

**Thiago Richter**

**ARQUITETURA DE SISTEMA INTELIGENTE  
PARA SENSORIAMENTO VIRTUAL DE  
OXIGÊNIO EM VEÍCULOS BICOMBUSTÍVEIS  
COM INJEÇÃO ELETRÔNICA**

Dissertação apresentada à Escola de Engenharia de São Carlos da Universidade de São Paulo, como parte dos requisitos para a obtenção do Título de Mestre em Engenharia Elétrica.

Orientador: Prof. Dr. Ivan Nunes da Silva

São Carlos

2009

# **Livros Grátis**

<http://www.livrosgratis.com.br>

Milhares de livros grátis para download.

AUTORIZO A REPRODUÇÃO E DIVULGAÇÃO TOTAL OU PARCIAL DESTE TRABALHO, POR QUALQUER MEIO CONVENCIONAL OU ELETRÔNICO, PARA FINS DE ESTUDO E PESQUISA, DESDE QUE CITADA A FONTE.

**Ficha catalográfica preparada pela Seção de Tratamento  
da Informação do Serviço de Biblioteca – EESC/USP**

R535a Richter, Thiago  
Arquitetura de sistema inteligente para sensoriamento virtual de oxigênio em veículos bicompostíveis com injeção eletrônica / Thiago Richter ; orientador Ivan Nunes da Silva -- São Carlos, 2009.

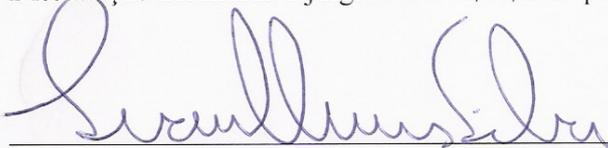
Dissertação (Mestrado-Programa de Pós-Graduação em Engenharia Elétrica e Área de Concentração em Sistemas Dinâmicos -- Escola de Engenharia de São Carlos da Universidade de São Paulo, 2009.

1. Redes neurais artificiais. 2. Sensoriamento virtual. 3. Sensor de oxigênio virtual. 4. Veículos bicompostíveis. 5. Injeção eletrônica de combustível. I. Título.

**FOLHA DE JULGAMENTO**

Candidato: Bacharel **THIAGO RICHTER**.

Dissertação defendida e julgada em 12/08/2009 perante a Comissão Julgadora:



Prof. Associado **IVAN NUNES DA SILVA (Orientador)**  
(Escola de Engenharia de São Carlos/USP)

APROVADO



Prof. Dr. **ALEDIR SILVEIRA PEREIRA**  
(Universidade Estadual Paulista "Julio de Mesquita Filho"/UNESP/campus de São José do Rio Preto)

APROVADO



Dr. **ANTÔNIO VANDERLEI ORTEGA**  
(Racetrax Engenharia Industrial Ltda)

Aprovado



Prof. Titular **GERALDO ROBERTO MARTINS DA COSTA**  
Coordenador do Programa de Pós-Graduação em Engenharia Elétrica e  
Presidente da Comissão de Pós-Graduação

Protocolo nº 12.08.2009  
Assessoria



Dedico este trabalho a três pessoas muito importantes para mim e que estavam sempre presentes durante a realização desta dissertação:

Marilda Fávero, Lucas Richter e Tatiane Veronese.



## **Agradecimentos**

Inicialmente aos meus pais, Oswaldo e Marilda, por me darem força, conhecimento e determinação para percorrer mais esta etapa de minha vida.

Ao meu irmão Lucas, e a minha mãe, pelo apoio, compreensão e, principalmente, paciência em todos os momentos.

À minha noiva Tatiane, pela paciência, compreensão, apoio, confiança e carinho nos melhores ou piores momentos.

Ao meu professor e orientador Dr. Ivan Nunes da Silva, por acreditar no projeto, pela ajuda na elaboração deste trabalho e, principalmente, pela transmissão de experiência e conhecimento. Sem dúvida um excelente professor e uma excelente pessoa, profissional e pessoalmente.

À Delphi Automotive Systems do Brasil, por permitir e apoiar o desenvolvimento deste trabalho. À Elisabete M. D. Oliveira, Fábio A. M. Fantinato, Amaury F. de Oliveira, Patrícia F. S. Veras, Roberto M. Stein, Orlando Volpato Filho, Felipe Campos, Sandro Souza e muitos outros.

A todos os professores que encontrei ao longo de todos esses anos de estudo e que sempre acreditaram em mim. A todas as instituições por onde transitei - à Universidade de São Paulo, à Escola de Engenharia de Piracicaba, ao Exército Brasileiro, à Escola Técnica Estadual Cel. Fernando Febeliano da Costa e ao SESI 185.

Aos meus avós, tios e tias, primos e primas, amigos, enfim, a todos que, direta ou indiretamente, contribuíram ao longo deste trabalho.



*“Tudo o que sei é que nada sei.”*

Sócrates

*“Forte é permanecer quieto.”*

João Guimarães Rosa



## **Resumo**

**RICHTER, T. (2009). Arquitetura de Sistema Inteligente para Sensoriamento Virtual de Oxigênio em Veículo Bicombustível com Injeção Eletrônica.** Dissertação (Mestrado) – Escola de Engenharia de São Carlos, Universidade de São Paulo, 2009.

A indústria automobilística é um dos mais importantes setores da economia no Brasil e no mundo. Nos últimos anos viu-se praticamente obrigada a melhorar o desempenho de seus veículos produzidos e reduzir seus custos. Um dos marcos desta transformação foi o desenvolvimento do sensor de oxigênio, sendo este um dos principais elementos dos sistemas gerenciadores de motor. Esta dissertação propõe o estudo de arquiteturas de sistemas inteligentes para sensoriamento virtual de oxigênio em veículos bicombustíveis, utilizando-se redes neurais artificiais supervisionadas, com arquitetura Perceptron multicamadas. As topologias implementadas atingiram resultados com erros relativos médios menores que 1% em centenas de topologias. Verificou-se também que para o sensoriamento virtual de oxigênio em veículos bicombustíveis, a abordagem de se realizar treinamentos com todos os tipos de combustíveis, segmentando conjuntos de todo o universo de dados, mostra-se a mais adequada.

Palavras-chave: Redes Neurais Artificiais, Sensoriamento Virtual, Sensor de Oxigênio Virtual, Veículos Bicombustíveis, Injeção Eletrônica de Combustível.



## **Abstract**

**RICHTER, T. (2009). Intelligent System Architecture for Virtual Sensing of Oxygen in Bi-Fuel Vehicle with Electronic Fuel Injection.** Dissertation (Master's Degree) – *Escola de Engenharia de São Carlos, Universidade de São Paulo*, 2009.

The automotive industry is one of the most important sectors in Brazilian's economy and in the world. In recent years, this industry has been forced to improve the performance of their produced vehicles and to reduce their costs. One of the landmarks of this transformation was the development of the oxygen sensor, which is one of the main elements of the Engine Management Systems. This dissertation proposes the use of intelligent systems architectures for virtual oxygen sensing of bi-fuel vehicles, using multilayer Perceptron artificial neural networks. The implemented topologies reach results with mean relative errors less than 1% in hundreds of topologies. It was also noted that the approach to train the neural network with all types of fuels, using subsets of data universe, it is the most appropriate to have a virtual sensing of oxygen in bi-fuel vehicles.

**Keywords:** Artificial Neural Networks, Virtual Sensing, Virtual Oxygen Sensor, Bi-Fuel Vehicles, Electronic Fuel Injection.



## Lista de ilustrações

<i>Figura 2.1: Utilização industrial da injeção eletrônica de combustível (AMEY, 1995).</i>	32
<i>Figura 2.2: Mapa do avanço da ignição (BOSCH, 2005).</i>	37
<i>Figura 2.3: Exemplo de entradas e saídas de um EMS (DELPHI, 2002).</i>	38
<i>Figura 3.1: Sensor de Oxigênio de Banda Larga (BOSCH, 2005).</i>	40
<i>Figura 3.2: Curva de resposta do sensor de oxigênio de banda larga (DELPHI, 2007).</i>	40
<i>Figura 3.3: Ilustração do funcionamento do sensor de oxigênio.</i>	41
<i>Figura 3.4: Curva de resposta do sensor de oxigênio (DELPHI, 2005).</i>	42
<i>Figura 3.5: Sensor de oxigênio planar (DELPHI, 2005).</i>	43
<i>Figura 3.6: Emissões do motor com gasolina comum (DELPHI, 2005).</i>	46
<i>Figura 3.7: Típicas localizações do Sensor de Oxigênio (BOSCH, 2005).</i>	47
<i>Figura 3.8: Eficiência da conversão do catalisador de três vias (DELPHI, 2005).</i>	48
<i>Figura 4.1: Neurônio biológico (LNCC, 2008).</i>	52
<i>Figura 4.2: Principais funções de ativação.</i>	54
<i>Figura 4.3: Representação de um neurônio artificial.</i>	55
<i>Figura 4.4: Comparação entre o neurônio biológico e o artificial.</i>	55
<i>Figura 4.5: Rede feedforward (com camada única).</i>	57
<i>Figura 4.6: Rede feedforward com duas camadas intermediárias e uma camada de saída.</i>	58
<i>Figura 4.7: Rede recorrente com realimentação.</i>	58
<i>Figura 4.8: Fluxo do processamento do algoritmo back-propagation.</i>	65
<i>Figura 4.9: Modelo de TDNN.</i>	68
<i>Figura 4.10: Modelo da PMC recorrente.</i>	69
<i>Figura 5.1: Modelo de PMC com uma CNE usado.</i>	72
<i>Figura 5.2: Modelo de PMC com duas CNE usado.</i>	73
<i>Figura 5.3: Exemplo de variáveis normalizadas usadas no sistema (I).</i>	74
<i>Figura 5.4: Exemplo de variáveis normalizadas usadas no sistema (II).</i>	75
<i>Figura 5.5: Exemplo da variável lambda (saída desejada).</i>	75
<i>Figura 5.6: Detalhe da saída desejada.</i>	76
<i>Figura 6.1: Representação do desenvolvimento dos testes.</i>	82
<i>Figura 6.2: Exemplo de resultado invalidado de treinamento e teste com E20-Segmentado.</i>	87
<i>Figura 6.3: Exemplo de resultado de treinamento e teste com E20-Segmentado (Situação I).</i>	88
<i>Figura 6.4: Exemplo de resultado de treinamento e teste com E20-Segmentado (Situação II).</i>	89
<i>Figura 6.5: Exemplo de resultado de treinamento e teste com E20-Segmentado (Situação III).</i>	90

<i>Figura 6.6: Exemplo de resultado de treinamento e teste com E20-Segmentado (Situação IV).</i>	91
<i>Figura 6.7: Exemplo de resultado de treinamento e teste com E20-Segmentado (Situação V).</i>	91
<i>Figura 6.8: Exemplo de resultado de treinamento e teste com E20-Segmentado (Situação VI).</i>	92
<i>Figura 6.9: Exemplo de resultado invalidado de treinamento e teste com E20-Completo.</i>	93
<i>Figura 6.10: Exemplo de resultado de treinamento e teste com E20-Completo (Situação I).</i>	94
<i>Figura 6.11: Detalhes dos resultados apresentados na Figura 6.10.</i>	94
<i>Figura 6.12: Exemplo de resultado de treinamento e teste com E20-Completo (Situação II).</i>	95
<i>Figura 6.13: Exemplo de resultado de treinamento e teste com E100-Segmentado (Situação I).</i>	96
<i>Figura 6.14: Exemplo de resultado invalidado de treinamento e teste com E100-Segmentado.</i>	97
<i>Figura 6.15: Exemplo de resultado de treinamento e teste com E100-Segmentado (Situação II).</i>	98
<i>Figura 6.16: Detalhes dos resultados apresentados na Figura 6.14.</i>	99
<i>Figura 6.17: Exemplo de resultado de treinamento e teste com E100-Completo (Situação I).</i>	100
<i>Figura 6.18: Detalhes dos resultados apresentados na Figura 6.17.</i>	100
<i>Figura 6.19: Exemplo de resultado de treinamento e teste com E100-Completo (Situação II).</i>	101
<i>Figura 6.20: Exemplo de resultado de treinamento e teste com E100-Completo (Situação III).</i>	102
<i>Figura 6.21: Exemplo de resultado invalidado de treinamento com E20-Segmentado e teste com E20-Completo (Situação I).</i>	103
<i>Figura 6.22: Exemplo de resultado invalidado de treinamento com E20-Segmentado e teste com E20-Completo (Situação II).</i>	103
<i>Figura 6.23: Exemplo de resultado invalidado de treinamento com E100-Segmentado e teste com E100-Completo (Situação I).</i>	104
<i>Figura 6.24: Exemplo de resultado invalidado de treinamento com E100-Segmentado e teste com E100-Completo (Situação II).</i>	104
<i>Figura 6.25: Exemplo de resultado invalidado de treinamento com E20-Segmentado e teste com E100-Segmentado.</i>	105
<i>Figura 6.26: Exemplo de resultado invalidado de treinamento com E20-Completo e teste com E100-Completo.</i>	106
<i>Figura 6.27: Exemplo de resultado invalidado de treinamento com E100-Segmentado e teste com E20-Segmentado.</i>	107
<i>Figura 6.28: Exemplo de resultado invalidado de treinamento com E100-Completo e teste com E20-Completo.</i>	108
<i>Figura 6.29: Exemplo de resultado de treinamento com E20 e E100-Segmentado e teste com E20-Segmentado (Situação I).</i>	109

