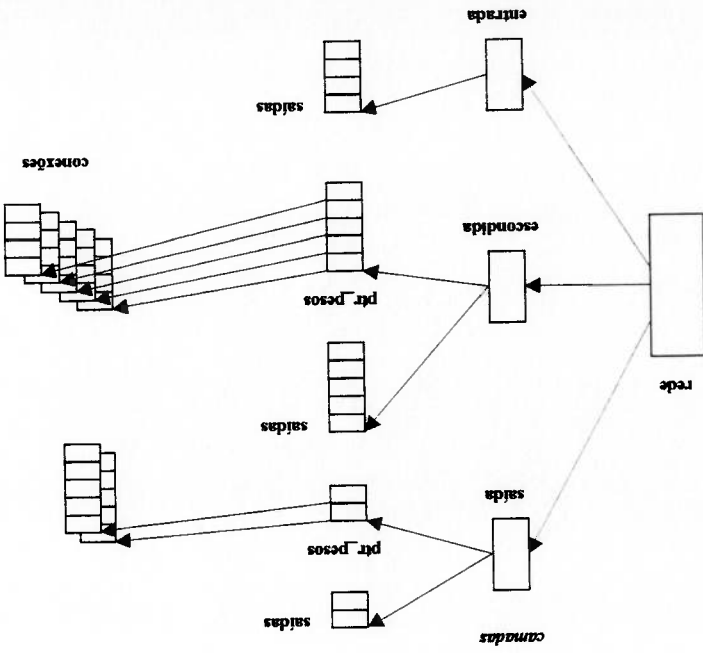


Fig: 2.9: Estrutura de dados para a implementação de redes neurais.

Por último considera-se a forma em que poderia se modelar uma rede completa, como decidiu-se que qualquer rede poderá ser construído começando de um conjunto de camadas, será modelada a rede na forma de um registro que contenha tantos dados globais como ponteiros para localizar os primeiros e últimos elementos de um *array* de registros de camadas reservadas mediante alocação dinâmica de memória. Esta aproximação permite criar uma rede de tamanho arbitrário ao mesmo tempo, oferecendo uma forma de acesso imediato às duas camadas mais frequentes: a camada de entrada e a camada de saída.

Esta representado na (Fig. 2.9) uma destas estruturas de dados, junto com a estrutura da rede. Ao modelar os dados desta maneira cria-se possíveis redes de qualquer tamanho e complexidade, otimizando as estruturas de dados para um funcionamento eficiente da malha interna durante a execução, isto é, para calcular a saída total para cada nó da rede.

2.5.3 Implementação do simulador de *Backproagation*

No projeto de implementação do *software* para a simulação de redes neurais artificiais foram tomadas em consideração as seguintes características:

- Possibilidade de se especificar o número e tamanho de todas as camadas;
- Possibilidade de se utilizar uma ou mais camadas escondidas;
- Capacidade de gravar e restaurar o estado de uma rede previamente treinada;
- Possibilidade de se ajustar parâmetros da rede e do simulador, e
- Visualização das informações dos resultados produzidos pela rede.

O simulador permite dois modos de operação, modo treinamento e modo teste. No modo treinamento é utilizado um arquivo que contém pares de padrões, cada padrão tem um conjunto de entradas seguido por um conjunto de saídas. Cada dado é separado por um ou mais espaços em branco, além disso há um outro arquivo de pesos.

Uma vez que o simulador atinge a tolerância do erro que foi especificado previamente, ou o máximo número de ciclos, o simulador grava o estado da rede, salvando todos os pesos das conexões no arquivo de pesos. Este arquivo pode subsequentemente ser utilizado para rodar o simulador em modo teste.

No modo teste o usuário provê dados de teste ao simulador no arquivo *test.dat*. Este arquivo só contém padrões de entrada, quando este arquivo é aplicado a uma rede previamente treinada, um arquivo de saída é gerado contendo as saídas produzidas pela rede para todos os padrões de entrada. Neste modo de operação a rede percorre todo o arquivo em um ciclo de operação cobrindo todos os seus padrões. Inicialmente a rede lê o arquivo de pesos para inicializar o estado da rede.

3 MODELAGEM BLACK-BOX NÃO LINEAR EM

IDENTIFICAÇÃO DE SISTEMAS

Neste capítulo é apresentada a fundamentação matemática da modelagem *black box* não linear. Esta técnica é utilizada nas tarefas de identificação e modelagem de sistemas através de redes neurais desenvolvidas nesta dissertação.

Uma estrutura black-box não linear para um sistema dinâmico, é um modelo de estrutura que é preparada para descrever virtualmente qualquer dinâmica não linear. Uma estrutura não linear pode ser vista como uma concatenação de dados observados a um vetor de regressão e um mapeamento não linear de um regressor de espaço a um espaço de saída (SJÖBERG et al., 1995 apud JUDITSKY et al., 1995).

3.1 Introdução

A chave do problema na identificação de sistemas é encontrar uma estrutura apropriada do modelo dentro do qual o melhor modelo seja obtido. Ajustar um modelo dentro de uma estrutura dada (estimação de parâmetros), é em muitos casos o menor problema. Uma regra básica na estimação é *não estimar aquilo que já se conhece*. Em outras palavras poderia-se utilizar o conhecimento prévio e físico do sistema quando selecionamos o modelo da estrutura. Isto é customizando para distinguir entre três níveis de conhecimento prévio (SJÖBERG et al., 1995), os quais são codificados seguindo o seguinte código de cores:

• Modelos white box: No caso do modelo ser perfeitamente conhecido, é possível ser construído completamente a partir do conhecimento prévio e conhecimento físico profundo.

• Modelos grey box: Neste caso algum conhecimento físico é disponível, mas alguns parâmetros deverão ser determinados de dados observados. Isto é muito utilizado considerando dois subcasos.

- Modelagem físico: O modelo da estrutura pode se construir sobre uma base física, que tem um certo número de parâmetros a serem estimados a partir dos dados.

- Modelagem semi-físico: O conhecimento físico é usado para sugerir algumas combinações não lineares da dados medidos do sinal. Esses novos sinais são sujeitos a modelos de estruturas de caracter *black box*.

- Modelos Black Box: O conhecimento físico profundo não é disponível para ser usado, mas a eleição do modelo da estrutura pertencem às famílias que são conhecidas por ter boa flexibilidade e tem demonstrado “sucesso no passado”.

3.1.1 Modelos Black Box

Para modelos *black box* lineares, a tarefa é realmente descrever/aproximar a resposta em frequência do sistema (ou resposta impulso), o que é o mapeamento de \mathfrak{R} a \mathfrak{R}^{pm} , onde p é o número de saídas e m é o número de entradas.

A situação não linear *black box* é muito mais difícil. A principal razão para isto é que estes de nenhuma maneira são excluídos, e um rico espectro de possíveis descrições dos modelos podem ser manejados.

3.1.2 Estruturas Não Lineares Black Box

O problema na identificação de sistemas começa observando-se as entradas $u(t)$, e as saídas $y(t)$ de um sistema dinâmico genérico mostradas a seguir:

$$u^t = [u(1), u(2), \dots, u(t)] \quad (3.1)$$

$$y^t = [y(1), y(2), \dots, y(t)] \quad (3.2)$$

Podem-se obter uma relação funcional entre observações no passado $[u^{t-l}, y^{t-l}]$ e saídas futuras, $y(t)$:

$$y(t) = g(u^{t-1}, y^{t-1}) + v(t) \quad (3.3)$$

O termo aditivo $v(t)$ indica que a próxima saída $y(t)$ não será uma função exata dos dados passados. No entanto o sucesso está em que $v(t)$ seja pequeno, de maneira a obter $g(u^{t-1}, y^{t-1})$ como uma boa predição de $y(t)$, obtida de dados passados.

A equação (3.3) é um modelo geral de sistemas dinâmicos em tempo discreto. Como os sistemas estáticos podem ser vistos como um caso particular de sistemas dinâmicos, o presente trabalho focaliza-se em sistemas dinâmicos.

Agora para encontrar a função g na equação (3.3), se terá que procurar g dentro duma família de funções. Parametrizando esta família de funções com um parâmetro dimensionalmente finito, que é o vetor θ , temos:

$$g(u^{t-1}, y^{t-1}, \theta) \quad (3.4)$$

Parametrizando a função g com um vetor θ dimensionalmente finito é usualmente uma boa aproximação. Depois de ter escolhido a estrutura a ser usada e tendo coletado o conjunto de dados $[n_N, \gamma_N^N]$, a qualidade de θ pode naturalmente ser acessada por médio do ajuste entre o modelo e o registro de dados:

$$(3.5) \quad \sum_{t=1}^N \|y(t) - g(n_{t-1}, \gamma_{t-1}, \theta)\|_2^2$$

A norma é o atual meio para alcançar ou tentar alcançar o mínimo valor de θ , no entanto outros meios podem ser usados, mas muitos dos esquemas de identificação de sistemas seguem este conceito.

O modelo da estrutura da família da equação (3.4) é muito geral e por isso não é apto para escrever g como uma concatenação de dois mapeamentos: o primeiro toma os incrementos das observações passadas n_t, γ_t e os mapea dentro de um vetor $\phi(t)$ de dimensão fixa e o segundo leva este vetor para o espaço de saída:

$$(3.6) \quad g(n_{t-1}, \gamma_{t-1}, \theta) = \mathcal{G}(\phi(t), \theta)$$

onde:

$$(3.7) \quad \phi(t) = \phi(n_{t-1}, \gamma_{t-1})$$

Chamando este vetor como *vetor de regressão* e seus componentes como *regressores*. Um caso mais geral para a formação de regressores, são os auto-parametrizáveis, equação (3.8).

(3.10)

$$\hat{y}(t|\theta) = \theta^T \phi(t, \theta)$$

linear”.

Um caso especial da equação (3.9) é conhecido como modelo Box-Jenkins (BJ) se (A=1), modelo ARMAX (F=D=1), modelo *Output Error* (A=C=D=1), e o modelo ARX (F=C=D=1). O preditor associado com (3.9) pode ser visto como uma regressão “pseudo-

(3.9)

$$A(q)y(t) = \frac{B(q)}{C(q)}u(t) + \frac{D(q)}{C(q)}e(t)$$

família:

As estruturas *black-box* mais usadas na prática podem ser resumidas pela seguinte

3.1.3 Modelos Lineares *Black-box*

espaco de saída.

1. Como escolher o vetor de regressão $\phi(t)$ de entradas e saídas passadas, e
2. Como escolher o mapeamento não linear $g(\phi)$ de um regressor de espaco ao

problemas parciais para sistemas dinâmicos.

A escolha do mapeamento não linear na equação (3.4) foi descomposta em dois

explicitamente só quando for essencial.

Para resumir escreve-se como $\phi(t, \eta)$. As vezes $\eta = \theta$, e o vetor de regressão depende dos parâmetros do modelo. Para simplificar, o argumento extra η será usado

(3.8)

$$\phi(t) = \phi(u^{t-1}, y^{t-1}, \eta)$$

rede produza a saída estimada $\hat{y}(t-k|\theta)$. A resposta é que a saída do modelo apresentada

(3.11). Outra questão que se apresenta é como calcular a saída simulada $\hat{y}^n(t-k|\theta)$, se a

black-box $\hat{y}(t-k|\theta)$, de modo que pode-se escrever $\phi(t, \theta)$ no lugar de $\phi(t)$ na equação

serem incluídos. Dos três restantes todos estão baseados nas saídas prévias do modelo

dois primeiros $u(t-k)$ e $y(t-k)$, são variáveis mensuráveis e não causam problemas para

similares aos descritos anteriormente como regressores. Para o caso de entrada-saída, os

onde g é alguma função não linear parametrizada por θ , e os componentes de $\phi(t)$ são

$$\hat{y}(t|\theta) = g(\phi(t), \theta) \quad (3.11)$$

A estrutura utilizada neste classe de modelagem é do tipo:

3.2 Regressores para Modelos dinâmicos Não Lineares Black-Box

$\varepsilon^n(t-k) = y(t-k) - \hat{y}^n(t-k|\theta)$ Simulação de erros (associado com o polinômio D)

$\varepsilon(t-k) = y(t-k) - \hat{y}(t-k|\theta)$ Predição de erros (associado com o polinômio C)

$\hat{y}^n(t-k|\theta)$ Saídas simuladas de valores passados, só n (associado com o polinômio F)

$y(t-k)$ (associado com o polinômio A)

$u(t-k)$ (associado com o polinômio B)

Os regressores (os componentes de $\phi(t, \theta)$), são neste caso geral dados por:

na equação (3.11) é igual a $y_n^*(t|\theta)$, se todas as saídas medidas $y^*(t-k|\theta)$ nos regressores são substituídos pelo último valor calculado $y_n^*(t-k|\theta)$.

A seguir são descritos os modelos não lineares:

- Modelos NFIR, os quais usam só $u(t-k)$ como regressores;
- Modelos NARX, também chamados série-paralelos, usam $u(t-k)$ e $y(t-k)$ como regressores;
- Modelos NOE, também chamados paralelos, utilizam $u(t-k)$ e $y_n^*(t-k|\theta)$ como regressores;

- Modelos NARMAX, utilizam $u(t-k)$, $y(t-k)$ e $\varepsilon(t-k|\theta)$ como regressores; Modelos NBI, os quais utilizam $u(t-k)$, $y_n^*(t-k|\theta)$, $\varepsilon(t-k|\theta)$, e $\varepsilon_n(t-k|\theta)$ como regressores. Neste caso a saída simulada y_n^* é obtida como saída da equação (3.11), usando a mesma estrutura, substituindo ε e ε_n por zeros no vetor de regressão $\phi(t, \theta)$;

- Modelos de Espaço de Estado Não Linear, os quais usam componentes passados de saídas virtuais, por exemplo, valores do sinal em nós internos da rede.

3.3 Mapeamento Não Linear

Seja o seguinte mapeamento não linear:

(3.12)

$$g(\phi, \theta)$$

onde para qualquer θ que vá de \mathcal{M} a \mathcal{M} . Neste ponto não interessa como foi construído o vetor de regressão $\phi = (\phi_1, \dots, \phi_d)^T$. É só um vetor que pertence a \mathcal{M} . Então é natural pensar na família da função parametrizada como uma expansão da seguinte função:

(3.13)

$$g(\phi, \theta) = \sum \alpha_k g_k(\phi)$$

Defina-se a g_k como uma *função base*. Agora a pergunta chave é como escolher a

função base g_k .

Muitas das estruturas dos modelos *black-box* são compostas de g_k obtido por uma única "função base mãe" que genericamente é denotada por $k(x)$. Em tais situações

geralmente escreve-se:

(3.14)

$$g(\phi) = k(\phi, \beta_k, \gamma_k) = k(\beta_k(\phi - \gamma_k))$$

onde β_k e γ_k denotam parâmetros de diferente natureza. Tipicamente β_k é relacionado com a escala ou a alguma propriedade direcional de $g_k(\phi)$, e γ_k é algum parâmetro de posição ou translação.

3.4 Questões Acerca de Redes Neurais

Até agora só foram vistos os modelos das estruturas como *expansões de funções base*, mesmo que sejam como funções base ajustáveis. Tais estruturas frequentemente são referidas como *redes* desde que uma típica função base mãe k é repetida um grande número de vezes na expansão.

3.4.1 Redes Multicamadas

Sejam as saídas das funções base denotadas por:

$$\phi_{(2)}^k(t) = g^k(\phi(t)) = k(\phi(t); \beta^k, \gamma^k) \quad (3.15)$$

e coletando elas dentro do vetor:

$$\phi_{(2)} = [\phi_{(2)}^1(t), \dots, \phi_{(2)}^n(t)] \quad (3.16)$$

Agora, no lugar de tomar uma combinação linear dos $\phi_{(2)}^i$ como as saídas do modelo,

equação (3.13), todas elas serão tratadas como novos regressores e incluindo-os dentro de

outra "camada" de uma função base formando uma segunda expressão:

$$g(\phi, \theta) = \sum_{l=1}^l \alpha_{(2)}^l k(\phi_{(2)}; \beta_{(2)}^l, \gamma_{(2)}^l) \quad (3.17)$$

onde θ denota a coleção completa dos parâmetros envolvidos: $\alpha_{(2)}, \beta_{(2)}, \gamma_{(2)}, \alpha_{(2)}, \beta_{(2)}, \gamma_{(2)}$.

Dentro da terminologia usada em redes neurais, a equação (3.17) é chamada uma

rede neural com *duas camadas ocultas*. As funções base $k(\phi(t); \beta_{(2)}, \gamma_{(2)})$ então constituem a

primeira camada oculta, enquanto $k(\phi_{(2)}; \beta_{(2)}^l, \gamma_{(2)}^l)$ formão a segunda camada. As

camadas são chamadas "ocultas" porque elas não podem ser ajustadas explicitamente na

saída $g(\phi, \theta)$, equação (3.17), mas elas são de fato disponíveis para o usuário vide (Fig.

3.1), a questão de utilizar uma o mais camadas não é nada fácil. Em princípio para muitas

funções base uma camada oculta é suficiente para muitos sistemas.

flexibilidade na sua utilização.

que a estrutura seja consideravelmente mais complexa, contudo oferecem completa interior a um prévio instante de tempo t . Tais modelos dependem de regressores o que faz componentes $\phi_j(t)$ do regressor, no momento t são obtidos como o valor de algum nó

A (Fig. 3.2) mostra este tipo de rede. Este seria também o caso de alguns

$$\phi_k(t) = g(\phi(t-k), \theta)$$

(3.18)

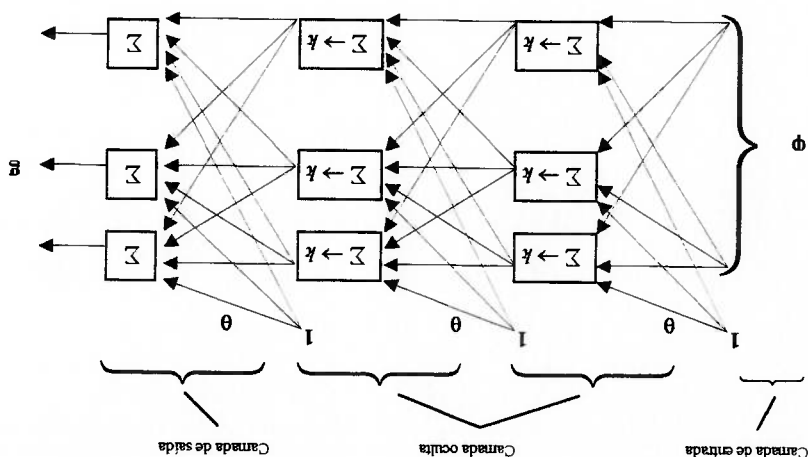
serem as saídas do modelo da estrutura no prévio instante de tempo:

recorrentes. Estas referem-se à situação de que alguns regressores usados no tempo t

Outro conceito muito importante para aplicações a sistemas dinâmicos são as redes

3.4.2 Redes Recorrentes

Fig. 3.1: Rede neural com duas camadas ocultas



As redes baseadas em espaço de estado tem maior poder comparadas com redes baseadas em entradas/saídas, embora a diferença não seja muito significativa no caso não linear.

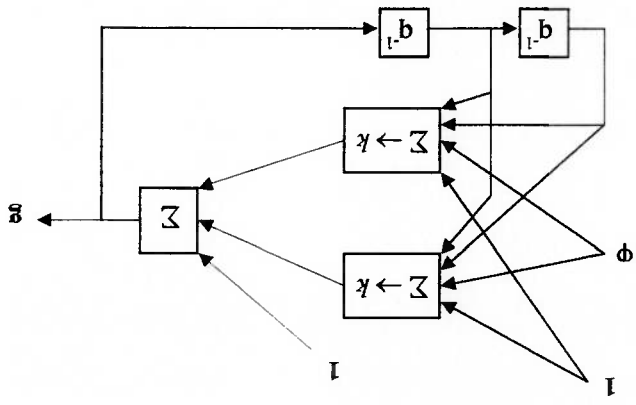


Fig: 3.2: Exemplo de rede recorrente, q^{-1} retarda o sinal uma amostra de tempo.

3.5 Redes Neurais como Modelos Estatísticos

A rede neural multicamada (MLP), é um modelo estatístico do mundo real (SWINGLER, 1996). Esta é construída pela sintonia de um conjunto de parâmetros próprios da rede. Esses parâmetros são conhecidos como *pesos* que descrevem um modelo, ou que vem do mapeamento de um conjunto de valores conhecidos como entradas para ser associadas a um conjunto conhecido como saídas. O processo de sintonia dos pesos para valores corretos de treinamento é levado a cabo fornecendo à rede exemplos de pares de entrada-saída através do modelo e ajustando os pesos no sentido de minimizar o erro entre a saída produzida pela rede e a saída desejada. Uma vez que os pesos foram ajustados, o modelo é capaz de produzir respostas para valores de entrada os quais não foram incluídos nos dados de treinamento.