<i>Figura 6.30: Exemplo de resultado de treinamento com E20 e E100-Segmentado e teste com E20-Segmentado (Situação II).</i>	110
<i>Figura 6.31: Exemplo de resultado de treinamento com E20 e E100-Completo e teste com E20-Completo (Situação I).</i>	111
<i>Figura 6.32: Exemplo de resultado de treinamento com E20 e E100-Completo e teste com E20-Completo (Situação II).</i>	111
<i>Figura 6.33: Exemplo de resultado de treinamento com E20 e E100-Segmentado e teste com E100-Segmentado (Situação I).</i>	112
<i>Figura 6.34: Exemplo de resultado de treinamento com E20 e E100-Segmentado e teste com E100-Segmentado (Situação II).</i>	113
<i>Figura 6.35: Exemplo de resultado de treinamento com E20 e E100-Completo e teste com E100-Completo (Situação I).</i>	113
<i>Figura 6.36: Exemplo de resultado de treinamento com E20 e E100-Completo e teste com E100-Completo (Situação II).</i>	114
<i>Figura 6.37: Exemplo de resultado invalidado de treinamento com E20 e E100-Segmentado e teste com E20-Completo.</i>	115
<i>Figura 6.38: Exemplo de resultado invalidado de treinamento com E20 e E100-Segmentado e teste com E100-Completo.</i>	116
<i>Figura 6.38: Exemplo de resultado de teste com topologias iguais - E20-Completo (Situação I).</i>	117
<i>Figura 6.40: Exemplo de resultado de teste com topologias iguais – E100-Completo (Situação I).</i>	118
<i>Figura 6.41: Exemplo de resultado de teste com topologias iguais - E20-Completo (Situação II).</i>	119
<i>Figura 6.42: Exemplo de resultado de teste com topologias iguais – E100-Completo (Situação II).</i>	119
<i>Figura A.1: Participação dos veículos flex nas vendas de veículos leves.</i>	134



## Lista de Tabelas

<i>Tabela 3.1: Estequiometria de diversos combustíveis (DELPHI, 2005).</i>	45
<i>Tabela 5.1: Variáveis utilizadas nos treinamentos</i>	78
<i>Tabela 5.2: Características utilizadas nas topologias desenvolvidas.</i>	79
<i>Tabela 6.1: Melhores desempenhos – Treinamentos e Testes utilizando mesmos mapas de dados.</i>	83
<i>Tabela 6.2: Melhores desempenhos – Treinamentos e Testes utilizando mapas de dados distintos.</i>	84
<i>Tabela 6.3: Melhores desempenhos – Treinamentos e Testes utilizando tipos de discretização ou mapas de dados distintos.</i>	85
<i>Tabela 6.4: Melhores desempenhos – Treinamentos utilizando ambos os mapas de dados e testes utilizando apenas um dos mapas.</i>	86
<i>Tabela 6.5: Resumo dos resultados de testes – Treinamentos e Testes utilizando mesmos mapas de dados.</i>	120
<i>Tabela 6.6: Resumo dos resultados de testes – Treinamentos e Testes utilizando mapas de dados distintos.</i>	120
<i>Tabela 6.7: Resumo dos resultados de testes – Treinamentos e Testes utilizando tipos de discretização ou mapas de dados distintos.</i>	121
<i>Tabela 6.8: Resumo dos resultados de testes – Treinamentos utilizando ambos os mapas de dados e testes utilizando apenas um dos mapas.</i>	121
<i>Tabela A.1: PROCONVE – Limites de emissão para veículos leves novos.</i>	140



## Lista de Abreviaturas ou Siglas

- ADALINE** – *Adaptive Linear Element*, Elemento Linear Adaptativo;
- ANFAVEA** – Associação Nacional dos Fabricantes de Veículos Automotores;
- ART** – *Adaptive Resonance Theory*, Teoria da Ressonância Adaptativa;
- CAFE** – *Corporate Average Fuel Economy*, Economia Média Unificada de Combustível;
- CETESB** – Companhia de Tecnologia de Saneamento Ambiental do estado de São Paulo;
- CNE** – Camada Neural Escondida;
- EMS** – *Engine Management System*, Sistema de Gerenciamento de Motor;
- ESALQ** – Escola de Agricultura Luiz de Queiros;
- EUA** – Estados Unidos da América;
- FIS** – *Fuzzy Inference System*, Sistema de inferência Fuzzy;
- GM** – General Motors;
- GNV** – Gás Natural Veicular;
- IPT** – Instituto de Pesquisas Tecnológicas de São Paulo;
- MLP** – *Multilayer Perceptron*, Perceptron Multicamadas;
- PIB** – Produto Interno Bruto;
- PMC** – Perceptron Multicamadas;
- PROCONVE** - Programa de Controle da Poluição do Ar por Veículos Automotores;
- RBF** – *Radial Basis Function*, Função de Base Radial;
- RNA** – Redes Neurais Artificiais
- RTRL** – *Real-Time Recurrent Learning*, Treinamento recorrente em tempo-real;
- SAE** – *Society of Automotive Engineers*, Sociedade da Engenharia da Mobilidade;
- TDNN** – *Time-Delay Neural Network*, Rede Neural com Atraso de Tempo;
- TSK** – Modelo Takagi, Sugeno, Kang;
- UCE** – Unidade de Controle Eletrônico;
- UNICA** – União da Indústria de Cana-de-açúcar;
- VW** – Volkswagen.



## Lista de Símbolos e Fórmulas

°C – Grau Celsius;

*A/F* – *Air/Fuel ratio*, razão Ar/Combustível;

CO – Monóxido de Carbono;

CO<sub>2</sub> – Dióxido de Carbono;

$\bar{e}_r$  – Erro Relativo Médio;

**E100** – Combustível contendo 100% de álcool;

**E20** – Combustível contendo a mistura de 24% de álcool à gasolina;

**g** – grama;

**g/g (A/F)** – grama de ar por grama de combustível;

**g/s** – grama por segundo;

HC – Hidróxido de Carbono;

**IPT** – Instituto de Pesquisas Tecnológicas de São Paulo;

**KPa** – Kilopascal;

$\lambda$  – Valor Lambda.

**L** – Litro;

*MAF* – *Mass of Air Flow*, Fluxo da massa de ar;

*MAP* – *Manifold Absolute Pressure*, Pressão absoluta do coletor de admissão;

**ms** – milissegundos;

**MTBE** - Metil Tércio Butil Éter;

**Nm** – Newton metro;

**NO<sub>x</sub>** – Óxidos de Nitrogênio;

**O<sub>3</sub>** – Ozônio;

**rpm** – Rotações por minuto;

**s** – Segundo;

**SO<sub>2</sub>** – Dióxido de Enxofre;

**SrTiO<sub>3</sub>** – Titanato de Estrôncio

**TiO<sub>2</sub>** – Dióxido de Titânio

**V** – Volt, Tensão;



## Sumário

Capítulo 1 - Introdução .....	25
<b>1.1 - Motivação e Relevância do Trabalho</b> .....	<b>25</b>
<b>1.2 - Proposta e Justificativa da Dissertação</b> .....	<b>27</b>
<b>1.3 - Organização da Dissertação</b> .....	<b>28</b>
Capítulo 2 - Aspectos de Sistemas de Gerenciamento de Motor .....	29
<b>2.1 - Introdução</b> .....	<b>29</b>
<b>2.2 - Evolução dos EMS</b> .....	<b>29</b>
<b>2.3 - Sistemas de Controle dos Motores</b> .....	<b>35</b>
Capítulo 3 - Aspectos de Sensores de Oxigênio.....	39
<b>3.1 - Introdução</b> .....	<b>39</b>
<b>3.2 - Tipos de Sensores Para Estimação de Oxigênio</b> .....	<b>39</b>
<b>3.3 - Fundamentos de Estequiometria</b> .....	<b>44</b>
3.3.1 -Efeitos da Taxa Ar/Combustível na Combustão .....	44
3.3.2 -Misturas de Combustíveis .....	45
3.3.3 -Localizações dos Sensores de Oxigênio e Catalisadores.....	47
Capítulo 4 - Aspectos de Redes Neurais Artificiais .....	49
<b>4.1 - Modelo de Neurônio Biológico e Artificial</b> .....	<b>51</b>
<b>4.2 - Arquiteturas de Redes Neurais</b> .....	<b>56</b>
4.2.1 -Redes de Camada Única ( <i>Feedforward</i> ).....	56
4.2.2 -Redes de Múltiplas Camadas ( <i>Feedforward</i> ) .....	57
4.2.3 -Redes Recorrentes .....	58
<b>4.3 - Tipos de Treinamento</b> .....	<b>59</b>
<b>4.4 - Rede Perceptron Multicamadas</b> .....	<b>60</b>
4.4.1 -Treinamento de redes PMC .....	64
4.4.2 -Perceptron Multicamadas como aproximador universal .....	67
4.4.3 -Redes com atraso de tempo .....	68
<b>4.5 - Aplicações de Redes Neurais em Sensoriamento Virtual</b> .....	<b>69</b>
Capítulo 5 - Arquitetura de Sistema Inteligente Desenvolvida .....	71
<b>5.1 - Introdução</b> .....	<b>71</b>
<b>5.2 - Aspectos Básicos do Sistema Inteligente</b> .....	<b>72</b>
<b>5.3 - Detalhes da Arquitetura do Sistema Inteligente Desenvolvido</b> .....	<b>76</b>
Capítulo 6 - Resultados Experimentais .....	81
<b>6.1 - Introdução</b> .....	<b>81</b>
<b>6.2 - Resultados de Testes como Monocombustíveis</b> .....	<b>87</b>
6.2.1 -Treinamentos e Testes com E20.....	87
6.2.2 -Treinamentos e Testes com E100 .....	96
6.2.3 -Treinamentos com Mapas Segmentados e Testes com Completos .....	102

<b>6.3 - Resultados de Testes como Bicombustíveis .....</b>	<b>105</b>
6.3.1 - Treinamentos com E20 e Testes com E100 .....	105
6.3.2 - Treinamentos com E100 e Testes com E20 .....	107
6.3.3 - Treinamentos com E20 e E100 e Testes com E20 .....	109
6.3.4 - Treinamentos com E20 e E100 e Testes com E100 .....	112
6.3.5 - Treinamentos com Mapas Segmentados e Testes com Completos .....	115
6.3.6 - Testes de redes com topologias iguais para E20 e E100 .....	117
<b>6.4 - Síntese dos Resultados de Testes.....</b>	<b>120</b>
Capítulo 7 - Conclusões Gerais e Trabalhos Futuros.....	123
<b>7.1 - Conclusões Gerais .....</b>	<b>123</b>
<b>7.2 - Trabalhos Futuros.....</b>	<b>125</b>
Referências Bibliográficas .....	127
Apêndice A - Veículos Bicombustíveis.....	131
Apêndice B - Métodos Levenberg-Marquardt e Levenberg-Marquardt com Regularização Bayesiana .....	141

## Capítulo 1 - Introdução

### 1.1 - Motivação e Relevância do Trabalho

A indústria automobilística é um dos mais importantes setores da economia no Brasil e no mundo, tendo um faturamento líquido de mais de US\$ 52 bilhões apenas na produção brasileira de veículos em 2007, ou seja, considerando somente as montadoras, a participação da indústria automobilística no PIB industrial corresponde a 18% (ANFAVEA, 2008).

O início da operação de montadoras no Brasil aconteceu em maio de 1919, quando a Ford Motors Company iniciou suas atividades na cidade de São Bernardo do Campo (EMBAIXADA AMERICANA, 2007). O país tem atualmente 40 fábricas de veículos leves, incluindo as maiores montadoras norte americanas, asiáticas e européias, ou seja, General Motors, Ford, Mercedes-Benz, Honda, Toyota, Fiat, Renault e Peugeot Citroën.

O Brasil é o sexto maior produtor de veículos do mundo, com uma produção de 3,22 milhões de veículos leves em 2008 (OICA, 2009), sendo que mais de 87% de todos os automóveis licenciados no Brasil foram de veículos equipados com a tecnologia bicombustível (ANFAVEA, 2009).