4 ESTIMAÇÃO DE SÉRIES TEMPORAIS ATRAVÉS DE REDES

NEURAIS ARTIFICIAIS

4.1 Aspectos Teóricos das Séries Temporais

4.1.1 Funções Estáticas e Funções Dependentes do Tempo

As redes neurais são hábeis para aprender dependências temporais de um sistema. Um sistema que não contém dependências temporais é referenciado como estático.

Sistemas estáticos: Os sistemas estáticos não são afetados por estados prévios do sistema isto é, tais sistemas não têm memória e são afetados unicamente pelas entradas atuais. Isto não é uma regra de saída para sistemas que tem alguma mudança no tempo ou são afetadas por certas considerações temporais. Por exemplo, as vendas de uma loja naturalmente serão altas nos sábados, mas isto não é uma dependência das vendas nos dias anteriores, é somente um evento cíclico.

Sistemas dependentes do tempo: Tarefas dependentes do tempo são opostas as tarefas estáticas, ou seja não se poderia conhecer o efeito que teria uma entrada dada, na saída conhecida do estado atual ou de um comportamento passado do sistema; por essa razão o sistema ou um modelo parecido com sistema requer memória. Por exemplo mudanças nos preços de produtos poderiam ser vistos como tais sistemas; uma máquina para mistura de tintas é um caso similar, pois o efeito de adicionar uma certa cor à mistura dependerá das cores que foram previamente incluídas.

Por exemplo, na tarefa de aprendizado de uma onda senoidal para aprender esta função com respeito ao tempo, teria que ser solucionado $x_t = \sin(t)$. A sequência de aprendizado tendo como referência os valores prévios é equivalente à solução da equação diferencial de segunda ordem $x_t = f(x_{t-1}, x_{t-2})$. No entanto ao tentar aprender a função com respeito a t , teria que se começar com uma faixa de valores de entrada para o qual t é não limitado, assim teria que fazer-se a redução para a faixa de zero a um na entrada da rede, limitando t por causa da normalização dos dados. Em seguida surgem outras questões como por exemplo: quantos dados de treinamento deve-se utilizar, em que valor t deveria parar?. No lugar de isto, para tentar aprender a estimar valores subsequentes

crescimento de t ser muito grande.

Na teoria frequentemente é fácil definir uma série pelo tempo transcorrido t como uma função. Embora a prática não seja tão simples, e especialmente por isto são utilizadas redes neurais, utilizar t como uma entrada não é realista. Tal dificuldade deve-se de um lado a que t não é tão importante, e por outro lado, devido ao poder de

valores prévios na série (DEO e NAIDU, 1999).

Ha dois tipos de variáveis de referência ao definir-se uma série temporal: tempo e número, e pode-se defini-la como: 1. Em termos do tempo transcorrido desde o início da série e, 2. Em termos de um número desde o início da série ou como um número de

4.1.2 Definição de Termos para Séries Temporais

(SWINGLER, 1996).

“Diz-se que um sistema é dependente do tempo, se para uma dada entrada o sistema fornece uma resposta que depende das entradas prévias ou de estados prévios do sistema”

para os dois valores prévios nas séries, será preciso enxergar tão só exemplos de 2π para cobrir por completo a equação do espaço.

4.1.3 Dimensionalidade Inerente da Série

Quando um modelo para séries temporais é testada, a sua construção é importante caracterizar a dimensão inerente da série. A dimensão inerente define o número de palavras é equivalente à ordem de uma equação diferencial a qual define a série com respeito aos valores prévios. Isto não diz sobre quais valores são importantes, tão só indica quantos valores são necessários. No caso simples assume-se que o valor mais recente determina o próximo valor.

Uma série temporal na qual dois valores prévios são suficientes para determinar o próximo valor, como o exemplo da função seno, precisa-se do último valor para conhecer onde esta a onda e o valor anterior para conhecer se o deslocamento da onda foi na subida ou na descida. Esta série têm uma dimensão inerente de dois.

4.1.4 O Espaço de Estado das Séries Temporais

TAKENS (1981), provou que a partir de um vetor suficientemente grande levantando valores passados da série temporal, pode-se reconstruir completamente uma estrutura de um sistema dinâmico que produz a sequência original.

Desta maneira, tomando-se um grande número de valores passados, permite um número impressionantemente grande de graus de liberdade no sistema.

4.1.5 Redes Neurais de Atrasos de Tempo (Time Delay Neural Network)

A combinação de valores passados dentro de vetores estáticos é conhecido como “atrasos de tempo” ou “janelamento temporal”. Nesta rede é utilizada uma janela com atrasos de tempo que se movimenta por toda a série, e que serve como entrada para a rede. As redes que utilizam esta técnica (Fig. 4.1), são chamadas redes neurais de atrasos de tempo (TDNN).

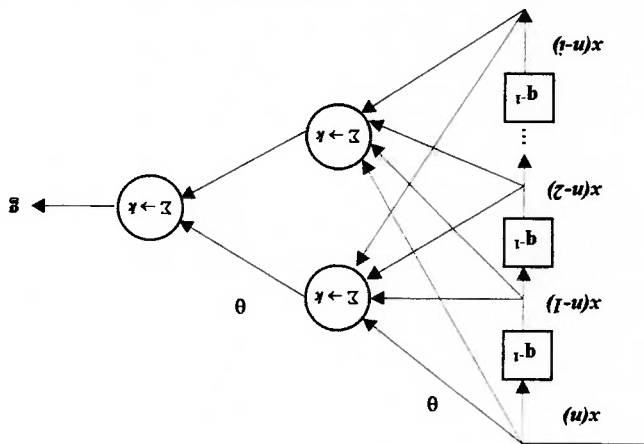


Fig. 4.1: Rede neural de atrasos de tempo.

As redes TDNN podem ser tomadas como uma extensão da rede neural multicamada (MLP) com uma linha de atrasos de tempo, ou uma extensão de um combinador linear com não linearidades.

A maior vantagem deste tipo de rede é treiná-la usando o algoritmo de *backpropagation*, o que será utilizado nos experimentos desenvolvidos neste trabalho. As principais aplicações das TDNN são:

- Redes TDNN para o reconhecimento de padrões temporais;

O esforço para utilizar redes neurais, ou modelos conexionistas nas tarefas de previsão já vem de algum tempo. WIDROW (1963) realizou previsões meteorológicas

4.2.1 Histórico

4.2 Aplicação de Redes Neurais Artificiais em Previsão

Estimar mais que um passo em frente numa série temporal com ruído, não linear ou caótica pode ser uma tarefa perigosa. Um método simples toma a última saída e a realimenta de novo como uma entrada no próximo instante de tempo. Os erros naturalmente são ampliados e assim é fácil que a rede comece a crescer exponencialmente no tempo. REFENES (1991) apresentou uma modelo para mapear uma janela com atrasos de tempo na entrada a uma janela com atrasos na saída, contendo mais que uma predição, esta metodologia tem demonstrado bons resultados.

4.1.7 Estimando Mais que Um Passo em Frente no Futuro

As redes neurais com atrasos de tempo (TDNN) relacionam o vetor de entrada com atrasos de tempo e a saída estimada explicitamente. O espaço de estado é construído diretamente do vetor com atrasos de tempo. As redes recorrentes, no entanto constroem a representação do espaço de estado na camada oculta.

4.1.6 Modelagem de Espaço de Estado com Redes Neurais

- Redes TDNN para previsão não linear.
- Redes TDNN para identificação de sistemas não lineares, e

em San Francisco – EUA para o “*US Weather Bureau*” utilizando o modelo de redes denominado *Adaline*.

Mais recentemente, o modelo de Retropropagação tem sido escolhido por vários pesquisadores para realização de estudos de previsão. WERBOS e TITUS (1978) utilizaram o modelo de *backpropagation* para previsão de conflitos na América Latina, e posteriormente WERBOS (1989) utilizou o mesmo modelo para produzir as previsões oficiais de energia da EIA (*Energy Information Administration – EUA*) e ainda segundo NASH (1990), a *Frontier Financial Corp.* em *Houston* utiliza redes neurais quando compra e vende em mercados futuros tais como moedas, óleo cru e apólices do tesouro americano.

RUMELHART et al. (1990) escolheram a série temporal de manchas solares, usada como “benchmark” por vários analistas de séries temporais, para realização de experimentos.

4.2.2 Arquitetura

As redes utilizadas na tarefa de estimação foram as redes neurais com atrasos de tempo (TDNN), utilizando o algoritmo de aprendizado de *backpropagation*, conforme se mostra na (Fig. 4.1).

O uso de não linearidades nas unidades de processamento distingue a abordagem neural dos modelos lineares tais como os auto-regressivos. Assim, podemos imaginar o funcionamento da rede TDNN, como uma espécie de filtro adaptativo não linear.

Para a previsão dos esforços produzidos nas linhas de ancoragem dos FPSO, não há uma obrigatoriedade quanto à rede a ser utilizada. Podemos escolher dentre diversos tipos de funções de ativação, conectividade e aprendizado da rede. Na especificação do tipo da função de ativação pode-se optar entre funções de ativação lineares, sigmoidais, tangente hiperbólica, *hard limit*, etc. A topologia também pode variar, como por exemplo considerar mais de uma camada escondida. No que diz respeito ao modelo de aprendizado, apesar de ser utilizado o método de *backpropagation*, que constitui-se num método de aprendizado por minimização de erro, pode-se obter diferentes variações através de modificações no algoritmo original.

4.2.3 Escolha da Topologia.

Tipicamente a entrada de uma única série temporal dentro de uma rede neural é feita como se mostra na figura (4.2). A variável é amostrada numa frequência apropriada para evitar o *aliasing*, e as amostras sequenciais são introduzidas na camada de entrada de maneira similar a um filtro transversal. A cada incremento de tempo Δt , uma nova amostra é introduzida no neurônio da camada de entrada que está mais para a direita e o valor que está mais à esquerda é descartado.

Nesse sentido a rede será treinada para prever o próximo valor no tempo da série temporal baseada nos valores das amostras anteriores.

Diferentemente de outros métodos de previsão, onde se realizam testes de correlação para se indicar quais valores passados são mais significativos na determinação de um valor futuro, o mesmo não é frequentemente feito quando se trabalha com redes neurais. Assim a rede construída também levaria em conta os valores x_{t-2} e x_{t-3} , com a camada de

entrada assumindo a forma: $x_{t-6}, x_{t-5}, x_{t-4}, x_{t-3}, x_{t-2}$ e x_{t-1} . O argumento que justificaria tal procedimento é de que a própria rede deveria descobrir que a informação de x_{t-2} e x_{t-3} não é relevante na determinação de x_t . Assim, após o treinamento, os pesos das conexões que partem de x_{t-2} e x_{t-3} deveriam ser praticamente nulos, ou seus efeitos anulados durante a fase de execução (previsão).

Em contrapartida, espera-se um aumento no tempo de treinamento com a utilização de mais unidades de entrada.

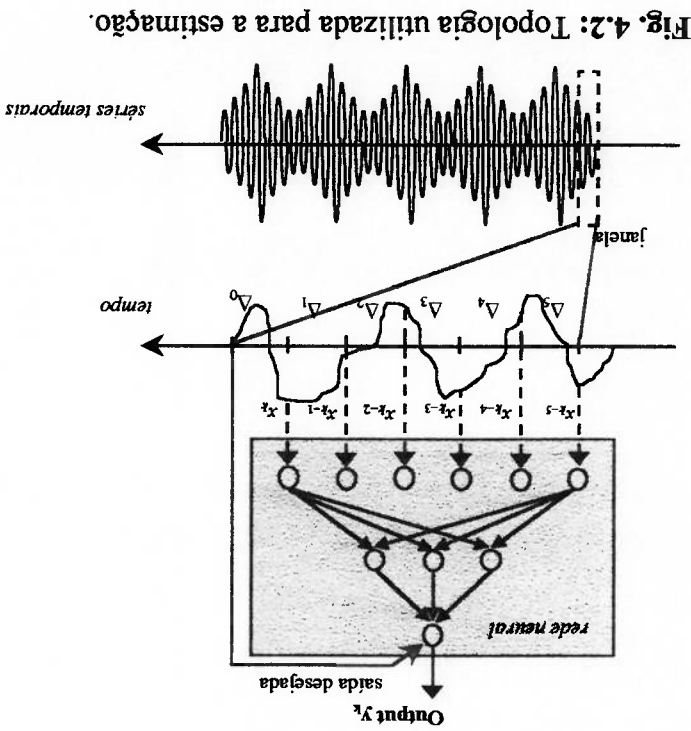


Fig. 4.2: Topologia utilizada para a estimação.

4.3 Modelagem de Redes Neurais para Previsão de Séries Temporais

4.3.1 Fatores Determinantes no Desempenho das Redes Neurais

O desempenho das redes neurais é, em geral avaliado em função de três fatores determinantes: convergência, generalização e escalabilidade.

A padronização dos dados da série podem frequentemente melhorar o desempenho das redes neurais através da remoção de características insignificantes do conjunto de dados que podem obscurecer a tarefa das redes. As informações relevantes das séries são geralmente independentes de valores de compensação "offset" e de desvios padrão.

4.3.2 Padronização

Escalabilidade, que diz respeito a ambos; convergência e generalização. É relativa ao procedimento de ajustar os parâmetros das redes neurais para que estas consigam convergir e generalizar bem. Sabe-se que para uma rede neural artificial possa ter um bom desempenho, ela deve ser grande o suficiente para aprender o problema (convergir), mas também pequena o bastante para generalizar bem.

Generalização: diz respeito à habilidade das redes neurais de reconhecer padrões fora do conjunto de treinamento. Esta é a principal propriedade que deve ser exigida de um sistema neural. Ela determina a quantidade de dados necessário para treinar o sistema para que este responda corretamente a padrões fora do conjunto de treinamento.

Convergência: diz respeito à capacidade das redes neurais de aprender um determinado conjunto de dados definido no procedimento de treinamento. A convergência é o que indica quando o procedimento de treinamento é capaz de aprender a classificação definida nos padrões apresentados, sob que condições das redes e sob quais exigências computacionais.

4.3.3 Janelamento dos Dados

A arquitetura das redes neurais é influenciada pelo método de identificação das regularidades da série. O método utilizado será o "janelamento" ou "windowing". A idéia básica deste método é utilizar duas janelas W_i e W_o de tamanhos fixos JF e JS respectivamente e deslocá-las no tempo (Fig. 4.3).

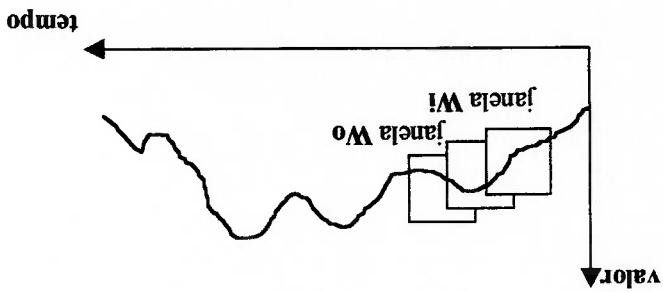


Fig. 4.3: Janelamento do sinal.

A utilização deste método é feita utilizando os pares de janelas W_i e W_o como vetores de treinamento, onde os dados das janelas W_i 's formam os vetores de entrada e os dados das janelas W_o 's formam os vetores alvo correspondentes. Com isto o número de neurônios de entrada e o número de neurônios da camada de saída são determinados pelo tamanho das janelas W_i e W_o respectivamente.

Durante o estudo de caso foram feitas estimações "single step" e "multi step" comparando a precisão da rede na estimação de horizontes maiores. Nestes experimentos foram manipulados vários parâmetros das redes neurais em busca de resultados mais otimizados.

derivada mas não criada.

Daqui que é importante lembrar que é impossível criar informação, ela só poderá ser

ordenação para ser mapeados em regiões comuns na saída.

objetos no espaço de entrada devem ocupar regiões em comum para sua própria

são mapeadas dentro de um novo espaço, que é o espaço de saída (ou de decisão). Os

seja representado como pontos em algum espaço multidimensional. As áreas deste espaço

As redes neurais codificam características que descrevem um objeto de modo que

4.3.5 Padrões de Qualidade

teste na realização dos experimentos na totalidade do trabalho.

formação do conjunto de teste. Este procedimento é adotado na geração do conjunto de

Uma prática freqüente é retirar uma amostra de 20% do universo de dados para a

período de 25 segundos.

comportamento do navio durante 125 minutos. As amostras foram tomadas com um

O universo dos dados é composto por 301 amostras, correspondentes ao

deste simulador são apresentados no capítulo 5.

navio e as forças nas linhas de amarração, maiores detalhes da modelagem matemática

oceânicos Dynasim (FUCATU, 1998) fornecendo a série temporal do movimento do

Os dados para treinamento das redes foram gerados através do simulador de sistemas

4.3.4 Coleta dos Dados

Nesse sentido deve-se treinar a rede com a quantidade correta dos dados. Dado que o sistema estudado pertence a um sistema não linear de sexta ordem, a modelagem para a estimação das séries temporais utilizaram o método de atrasos de tempo para reconstruir o espaço de saída a partir de vetores com atrasos de tempo Δt , definidos como:

$$y_i = (x_i, x_{i-1}, \dots, x_{i-m+1})^T \quad i=1, 2, \dots, N \quad (4.1)$$

onde τ é um atraso escolhido apropriadamente que multiplica o intervalo Δt , m é um inteiro relacionado à dimensão inerente da série (KULKARNI, 1997).

Na tarefa de estimação, a largura do vetor y_i determinará o tamanho da janela de entrada, onde $m = 6$, o intervalo de tempo Δt para os atrasos será igual a uma amostra (25 segundos) e $\tau = 1$. Obtendo o vetor de entrada para o instante i , temos:

$$y_i = (x_i, x_{i-1}, x_{i-2}, x_{i-3}, x_{i-4}, x_{i-5}) \quad (4.2)$$

Todos os experimentos são realizados na estimação dos esforços no cabo de ancoragem número um e do cabo *hawser*.

4.3.6 Treinamento das Redes Neurais para a Estimação de Séries Temporais

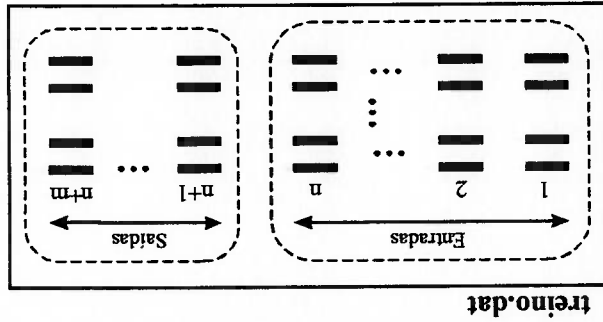
Uma vez que a rede neural foi projetada e os dados do conjunto de treinamento foram coletados e devidamente codificados, a rede está pronta para ser treinada. Uma

No projeto de uma rede para uma dada tarefa são várias as escolhas a serem feitas. Não existe um método definitivo que indique qual a rede ótima ou como se chegar à mesma. Para evitar uma explosão combinatória, os experimentos foram feitos mantendo-se alguns aspectos constantes, enquanto outros foram modificados para permitir análises

4.3.7 Validação dos Resultados

onde : n : número de entradas e m : número de passos em frente a estimar-se.

Fig. 4.4: Configuração do arquivo de treinamento.



em formato texto, como mostra a (Fig. 4.4).

Os arquivos de treinamento tem o seguinte formato: O número de colunas é igual ao número de entradas mais o número de saídas da rede separadas com espaços em branco

caso, requerendo assim ensaios de prova e erro. Para o processo de aprendizado realizaram-se vários experimentos pelo fato de não existir uma regra fixa que governe a configuração das redes neurais usadas para cada

durante a fase de treinamento.

única solução final deve ser escolhida de uma população de redes neurais produzidas

e comparações. É claro que essa abordagem não necessariamente leva à construção da rede ótima, mas da grandes chances para permitir construir uma rede que atenda de maneira mais satisfatória a tarefa de previsão. Mais do que isso, a observação focalizada sobre um aspecto de projeto permite especular sobre a influência do mesmo no comportamento da rede.

Assim, inicialmente realizou-se uma série de experimentos no sentido de orientar a escolha de uma rede que permitisse realizar bons ajustes das curvas dos esforços nas linhas de amarração. Em seguida verificou-se a adequação da rede e o desempenho da mesma na tarefa de previsão, nesse sentido para avaliar os resultados produzidos pelas redes neurais, foram realizados experimentos com modificações em:

- Número de neurônios na camada de entrada;
- Número de neurônios na camada de escondida;
- Estimação um passo em frente;
- Estimação dois passos em frente;
- Estimação três passos em frente, e
- Estimação seis passos em frente.