Nos últimos anos a indústria automobilística viu-se praticamente obrigada a melhorar o desempenho de seus veículos produzidos ano após ano e a reduzir

seus custos vertiginosamente. Um dos marcos desta transformação foi o desenvolvimento do sensor de oxigênio, em que foi possível implementar os Sistemas Gerenciadores de Motor (*EMS – Engine Management System*) que possibilitam, entre suas diversas funções, controlar a combustão e, conseqüentemente, as emissões de um veículo. Outro destaque na indústria, em especial a brasileira, foi o desenvolvimento de motores bicomustíveis, possibilitando o uso de dois tipos de combustíveis em qualquer proporção, permitindo aos proprietários a escolha dos mesmos. Dependendo desta escolha, a redução na emissão de gases prejudiciais ao meio ambiente, outro grande balizador da indústria, poderá ser grande, além também de permitir ao proprietário uma economia financeira com combustíveis.

Paralelamente, considerando os últimos anos, as aplicações de sistemas inteligentes na indústria automotiva vêm também aumentando. Diversas soluções para os mais variados sistemas veiculares têm sido desenvolvidos, de controle neural de marcha lenta (VICENTE et al., 2007) à estimativa de consumo de combustível (PARLAK et al., 2006), de redução no tempo de desenvolvimento do *EMS* (SHAYLER et al., 2000) ao controle da taxa A/F em veículos bicomustíveis (GNANAM et al., 2003), etc.

Assim, o desenvolvimento deste trabalho é motivado pela busca constante da indústria automobilística por aumento de desempenho, redução de custos e avanços tecnológicos, assim como pelas diversas inovações que se têm testemunhadas quando da utilização de sistemas inteligentes na indústria automotiva.

## 1.2 - Proposta e Justificativa da Dissertação

O presente trabalho propõe o estudo de arquiteturas de sistemas inteligentes para sensoriamento virtual de oxigênio em veículos bicombustíveis, podendo assim, substituir um sensor físico por um sensor virtual. A partir de valores estimados do sensor de oxigênio, implementa-se um sensor de oxigênio virtual por intermédio do uso de redes neurais artificiais supervisionadas.

Em relação aos estudos sobre sensores de oxigênio virtuais encontrados na literatura (AMATO et al., 2005; KAMAT et al., 2006), a principal inovação deste trabalho será o estudo deste tema para veículos bicombustíveis e não somente para monocombustíveis (diesel ou gasolina).

Com o desenvolvimento de um sistema inteligente para estimação da leitura do sensor, a indústria poderá deixar de utilizar o dispositivo físico, permitindo então a redução de custos na produção de veículos.

As redes neurais pesquisadas terão como entradas os mapas de dados fornecidos pelo *EMS*, assim como a saída desejada, e a resposta da rede corresponderá ao valor estimado para o sensor em função das entradas fornecidas. Os resultados das diversas topologias e configurações treinadas serão analisados e as melhores configurações serão então selecionadas para cada mapa de dados, ou seja, usando o mapa de E20 (24% de etanol misturado à gasolina) serão selecionadas as melhores configurações, e usando 100% de etanol (E100) serão selecionadas outras. Posteriormente, serão então verificadas as aplicações

das melhores configurações do mapa E20 no E100 e vice-versa, permitindo-se a seleção de configurações aplicáveis a ambos os mapas.

### 1.3 - Organização da Dissertação

Apresenta-se nesta seção uma introdução sobre o conteúdo abordado em cada capítulo.

No Capítulo 2 será apresentado o tema sobre Sistemas de Gerenciamento de Motores em função da importância que os mesmos têm para o controle de veículos bicombustíveis.

Um dos principais sensores usados no *EMS*, ou seja, sensores de oxigênio, será abordado no Capítulo 3.

Aspectos de redes neurais artificiais e as principais características da arquitetura perceptron multicamadas, além de algumas aplicações em estimação de valores, serão apresentadas no Capítulo 4.

A arquitetura do sistema desenvolvido, incluindo aspectos básicos e detalhes da arquitetura, será abordada no Capítulo 5.

No Capítulo 6 serão apresentadas tabelas com os melhores resultados, diversos exemplos de simulações, como sendo monocombustíveis ou bicombustíveis, assim como uma síntese dos mesmos.

Finalizando, no Capítulo 7, serão apresentadas as principais conclusões da arquitetura desenvolvida para veículos bicombustíveis e, também, os trabalhos futuros que poderão ser realizados mediante a continuação desta investigação.

## Capítulo 2 - Aspectos de Sistemas de Gerenciamento de Motor

Neste capítulo há uma breve exploração dos sistemas de controle eletrônico utilizados em motores de combustão interna de ciclo Otto.

### 2.1 - Introdução

Os Sistemas de Gerenciamento de Motor (*EMS – Engine Management Systems*) atuam de maneira integrada com diversos sistemas veiculares (sistema de injeção, de ignição, de recirculação de gases do escapamento, etc), os quais têm como missão gerenciar tais sistemas.

O uso do *EMS* possibilita melhor controle, melhor sensibilidade, maior eficiência dos combustíveis e do motor, assim como menores níveis de poluição. Sistemas mais modernos podem controlar outros componentes ou sistemas automotivos, como por exemplo, o turbo ou a exaustão, e se comunicam com outros sistemas, tais como aquele de transmissão.

### 2.2 - Evolução dos *EMS*

Os primeiros sistemas de injeção de combustível para motores de ciclo Otto começaram a ser desenvolvidos em meados de 1950, sendo a maioria deles sistemas mecânicos (BOSCH, 2005). A fábrica alemã Mercedes-Benz lançou em 1954 seu modelo esportivo 300SL (GIVENS, 1976), que utilizava um sistema de

injeção direta Bosch, ou seja, injetava o combustível diretamente na câmara de explosão. A partir de 1957, o sistema de injeção passou a ser de maneira indireta no coletor de admissão. Neste mesmo ano, foi apresentado no encontro anual da SAE (*Society of Automotive Engineers* – Sociedade de Engenheiros da Mobilidade), em Detroit, EUA, pela GM (General Motors) e pela Bendix os seus respectivos sistemas de injeção para motores de ciclo Otto. Porém, nesta mesma data, em Cleveland, EUA, a Bosch apresentava seu sistema no encontro da SAE. Segundo Nystrom (1958 apud MILHOR, 2002), outros sistemas foram apresentados no mesmo encontro, como o sistema Borg-Warner, semelhante ao da Bosch.

O sistema apresentado pela GM mostrou em testes realizados nos dinamômetros que, em condições de regime, o sistema era pouco superior que o carburador convencional. Entretanto, no regime de aceleração, o sistema de injeção da GM tinha um rendimento bem superior comparado ao do carburador convencional, com uma resposta mais rápida e sem falhas. Winkler e Sutton (1957 apud MILHOR, 2002) apresentaram na mesma conferência o sistema da Bendix de injeção sequencial, o Electrojector, proporcionando melhores resultados para torque, potência e, o mais importante, menor consumo de combustível quando comparado com um mesmo motor equipado com carburador convencional.

Em Cleveland, Nystrom (1958 apud MILHOR, 2002), apresentou um sistema de injeção semelhante ao da GM; porém, não era um sistema de injeção de fluxo de combustível constante como o da GM, pois tratava-se de um sistema

temporizado. A quantidade de combustível que a bomba injetora fornecia para as válvulas injetoras era determinada por um controlador de acordo com a condição de operação do motor. Os testes de dinamômetro, com motor V-8, mostraram que um motor com este tipo de injeção apresentava uma curva de torque melhor e, além disso, o consumo de combustível também foi menor, o qual foi atribuído ao corte de combustível durante as desacelerações.

O uso de sistemas de injeção eletrônicos visa fornecer uma mistura ar-combustível mais eficiente para o motor, diminuindo-se o consumo de combustível e eliminando problemas de partida a frio e aquecimento. Segundo Amey (1995), o desenvolvimento da tecnologia da injeção eletrônica de combustível e gerenciamento de motores foi liderado pelos europeus na década de 1960 e pelos japoneses em meados da década de 1970. Após uma estagnação no desenvolvimento, motivada pela rejeição dos usuários, a partir do fim da década de 1970, os americanos passaram a acreditar que os carros europeus e japoneses eram mais avançados tecnologicamente e de qualidade superior do que a maioria dos carros americanos e começaram a requisitá-los, retomando-se assim o desenvolvimento tecnológico.

Na Figura 2.1 (AMEY, 1995) são evidenciados os avanços no desenvolvimento da tecnologia de injeção eletrônica citados anteriormente, podendo-se notar a interrupção da adoção pelos EUA; porém, a Europa manteve sua adoção e desenvolvimento.

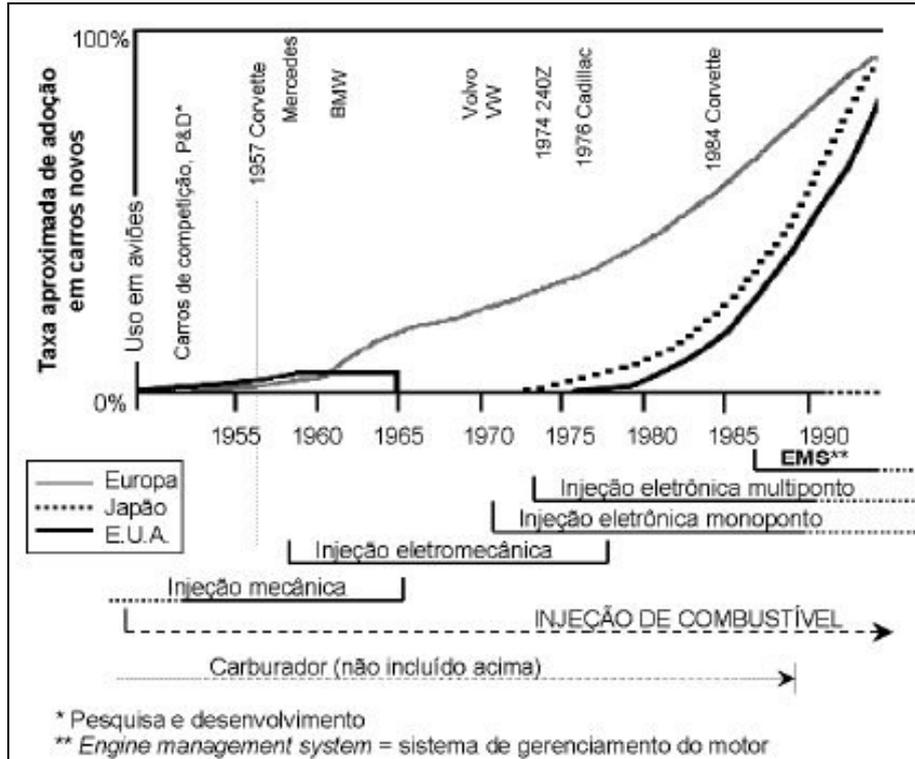


Figura 2.1: Utilização industrial da injeção eletrônica de combustível (AMEY, 1995).

O uso do controle eletrônico tornou-se interessante, segundo Ribbens (2003), devido a dois requisitos governamentais vindos do governo dos EUA: a redução dos níveis de emissões de gases poluentes e a redução dos níveis de consumo de combustível.

De acordo com Ribbens (2003), além do nível de emissões de gases poluentes, a legislação passou a regulamentar o consumo de combustível por meio do CAFE (*Corporate Average Fuel Economy* - Economia Média Unificada de Combustível), onde se estipulou a eficiência do motor para a produção de todos os veículos.

Controles mecânicos não teriam a capacidade de reproduzir funções de forma precisa para a larga gama de veículos em produção durante toda a vida do

veículo, considerando todas as condições de operação, e permanecer dentro das tolerâncias para cumprir as metas governamentais (PASSARINI, 1993). Devido a estas exigências, a indústria automobilística americana retomou o desenvolvimento do sistema de injeção de combustível, conforme a Figura 2.1.

De acordo com Glöckler e Manger (1981 apud MILHOR, 2002), a Bosch lança o sistema com controle eletrônico analógico JETRONIC em 1967, sendo comandado pela pressão no coletor de admissão. Outras versões surgiram baseadas no sistema desenvolvido pela Bosch, dando então início a uma nova era na injeção de combustível.

Em 1973 a Cadillac começou um programa de implementação de sistemas de injeção eletrônica em seus veículos (GIVENS, 1976), passando a ser instalado como equipamento de série nos modelos Seville, tratando-se de um sistema desenvolvido pela Bosch e Bendix, em que a maior inovação estava no envio do sinal para ativar as válvulas injetoras, onde era enviado diretamente pela UCE (Unidade de Controle Eletrônico). Esta UCE, que era um computador analógico pré-programado, recebia os sinais dos sensores e gerava um pulso com duração necessária para ativar os bicos injetores, controlando-se também a recirculação dos gases do escapamento e outras operações. Também em 1973, a Bosch introduziu o K-JETRONIC, sistema com controle mecânico-hidráulico e com medição do volume de ar. Após o K-JETRONIC, foi apresentado pelo mesmo fabricante, o sistema L-JETRONIC com controle eletrônico e injeção de combustível intermitente.

Com o desenvolvimento da eletrônica, o uso de circuitos integrados passou a substituir os circuitos analógicos e os sistemas de injeção de combustível tiveram um grande avanço, tornando-se possível a incorporação de outros sistemas.