Para a estimação um passo em frente foram realizadas diversas modificações, como o número de neurônios na entrada e o número de neurônios na camada escondida, mantendo sempre um neurônio na saída. Por exemplo uma rede (5-35-1) indica 5 neurônios na camada de entrada, 35 neurônios na camada escondida e 1 neurônio na camada de saída.

Inicialmente as redes treinadas no modo "single step", testando-se vários *frames* para a janela de entrada, confirmando-se que o tamanho escolhido na seção anterior, equação (4.2) com $m = 6$, apresenta a melhor precisão nos resultados como mostra a tabela 1.

Onde o erro médio absoluto é o valor absoluto médio dos erros em todos os padrões de teste. Mostra-se também o coeficiente de correlação que proporciona uma medição estatística da robustez da relação existente entre a entrada e a saída estimada. Este coeficiente está dentro da faixa de -1 a $+1$, o valor mais próximo a 1 mostra uma forte relação linear positiva, quando este coeficiente é zero então diz-se que não existe nenhuma relação linear.

TABELA 1: CONFIGURAÇÃO DAS REDES UTILIZADAS PARA A ESTIMAÇÃO NO MODO "SINGLE STEP"

Arquivo de treino	No. de camadas	Config. da rede	Coef. de aprendizado	Erro médio absoluto (%)	Coef. de correlação
3UM	3	3-8-1	0.3	14.91	0.95
4UM	3	4-10-1	0.15	10.78	0.96
6UM	3	6-12-1	0.2	7.61	0.97
7UM	3	7-15-1	0.25	8.69	0.96
8UM	3	8-16-1	0.3	8.42	0.96
12UM	3	12-23-1	0.3	8.48	0.96

Nas curvas a seguir são mostradas os resultados para as 300 primeiras amostras para uma melhor visualização da estimação produzida pela rede junto com a série temporal real, de modo a fazer uma comparação qualitativa do desempenho das redes neurais artificiais utilizadas para a estimação.

	1 passo	2 passos	3 passos	4 passos	5 passos	6 passos
Coeff. de correlação	0.99	0.97	0.93	0.88	0.84	0.81
Erro médio absoluto (%)	5.47	8.68	12.65	15.99	18.32	19.77
Dados dentro do 5% do erro	100	98.62	91.03	83.45	75.86	72.07

TABELA 2: CONFIGURAÇÕES PARA ESTIMAÇÃO NO MODO "MULTI STEP"

As estimações produzidas com múltiplos passos em frente são mostradas nas (Fig. 4.7) e (Fig. 4.9), apresentando estimações de 3 e 6 passos em frente, respectivamente, também é mostrado a diferença entre a saída real e a saída produzida pela rede. Neste ponto é importante mencionar que o incremento do tamanho da janela de saída melhorou notavelmente os resultados, e é assim que o coeficiente de correlação alcançou o valor de 0,99 na estimação um passo em frente como mostra a tabela 2

A (Fig. 4.5) mostra a estimação dos esforços (cabo 1) um passo em frente, neste caso a rede realiza a previsão com bastante precisão e consegue representar a série temporal sem maiores problemas, demonstrando que a escolha do tamanho da janela de entrada foi correta. Também é mostrado o erro produzido na estimação (Fig. 4.6), aqui o erro é definido como a diferença entre a saída real e a saída estimada.

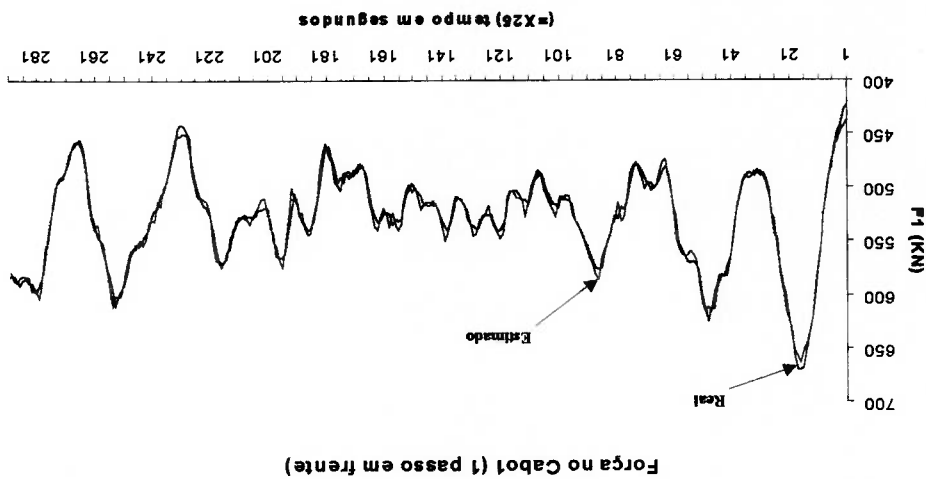


Fig. 4.5: Saída real e saída produzida pela rede (estimação 1 passo = 25 seg.).

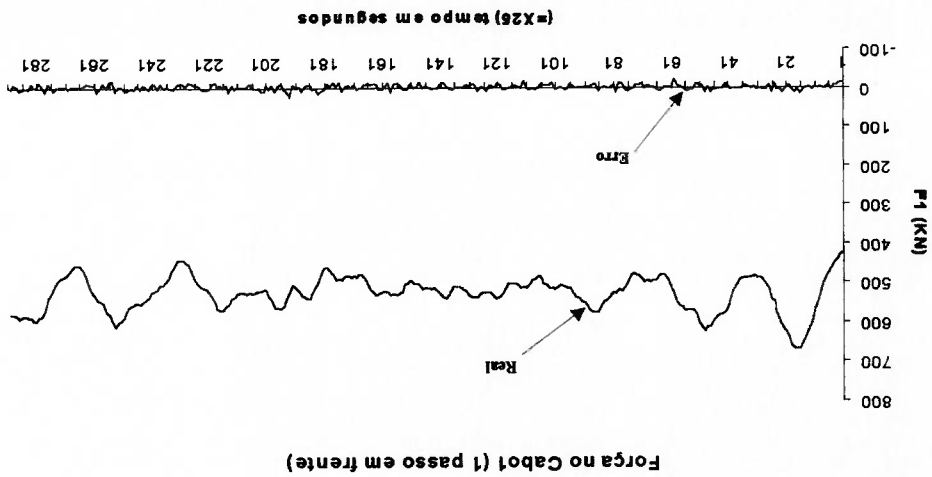


Fig. 4.6: Saída real e diferença (real-estimada) na estimação 1 passo em frente.

A estimação de múltiplos passos em frente (*multi step*), é mostrada na (Fig. 4.7) para a previsão de três passos em frente. A representação da série temporal produzida pela rede continua sendo excelente, embora seja observada uma leve queda na precisão alcançada pela rede (Fig. 4.8), devido ao incremento no horizonte de previsão.

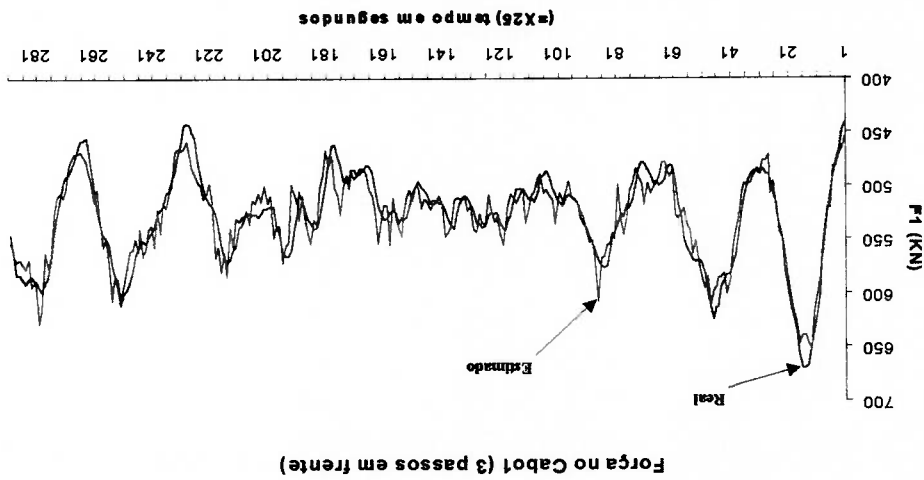


Fig. 4.7: Saída real e saída produzida pela rede (estimação 3 passos = 75 seg.).

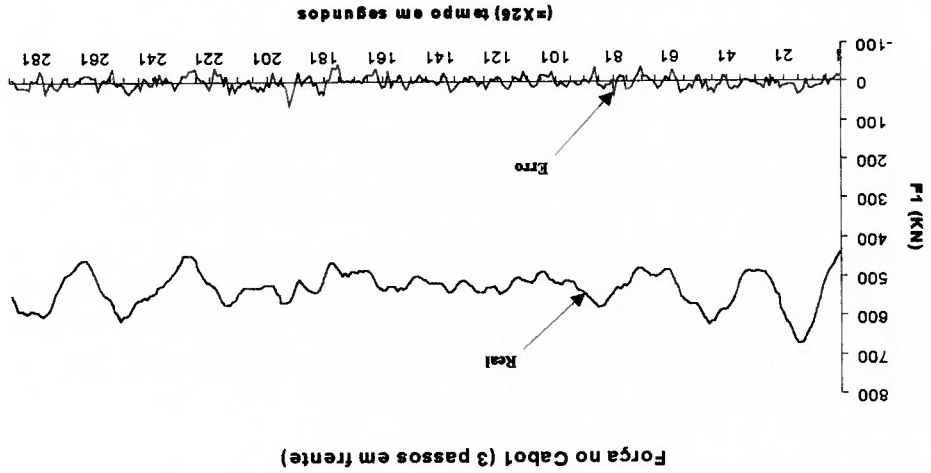


Fig. 4.8: Saída real e a diferença (*real-estimada*), estimação 3 passos em frente.

Continuando no modo "multi step" a (Fig. 4.9), mostra a estimação olhando no futuro seis passos em frente, e o desempenho da rede continua sendo bom. Como era de esperar existe uma diminuição na precisão da rede, em consequência observa-se um incremento não muito significativo do erro (Fig. 4.10).

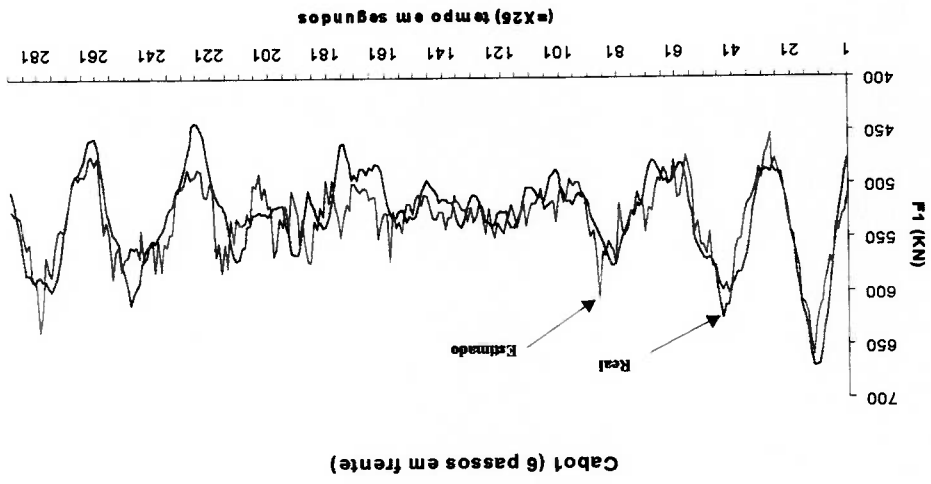


Fig. 4.9: Saída real e saída produzida pela rede (estimação 6 passos = 150 seg.).

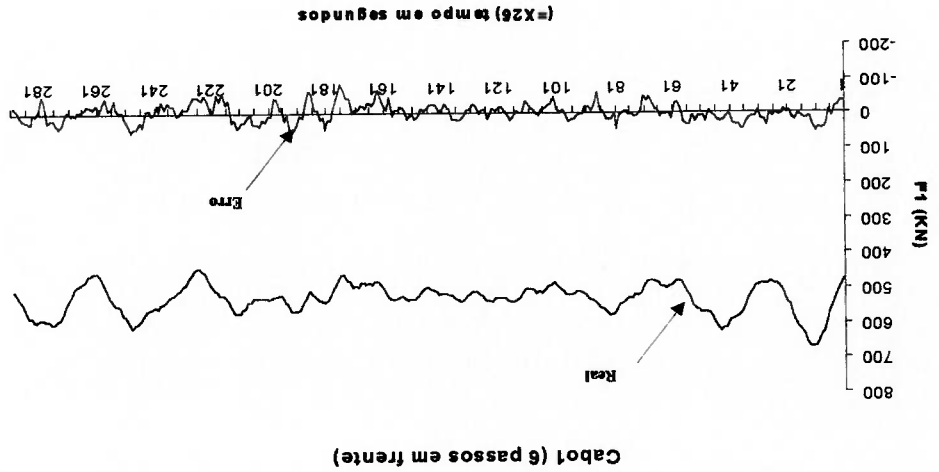


Fig. 4.10: Saída real e diferença (real-estimada) na estimação 6 passos em frente.

A estimação um passo em frente dos esforços produzido no *hawser* são mostrados na (Fig. 4.11), neste ponto é importante ressaltar que precisão obtida pela rede é menor (Fig. 4.12) do que a alcançada no caso do Cabo 1, devido à função complexa que representa o cabo *hawser*. Isto é consequência do acoplamento mútuo que existe entre os dois navios, além de representar uma função com múltiplas descontinuidades, provenientes das simplificações realizadas no modelo matemático do simulador, o qual não acontece num sistema físico real.

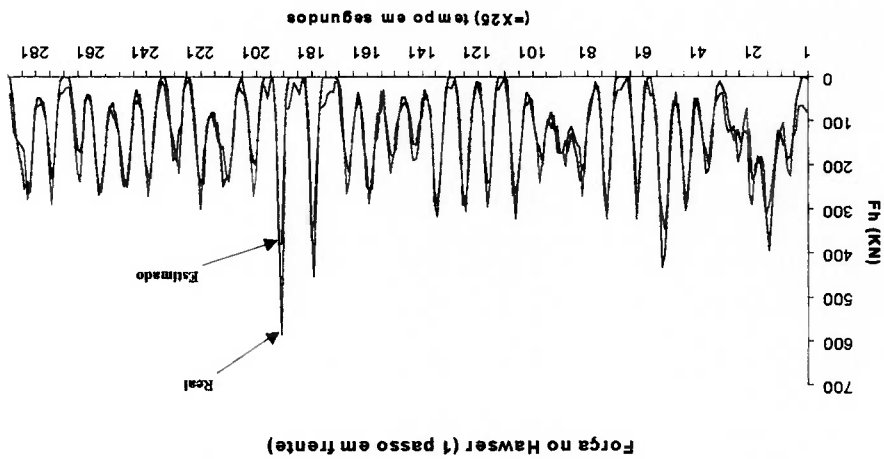


Fig. 4.11: Saída real e produzida pela rede no *hawser* (estimação 1 passo = 25seg.).

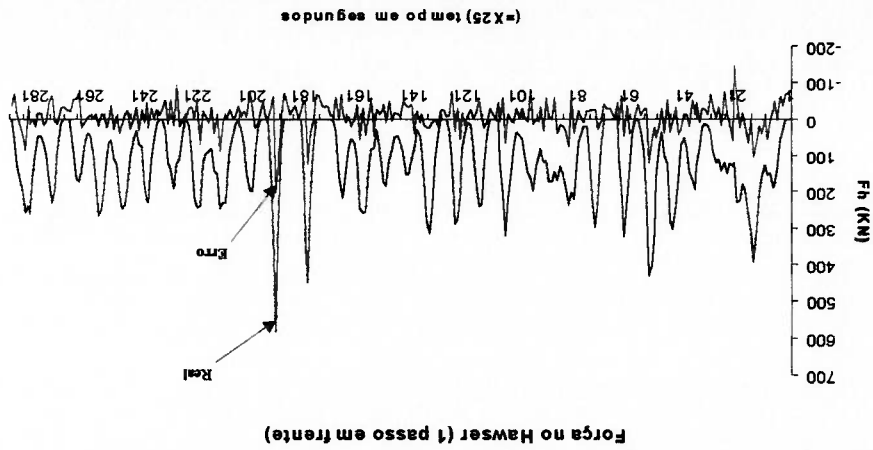


Fig. 4.12: Saída real e diferença (real-estimada) no *hawser*.

A seguir são mostrados os diagramas de dispersão das estimações realizadas, isto pode ser visto como uma representação gráfica do coeficiente de correlação. As (Fig. 4.13) e (Fig. 4.14) mostram os diagramas de dispersão para a estimação 1 e 3 passos em frente respectivamente. Aqui pode-se observar claramente que a rede conseguiu uma forte relação linear entre a entrada e a saída desejada.

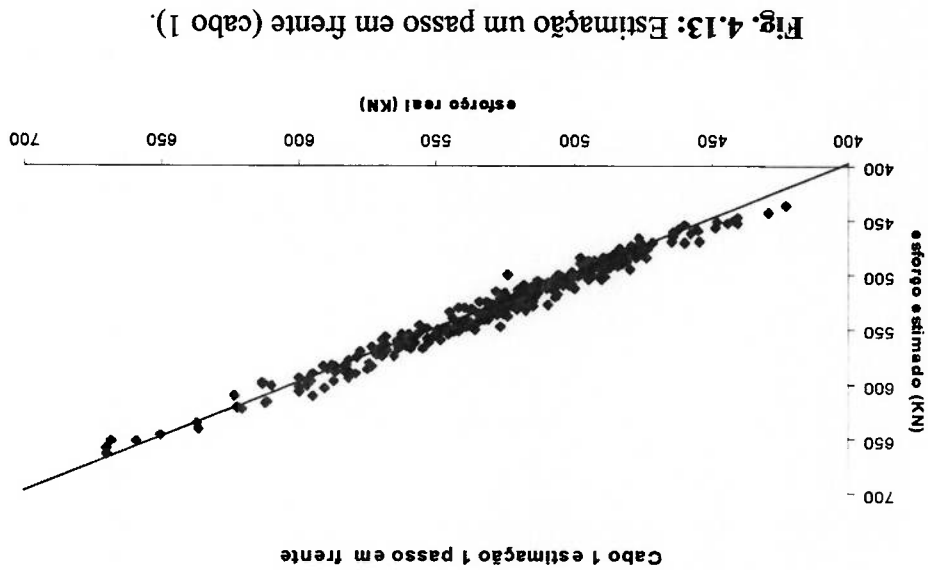


Fig. 4.13: Estimação um passo em frente (cabo 1).

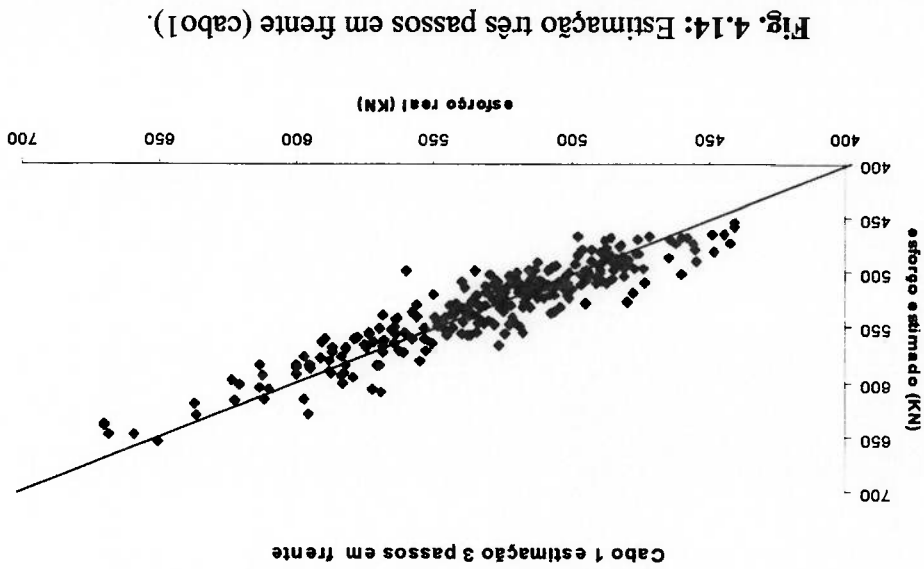


Fig. 4.14: Estimação três passos em frente (cabo 1).