Segundo Givens (1976), um sistema de ignição microprocessado, chamado MISAR, desenvolvido pela Delco Remy (ex-subsidiária da GMB e posteriormente da Delphi Automotive Systems) passou a ser usado em veículos como item de série. O sistema controlava o momento exato da ignição e a duração do tempo de energização da bobina.

Em 1979 foi introduzido pela Bosch o sistema MOTRONIC, o qual fazia o controle das funções do motor por meio de processamento digital. Este sistema foi desenvolvido a partir da união do sistema L-JETRONIC com uma ignição eletrônica mapeada. Começava então uma nova era de controle, pois com a integração dos sistemas, o mesmo agora era responsável pelo gerenciamento do motor como um todo, tendo-se então o início dos chamados *EMS (Engine Management Systems – Sistema de Gerenciamento de Motores)*, os quais uniam os subsistemas de ignição, injeção, recirculação de gases, entre outras (BOSCH, 1999).

De acordo com Bartolomias Jr. (1989 apud SILVA et al., 2007), no Brasil, em 1988, a Volkswagen (VW) passou a utilizar em seus veículos o controle eletrônico de motores, equipando um de seus modelos esportivos com um

sistema analógico desenvolvido pela Bosch, chamado LE-JETRONIC, e tendo um módulo separado para o controle da ignição.

A GM passou a utilizar o sistema de injeção eletrônica no ano seguinte, iniciando-se com um sistema de injeção central (apenas uma válvula de injeção) e, mais tarde, a injeção multiponto passou a ser utilizada nos modelos esportivos e nos modelos de luxo. A Autolatina (união entre Ford e VW) usava o sistema de injeção eletrônica nos modelos mais sofisticados (Gol GTi e Santana Executivo), e continuava empregando o carburador com auxílio eletrônico, chamado de carburador eletrônico, nos modelos mais simples. A FIAT passou a empregar um sistema de injeção combinado com um distribuidor diferenciado do convencional. A partir de 1991 todas as montadoras nacionais já estavam utilizando a injeção eletrônica.

Com o crescimento da produção brasileira de veículos movidos à gasolina no início da década de 90, o uso de sistemas de gerenciamento de motores de combustão interna teve um aumento e, em 1997, como resultado da legislação de emissões e melhor relação custo/benefício, todos os carros produzidos no país já dispunham de injeção eletrônica.

### 2.3 - Sistemas de Controle dos Motores

O sistema de injeção eletrônica é a principal parte do sistema de controle dos motores, pois o mesmo é responsável por controlar a quantidade de combustível ideal para cada condição de operação do motor. Os sistemas de controle de ignição, recirculação de gases e outros sistemas completam o sistema

de controle do motor, que atuando de maneira integrada, melhoram o desempenho do motor, minimizando o consumo de combustível, as emissões e maximizando a dirigibilidade, o desempenho e a vida útil do motor.

A UCE é a unidade física responsável pelo gerenciamento do motor, pois a mesma recebe os sinais dos diversos sensores instalados no motor e no veículo, processa esses dados de acordo com uma programação específica e aciona os atuadores de recirculação dos gases de exaustão, de controle de combustível, entre outros.

O sistema de controle dos motores de combustão interna utiliza um mapeamento pré-definido, sendo obtido experimentalmente, onde são levantadas curvas de potência, consumo específico, torque do motor, níveis de emissões desejados, entre outras. Com estes dados, são então compilados gráficos tridimensionais de carga *versus* rotação *versus* ponto de ignição, carga *versus* rotação *versus* tempo de injeção, entre outras, inclusive bidimensionais, como temperatura do motor *versus* tempo de injeção, temperatura do motor *versus* ponto de ignição, etc.

Estas curvas são armazenadas em forma de tabelas na UCE. Em funcionamento, o *EMS* recuperará as informações contidas na UCE, ponto a ponto, conforme a condição de operação do motor. A Figura 2.2 (BOSCH, 2005) ilustra um mapa de avanço da ignição em função da carga e rotação do motor.

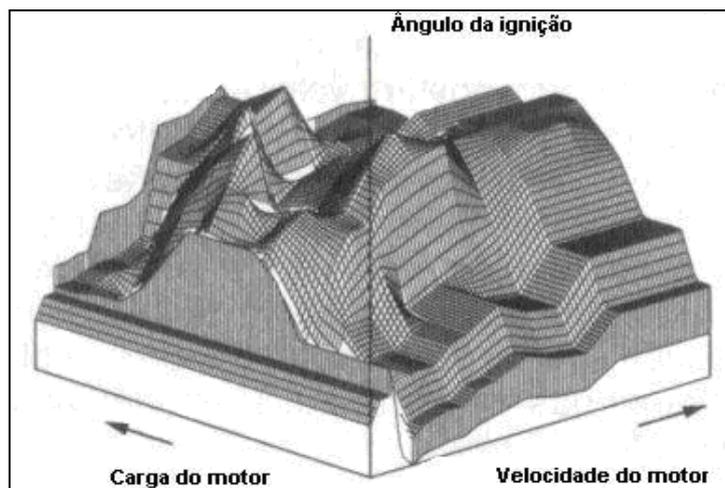


Figura 2.2: Mapa do avanço da ignição (BOSCH, 2005).

Para cada condição de operação do motor define-se um modo de controle, onde cada um deles corresponde a uma determinada rotina realizada pelo *EMS* e que é ativado a partir dos sinais recebidos pelos sensores.

Para fins de ilustração, é apresentado na Figura 2.3 um exemplo das interações entre os sensores, atuadores e a *ECU* (DELPHI, 2002), em que as linhas azuis indicam as conexões dos diversos sensores que alimentam a *ECU* com dados, as linhas vermelhas indicam as conexões da *ECU* com os diversos atuadores que devem ser controlados e, dentro da *ECU*, são indicadas funções disponíveis no *EMS*.

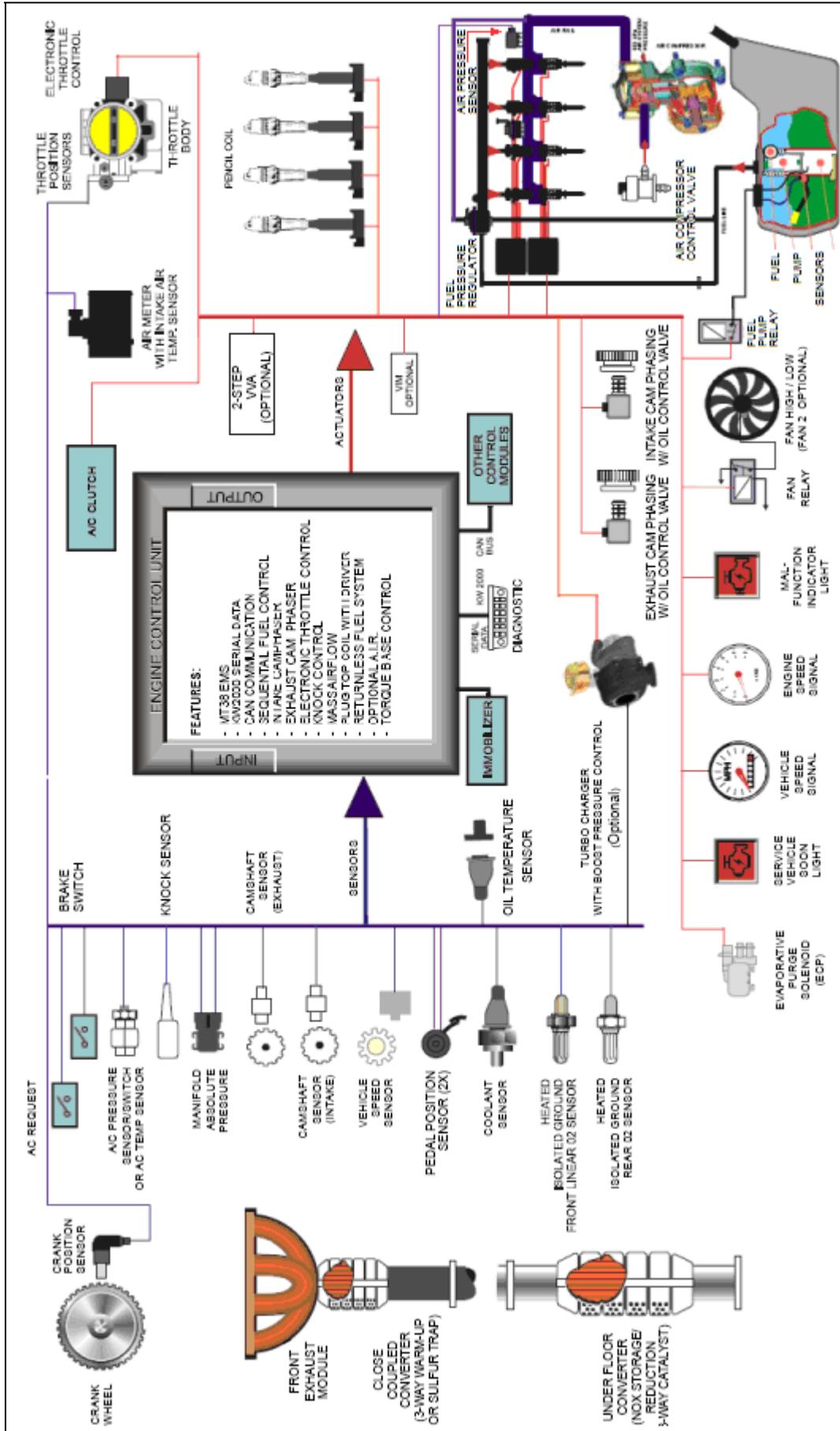


Figura 2.3: Exemplo de entradas e saídas de um EMS (DELPHI, 2002).

## Capítulo 3 - Aspectos de Sensores de Oxigênio

### 3.1 - Introdução

Conversores catalíticos de três vias requerem gases de exaustão com uma composição muito controlada para seu funcionamento ideal. Determinar esta composição ideal requerida é possível por meio do uso de sensores de oxigênio, que é um dispositivo desenvolvido para monitorar o oxigênio residual da exaustão de um motor de combustão interna. O sensor de oxigênio é o principal elemento para que o controle em malha fechada ar/gasolina (*A/F – Air/Fuel*) do motor resulte em uma mistura ótima para o conversor, tornando então este componente essencial para o *EMS*.

### 3.2 - Tipos de Sensores Para Estimação de Oxigênio

De acordo com a DELPHI (2005), sensores de oxigênio podem ser divididos em duas grandes categorias:

- Banda larga e
- Estequiométricos.

O sensor de banda larga fornece uma leitura contendo uma ampla escala da taxa *A/F* (com valores entre 0,7 e 5,0 mA), tendendo-se a ter maiores benefícios nos sistemas de controle que usam injeção direta ou em veículos com emissões super baixas. Este sensor é basicamente a combinação de um sensor

convencional (estequiométrico), que atua como uma célula galvânica (sensor Nemst), com uma outra célula chamada de célula “bomba”, cuja função é bombear oxigênio para dentro ou fora do sensor, estabelecendo-se uma corrente proporcional à diferença na concentração de oxigênio em ambas as células, conforme pode ser observado na Figura 3.1 (BOSCH, 2005).

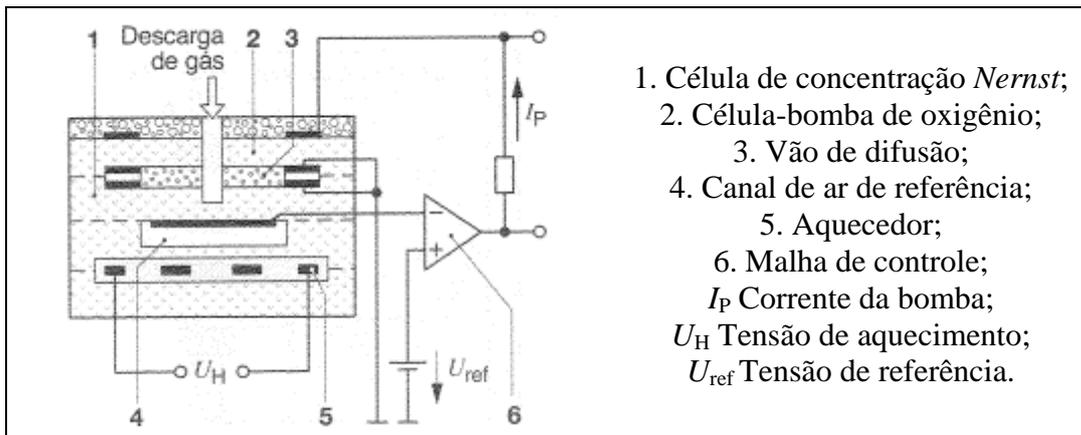


Figura 3.1: Sensor de Oxigênio de Banda Larga (BOSCH, 2005).