A (Fig. 4.15) mostra a curva de dispersão para a estimação seis passos em frente, observando-se que existe uma maior dispersão dos pontos conforme incrementa-se o horizonte de estimação. No caso da estimação dos esforços no *hawser*, existe uma leve dispersão dos pontos, mas ainda a rede consegue representar uma relação linear

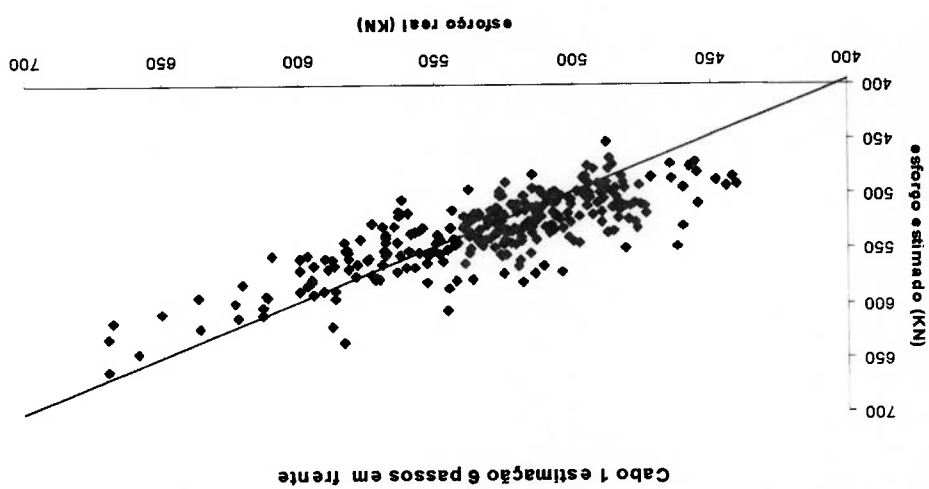


Fig. 4.15: Estimação seis passos em frente (cabo 1).

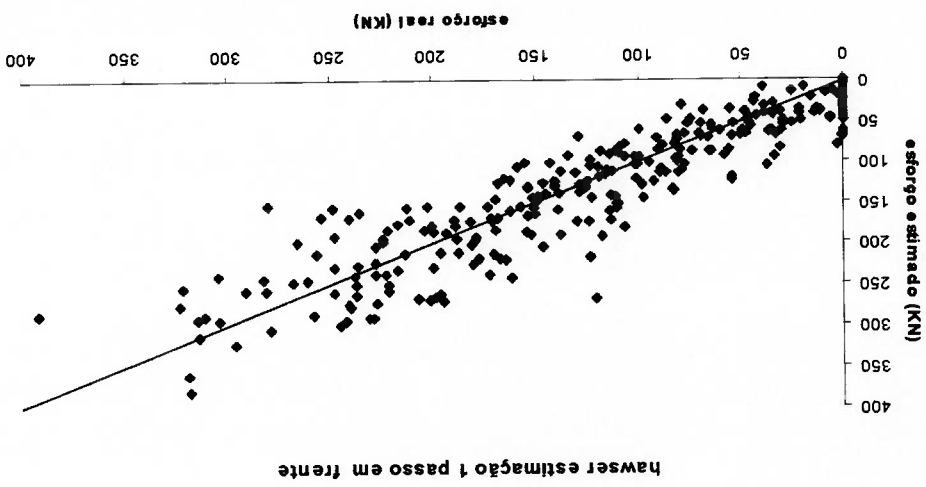


Fig. 4.16: Estimação um passo em frente (*hawser*).

A precisão atingida pelas redes e apresentada a seguir na forma gráficos de barras, mostrando o número de padrões de teste que estão dentro de uma faixa de erro para as estimações: um, dois, três, quatro, cinco e seis passos em frente. Para uma melhor visualização foram definidas os seguintes intervalos de erro: [0-5%], [5-10%], [10-20%], [20-30%] e [30-100%].

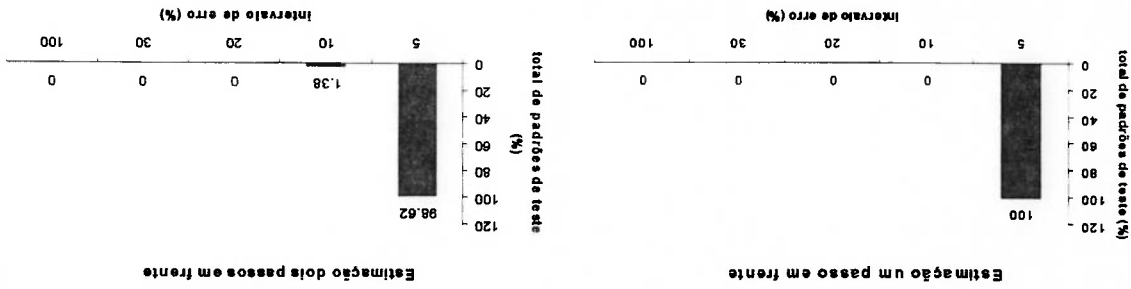


Fig. 4.17: Precisão atingida pelas redes (a) um passo e (b) dois passos em frente.

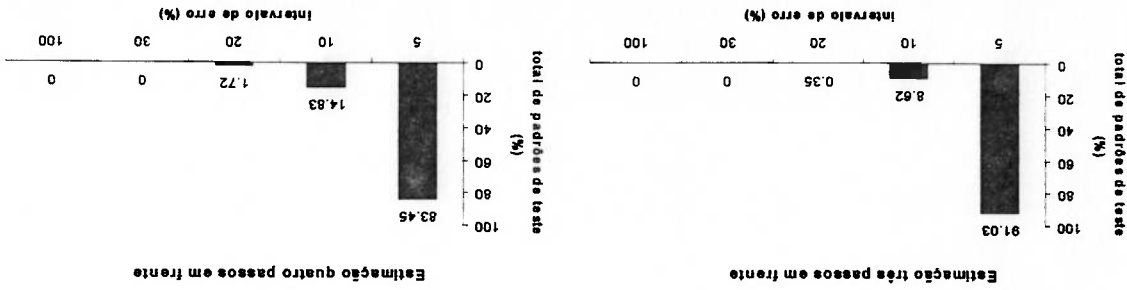


Fig. 4.18: Precisão atingida pelas redes (a) três passos e (b) quatro passos em frente.

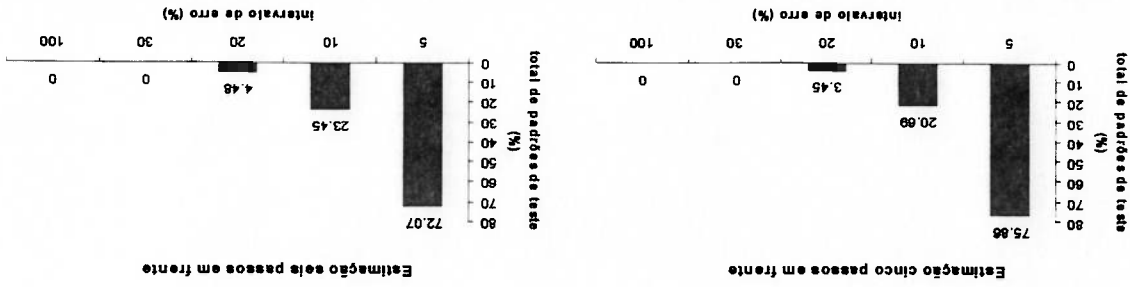


Fig. 4.19: Precisão atingida pelas redes (a) cinco passos e (b) seis passos em frente.

5 MODELAGEM E IDENTIFICAÇÃO DO SISTEMA OCEÂNICO

5.1 Modelo Matemático

Nesta primeira parte é descrita a formulação geral do modelo matemático e a dinâmica do movimento de navios ancorados (FUCATU, 1998), e em seguida são detalhadas cada uma das equações matemáticas das forças ambientais, as que serviram como geradoras do conjunto de treinamento das redes neurais para a tarefa de modelagem dinâmica.

5.1.1 Sistema de Coordenadas

Para o estudo dos movimentos do navio adotou-se três sistemas de coordenadas como ilustra a (Fig. 5.1). O primeiro eixo $OXYZ$ está fixo à terra. O segundo e o terceiro Gx_1y_1 ($i=1,2$), são solidários à embarcação e seus eixos coincidem com os principais de inércia e a sua origem está localizada no centro de gravidade do navio.

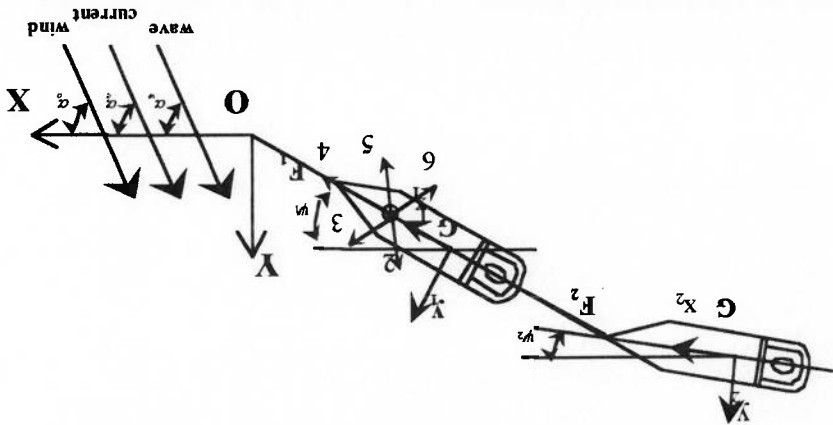


Fig. 5.1: Sistema de coordenadas

5.1.2 Equações de Movimento

A equação do movimento de cada navio em relação ao referencial inercial, já considerando as forças de reação hidrodinâmicas, são dadas por:

$$(m - m_{11})\ddot{u} = (m - m_{22})v\dot{r} - (m_{11} - m_{22})v^2\dot{r} + X \quad (5.1)$$

$$(m - m_{22})\dot{v} = (m_{11} - m)vr - (m_{11} - m_{22})u^2\dot{r} + Y \quad (5.2)$$

$$(I_z - m_{66})\dot{r} = N \quad (5.3)$$

onde: m é a massa do veículo; m_{ii} , $i = 1, 2, 6$ são, respectivamente, as massas adicionais em avanço, deriva e guinada; u e v são as velocidades da corrente relativas às direções GX e GY respectivamente; r é a velocidade angular; I_z é o momento de inércia em relação ao eixo GZ ; X , Y e N representam as forças externas totais respectivamente nas direções de avanço, deriva e guinada; o ponto sobre a variável representa a derivada em relação ao tempo.

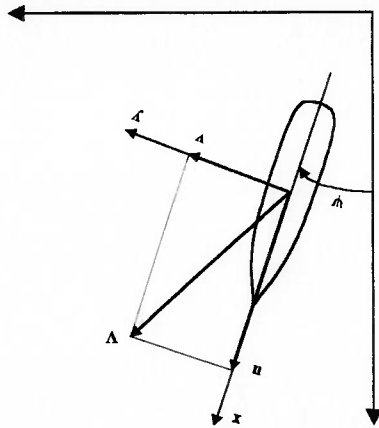


Fig. 5.2 : Posição do navio em relação ao referencial inercial

A posição e o aprornamento de cada embarcação em relação ao referencial inercial são obtidas pelas seguintes equações (Fig. 5.2):

$$x_o = n \cos \psi - v \operatorname{sen} \psi \quad (5.4)$$

$$y_o = n \operatorname{sen} \psi + v \cos \psi \quad (5.5)$$

$$\psi = r \quad (5.6)$$

onde x_o e y_o são as velocidades da embarcação nas direções OX e OY respectivamente, e ψ é o ângulo de aprornamento do veículo.

As componentes n_c e v_c da corrente são calculadas através de:

$$n_c = V_c \cos(\psi_c - \psi^T) \quad (5.7)$$

$$v_c = V_c \operatorname{sen}(\psi_c - \psi^T) \quad (5.8)$$

onde: V_c e ψ_c são a amplitude e a direção da velocidade da corrente, respectivamente.

As forças X , Y e o momento N , embora dependam do navio, podem ser expressas de uma forma geral, através do seguinte conjunto de equações:

$$X = X^g + X^h + X^o + X^i + X^j \quad (5.9)$$

$$Y = Y^g + Y^h + Y^i + Y^j + Y^k + Y^l \quad (5.10)$$

(5.11)

$$N = N^c + N^v + N^o + Y^v e^v + Y^h e^h + Y^i e^i$$

onde: $i = 1, 2$, referem-se, respectivamente, aos navios aliviador e ao navio FPSO-*Turret*; os índices c, v, o, h e i referem-se, respectivamente, à corrente, vento, onda, *hawser* e *turret*; e_h, e_v e e_i são, respectivamente, as distâncias entre o centro de gravidade e os pontos de aplicação das forças do *hawser*, do vento e do *turret*.

5.1.2.1 Força da corrente

As forças e os momentos hidrodinâmicos que atuam no navio, segundo TAKASHINA (1986), foram obtidas a partir de ensaios realizados com modelos reduzidos e são dadas por:

(5.12)

$$X^c = \frac{1}{2} \rho L D U^2 (X^n n^i + X^w w^i r^i)$$

(5.13)

$$Y^c = \frac{1}{2} \rho L D U^2 (Y^v v^i + Y^w w^i + Y^o o^i + Y^h h^i + Y^i i^i + Y^r r^i |r^i| + Y^v v^i |r^i|)$$

(5.14)

$$N^c = \frac{1}{2} \rho L D U^2 (N^v v^i + N^w w^i + N^o o^i + N^h h^i + N^i i^i + N^r r^i |r^i| + N^w w^i |r^i|)$$

onde: n^i, v^i e r^i são as velocidades não dimensionais de avanço, deriva e guinada,

respectivamente, obtidas através de:

(5.15)

$$n^i = A/n$$

(5.16)

$$v^i = A/v$$

$$r' = rL/A \quad (5.17)$$

$$V = \left(u^2 + v^2 \right)^{1/2} \quad (5.18)$$

As forças e os momentos devido à corrente são levadas em conta no modelo substituindo-se as velocidades absolutas de avanço e deriva, u e v nas equações (5.12, 5.13, 5.14) pelas correspondentes velocidades relativas, u_r e v_r do navio em relação ao meio fluido.

5.1.2.2 Força do vento

As componentes da força do vento atuantes na parte não submersa do casco e na superestrutura do navio são calculadas através das seguintes fórmulas:

$$X_v = \frac{1}{2} \rho_a C_v^x (\psi^w) (A_i + (H - D) B) W_w^2 \quad (5.19)$$

$$Y_v = \frac{1}{2} \rho_a C_v^y (\psi^w) (A_i + (H - D) L) W_w^2 \quad (5.20)$$

$$N_v = \frac{1}{2} \rho_a C_v^z (\psi^w) (A_i + (H - D) L) L V_w^2 - F_v^{(2)} x_v \quad (5.21)$$

onde A_i é a área transversal da superestrutura; A_l é a área longitudinal da superestrutura; H é o pontal; D é o calado; B é a boca e L é o comprimento do navio.

5.1.2.3 Força da onda

Foram consideradas parcelas de deriva média e deriva lenta e deriva da força da onda.

Força de deriva média das ondas. Depende da incidência da onda e é obtida para um

dado espectro a partir da seguinte expressão:

$$F_{dmi}^{\chi}(\chi) = 2 \int_{-\infty}^{\infty} S(\omega) \kappa_{dmi}^{\chi}(x, \omega) d\omega \quad (5.22)$$

onde: $F_{dmi}^{\chi}(\chi)$, $i = 1, 2, 6$ representam as forças/momento de deriva média na direção dos

eixos G_x , G_y e G_z .

Força de deriva lenta da onda. Para cada incidência de onda os espectros de forças de

deriva lenta podem ser obtidos pelo cruzamento espectral como mostrado abaixo:

$$Sx'_i(\chi, n) = 8 \int_{-\infty}^{\infty} S(\omega) S(\omega + \mu) \left[F_{dmi}^{\chi} \left(\chi, \omega + \frac{\mu}{2} \right) \right]^2 d\omega \quad (5.23)$$

A expressão (5.23) pode ser simplificada para:

$$Sx'_i(\chi, \mu) = 8 \int_{-\infty}^{\infty} S^2(\omega) x_{dmi}^{\chi}(\chi, \omega) d\omega \quad (5.24)$$

Como os espectros de força de deriva lenta, as forças de deriva lenta no domínio do

tempo (F_{di}^{χ}) podem ser calculadas aplicando-se a transformada inversa de Fourier.

$$F_{di}^{\chi}(\chi, t) = \sum_{i=1}^6 \sqrt{2} Sx'_i(\chi, \mu) d\mu \cos(\mu t + \phi_i) \quad \text{para } i = 1, 2, 6 \quad (5.25)$$

onde: $d\mu = d\omega$ e $\phi =$ fase aleatória ($0 \sim 2\pi$).

Especificamente neste ponto, o trabalho está baseado na possibilidade de obtenção de uma rede neural que tenha a capacidade de estimar de forma rápida e instantânea as magnitudes das forças nas linhas de ancoragem. Estas atuam sobre o navio devido à ação conjunta das ondas, das correntezas e do vento.

5.2.1 Introdução

Artificiais

5.2 Modelagem da Dinâmica de Navios Ancorados Através de Redes Neurais

Devido as características construídas do *turret*, as forças horizontais das linhas atuantes na periferia do mesmo não causam momento em relação ao seu centro. Assim, este foi modelado de uma forma mais simplificada, onde considerou-se as linhas conectadas num único ponto, equivalentes ao centro do mesmo.

hawser em cada uma das direções do sistema local.

A força de amarração em cada direção é calculada projetando-se a força de tração do

$$\Delta l = l_0 \ln \left(\frac{F}{31.53F} + 1 \right) + 0.005488 F_{np} \quad (5.26)$$

A curva de elasticidade do *hawser* é calculada em função da força de ruptura e do comprimento inicial do mesmo conforme equação abaixo.

A força de amarração, independente do sistema adotado, é calculada num regime quase estático sem considerar o efeito dinâmico nas linhas de ancoragem e no *hawser*.

5.1.2.4 Força de Amarração

O treinamento e o subsequente uso do estimador dos esforços nas linhas de ancoragem são descritos mais adiante. Para o caso atual, os dados de treinamento foram gerados por cálculos realizados através do simulador de sistemas oceânicos *Dynasim* (FUCATU, 1998), tomando-se cuidado para manter a configuração e a arquitetura da rede, de maneira que, se possa trabalhar independentemente num sistema embarcado com dados medidos em situações reais.

linear (NARENDRA e PARTHASARATHY, 1990) e (PAO et al., 1991).
representar modelos complexos não lineares como se fossem simplesmente um sistema do sistema” ou “modelamento da planta”. As redes neurais têm melhor capacidade para funcional. Na linguagem de sistemas de controle esta ação é chamada de “identificação suficiente de dados, pode-se fazer com que a rede aprenda a representar a relação produzido nas linhas de ancoragem numa seqüência de tempo. Tendo-se um número deslocamento do navio, a altura das ondas e a velocidade do vento, sendo o esforço pares de dados (entrada/saída) constituem um exemplo da relação funcional geral entre o Considerando a utilização de redes neurais artificiais sob o ponto de vista geral,

esforço no *hawser*, que liga o navio Mãe (FPSO) e o navio Filho (*Shuttle*).
forças nas linhas de amarração em um navio FPSO tipo *Turret* e o correspondente magnitude destas forças. Desta maneira, o trabalho está focalizado sobre a estimação da O modelamento neural desta relação funcional nos capacita estimar em tempo real a

do navio, a velocidade do vento e a força resultante nas linhas de ancoragem.
1992) que realize a descrição causa-efeito entre as elevações das ondas, o deslocamento A primeira aproximação é feita no sentido de construir uma rede neural (GU et al.,

5.2.2 Aquisição e Medição das Condições Ambientais

O vento, a correnteza, e as ondas contribuem forçças permanentes de deriva sobre o navio, porém, de todas elas, as ondas são as mais significativas (GU et al., 1992). Na implementação a bordo, as elevações das ondas serão medidas com elevada precisão através de sensores acústicos. Para o modelamento, precisa-se descrever as ondas de uma maneira apropriada às circunstâncias para as mais variadas condições do mar.

A configuração para a aquisição dos dados foi baseada no projeto (RECOPE) de monitoração do FPSO "P. P. Moraes" (Fig. 5.4), o qual é composto pelos seguintes elementos:

1. (*Differential GPS*) DGPS para a determinação dos movimentos de *sway* e *surge* (Fig. 5.3) do navio;

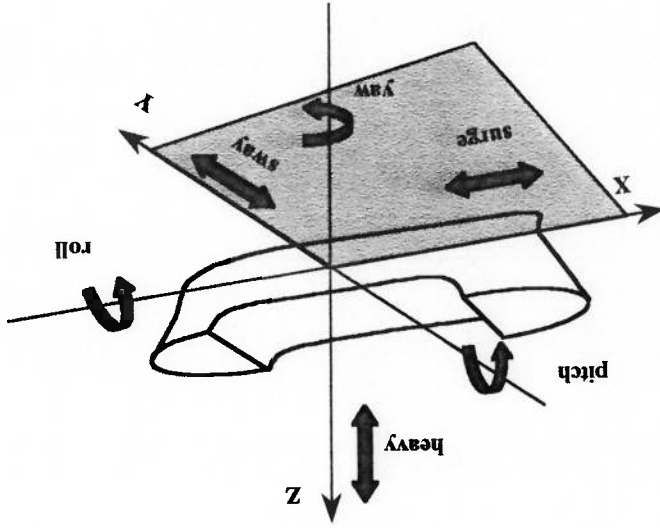
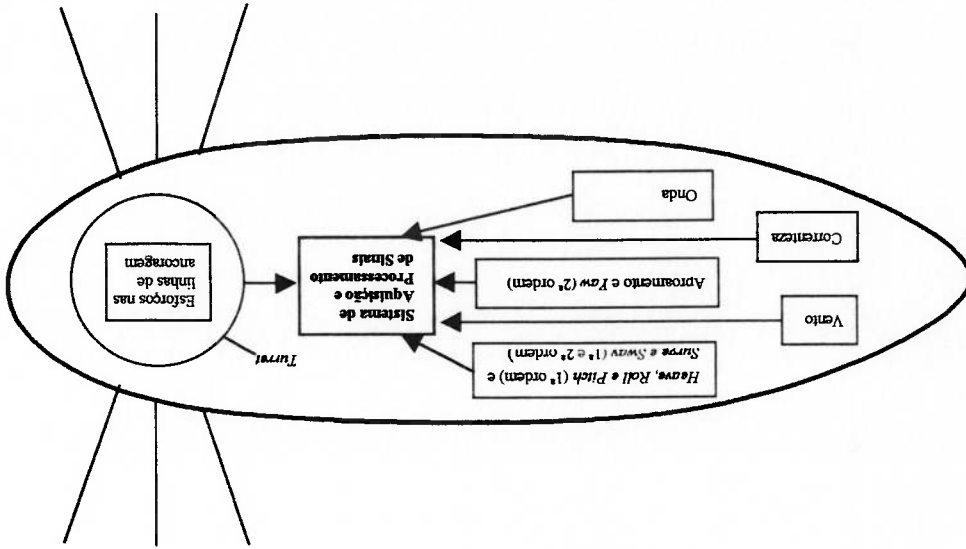


Fig. 5.3: Movimentos do navio

2. Compasso tipo transdutor para a determinação do aprumamento do navio e dos movimentos em *yaw*;

Os transdutores eletricos são: DGPS, acelerômetros, compasso, correntômetro e anemômetro. Os esforços de tração nas amarras deverão ser obtidos a partir da inclinação das linhas no ponto medido e o estado de mar será obtido a partir dos transdutores de

Fig. 5.4 : Esquema de monitoração do FPSO P.P. Moraes



6. Anemômetro para medir velocidade e direção do vento.

que, é uma medida indireta do estado do mar, e

inferência do espectro direcional de mar. O resultado obtido é aproximado, visto

variações de pressão, medidas simultaneamente nestes 4 pontos, é possível a

popa do navio. Conhecendo-se as funções de transferência do navio e as suas

5. Sensores de ondas através da medição da pressão na proa, na seção mestra e na

de contato da linha com o navio que permite a avaliação das forças de tração;

ancoragem. Este tipo de sensor mede um parâmetro, neste caso, o ângulo na seção

4. Sensores de inferência das forças de tração e sua variação nas linhas de

3. Acelerômetros para a determinação dos movimentos de primeira ordem do navio;

pressão existentes no navio. Tanto os transdutores quanto os outros componentes do *hardware* do sistema de aquisição de dados possuem especificações apropriadas para operação contínua em ambiente marinho.

5.2.3 Modelagem Baseada em Redes Neurais

Os métodos clássicos para simulações hidrodinâmicas são baseados em modelos matemáticos que interpolam dados experimentais obtidos mediante ensaios de modelos em escala reduzida. Desta forma, obtêm-se as características hidrodinâmicas dos mesmos dependendo de aproximações matemáticas.

Como a utilização de modelagem matemática clássica é difícil de se aplicar em um sistema real, uma metodologia baseada em redes neurais facilita a utilização de medições reais.

5.2.4 Modelagem e Identificação

Os modelos hidrodinâmicos são tipicamente não lineares e geralmente devem ser linearizados, com isso a sua eficácia é restrita a uma pequena faixa de operação.

O modelo apresentado neste capítulo identifica características estruturais de modelos de sistemas reais, isto inclui a identificação do modelo da estrutura na forma de equações diferenciais lineares e não lineares.

Como este sistema dinâmico tem comportamento temporal, as saídas com retardo de tempo são realimentadas. Isto é necessário para modelar apropriadamente o sistema e incorporar a dinâmica como mostra a (Fig. 5.5).

A posição global do navio é estimada através de uma rede neural que desenvolve a transformação de coordenadas do sistema local para o sistema inercial, conforme mostra a (Fig. 5.6). A rede tem como entrada a posição atual do navio no sistema global e tem como saída os dados para o cálculo das forças que interagem com o mesmo.

Como as forças presentes no sistema estão ligadas internamente à posição do navio, seu cálculo será feito de duas formas distintas. A primeira, aplica-se às forças com características lineares, cujos resultados são obtidos através de simples produtos algébricos, não sendo necessário a utilização de redes neurais. Estas forças são: F_i força inercial, F_v força do vento, F_{res} força de restauração hidrostática, F_{poi} força de amortecimento. A Segunda, aplica-se no caso das forças não lineares, F_o força da onda de primeira ordem, F_{dl} força de deriva lenta, F_{dm} força de deriva média, F_{am} força de amortecimento por deriva de onda. No segundo caso as redes neurais mostram sua grande vantagem. As forças atuantes no sistema são simulados conforme a rede neural estruturada na (Fig. 5.6).

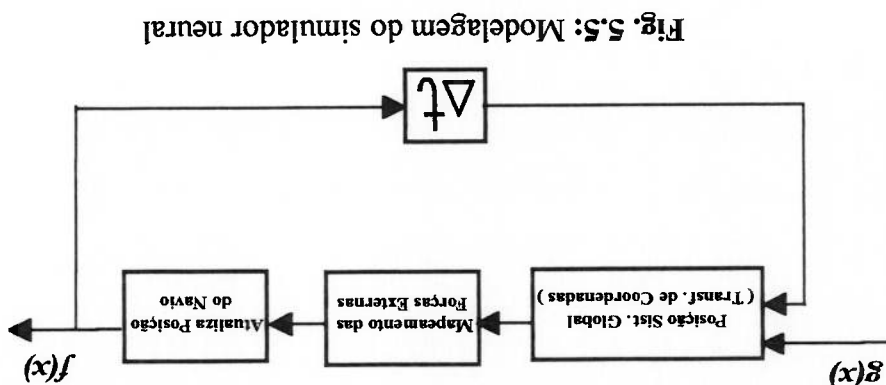


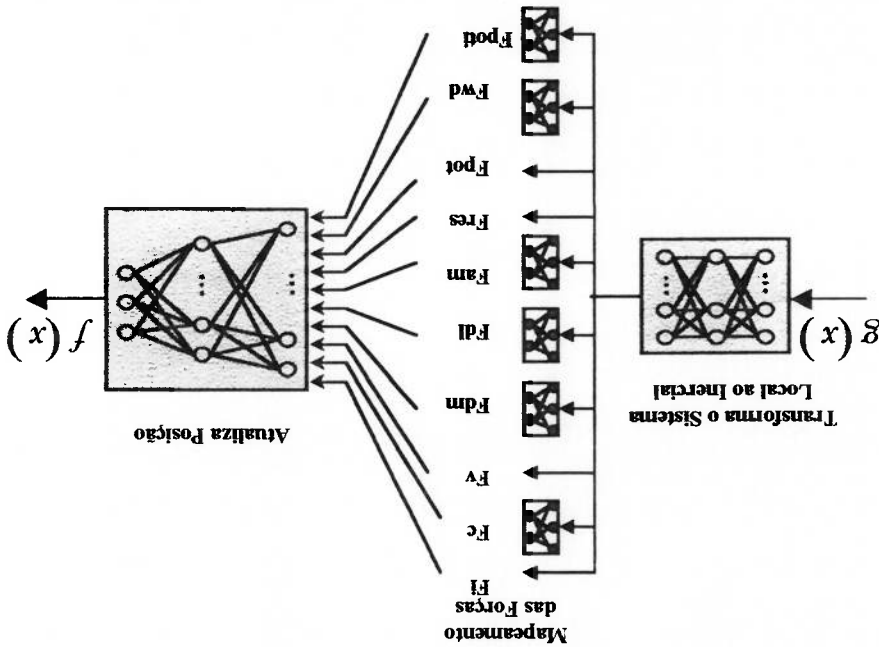
Fig. 5.5: Modelagem do simulador neural

O simulador pode ser definido como segue:

conforme a rede neural estruturada na (Fig. 5.6).

Como as forças presentes no sistema estão ligadas internamente à posição do navio, seu cálculo será feito de duas formas distintas. A primeira, aplica-se às forças com características lineares, cujos resultados são obtidos através de simples produtos algébricos, não sendo necessário a utilização de redes neurais. Estas forças são: F_i força inercial, F_v força do vento, F_{res} força de restauração hidrostática, F_{pot} força de amortecimento. A Segunda, aplica-se no caso das forças não lineares, F_o força da onda de primeira ordem, F_{dl} força de deriva lenta, F_{dm} força de deriva média, F_{am} força de amarragem F_{wd} força de amortecimento por deriva de onda. No segundo caso as redes neurais mostram sua grande vantagem. As esforços atuantes no sistema são simulados

Fig. 5.6: Forças atuantes no simulador neural para a configuração *tandem*



A combinação de três redes neurais é usada na modelagem seguindo a configuração mostrada na (Fig. 5.5). Esta, torna possível a livre variação dos parâmetros *feed-forward* e *feedback* do sistema de acordo com o estado do navio. Levando-se em conta as características não lineares das forças hidrodinâmicas em cada estágio, as redes neurais adquirem o conhecimento necessário para a modelagem através do aprendizado repetitivo pelo método de *backpropagation*.

5.2.5 Operação do Simulador

F_{1.6}: Esforços nas linhas de ancoragem.

movimento em *sway* e x_6 movimento em *yaw*)

$f(x) = f(x_1, x_2, x_6)$: Deslocamento do navio (x_1 movimento em *surge*, x_2

Saída:

$$g(x) = [f(x + \Delta t), C]$$

$f(x + \Delta t)$: Deslocamento do navio no seguinte intervalo de tempo

C: Condições ambientais (Vento, correnteza e ondas)

Entrada: $g(x)$

5.2.6 Configuração das Redes para a Simulação

Devido aplicações futuras do simulador como um sistema embarcado para estimar as forças de ancoragem, as entradas serão especificamente aquelas encontradas no ambiente médio marinho, as quais serão medidas através de sensores instalados no navio como

mostra a (Fig. 5.4).

A identificação começa com a aquisição das variáveis de entrada (condições ambientais e deslocamento inicial do navio), que são anexadas ao sistema através da *Rede I*. Esta rede encarrega-se de produzir os esforços nas linhas de amarração do navio mãe (FPSO) como consequência do deslocamento inicial do navio (x_{1mi} , x_{2mi} , x_{6mi}), altura da onda (h_w), direção e velocidade do vento (v_w , θ_w).

Logo após ser produzido o primeiro esforço nas linhas de amarração, como resultante das forças ambientais, adiciona-se na entrada a força do *Hawser* (cabo de ligação entre os dois navios), esta é devido a magnitude da força que intervem no sistema navio mãe. A *Rede II* realiza a ação de restituição sobre o navio fornecendo na saída o próximo deslocamento do navio x_{1mi+1} , x_{2mi+1} , x_{6mi+1} , que será realimentado na *Rede I* depois de passar por um retardo de tempo para introduzir a dinâmica ao sistema navio mãe, vide figura (5.7).

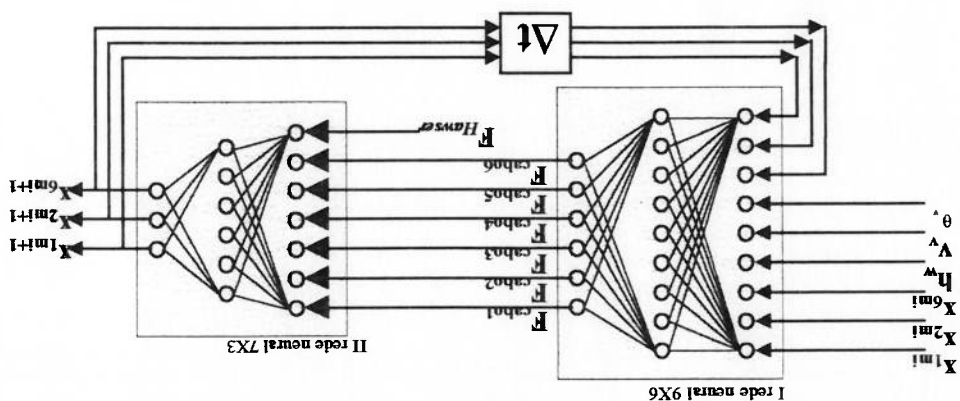
Com os novos deslocamentos no sistema local (navio filho) e as condições ambientais da *Rede IV* realiza-se o cálculo do esforço no *hawser*. Por ultimo a *Rede V* encarrega-se de produzir o próximo deslocamento do navio filho $x_{1f+1}, x_{2f+1}, x_{6f+1}, x_{2mf+1}, x_{6mf+1}$. Da mesma forma que se fez no navio mãe, é introduzido um retardo de tempo para fornecer a dinâmica ao sistema navio filho, como é mostrado na (Fig. 5.8).

Com os novos deslocamentos no sistema local (navio filho) e as condições ambientais da *Rede IV* realiza-se o cálculo do esforço no *hawser*. Por ultimo a *Rede V* encarrega-se de produzir o próximo deslocamento do navio filho $x_{1f+1}, x_{2f+1}, x_{6f+1}, x_{2mf+1}, x_{6mf+1}$. Da mesma forma que se fez no navio mãe, é introduzido um retardo de tempo para fornecer a dinâmica ao sistema navio filho, como é mostrado na (Fig. 5.8).

filho) através da *Rede III*.

No caso do navio filho existe ampla dependência dos deslocamentos do navio mãe, no qual precisa-se realizar a transformação de coordenadas do sistema inercial (origem centro de gravidade do navio mãe) para o sistema local (centro de gravidade do navio filho) através da *Rede III*.

Fig. 5.7: Representação do navio mãe para a simulação



Com os novos deslocamentos no sistema local (navio filho) e as condições ambientais da Rede IV realiza-se o cálculo do esforço no *hawser*. Por último a Rede V encarrega-se de produzir o próximo deslocamento do navio filho $x_{1f+1}, x_{2f+1}, x_{6f+1}$. Da mesma forma que se fez no navio mãe, é introduzido um retardo de tempo para fornecer a dinâmica ao sistema navio filho, como é mostrado na (Fig. 5.8).

Devido a forte interação entre os dois sistemas (mãe/filho), o deslocamento no navio mãe afetará consideravelmente os deslocamentos no navio filho, com isso, os esforços produzidos nas linhas de amarração ficam ligados internamente às condições ambientais e ao acoplamento mútuo dos navios. Nesse sentido, uma magnitude ambiental (vento, correnteza e ondas), causará um deslocamento no navio que, por sua vez, produzirá um esforço nas linhas de amarração como consequência do novo esforço apresentado nas linhas, produzindo um novo deslocamento do navio.

A (Fig. 5.9) mostra o sistema completo para a modelagem do sistema de amarração dos FSSO em configuração *Tandem* através de redes neurais.

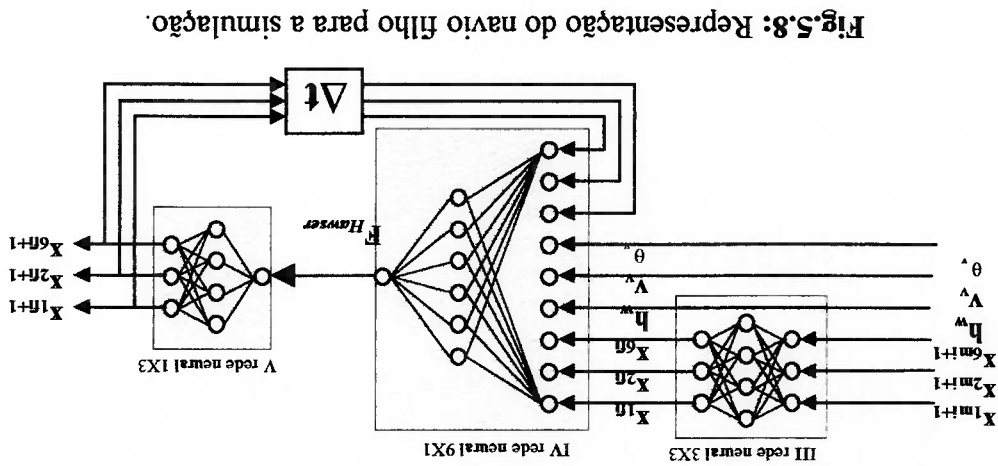
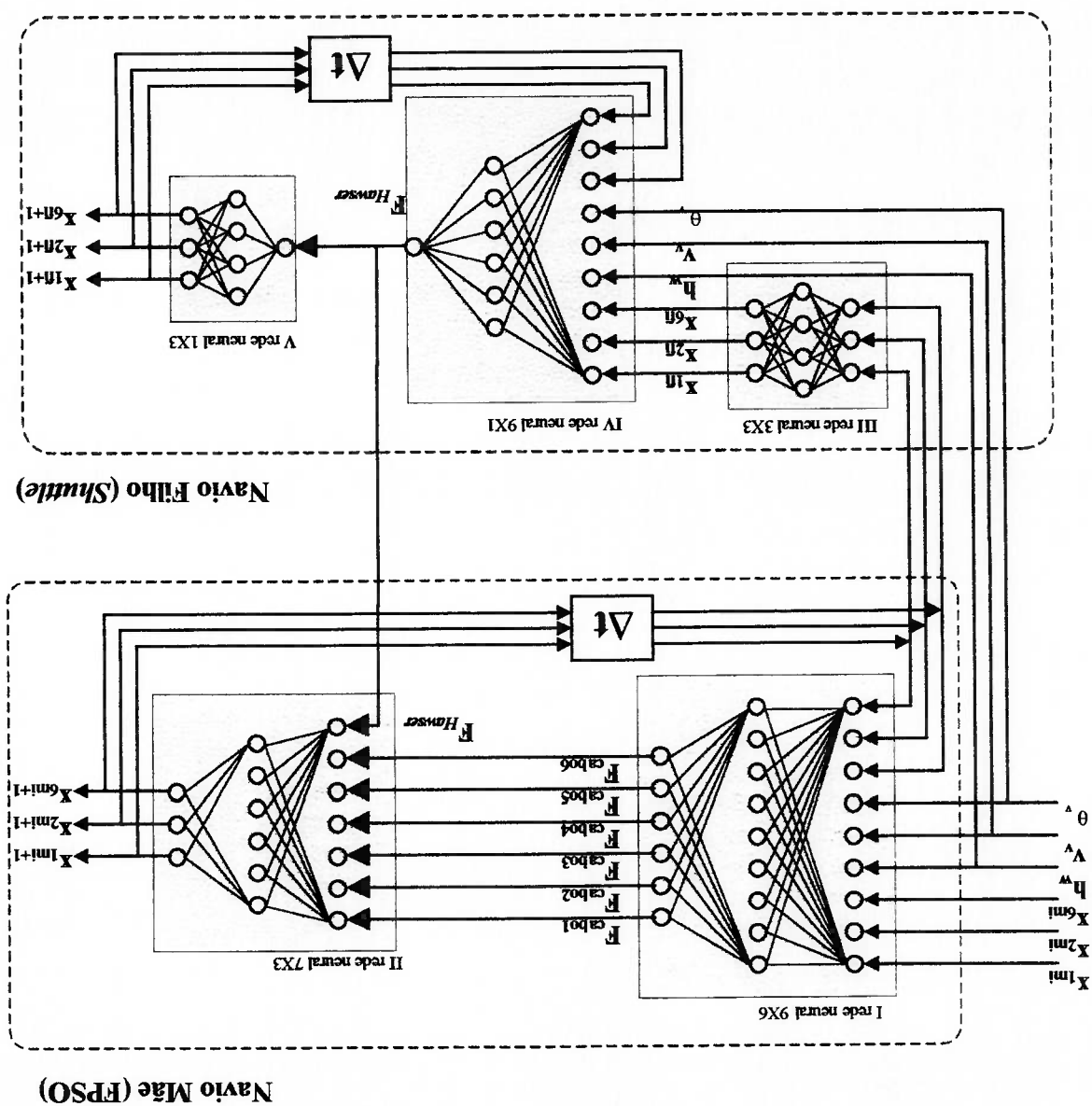


Fig.5.8: Representação do navio filho para a simulação.