Havendo bombeamento de oxigênio para dentro do sensor, significa que a mistura está pobre de ar; se o bombeamento for para fora, a tensão é revertida, significando que a mistura está rica em ar. Na Figura 3.2 pode-se observar o sinal de saída de um sensor deste tipo (DELPHI, 2007).

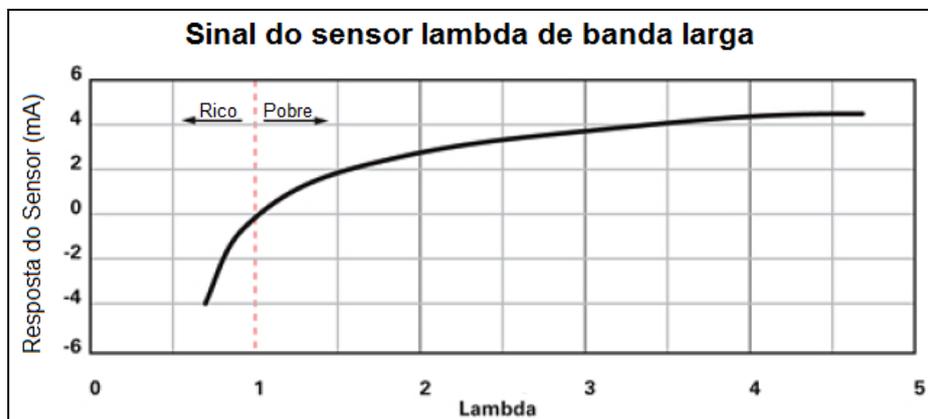


Figura 3.2: Curva de resposta do sensor de oxigênio de banda larga (DELPHI, 2007).

O sensor estequiométrico fornece uma ampla variação no sinal de saída na taxa estequiométrica  $A/F$ . Estes sensores podem se basear nos princípios resistivos ou galvânicos.

Devido à maior aplicação do sensor de oxigênio estequiométrico galvânico, e por ser este o sensor aplicado neste trabalho, a seção seguinte tratará do sensor resistivo de forma resumida e do sensor galvânico de maneira mais detalhada.

Os sensores galvânicos produzem uma tensão de saída em forma de degrau (Figura 3.4). Esta tensão está relacionada ao diferencial entre a falta de oxigênio na parte do elemento exposto ao gás de exaustão e a parte do elemento exposto ao ar de referência (Figura 3.3). O ar atmosférico é usado como referência e é considerado que contenha aproximadamente 21% de oxigênio. O material mais utilizado como elemento base para tais sensores é o óxido de zircônia, estabilizado com outros óxidos como a Ytria ou Calcia.

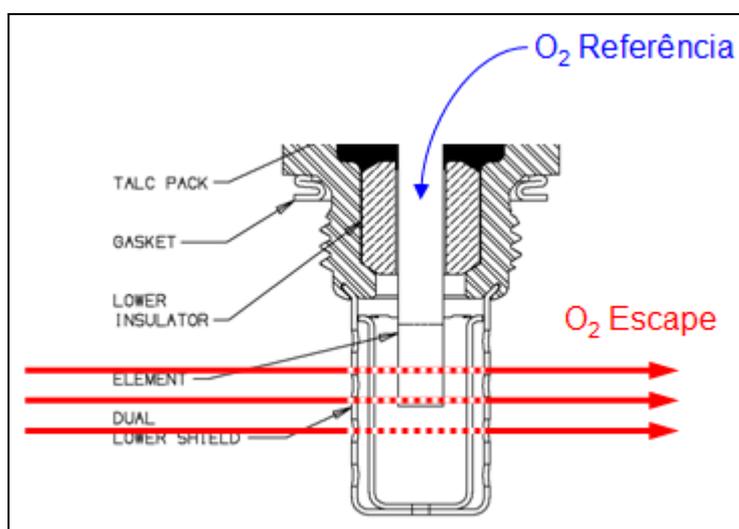


Figura 3.3: Ilustração do funcionamento do sensor de oxigênio.

Estes sensores produzem uma alteração significativa no sinal de saída quando a relação  $A/F$  se alterna entre rico para pobre e pobre para rico, sendo que também que precisam de uma temperatura mínima de operação de aproximadamente  $350^{\circ}\text{C}$  (DELPHI, 2005).

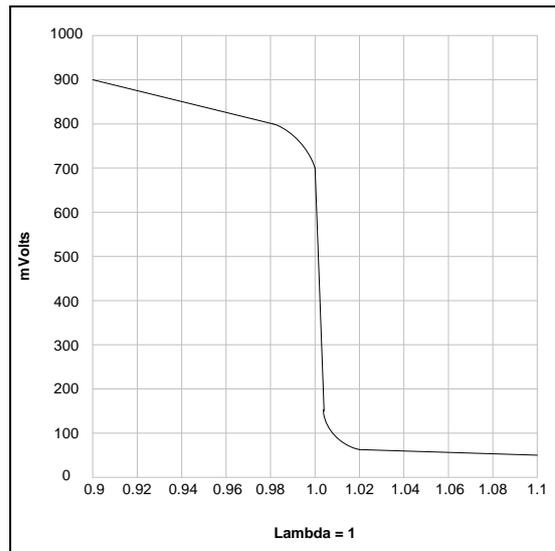


Figura 3.4: Curva de resposta do sensor de oxigênio (DELPHI, 2005).

Na maioria das aplicações, o calor do gás de exaustão é suficiente para eliminar a necessidade de um aquecimento adicional. As características das saídas não mudam significativamente com a temperatura.

Já os sensores de oxigênio resistivos, não utilizam o ar de referência como nos sensores galvânicos. São compostos de materiais óxidos metálicos semicondutores, como o dióxido de titânio ( $\text{TiO}_2$ ) ou titanato de estrôncio ( $\text{SrTiO}_3$ ), que indicam variações na resistência quando expostos a diferentes concentrações de oxigênio. Ao contrário dos sensores galvânicos, que geram tensão internamente, os resistivos necessitam de alimentação externa para permitir a medição e, como são dispositivos resistivos, tornam-se sensíveis à

temperatura, necessitando-se também de um aquecedor para estabilizar a temperatura (DELPHI, 1995).

Na Figura 3.5 é apresentado cortes em um sensor de oxigênio planar (DELPHI, 2005).

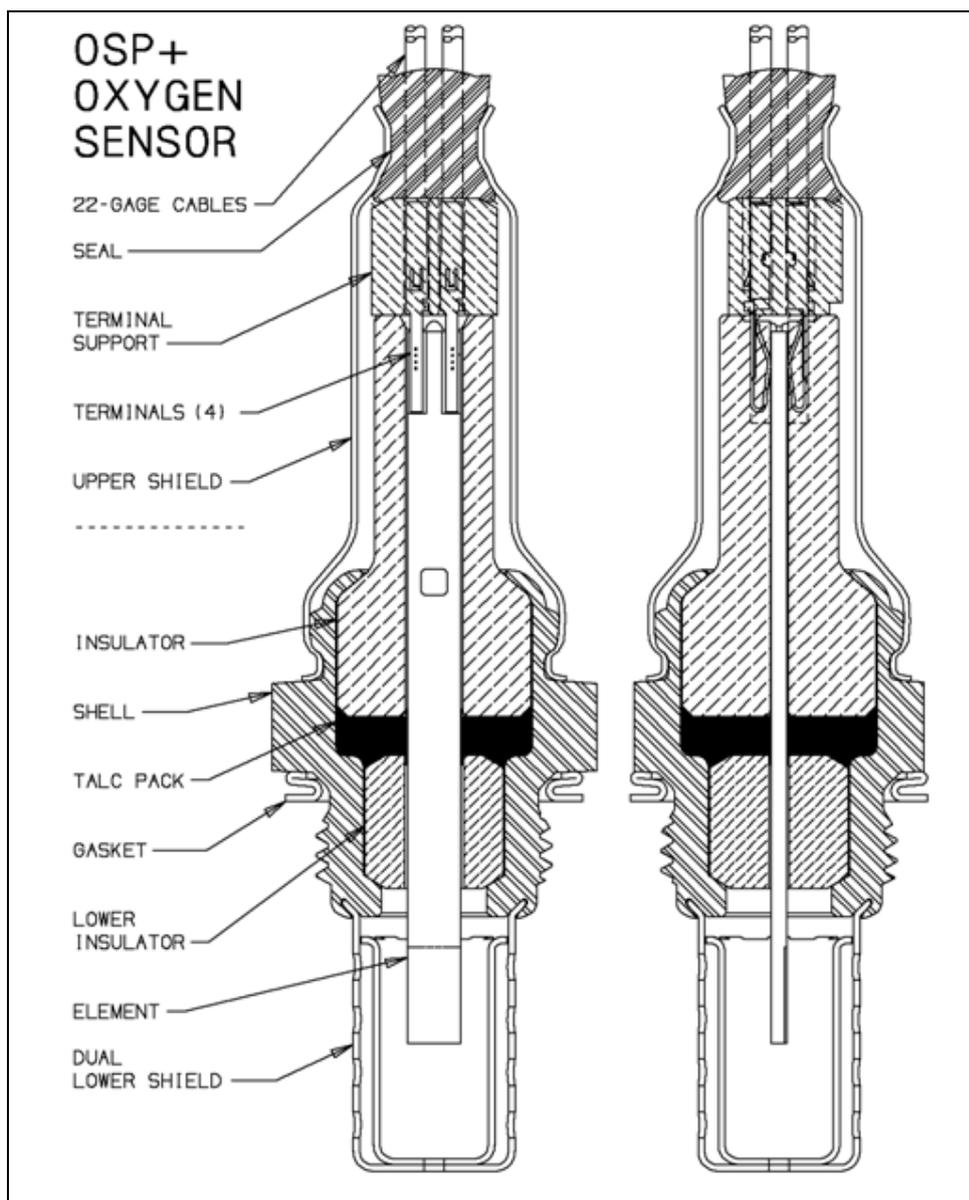


Figura 3.5: Sensor de oxigênio planar (DELPHI, 2005).

### 3.3 - Fundamentos de Estequiometria

A seguir serão apresentados alguns fundamentos básicos de combustão do motor de ciclo Otto, sendo que maiores detalhes podem ser vistos em Bosch (1999 e 2005) e em Zhao e Ladammatos (2001).

A combustão interna é um processo complexo que envolve a interação de muitos subsistemas do motor sobre grandes condições de mudanças.

#### 3.3.1 - Efeitos da Taxa Ar/Combustível na Combustão

Um dos principais objetivos do *EMS*, como mencionado no Capítulo 2, é fornecer as massas apropriadas de ar e combustível para atingir a taxa correta de mistura no momento da combustão. O ar e o combustível devem estar nas proporções ideais, e a esta proporção ideal se dá o nome de estequiometria Ar/Combustível, taxa ou relação Ar/Combustível. Tais proporções dependem da química do combustível (ZHAO; LADAMMATOS, 2001).

A taxa estequiométrica Ar/Combustível, comumente chamada de taxa *A/F* (*Air/Fuel* - Ar/Combustível), se refere à quantidade de ar e combustível que permitirá um processo químico de combustão equilibrado, assumindo-se que, quando as taxas *A/F* são fixadas como estequiométricas, referem-se então aos combustíveis não oxigenados. Combustíveis oxigenados, como o etanol, por exemplo, tem taxas estequiométricas mais baixas que a gasolina padrão, significando-se que para ocorrer uma combustão completa, mais combustível é necessário para a mesma massa de ar. Estas variações devem ser consideradas no desenvolvimento de *EMS* para veículos bicombustíveis.

### 3.3.2 - Misturas de Combustíveis

Para gasolina padrão, não oxigenada, os valores típicos para a taxa estequiométrica estão entre 14,2 e 14,8:1. Valores estequiométricos para vários combustíveis são apresentados na Tabela 3.1.

Tabela 3.1: Estequiometria de diversos combustíveis (DELPHI, 2005).

<b>Tipo de Combustível</b>	<b>Taxa estequiométrica <i>A/F</i></b>	<b>Lambda</b>
Gasolina típica sem chumbo	14,5 (faixa 14,2 – 14,8)	1
Mistura Etanol 10%	13,9	1
Mistura Etanol 24%	13,3	1
Mistura Etanol 85%	10,0	1
Etanol 100%	9,0	1
Mistura MTBE 15%	14,1	1
Propano	15,6	1
GNV / Metano	17,2	1

Como a taxa  $A/F$  dependerá do combustível usado, conforme tabela anterior, um método mais apropriado para referenciar a taxa é necessário. Este método é a normalização do valor da taxa. Usando-se o valor normalizado, a taxa estequiométrica  $A/F$  assumirá o valor unitário, independente do combustível usado. Dois termos são então usados para este valor normalizado, ou sejam:

Valor Lambda ( $\lambda$ ):

$$\lambda = \frac{A/F \text{ (atual)}}{A/F \text{ (estequiométrico)}} \quad (3.1)$$

onde:

Se  $\lambda < 1$ , então a mistura é rica (deficiência de oxigênio);

Se  $\lambda = 1$ , então a mistura é ideal (estequiométrica);

Se  $\lambda > 1$ , então a mistura é pobre (excesso de oxigênio).

Razão equivalente ( $\phi$ ):

$$\phi = \frac{F / A \text{ (atual)}}{F / A \text{ (estequiométrico)}} \quad (3.2)$$

onde:

Se  $\phi < 1$  então a mistura é pobre (excesso de oxigênio);

Se  $\phi = 1$  então a mistura é ideal (estequiométrica);

Se  $\phi > 1$  então a mistura é rica (deficiência de oxigênio).

Este trabalho utilizará o valor lambda para se referir ao valor normalizado da taxa estequiométrica  $A/F$ . Um exemplo do efeito de lambda na emissão de gases do motor pode ser visto na Figura 3.6 (DELPHI, 2005).

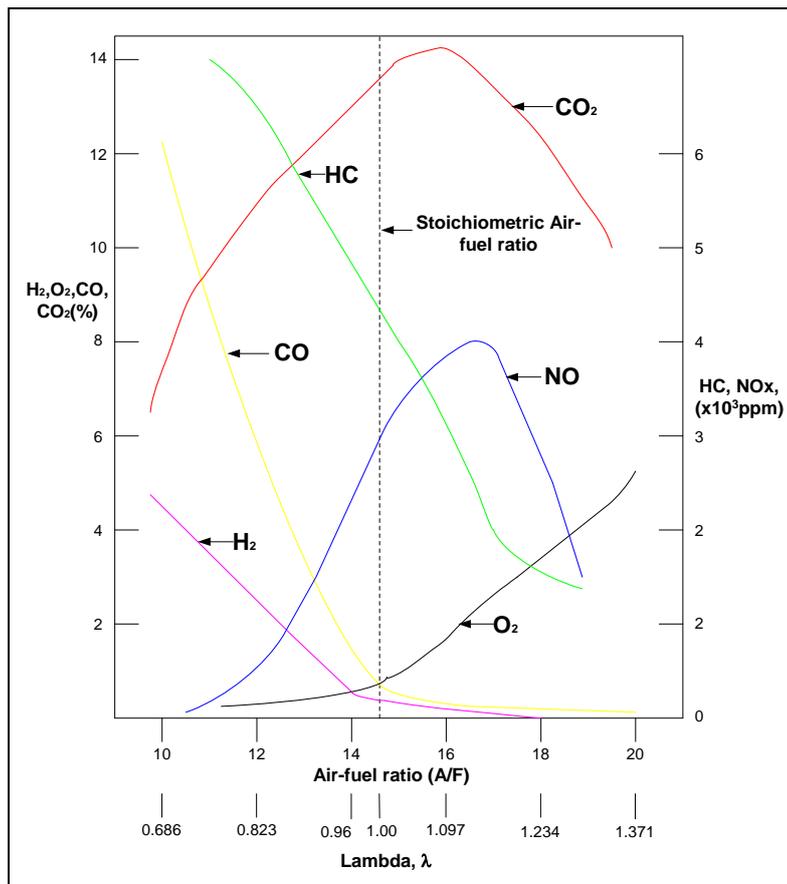


Figura 3.6: Emissões do motor com gasolina comum (DELPHI, 2005).

A válvula do corpo da borboleta (*throttle body*) controla o ar que entra no motor e o *EMS* usa estas informações para distribuir a massa correta de ar, podendo o *EMS* usar informações de outros sensores que também podem medir a massa de ar de entrada. Contudo que não haja vazamentos no motor ou no sistema de exaustão, não haverá adições ou subtrações de ar ou combustível, e o sensor de oxigênio detectará uma leitura atrasada da taxa *A/F* de entrada. O sensor de oxigênio reage às mudanças na taxa *A/F*, do lambda, com uma mudança acentuada na voltagem de saída.

### 3.3.3 - Localizações dos Sensores de Oxigênio e Catalisadores

A Figura 3.7 apresenta um esquema simplificado das possíveis localizações dos sensores de oxigênio. Os sensores localizados após os conversores catalíticos (catalisadores) têm, normalmente, a função de avaliar a eficácia destes conversores.

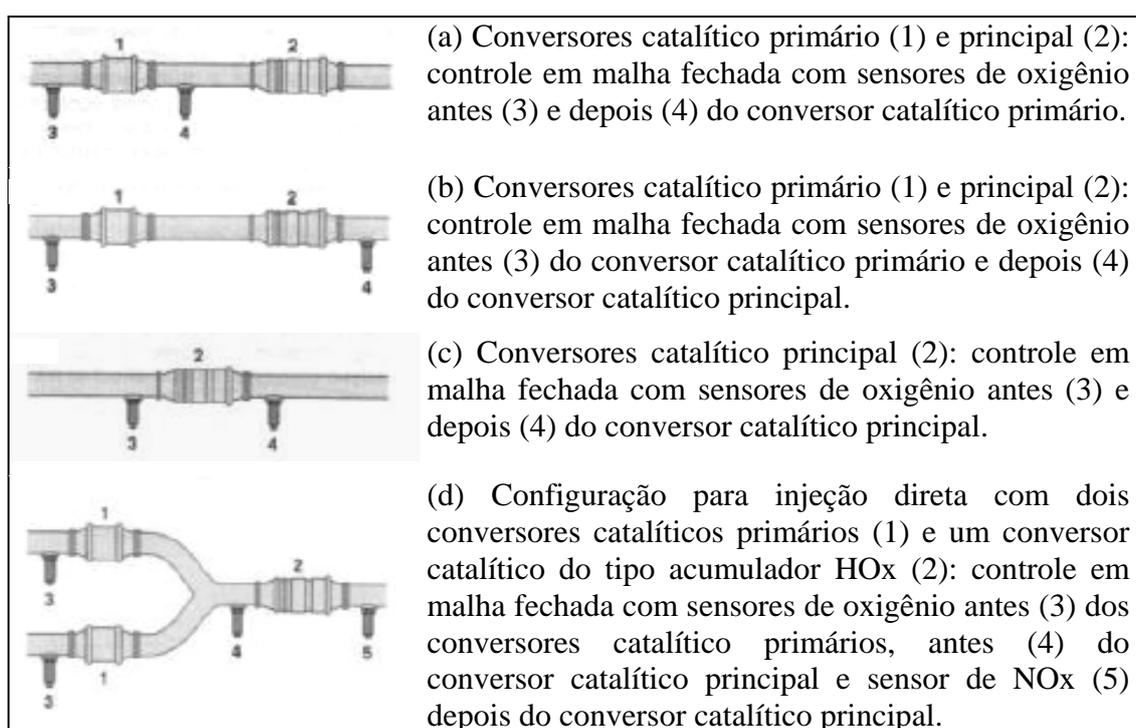


Figura 3.7: Típicas localizações do Sensor de Oxigênio (BOSCH, 2005).

Os catalisadores têm a função de converter os componentes poluentes HC (Hidróxido de Carbono), CO (monóxido de carbono) e NOx (óxidos de nitrogênio) em componentes não nocivos e, para obter a conversão mais eficiente possível para os três poluentes, a composição ideal da mistura a ser convertida deve ter lambda igual a 1 (BOSCH, 2005). É ilustrada na Figura 3.8, a conversão catalítica dos gases em função de lambda.

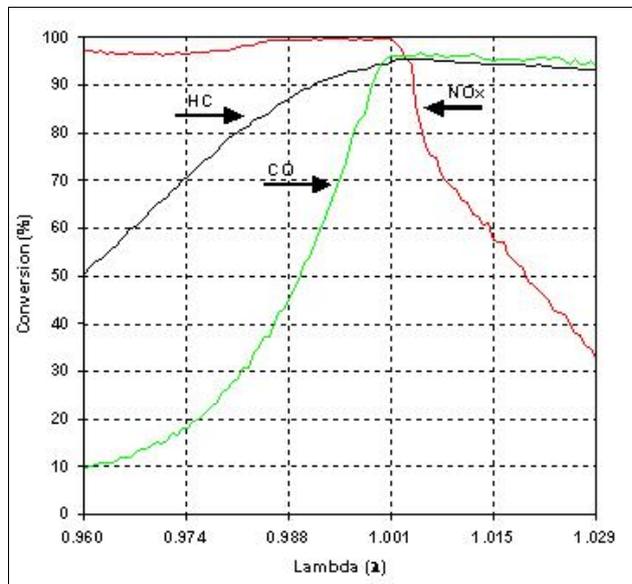


Figura 3.8: Eficiência da conversão do catalisador de três vias (DELPHI, 2005).

Conforme se pode observar na Figura 3.8, o desempenho de catalisadores modernos de três vias depende do controle da taxa  $A/F$  numa faixa muito pequena de lambda ( $\lambda$  próximo de 1).

## **Capítulo 4 - Aspectos de Redes Neurais Artificiais**

Redes neurais artificiais são modelos computacionais inspirados no cérebro e que possuem a capacidade de aquisição e manutenção de informações. O cérebro possui um sistema de processamento de informações altamente complexo, não-linear e paralelo. As redes neurais possuem um processamento paralelamente distribuído constituído de unidades de processamento simples, tendo a propensão natural de armazenar conhecimento experimental e posteriormente torná-lo disponíveis (HAYKIN, 1999).

O processo de aprendizagem e a utilização de pesos sinápticos é uma das semelhanças entre os neurônios do cérebro e as redes neurais artificiais. Tal processo é responsável pelo conhecimento adquirido pela rede, pois isso acontece porque as características necessárias do conjunto de dados apresentados à rede são extraídas. Os pesos sinápticos são usados como armazenamento do conhecimento adquirido.

As principais características da rede neural podem ser definidas como: capacidade de aprender por meio de exemplos, de se adaptar, de generalizar e de agrupar ou auto-organizar dados. A generalização está relacionada com a capacidade da rede aprender, com um conjunto reduzido de exemplos, e, posteriormente dar uma resposta coerente para dados não-conhecidos. A característica de agrupar ou auto-organizar está envolvida com a capacidade da

rede explorar semelhanças entre padrões, possibilitando a reunião dos mesmos em classes.

O primeiro modelo de rede neural artificial foi desenvolvido por McCulloch e Pitts, em 1943 (KOVÁCS, 1996). O modelo tinha uma única saída, e era uma função binária (*threshold*) da soma dos valores de entrada. Com esta característica binária, este modelo se comporta como um classificador de padrões que pode dividir duas classes linearmente separáveis.

Baseado na biologia (neurônios), por meio da regra de Hebb de 1949, Frank Rosenblatt pôde demonstrar, em 1958, o seu modelo conhecido como *perceptron*. Esse modelo possui três camadas, sendo que a primeira camada recebe os dados de entradas exteriores, possuindo conexões fixas; a segunda camada recebe impulsos da primeira camada por meio de conexões cuja eficiência de transmissão (peso) é ajustável, que por sua vez, envia valores para a terceira camada, onde será apresentada a resposta, sendo esta última a camada de saída (BRAGA et al., 2000).

Minsky e Pappert demonstraram em 1969 que o *perceptron* não era capaz de resolver algumas tarefas muito simples (HAYKIN, 1999), como o **ou-exclusivo**. Essa rede apenas resolvia problemas linearmente separáveis. Tal demonstração fez o número de pesquisadores que trabalhavam no desenvolvimento de redes neurais diminuir e, por alguns anos, o tema ficou adormecido; porém, alguns pesquisadores ainda continuaram a trabalhar na área.

John Hopfield publicou em 1982 um trabalho que reverteu a situação do baixo número de pesquisas, fazendo-se com que os grupos de pesquisa voltassem a pesquisar redes neurais. Ele mostrou a relação entre redes recorrentes auto-associativas e sistemas físicos. Depois disso, abriu-se caminho para outras pesquisas, como a do desenvolvimento do algoritmo *back-propagation* e das redes perceptron multicamadas.

#### 4.1 - Modelo de Neurônio Biológico e Artificial

O sistema nervoso humano é formado por células nervosas denominadas neurônios, que são divididas em três partes: o corpo celular, os dendritos e o axônio.

O centro dos processos metabólicos da célula nervosa é o corpo celular, medindo milésimos de milímetros e conhecido como soma, a partir do qual, projetam-se os dendritos e os axônios. Os dendritos são muitas vezes maiores do que o próprio corpo celular, e formam uma árvore de conexões chamada árvore dendrital. Eles são responsáveis pelo recebimento dos impulsos nervosos (informações) de outros neurônios e de conduzi-los das extremidades para o corpo celular. O corpo celular faz o processamento dos sinais recebidos pelos dendritos e libera novos impulsos, que serão transmitidos para a extremidade da célula por intermédio dos axônios. Sendo assim, os axônios podem ser definidos como dispositivos de saídas, os dendritos como de entradas e o corpo celular como processadores.

A célula controla a produção de impulsos elétricos por meio de um processo denominado função limiar. Em redes neurais artificiais é chamado de função de ativação, onde somente será produzida uma saída se a soma das entradas for maior ou igual a um determinado limiar.