Fig. 5.9 : Disposição das redes para a simulação



5.2.7 Validação dos Resultados

Um conjunto de dados não utilizados durante o aprendizado é utilizado para testar a validade do treinamento. Os resultados da simulação são comparados com os dados originais e os gerados pelas redes neurais. Além disso, é apresentado o erro produzido na identificação do sistema.

Só são mostrados os esforços e o erro nas linhas de ancoragem 1.º e 6.º, assim como o esforço no cabo de ligação entre os dois navios e os esforços produzidos nas linhas de ancoragem 2.º, 3.º, 4.º e 5.º respectivamente.

TABELA 3: PRECISÃO ATINGIDA PELAS REDES NA TAREFA DE MODELAGEM DO SISTEMA DE AMARRAÇÃO DO FPSO.

	Cabo 1	Cabo 2	Cabo 3	Cabo 4	Cabo 5	Cabo 6	Hawser
Coefficiente de correlação	0.99	0.96	0.95	0.97	0.94	0.97	0.83
Erro médio absoluto (%)	3.88	5.19	2.19	1.93	2.60	2.12	40.09
Padões de erro (com até 5%) (%)	96.68	95.35	98.34	98.01	97.34	98.34	10.70

A (Fig. 5.10) mostra a saída atual do sistema e a saída produzida pela rede neural para a identificação do esforço no cabo 1, também é mostrada a diferença entre a saída atual do sistema e a saída produzida pela rede (Fig. 5.11), onde pode-se observar que o erro produzido durante a tarefa de identificação é mínimo.

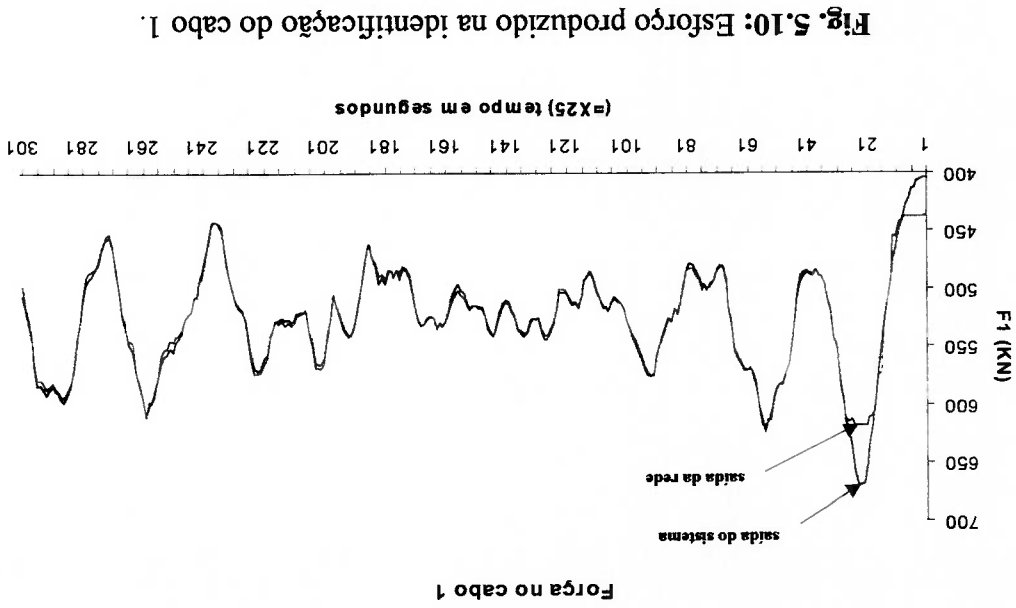


Fig. 5.10: Esforço produzido na identificação do cabo 1.

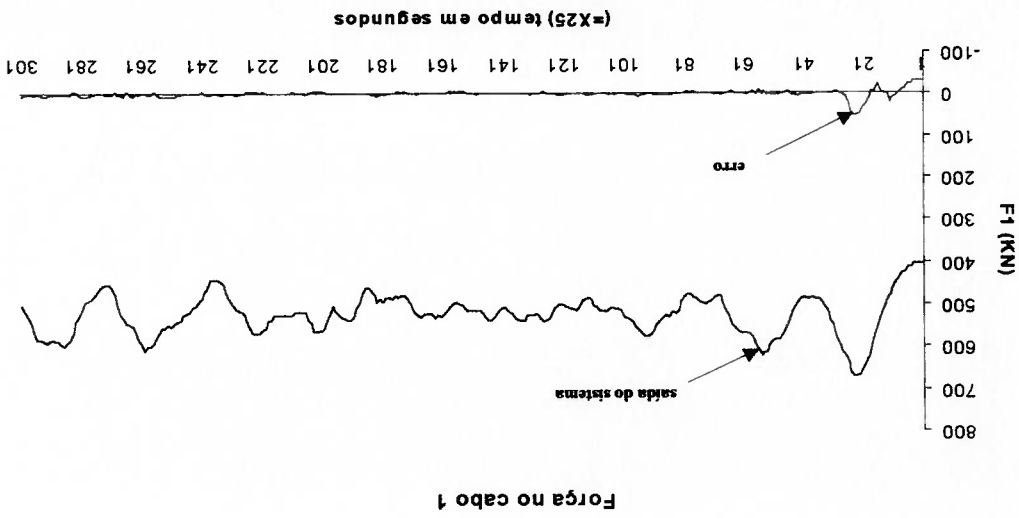
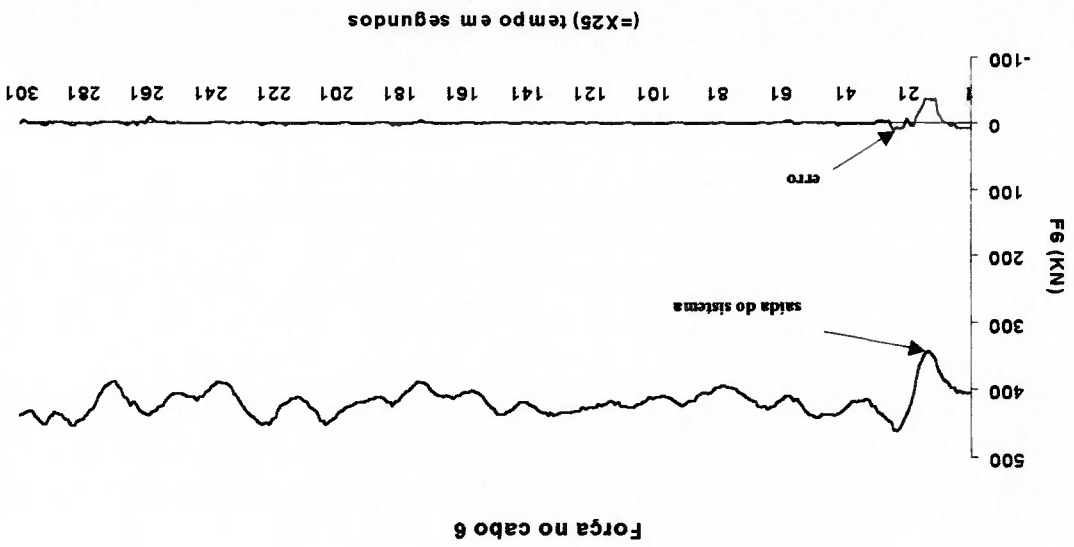
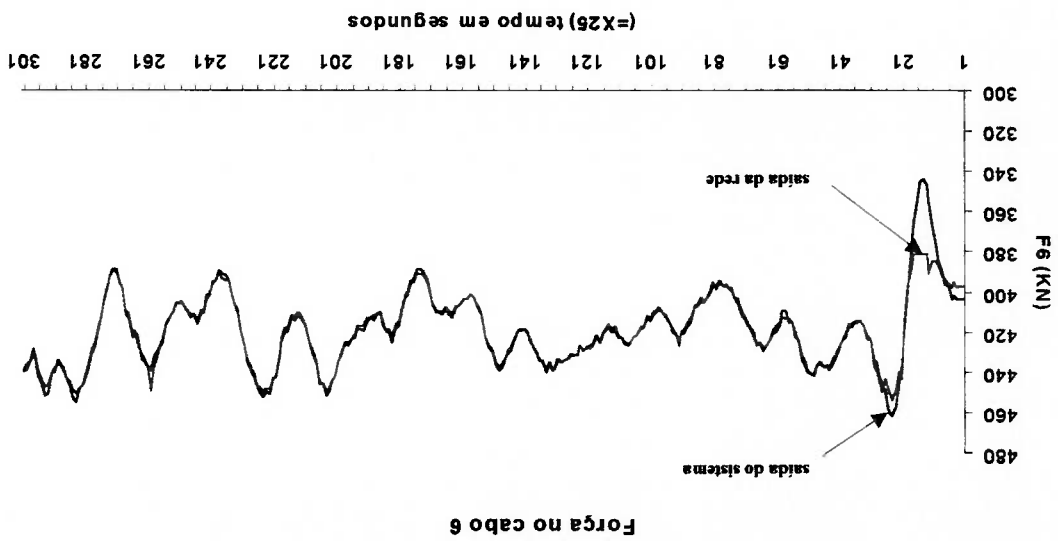


Fig. 5.11: Erro produzido na identificação do cabo 1.

O esforço produzido no cabo 6 é mostrado na (Fig. 5.12), onde pode-se observar que a rede identificou plenamente os sistemas, apresentando mínima quantidade de erro (Fig. 5.13), o erro é mais significativo no início devido ao regime transitório no começo da simulação.



A seguir serão mostrados os esforços nos demais cabos do sistema de ancoramento do navio mãe. A (Fig. 5.14) e a (Fig. 5.15) mostram os esforços produzidos nos cabos 2 e 3 respectivamente, neste caso, a rede também consegue representar o sistema sem maiores problemas.

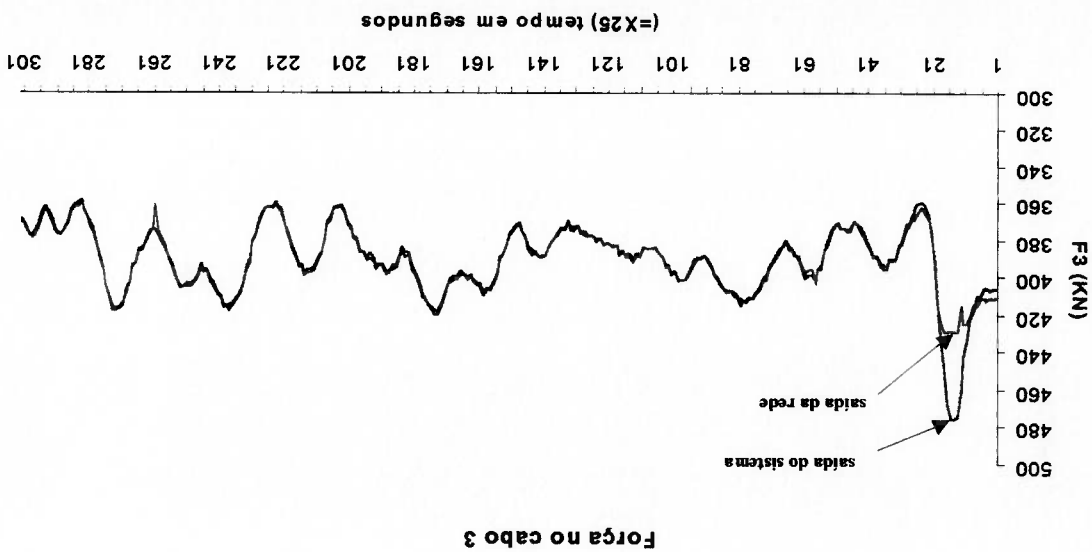


Fig. 5.15: Esforço produzido na identificação do cabo 3.

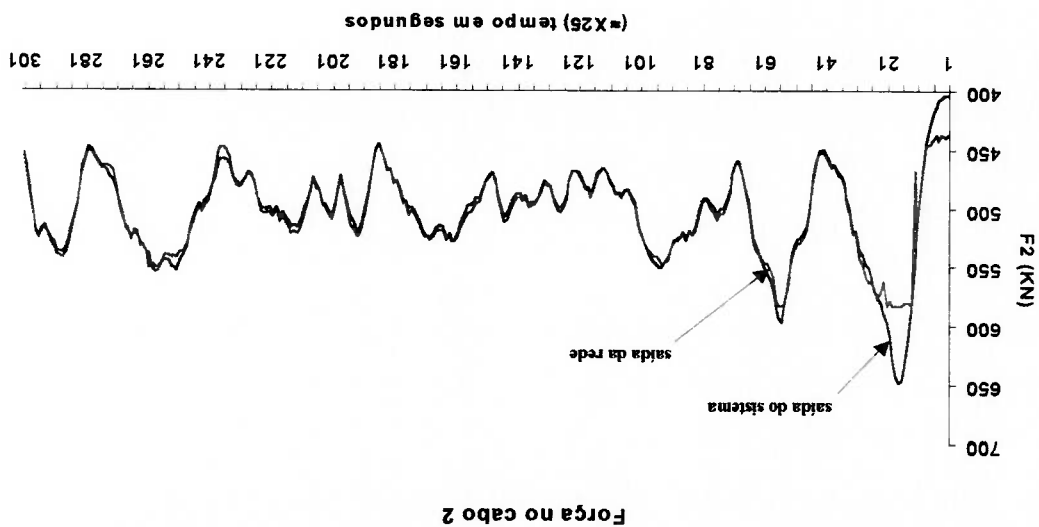


Fig. 5.14: Esforço na identificação do cabo 2.

A (Fig. 5.16) e a (Fig. 5.17) mostram os esforços produzidos nos cabos 4 e 5 respectivamente, completando a representação dos 6 cabos de ancoragem do navio mãe. A resposta da rede na identificação do sistema de ancoramento foi excelente, conseguindo um ótimo o mapeamento dos esforços nas linhas de amarração.

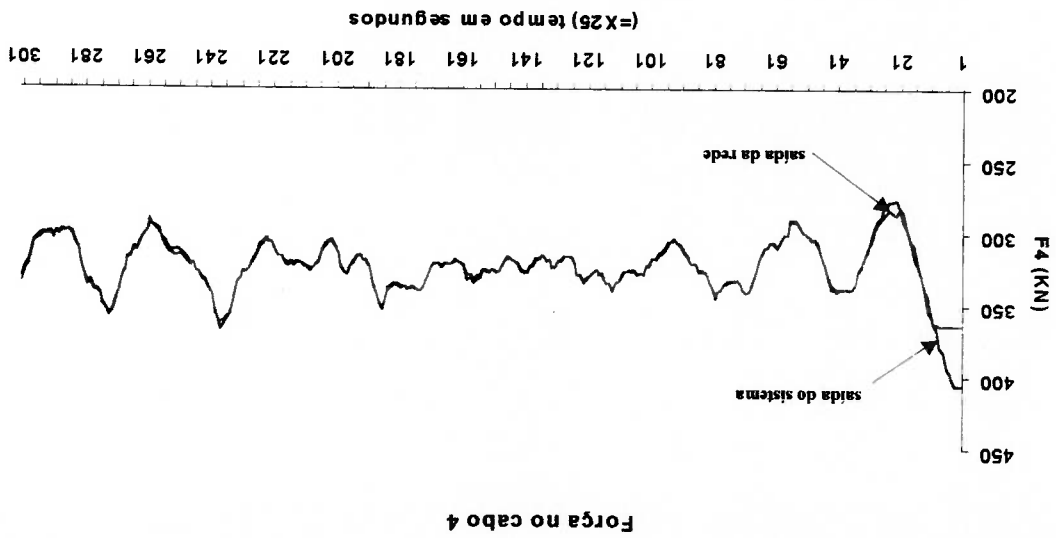


Fig. 5.17: Esforço produzido na identificação do cabo 4.

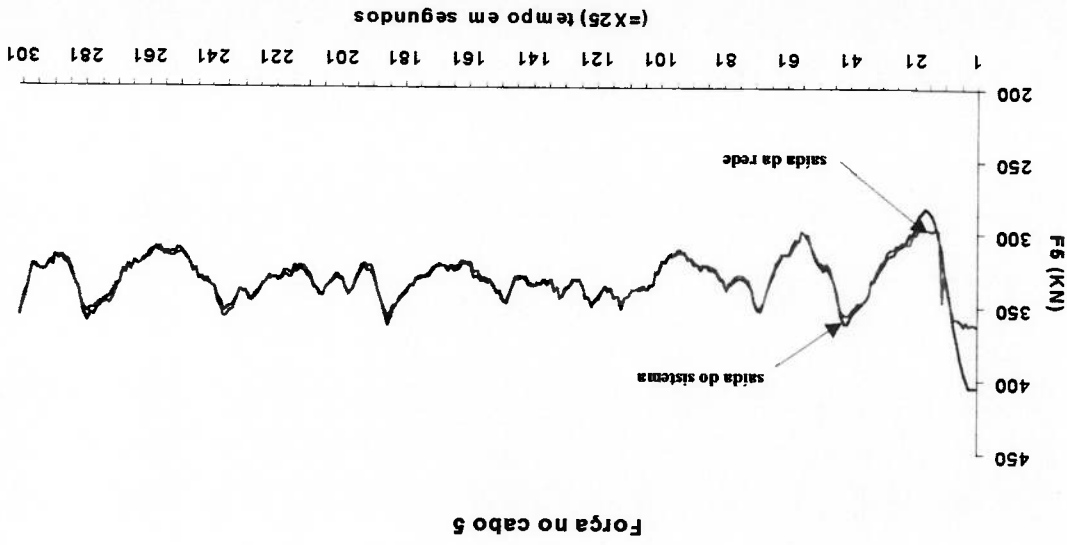


Fig. 5.18: Esforço produzido na identificação do cabo 5

O esforço produzido no cabo de ligação entre os navios (*hawser*) é mostrado na (Fig. 5.18), onde observa-se que a rede apresenta uma pequena dificuldade na identificação do esforço no *hawser*. Isto é devido a função que representa os esforços do *hawser* contém múltiplas descontinuidades, como foi mencionado no capítulo anterior.

Nas regiões com descontinuidades, a rede demonstra sua boa capacidade de generalização adaptando-se ao sistema e produzindo esforços negativos, o que indica a presença de um *offset* onde estariam consideradas: o próprio peso da linha, forças de arraste na linha e outras cargas impostas sobre a linha devido ao efeito sombra de água produzidas pelo navio mãe, as que não foram considerados no simulador *Dynasim*.

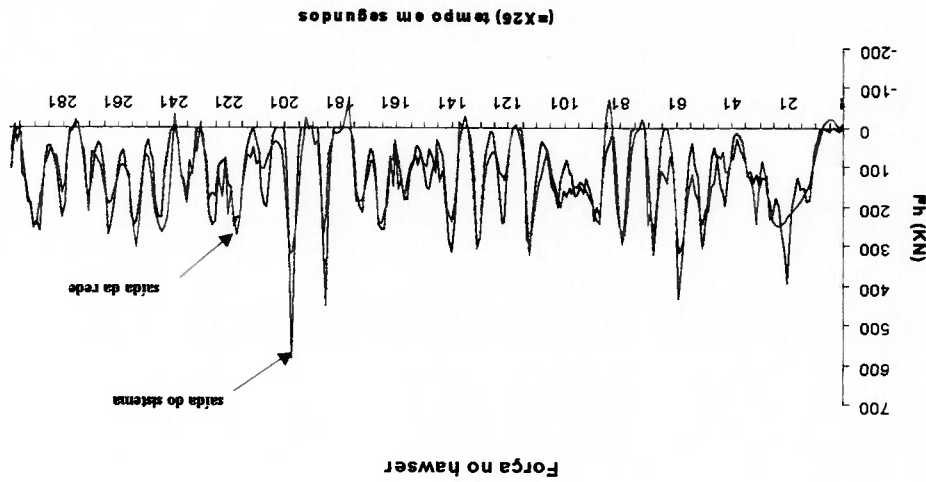


Fig. 5.19: Esforço produzido na identificação do *hawser*

A (Fig. 5.20) mostra a diferença entre a saída do sistema e a saída da rede na identificação dos esforços produzidos no *hawser*.