As extremidades do axônio são conectadas com os dendritos de outros neurônios pelas sinapses, permitindo-se assim a propagação dos impulsos nervosos de uma célula para outra. Com a união desses nós é então formada a rede neural. A Figura 4.1 ilustra de forma simplificada um neurônio biológico (LNCC, 2008).

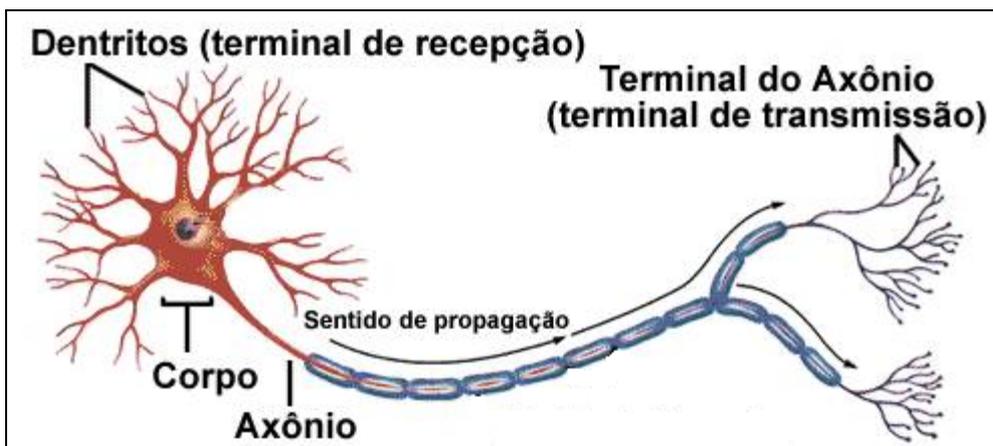


Figura 4.1: Neurônio biológico (LNCC, 2008).

Como se pode observar na Figura 4.3, o neurônio artificial criado por McCulloch e Pitts baseia-se em  $n$  terminais de entradas  $x_1, x_2, \dots, x_n$ , e seus pesos sinápticos  $w_{i1}, w_{i2}, \dots, w_{in}$ , cujos resultados podem ser positivos ou negativos. Tais resultados estão relacionados com os tipos de sinapses utilizadas pela rede. Se for utilizada uma sinapse inibitória, os neurônios terão dificuldades para realizar as trocas de informações. Já na sinapse excitatória, as passagens de informações entre os neurônios são permitidas.

Depois da multiplicação entre os terminais de entrada e os pesos ser efetuada, o corpo celular faz a soma ponderada dos valores recebidos e, assim como no neurônio biológico, o neurônio artificial decide se deve ou não disparar. A saída do neurônio poderá ser linear ou não-linear de acordo com a função de ativação utilizada.

Dentre as funções de ativação mais usadas estão as seguintes: degrau bipolar, linear, tangente hiperbólica, logística, degrau, rampa e base radial. Estas funções de ativação mais utilizadas são ilustradas na Figura 4.2.

Após a passagem pela função de ativação, o neurônio terá sua saída ativada quando:

$$\sum_{i=1}^n x_i w_i \geq \theta \quad (4.1)$$

onde:

$n \rightarrow$  Número de entrada.

$x_i \rightarrow$  Entrada da rede.

$w_i \rightarrow$  Peso associado à entrada  $x_i$ .

$\theta \rightarrow$  Limiar de excitação (*threshold*) do neurônio.

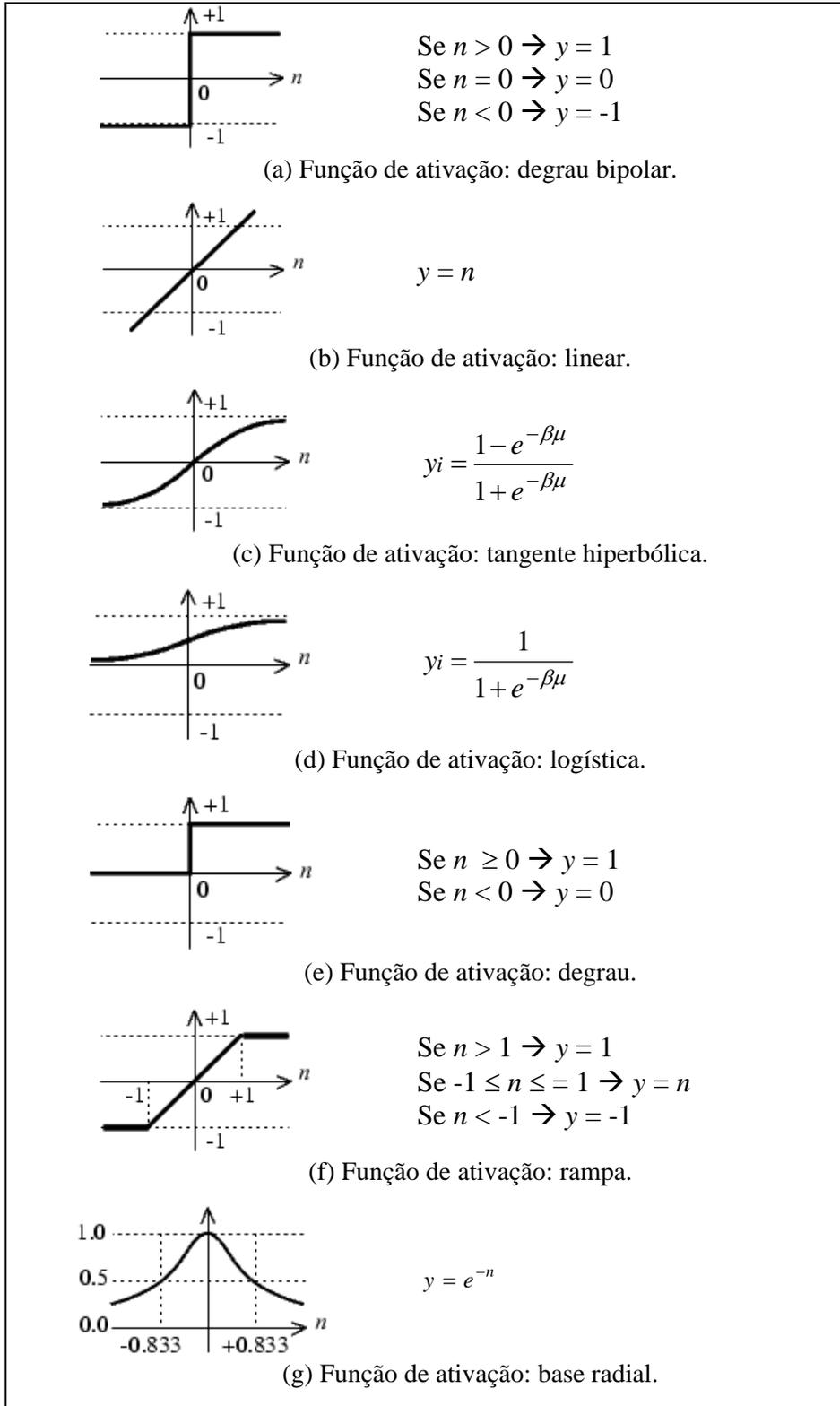


Figura 4.2: Principais funções de ativação.

É apresentado na Figura 4.3 o neurônio artificial idealizado por McCulloch e Pitts, constituído de  $n$  entradas ( $x_1, x_2, \dots, x_n$ ) e apenas uma saída  $y$

(BRAGA et al., 2000). A utilização de pesos sinápticos nos neurônios artificiais simula o funcionamento dos neurônios biológicos, fazendo-se com que as conexões dos mesmos se comportem como aquelas dos neurônios biológicos.

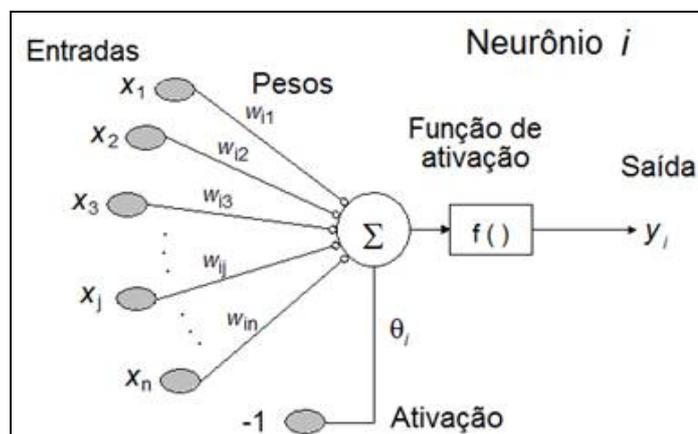


Figura 4.3: Representação de um neurônio artificial.

Na Figura 4.4 é apresentada de forma simplificada uma ilustração comparativa do neurônio biológico com o neurônio artificial, onde na parte superior são descritos como biológicos e na parte inferior, como artificiais.

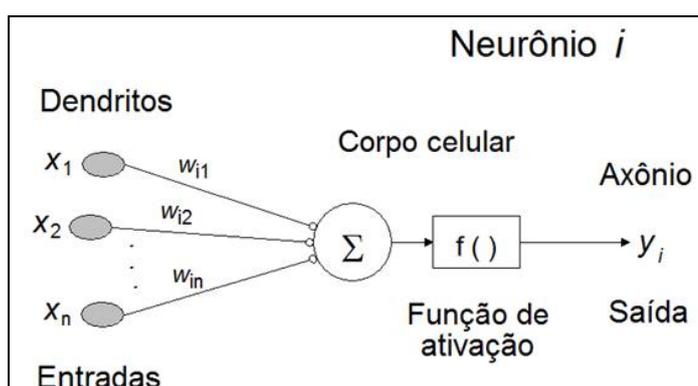


Figura 4.4: Comparação entre o neurônio biológico e o artificial.

No sistema biológico não existe um mecanismo que sincronize as ações dos neurônios, como no neurônio criado por McCulloch e Pitts, onde ocorrem

avaliações ao mesmo tempo, como também não há restrição quanto a ativação da saída em tempos discretos (BRAGA et al., 2000).

## 4.2 - Arquiteturas de Redes Neurais

A arquitetura da rede neural artificial está intimamente relacionada com o algoritmo utilizado para treinar a rede, incluindo parâmetros como os de número de camadas da rede, número de neurônios em cada camada, tipo de conexões entre os neurônios e topologia da rede.

Segundo Haykin (1999) existem três classes diferentes de arquiteturas. Essas classes determinam os tipos de problema que podem ser tratados pela rede neural.

### 4.2.1 - Redes de Camada Única (*Feedforward*)

A rede *feedforward* (alimentação para frente) de única camada é constituída de apenas uma camada de entrada e uma camada neural de saída.

Os neurônios da camada de entrada correspondem aos neurônios sensoriais ou lineares que recebem os sinais da rede. A camada de entrada não é uma camada neural, não ocorrendo processamento nas mesmas. Já na camada de saída é feito o processamento e apresentação dos resultados finais.

É ilustrado na Figura 4.5 uma rede *feedforward*, com  $n$  neurônios na camada de entrada e  $M$  neurônios na camada neural de saída.

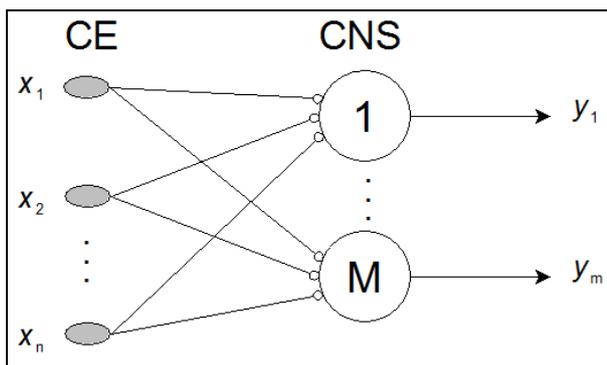


Figura 4.5: Rede *feedforward* (com camada única).

#### 4.2.2 - Redes de Múltiplas Camadas (*Feedforward*)

A utilização de uma camada intermediária escondida não-linear aumenta a capacidade de processamento da rede. A presença de uma ou mais camadas escondidas (ou intermediária) é que as distinguem da arquitetura anterior.

O sinal apresentado na camada de entrada é projetado junto com elementos do padrão de ativação na camada intermediária (segunda camada da rede e primeira camada neural), onde os dados são processados e o resultado é usado como atributo de entrada para a camada neural seguinte, e assim sucessivamente até à camada de saída da rede.

A Figura 4.6 ilustra uma rede *Feedforward* de múltiplas camadas, com  $n$  entradas,  $N1$  neurônios na primeira camada intermediária,  $N2$  neurônios na segunda camada intermediária e um na camada neural de saída.