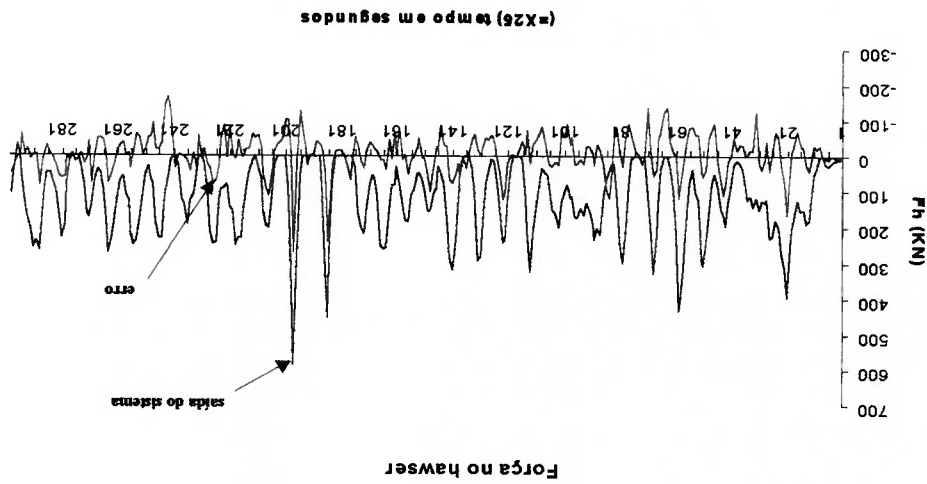


Fig. 5.20: Erro produzido na identificação do *hawser*

As (Fig. 5.21, 5.22 e 5.23) mostram graficamente em diagramas de barras, a precisão alcançada pela rede na identificação dos esforços no sistema de amarração dos FPSO, onde são definidos os intervalos de erro como [0-5%], [5-10%], [10-20%], [20-30%], e [30-100%].

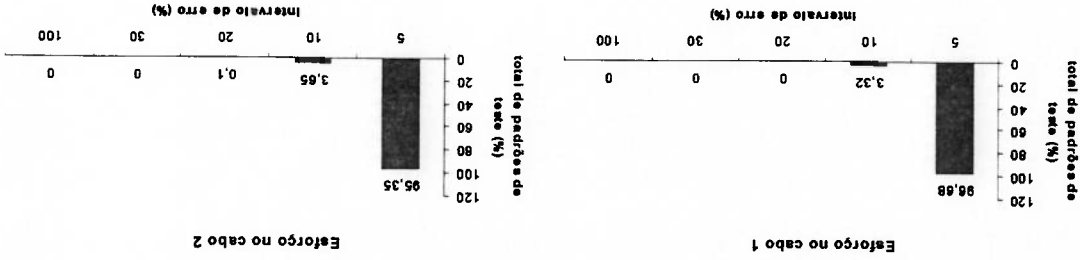


Fig. 5.21: Precisão alcançada pela rede na identificação do (a) cabo 1 e (b) cabo 2.

Fig. 5.23: Precisão alcançada pela rede na identificação do (a) cabo 5 e (b) cabo 6.

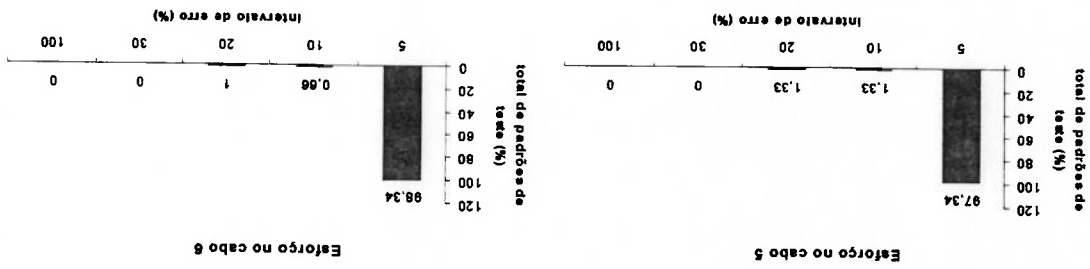
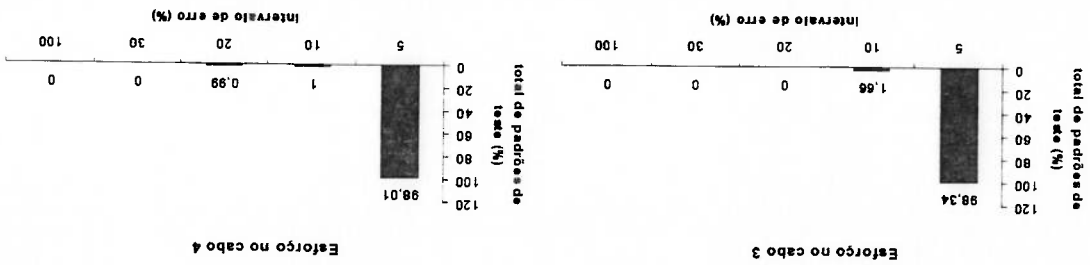


Fig. 5.22: Precisão alcançada pela rede na identificação do (a) cabo 3 e (b) cabo 4.



6 CONCLUSÕES E SUGESTÕES PARA TRABALHOS FUTUROS

6.1 Conclusões

- As redes neurais artificiais mostraram grande capacidade de maleabilidade na abordagem para a solução do problema e especificamente neste trabalho demonstraram grande eficiência tanto na interpolação (modelagem e identificação do sistema), como na extrapolação (estimação de séries temporais).

- Os resultados da simulação têm demonstrado que o problema da identificação dos esforços nos navios ancorados é resolvido com sucesso utilizando redes neurais. As vantagens da metodologia usada podem ser resumidos como:

- Com uma pequena quantidade de conhecimento acerca do processo, o sistema pode ser completamente identificado, e não é necessário ter conhecimentos das expressões matemáticas exatas da dinâmica do sistema.
- As características temporais e não lineares do navio não representam um problema na identificação do sistema através de redes neurais, porque as redes podem aprender o comportamento dinâmico de tais sistemas.

- Ambas metodologias, estimação da série temporal e modelagem dinâmica dos esforços nas linhas de amarração dos navios FPSO em configuração *Tandem* demonstraram bons resultados. A utilização do 1.º ou 2.º método dependerá da aplicação específica para uma tarefa dada. Por exemplo caso, deseja-se saber comportamentos futuros dos esforços nas linhas de amarração a escolha ótima será a estimação da série temporal de tais esforços. Mas se é necessário conhecer a influência que tem cada uma das variáveis de entrada (condições ambientais) sobre as

linhas de amarração, a escolha estará voltada para a modelagem através de redes neurais. A escolha destes métodos também dependerá da quantidade de dados disponíveis. Para o caso da estimação da série temporal será preciso dispor de uma quantidade maior de dados para obter melhores resultados. Em contrapartida a metodologia da modelagem precisa de uma quantidade menor de dados comparada com o primeiro método. No entanto será necessário alimentar a rede com mais variáveis de entrada.

- Outra conclusão observada no desenvolvimento do trabalho é o tempo de treinamento, para a estimação da série temporal deve ser menor para aproveitar ao máximo a capacidade de generalização da rede, evitando o processo de memorização devido ao sobretreinamento. Embora a propriedade de generalização ser importante na tarefa de modelagem, o tempo de treinamento requerido para a identificação do sistema deve ser um tanto maior para obter uma resposta mais precisa da rede.

- No processo de modelamento dinâmico através de redes neurais artificiais os resultados mostraram a capacidade das redes para modelar sistemas que variam no tempo com precisão e confiável resposta dinâmica, demonstrando que as redes neurais podem ser efetivamente implementadas para a simulação do comportamento dinâmico de navios amarrados.

- Comentários Finais sobre os resultados alcançados

Os resultados indicaram que o modelo de backpropagation pode ser uma ferramenta muito útil em estudos de modelagem e estimação.

Este trabalho não pretende sugerir a substituição dos métodos matemáticos para a simulação por técnicas de inteligência artificial, mas sim a incorporação desta nova metodologia aos métodos convencionais como o objetivo de obter melhores resultados, de modo que os sistemas dinâmicos possam se aproximar cada vez mais ao comportamento real de tais sistemas, contribuindo na segurança dos navios tanque FPSO.

6.2 Sugestões para Trabalhos Futuros

- Horizontes maiores de estimação poderão ser obtidos com a utilização de redes neurais hierárquicas (Fig. 6.1) de modo que a primeira rede realize previsões de curto prazo, a segunda rede previsões de meio prazo e a terceira rede previsões de longo prazo, todas alimentando uma quarta rede que realiza a previsão final, esta última rede deve ser treinada de modo a realizar uma soma ativa, de maneira similar a um amplificador operacional somador. Nesta configuração a rede poderá realizar previsões maiores com a precisão de estimções de curto prazo.

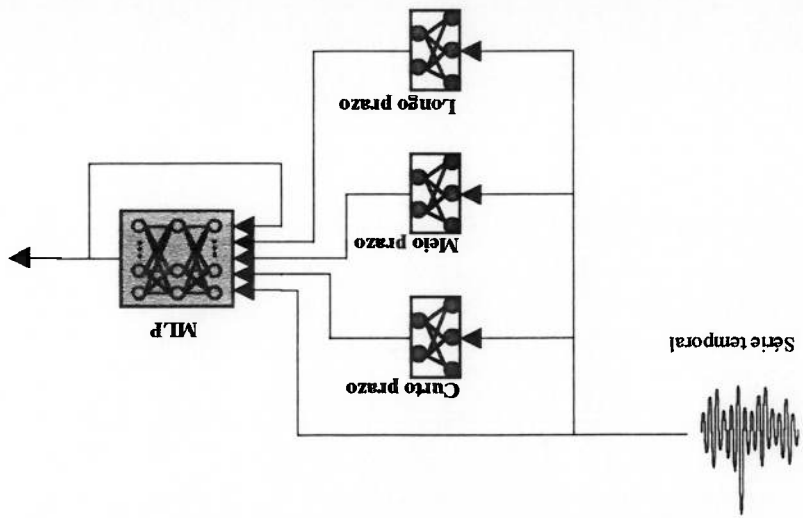


Fig. 6.1: Rede neural hierárquica

- Como sugestões para trabalhos futuros poderíamos relacionar um maior aprofundamento na pesquisa em:
 - Métodos mais eficazes na determinação da rede (topologia, tipos de funções de ativação, etc);
 - Métodos para determinação de um bom ponto de partida (pesos iniciais das conexões, topologia inicial, etc);
 - Métodos específicos (customizados) para as tarefas de estimação e modelagem, e Inclusão da noção de distribuição de probabilidades de forma que pudessem obter um intervalo de confiança além da previsão pontual.
- Na tarefa de identificação e modelagem de sistemas dinâmicos, a adição de redes neurais à modelagem clássica potencializaram o desempenho dos simuladores convencionais, tornando-o um sistema híbrido. De fato a integração de um sistema neural híbrido com uma base de conhecimento resultará no melhoramento de sistemas especialistas que na atualidade começam a controlar e identificar a maioria dos processos.

REFERÊNCIAS BIBLIOGRÁFICAS

- BARUSCO, P. J.; OZORIA DA CRUZ, R. M.; PETKOVIC, M. A. Application of Floating Systems in Deep Waters. Proceedings of MARIN workshops on: Advanced Vessels, Station Keeping, Propulsor-Hull Interaction, and Nautical Simulators, The Netherlands, May 1992, pp. 281-288.
- BHAT, N.; MCAVOY, T. Dynamic Process Modeling Via Neural Computing. AICHE Annual Meeting, San Francisco, USA, 1989.
- BOATMAN, L. T.; ETHERIDGE, C. O.; PORANSKI, P. F. Optimization of a Disconnectable Turret Mooring System. Proceedings of MARIN workshops on: Advanced Vessels, Station Keeping, Propulsor-Hull Interaction, and Nautical Simulators, The Netherlands, May 1992, pp. 393-398.
- BRENDLING, W. J.; WILSON, D. Theoretical and Experimental Investigation of the Low Frequency Motions of a Single Anchor Leg Moored Buoy and Tanker. Proceedings of a Workshop on Floating Structures and Offshore Operations, The Netherlands, 1987, pp. 109-124.
- CARD, J. P.; SNIDERMAN, D. L.; KLIMASAVSKAS, C. Dynamic Neural Control for a Plasma Etch Process. IEEE Transactions on Neural Networks, July 1997, Vol 8, pp. 883-901.
- CHAKRABARTI, S. K. Hydrodynamics of Offshore Structures. Springer, Berlin 1987.
- CHAKRABARTI, S. K. Nonlinear Methods in Offshore Engineering. Elsevier Science, Amsterdam 1990.
- COLINA-MORTES, E.; MORT, N. Inverse Model Neural Networks Based Control of Dynamic Systems. Control' 94, March 1994, N.º 389, pp. 955-960.
- D'HAUTEFUILLE, B. B.; DE BOOM, W.; PESCIO, L. Design Tools for Engineering of Mooring Systems. Proceedings of MARIN workshops on: Advanced Vessels, Station Keeping, Propulsor-Hull Interaction, and Nautical Simulators, The Netherlands, May 1992, pp. 387-392.
- DACONTA, M. C. C++ Pointers and Dynamic Memory Management. New York, A. Wiley - QED Publication, 1995.
- DE BOOM, W. C. Turret Moorings for Tanker Based FPSOs. Proceedings of a Workshop on Floating Structures and Offshore Operations, The Netherlands, 1987, pp. 143-154.
- DEO, M. C.; NAIDU, C. S. Real Time Wave Forecasting Using Neural Networks. Ocean Engineering, 1999, Vol 26, pp. 191-203.

- DERCKSEN, A. **Dynamic Analysis of Moored Tankers**. OMAE 1995.
- ECO, H. **Como se Faz uma Tese**. Ed. Perspectiva, São Paulo, 1999.
- FREEMAN, J. A.; SKAPURA, D. M. **Redes Neuronales: Algoritmos, Aplicaciones y Técnicas de Programación**. Addison-Wesley Iberoamericana, Wilmington Delaware, 1993.
- FUCATU, C. A. **Desenvolvimento de um Simulador Dinâmico para Análise de Navios Amarrados**. Dissertação de Mestrado, Escola Politécnica da Universidade de São Paulo, 1998.
- GU, X. M.; PAO, Y.; YIP, P. C.; **Neural-Net Computing for Dynamic Positioning at Sea**. Elsevier Science, Hydrodynamics: Computations, Model Test and Reality, 1992., pp.329-343.
- HADDARA, M. R.; XU, J. **On the Identification of Ship Coupled Heave-Pitch Motions Using Neural Networks**. Ocean Engineering, 1999, Vol 26, pp. 381-400.
- HINTON, G. E. **Learning in Parallel Networks: Simulating Learning in a probabilistic System**, BYTE, April 1995, pp. 265-273.
- HUANG, D.; VAN CAUWENBERGHE, A. R. **Neural Network Based Multiple Feedback Long-Range Predictive Control**. Elsevier Science, Neurocomputing, 1998, pp. 127-139.
- HUNTER, K. C.; DE SOUZA P. M. F. M.; PHAN, T. K. **Development of FPSO for Offshore Australia**. Proceedings of MARIN workshops on: Advanced Vessels, Station Keeping, Propulsor-Hull Interaction, and Nautical Simulators, The Netherlands, May 1992, pp. 229-242.
- KAT, J. O.; WICHERS, J.E.W. **The Behavior of a Moored Ship in Unsteady Current, Wind and Waves**. Marine Technology, September 1991.
- KULKARNI, D. R.; PARKH, J. C. **Dynamic plus Connectionist Approach to Time Series Prediction**. SPIE vol. 3077, pp.35-45.
- LANZAFAME, R.; SANTANGELO, G. **A Neural Logic Program as a New Method for The Development of Fluids Thermodynamic Properties**. Applied Modeling and Simulation, IASTED, 1997.
- MARTINS, C. A.; LEITE, A.; ARANHA, J. A. P. **Mooring System Optimization with Application to a Weather Vane Ship**. ASME 1995. pp. 231-236.
- MASTERS, T. **Advanced Algorithms for Neural Networks**. New York, John Wiley & Sons Inc, 1995.
- MASTERS, T. **Signal and Image Processing with Neural Networks**. New York, John Wiley & Sons Inc, 1996.
- McCLELLAND, J. L.; RUMELHART, D. E. **Explorations in Parallel Distributed Processing**. MIT Press, Cambridge, 1988.

- MELO, M. P. **Redes Neurais Artificiais: Uma Aplicação à Previsão de Preços de Derivados de Petróleo.** Dissertação de Mestrado, Pontifícia Universidade Católica do Rio de Janeiro, 1991.
- MERMA, J. L.; MORISHTA, H. M.; SIMÕES, M. G. **Análise Dinâmica de Sistemas de Produção Flutuante de Petróleo Através de Redes Neurais Artificiais.** 4.º Simpósio Brasileiro de Automação Inteligente, São Paulo, 1999.
- MILLER, W.; SUTTON, R.; WERBOS, P. **Neural Networks for Control.** MIT Press, Cambridge, USA, 1990.
- MORISHTA, H. M. **Estudo da Dinâmica e Controle de Multicorpos Flutuantes.** Escola Politécnica da Universidade de São Paulo, 1999.
- MORISHTA, H. M.; CORNET, B. J. J. **Dynamics of a Turret-FPSO and Shuttle Vessel due to Current.** IFAC Conference, Fukuoka, 1998.
- NAESS, T.; JOHNSEN, P. M. **Design of Floating Production Systems.** Proceedings of MARIN workshops on: Advanced Vessels, Station Keeping, Propulsor-Hull Interaction, and Nautical Simulators, The Netherlands, May 1992, pp. 215-222.
- NARENDRA, K. S.; PARTHASARATHY, K. **Identification and Control of Dynamical Systems Using Neural Networks.** IEEE Transactions on Neural Networks, 1990, Vol 1, pp. 4-26.
- NISHIMOTO, K.; BRINATI, H. L.; FUCATU, C. H. **Dynamic Analysis of FPSO with Tandem Offloading System.** Offshore Mechanics and Arctic Engineering, 1996.
- PAT, C.; HAGIWARA, I.; KAYABA, N.; MORISHTA, S. **Dynamic System Identification by Neural Network: A New, Fast Learning Method Based on Error Back Propagation.** Journal of Intelligent Material Systems and Structures, January 1994, Vol 5, pp. 127-135.
- PANDYA, A. S.; MACY, R. B. **Pattern Recognition with Neural Networks.** Boca Raton, Florida, IEEE Press, 1995.
- PATAKI, B. **Neural Network Based Dynamic Models.** Technical University of Budapest, Hungary, 1993.
- PESCE, C. P.; TANNURI, E. A. **Modelo Dinâmico de FPSO em Sistema Turret com Modelo de Forças Hidrodinâmicas de Asa Curta (Heurístico).** Escola Politécnica da Universidade de São Paulo, 1997.
- REFENES, A. N. **Constructive Learning and its Application to Currency Exchange Rate Forecasting.** Neural Networks Application in Investment and Finance Services. Probus Publishing, USA, 1991.
- RUDOLPH, G. L.; MARTINEZ T. R. **A Transformation Strategy for Implementing Distributed Multilayer Feedforward Neural Networks.** Elsevier Science, Future Generation Computer Systems, 1997, pp. 547-564.
- RUMELHART, D. E.; HINTON, G. F.; WILLIAMS, R. J. **Learning Internal Representations By Error Propagation in: Parallel Distributed Processing:**

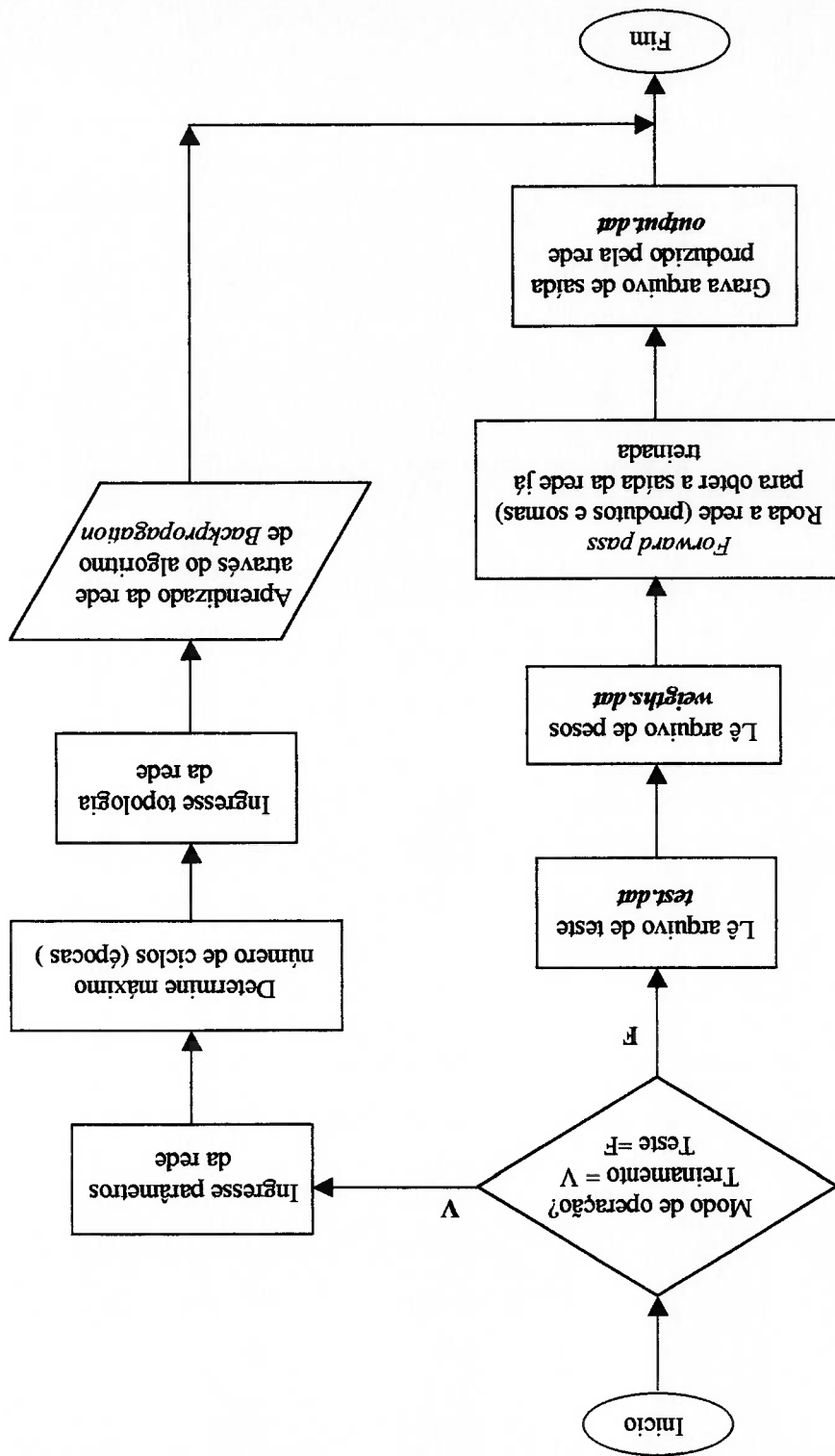
- Exploration in the Microstructure of Cognition. Foundations, PDP Research Group, MIT Press, Cambridge, Mass., 1986, Vol 1.
- SANNUM, H.; NAESS, T. Status and Trends in Design of Floating Productions Systems. Proceedings of a Workshop on Floating Structures and Offshore Operations, The Netherlands, 1987, pp.93-97.
- SHAW, I. S.; SIMÕES, M. G. Controle e Modelagem Fuzzy. São Paulo, Ed. Edgard Blücher Ltda, 1998.
- SHEFFIELD, R. Floating Drilling: Equipment and Its Use. Gulf Publishing, Houston USA, 1982.
- SHIOTSUKA, T.; YOSHIDA, K.; NAGOA, M. Active Control of Drive Motion of Four-Wheel Steering Car with Neural Network. SAE' 94, 1994, pp. 145-158.
- SHOSHA, A.; ASHOUR, M.; MOHAMED, F. Building I/O Neural Simulator of Nuclear Power Reactor. Applied Modeling and Simulation, IASTED, 1997.
- SIMÕES, M. G.; BOSE, B. K. Application of Fuzzy Neural Network in the Estimation Of Distorted Waveforms. IEEE International Symposium on Industrial Electronics, Warsaw, Poland, June 1996, pp. 415-420.
- SWINGLER, K. Applying Neural Networks: A Guide Practical. London, Academic Press, 1996.
- TAKASHINA, J. Ship Maneuvering Motion Due to Tug Moats and its Mathematical Model. Journal of Society of Naval Architects of Japan, Dec. 1986, Vol 160.
- TRELEA, I. C.; COURTOIS, F.; TRYSTRAM G. Dynamic Models for Drying and Wet Milling Quality Degradation of Corn Using Neural Networks. Drying Technology, 1997, pp. 1095-1102.
- TWOMEY, J. M.; SMITH, A. E. A Neural Network Model of the Dynamic Coefficient of Friction. 2nd Industrial Engineering Research Conference Proceedings, pp. 187-191.
- TSOUKALAS, L. H.; UHRIG, R. E. Fuzzy and Neural Approaches in Engineering. New York, A. Wiley - Interscience Publication, 1997.
- UITTENBOGAARD, R.; LANGE, F. Further Developments on Permanent Mooring Systems. Proceedings of a Workshop on Floating Structures and Offshore Operations, The Netherlands, 1987, pp. 155-176.
- VONALTROCK, C. Fuzzy Logic & NeuroFuzzy Applications Explained. Prentice-Hall Inc., Englewood Cliffs, NJ, 1995.
- WASSERMANN, P. D. Neuro Computing: Theory and Practice, New York, 1989.
- WELSTEAD, S. T. Neural Networks and Fuzzy Logic Applications. New York, John Wiley & Sons, 1994.
- WERBOS, P. Beyond Regression: New Tools for Prediction and Analysis in Behavioral Sciences. Ph.D. Thesis, Harvard University, 1974.

WICHERS, J. E. W. The Prediction of the Behavior of Single point Moored Tankers. Proceedings of a Workshop on Floating Structures and Offshore Operations, The Netherlands, 1987, pp. 125-142.

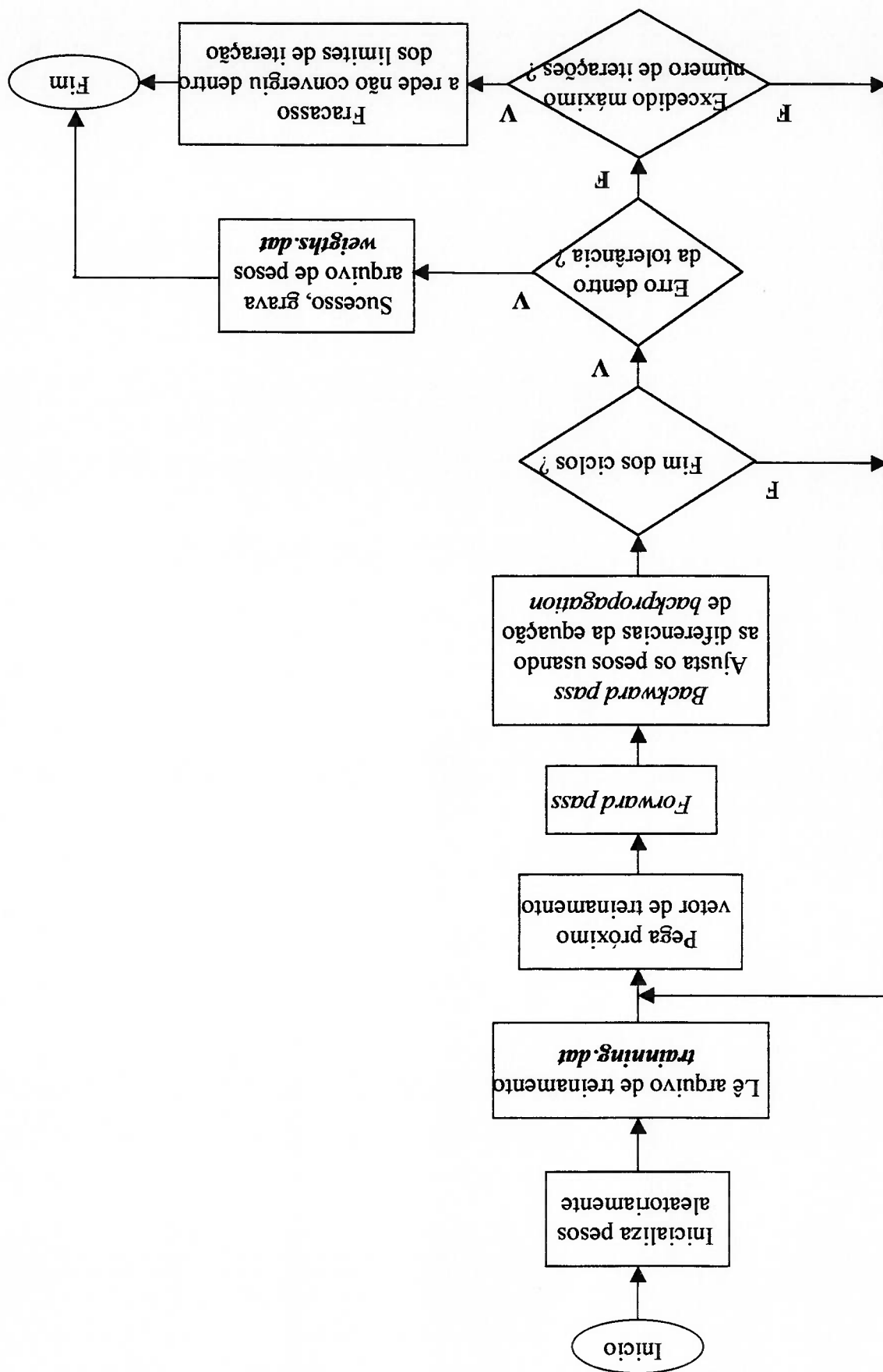
ZHANG, Y.; HEARN, G. E. SEN, P. A Multivariable Neural Controller for Automatic Ship Berthing. IEEE Transactions on Control Systems, August 1997, pp. 31-45.

FLUXOGRAMA DO PROGRAMA SIMULADOR DE REDES NEURAIS

APÊNDICE A



FLUXOGRAMA DO ALGORITMO DO BACKPROPAGATION



APÊNDICE B

Regra Delta Generalizada

Esta seção apresenta a descrição matemática formal do funcionamento de uma rede de retropropagação através do uso da regra delta generalizada. Uma rede neural é uma rede de correspondência, quando se demonstra sua capacidade para calcular alguma relação funcional entre entradas e saídas.

Por exemplo, se a entrada da rede é o valor de um ângulo θ e a saída é o cosseno do ângulo, a rede estabelece a correspondência $\theta \rightarrow \cos(\theta)$.

Suponha que tem-se um conjunto de treinamento de P pares de vetores $(x_1, y_1), (x_2, y_2), \dots, (x_p, y_p)$, que são exemplos de uma correspondência funcional,

$$(B.2) \quad y = \phi(x) : x \in R^M, y \in R^N$$

deseja-se treinar a rede para que aprenda uma aproximação

$$(B.3) \quad y' = \phi'(x)$$

Aplica-se um vetor $x_p = (x_{p1}, x_{p2}, \dots, x_{pN})$, na camada de entrada da rede. As unidades de entrada distribuem os valores às unidades da camada oculta. A *j*-ésima unidade oculta

é:

$$(B.4) \quad total_j = \sum_{h=1}^H w_{jh} x_{ph} + \theta_j$$

1. Aplica-se um vetor de entrada à rede, e calcula-se os correspondentes vetores de saída;
2. Comparam-se as saídas obtidas com as saídas corretas, e determina-se uma medida do erro;
3. Determina-se em qual direção (+ ou -) deve mudar cada peso com objetivo de reduzir o erro;
4. Determina-se a quantidade necessária para mudar cada peso;
5. Aplicam-se as conexões aos pesos, e

e:

onde o índice "o", refere-se à camada de saída. O procedimento básico para treinar a rede

$$o_{pk} = f_o^k(\text{total}_{pk})$$

(B.7)

$$\text{total}_{pk} = \sum_{j=1}^L w_{oj}^k i_{pj} + \theta_o^k$$

(B.6)

as equações para os nos de saída são os seguintes:

$$i_{pj} = f_h^j(\text{total}_h^j)$$

(B.5)

é igual à entrada *total*, então a saída deste nó é:

de polarização. O índice "h" refere-se a camada oculta. Supõe-se que a ativação deste nó

onde, w_{hj}^i é o peso da conexão procedente da *l-ésima* unidade de entrada e θ_{hj}^i é o termo

6. Repetem-se os passos 1 ao 5 com todos os vetores de treinamento, até que o erro para todos os vetores do conjunto de treinamento fique reduzido a um valor aceitável.

A lei iterativa que descreve a mudança dos pesos para redes sem unidades ocultas e com saídas lineares, denominada LMS (mínimos quadrados) ou regra delta é:

$$w(t+1)_i = w(t)_i + 2\mu e_k x_{ki} \quad (\text{B.8})$$

onde μ é uma constante positiva, x_{ki} é a i -ésima componente do k -ésimo vetor de treinamento e e_k é a diferença entre a saída obtida e o valor correto, $e_k = (d_k - y_k)$. Obtem-se uma equação similar quando a rede tem mais de duas camadas ou quando as saídas são não lineares.

APÊNDICE C

Atualização de Pesos da Camada da Saída

Ao derivar a Regra Delta, o erro do *k*-ésimo vetor de entrada é $e_k = (d_k - y_k)$, onde a entrada desejada é d_k e a saída real é y_k . Defina-se o erro de só uma unidade de saída na forma $\delta = (v^{pk} - o^{pk})$, onde o índice "p" refere-se ao *p*-ésimo vetor de treinamento, e "k" refere-se à *k*-ésima unidade de saída. Neste caso, y^{pk} é o valor de saída desejado, e o^{pk} é a saída obtida a partir da *k*-ésima unidade. O erro que se deseja minimizar é a soma dos quadrados dos erros de todas as unidades da saída:

$$E_p = \frac{1}{2} \sum_{k=1}^M \delta^{pk} \quad (C.1)$$

O fator $\frac{1}{2}$ da equação (C.1) aparece por conveniência para calcular derivadas mais adiante. Para determinar o sentido em que deve-se mudar os pesos, calcula-se o valor negativo da gradiente de E_p , VE_p relativo aos pesos w_{kj} . Para simplificar, considerou-se por separado cada componente de VE_p . Partindo da equação (C.1) e da definição de δ^{pk}

$$E_p = \frac{1}{2} \sum_{k=1}^K (v^{pk} - o^{pk})^2 \quad (C.2)$$

$$\frac{\partial E_p}{\partial w_{kj}} = (v^{pk} - o^{pk}) \left(\frac{\partial (total^{pk})}{\partial o^{pk}} \right) \frac{\partial o^{pk}}{\partial w_{kj}} \quad (C.3)$$

onde utilizou-se a equação (B.7) como valor de saída, o^{pk} , e a regra da cadeia para as derivadas parciais.

(C.9)

$$f_o^k = f_o^k (1 - o^{pk}) = o^{pk} (1 - o^{pk})$$

A função sigmoideal limita a saída, e é quase bi-estável e derivável.

(C.8)

$$f_o^k(t) = \frac{1}{1 + e^{-total_o^k}}$$

função sigmoideal ou função logística.

o fator η denomina-se parâmetro da velocidade de aprendizado. A função de saída é a

(C.7)

$$\Delta^p w_o^{hj} = \eta (V^{pk} - o^{pk}) f_o^k(t) total_o^{pk}$$

onde:

(C.6)

$$w_o^{hj}(t) = w_o^{hj}(t) + \Delta^p w_o^{hj}(t)$$

seguinte maneira:

ao gradiente negativo. Desta maneira, os pesos da camada de saída atualizam-se da

Com relação a amplitude da variação dos pesos, considerou-se que será proporcional

(C.5)

$$\frac{\partial E}{\partial w_o^{hj}} = (V^{pk} - o^{pk}) f_o^k(t) total_o^{pk}$$

Combinando as equações (C.3) e (C.4) tem-se para o gradiente negativo:

(C.4)

$$\frac{\partial total_o^{pk}}{\partial w_o^{hj}} = \sum_{j=1}^L w_o^{hj} I^{pj} + h_o^k = I^{pj}$$

(C.10)

$$w_{o}^{k} (t + 1) = w_{o}^{k} (t) + \eta (v^{pk} - o^{pk}) f_{pj}^{k}$$

Resumindo as equações de atualização dos pesos, define-se:

(C.11)

$$\delta_{o}^{pk} = (v^{pk} - o^{pk}) f_{o}^{k} \text{total}_{o}^{pk} = \delta_{pk} f_{o}^{k} \text{total}_{o}^{pk}$$

pode-se escrever a equação de atualização dos pesos na forma:

(C.12)

$$w_{o}^{k} (t + 1) = w_{o}^{k} (t) + \eta \delta_{o}^{pk} t_{pj}^{k}$$

independentemente da forma funcional da função de saída, f_{o}^{k} .

Neste ponto deseja-se fazer um comentário acerca da relação existente entre o

método de descida do gradiente, que foi descrito e a técnica dos mínimos quadrados. Se a

regra delta generalizada fosse completamente análoga ao método de mínimos quadrados,

não se mudaria nenhum dos valores dos pesos até que se houvessem apresentados à rede

todos os padrões de treinamento pelo menos uma vez. Simplesmente acumularam-se as

mudanças, na medida que fossem processados cada padrão, seria feita uma soma e a

atualização dos pesos.

Repete-se o processo até que o erro seja aceitavelmente baixo. O erro que minimiza

este processo é:

$$E = \sum_{p=1}^d E_p$$

(C.13)

onde P é o número de padrões do conjunto de treinamento.

APÊNDICE D

Atualizações dos Pesos das Camadas Ocultas

Se desejarmos repetir para a camada oculta o mesmo tipo de cálculo que se tem realizado para a camada de saída, surge um problema: quando se tenta determinar uma medida do erro das saídas para as unidades da camada oculta. Sabe-se qual é a saída obtida, e intuitivamente, o erro E_p deve estar relacionado de alguma maneira com os valores de saída da camada oculta. Pode-se comprovar a veracidade desta intuição voltando-se à equação (C.1), que pode ser escrita da seguinte forma:

$$(D.1) \quad E_p = \frac{1}{2} \sum_k (y^{pk} - o^{pk})^2$$

$$(D.2) \quad = \frac{1}{2} \sum_k (y^{pk} - f_o^k(\text{total}^{pk}))^2$$

$$(D.3) \quad = \frac{1}{2} \sum_k \left(y^{pk} - f_o^k \left(\sum_f w_{of}^{kj} i^{pj} + \theta_o^k \right) \right)^2$$

Sabe-se que i^{pj} depende dos pesos das camadas ocultas através das equações (B.5) e

(B.6). Pode-se aproveitar deste fato para calcular-se o gradiente de E_p com referência aos

pesos das camadas ocultas, conforme (D.4) e (D.5).

$$(D.4) \quad \frac{\partial E_p}{\partial w_h^j} = \frac{1}{2} \sum_k \frac{\partial w_h^j}{\partial (y^{pk} - o^{pk})} (y^{pk} - o^{pk})$$

(D.10)

$$\delta_h^{pj} = f_h^j \left(\text{total}_h^{pj} \right) x^{pj} \sum_{o^{pk}} \delta_o^{pk} w_o^{pk}$$

termo de erro para a camada oculta:

oculta para determinar as mudanças de peso adequadas a essa camada. Se definimos um seja os erros conhecidos da camada de saída se propagam de volta em sentido da camada os termos de erro, δ_o^{pk} da camada de saída. Assim surge a noção de *retropropagação*, ou Se observa que todas as atualizações dos pesos da camada oculta depende de todos

(D.9)

$$\Delta^p w_h^{pj} = \eta \cdot f_h^j \left(\text{total}_h^{pj} \right) x^{pj} \sum_{o^{pk}} \delta_o^{pk} w_o^{pk}$$

dada na seção anterior para escrever:

onde η é, mais uma vez a velocidade de aprendizado. Pode-se utilizar a definição de δ_o^{pk}

(D.8)

$$\Delta^p w_h^{pj} = \eta \cdot f_h^j \left(\text{total}_h^{pj} \right) x^{pj} \sum_{o^{pk}} \left(v^{pk} - o^{pk} \right) f_o^k \left(\text{total}_o^{pk} \right) w_o^{pk}$$

equação (D.6). Tem-se:

Atualizando-se os pesos da camada oculta proporcionalmente ao valor negativo da

(D.6)

$$\frac{\partial EP}{\partial w_h^{pj}} = - \sum_{o^{pk}} \left(v^{pk} - o^{pk} \right) f_o^k \left(\text{total}_o^{pk} \right) w_o^{pk} \left(\text{total}_h^{pj} \right) x^{pj} \sum_{o^{pk}} \delta_o^{pk} w_o^{pk}$$

equações anteriores. O resultado é o seguinte:

Cada um dos fatores da equação (D.5) podem ser calculados explicitamente das

(D.5)

$$= - \sum_{o^{pk}} \left(v^{pk} - o^{pk} \right) \frac{\partial \left(\text{total}_o^{pk} \right)}{\partial o^{pk}} \frac{\partial \left(\text{total}_o^{pk} \right)}{\partial i^{pj}} \frac{\partial \left(\text{total}_h^{pj} \right)}{\partial w_h^{pj}} x^{pj}$$

As equações da atualização dos pesos passam a ser análogas às correspondentes à

camada de saída:

$$(D.11) \quad w_h^{\#}(t+1) = w_h^{\#}(t) + \eta \delta_h^{\#} x_i$$

Por tanto se observamos a equação (C.12) Do apêndice B, como a equação (D.11) desta seção tem a mesma forma que a equação (B.8), que é a Regra Delta.