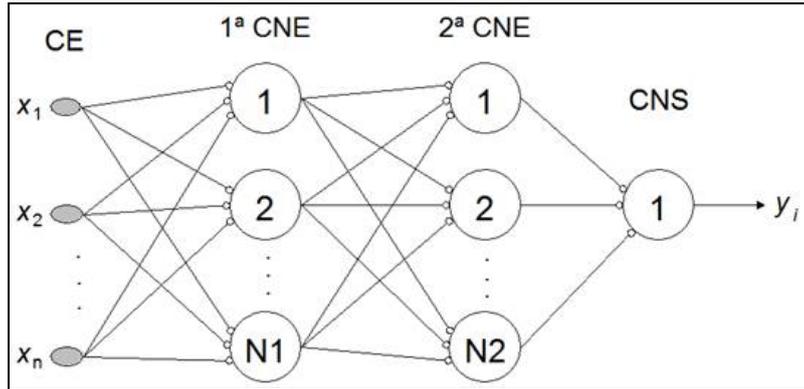


Figura 4.6: Rede *feedforward* com duas camadas intermediárias e uma camada de saída.

### 4.2.3 - Redes Recorrentes

As redes recorrentes distinguem-se das redes *feedforward* por ter no mínimo uma realimentação. Isso significa que os sinais de saída dos neurônios serão usados como entrada para os demais neurônios. A Figura 4.7 ilustra uma rede recorrente com três camadas neurais. A partir dos próprios neurônios saem os sinais de saída, que poderão ser utilizados como alimentação de entrada para todos os outros neurônios. Esta arquitetura pode lembrar entradas passadas e, conseqüentemente, processar seqüências de informações.

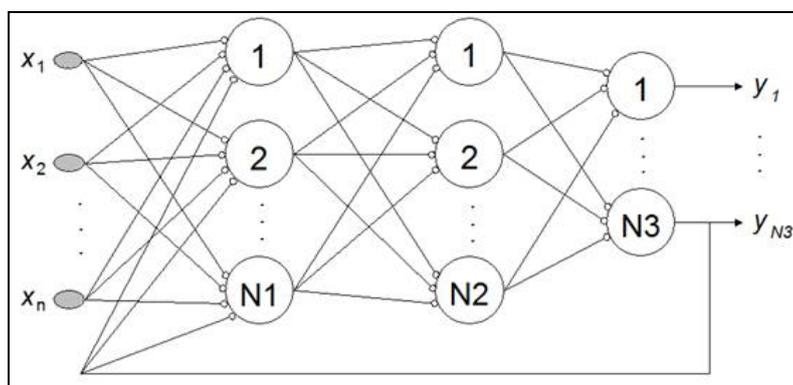


Figura 4.7: Rede recorrente com realimentação.

### 4.3 - Tipos de Treinamento

O treinamento ou processo de aprendizado da rede neural artificial consiste em ajustar os pesos sinápticos de forma que a aplicação de um conjunto de entradas produzirá um conjunto de saídas desejadas. Existem dois tipos básicos de treinamento (HAYKIN, 1999), ou sejam.

#### Treinamento Não Supervisionado:

Não há uma saída específica em relação aos estímulos de entrada. A rede se auto-organiza em relação a algum subconjunto de entrada que possua características similares. Como exemplo, têm-se as redes de Kohonen e *ART* (*Adaptive Resonance Theory*).

#### Treinamento Supervisionado:

Para cada padrão de entrada se tem o respectivo padrão de saída desejada. Como exemplo, as arquiteturas *Adaline*, *Perceptron*, *Perceptron* Multicamadas, *RBF* (*Radial Basis Function*) e Hopfield utilizam este tipo de treinamento.

O aprendizado do tipo supervisionado pode ser classificado como estático ou dinâmico. Os algoritmos estáticos não alteram a estrutura da rede, variando apenas os seus pesos. Os algoritmos dinâmicos alteram sua estrutura, aumentando ou diminuindo o tamanho da rede (BRAGA et al., 2000).

No tópico 4.4.1 será abordado com maiores detalhes o treinamento da rede *Perceptron* Multicamadas.

#### 4.4 - Rede Perceptron Multicamadas

As redes *perceptron* multicamadas (PMC, *Multilayer Perceptron – MLP*) são constituídas de uma camada de entrada, uma ou mais camadas neurais escondidas ou intermediárias e uma camada neural de saída. A camada intermediária tem a capacidade de detectar características dos padrões de entrada e utilizá-los para inferir os valores de saída da rede. A camada intermediária aumenta o poder computacional da rede. De acordo com o número de camadas intermediárias, a rede PMC tem a capacidade de resolver problemas de classificação de padrões não linearmente separáveis ou aproximar qualquer outro tipo de função.

O treinamento da PMC é efetuado por meio do algoritmo *backpropagation*, baseado na regra delta (Hebb) generalizada, e por ter um padrão de entrada para cada saída desejada, é denominado como sendo de treinamento supervisionado. Durante o treinamento o erro é propagado de trás para frente na rede, daí a origem do nome *backpropagation* (retro-propagação), sendo baseado na regra de aprendizado com correção do erro.

O aprendizado *backpropagation* constitui-se de dois passos principais: no primeiro passo (*forward*), um padrão de entrada é apresentado à rede e as informações são propagadas camada a camada até suas saídas, produzindo uma resposta atual da rede. Nesta fase, os pesos sinápticos da rede não se alteram, permanecendo fixos. No segundo passo (*backward*), a resposta atual produzida pela rede é subtraída da desejada, produzindo assim um sinal de erro. Esse sinal de erro é apresentado na camada de saída e será propagado de volta, camada a

camada até as suas entradas. Durante essa fase os pesos sinápticos da rede são ajustados, de acordo com a regra de correção de erro.

O sinal de erro na saída do neurônio é calculado pela Expressão (4.2).

$$E_m(k) = d_m(k) - y_m(k) \quad (4.2)$$

onde:

$E_m$  → Sinal de erro na saída do neurônio  $m$  em relação à  $k$ -ésima iteração.

$d_m$  → Resposta desejada na saída do neurônio  $m$ .

$y_m$  → Resposta produzida na saída do neurônio  $m$ .

O valor instantâneo é o somatório do erro quadrático sobre todos os neurônios na camada de saída, e pode ser calculado diretamente conforme a Expressão (4.3).

$$E(k) = \frac{1}{2} \sum_{m=1}^N (d_m(k) - y_m(k))^2 \quad (4.3)$$

onde:

$E(k)$  → Soma instantânea do erro quadrático na iteração  $k$ .

$N$  → Número de neurônios na camada neural de saída.

O valor do erro quadrático médio é obtido pelo somatório dos erros quadráticos, ou seja.

$$E_M = \frac{1}{P} \sum_{k=1}^P E(k) \quad (4.4)$$

onde:

$E_M$  → Erro quadrático médio.

$P$  → Número total de padrões.

O erro instantâneo e, conseqüentemente, o erro quadrático médio, é uma função de todos os parâmetros livres da rede, relativos aos pesos sinápticos e aos seus limiares. O objetivo do processo de treinamento é ajustar os parâmetros livres da rede e minimizar o erro quadrático médio (HAYKIN, 1999).

O número de vezes que a rede repete o seu treinamento, efetuando a correção de seus pesos, é denominado de época, sendo que o ajuste dos pesos é feito de acordo com o respectivo erro calculado para cada padrão apresentado à rede (HAYKIN, 1999).

A rede PMC possui funções de ativação não-lineares e diferenciáveis para que se possa ser calculado o gradiente e com isso fazer o ajuste dos pesos sinápticos. A função de ativação logística (4.5) é comumente utilizada em redes PMC executadas apenas em software, ou seja, que não são embutidas em hardware e é definida pela Expressão (4.5) (BRAGA et al., 2000).

$$y_i = \frac{1}{1 + e^{-\beta\mu}} \quad (4.5)$$

onde:

$y_i \rightarrow$  Saída do neurônio.

$\beta\mu \rightarrow$  Ângulo de inclinação onde há a inflexão ( $\beta$ ) com deslocamento horizontal de  $\mu$ .

Já a função tangente hiperbólica (4.6) é normalmente utilizada em hardware, pois, em se tratando de função ímpar, basta-se implementar metade desta função de ativação.

$$y_i = \frac{1 - e^{-\beta\mu}}{1 + e^{-\beta\mu}} \quad (4.6)$$

onde:

$y_i \rightarrow$  Saída do neurônio.

$\beta\mu \rightarrow$  Ângulo de inclinação onde há a inflexão ( $\beta$ ) com deslocamento horizontal de  $\mu$ .

É comum aumentar o número de camadas intermediárias para facilitar o treinamento da PMC, mas deve-se tomar cuidado quanto a essa alteração, pois o erro medido durante o treinamento se torna menos preciso conforme o mesmo vai sendo retro-propagado. Isso acontece porque toda vez que o erro é retro-propagado é também calculado a sua estimativa camada-a-camada.

Pode-se aumentar a capacidade de mapeamento não-linear da rede por incrementar o número de neurônios nas camadas intermediárias (KOVÁCS, 1996). Entretanto, se o número de neurônios for muito grande, o modelo pode se ajustar na presença de ruídos da amostra de treinamento, memorizando os padrões de treinamento ao invés de extrair as características. Esse efeito é conhecido como *overfitting*. Mas se o número de neurônios na camada intermediária for pequeno, a rede pode não ser capaz de realizar o mapeamento desejado, pois sobrecarrega os neurônios que estão sendo utilizados e, conseqüentemente, a rede não converge. Esse efeito é conhecido como *underfitting*.

#### 4.4.1 - Treinamento de redes PMC

O tipo de aprendizado utilizado pela PMC é chamado de treinamento supervisionado, pois, como citado anteriormente, para cada padrão de entrada se tem o respectivo padrão de saída desejada.

São apresentados à rede sucessivos padrões de entrada e seus respectivos padrões de saída. Durante este processo, a rede realiza a atualização dos pesos de suas conexões, de acordo com uma determinada regra de aprendizagem. Isso acontece até que o erro entre os padrões de saída gerados pela rede e os padrões desejados alcance um valor mínimo desejado.

O algoritmo utilizado para treinar redes do tipo PMC é o *backpropagation*, sendo o mesmo composto de duas fases. Essas fases são chamadas de *forward* e *backward*. Na primeira fase (*forward*), um padrão é apresentado às unidades da camada de entrada, o estímulo fornecido por essa camada é propagado camada-a-camada, até que a resposta seja produzida pela camada de saída. Como os valores de saídas desejados são conhecidos é então possível estimar os erros na camada de saída.

Na segunda fase (*backward*), a saída obtida pela rede é comparada à saída desejada dos padrões apresentados. Se as saídas forem diferentes, o erro é então calculado e a estimativa do erro é usada como referência para a correção dos pesos. A propagação acontece da camada de saída até a camada de entrada, e os pesos das camadas intermediárias vão sendo atualizados conforme o erro é retro-propagado, o que caracteriza o treinamento como *back-propagation* (HAYKIN,

1999). É ilustrado na Figura 4.8 o sentido do treinamento destas duas fases, *forward* e *backward*.

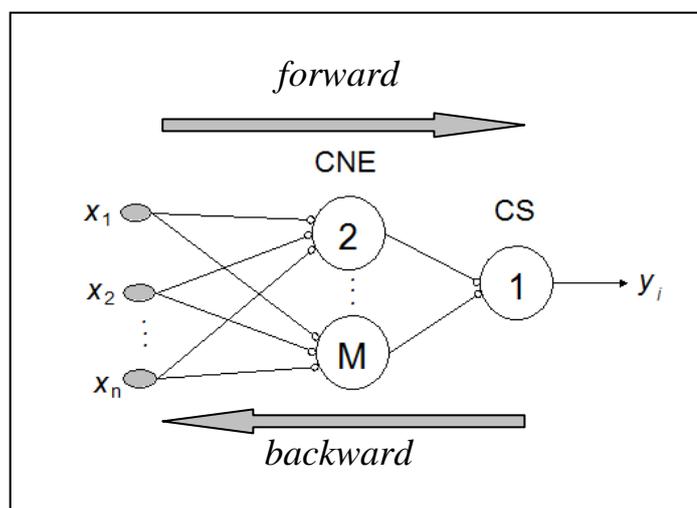


Figura 4.8: Fluxo do processamento do algoritmo *back-propagation*.

A seguir será apresentado o algoritmo de treinamento do *perceptron*, separando-se as fases *forward* e *backward*.

Passos que envolvem a fase *forward*:

- Apresenta-se a entrada à primeira camada da rede e propaga-se em direção às saídas.
- Calculam-se as entradas ponderadas dos neurônios da camada intermediária  $i$  que serão propagados para a camada intermediária  $i + 1$ .
- Calculam-se os sinais de saída da última camada intermediária e os envia à camada de saída.
- Comparam-se as saídas produzidas pelos neurônios da última camada com as saídas desejadas.

Caso a saída gerada esteja dentro da precisão desejada, ignora-se então a fase *backward* e apresenta-se a resposta; caso isso não ocorra, ativa-se a fase *backward* (a seguir).

Passos que envolvem a fase *backward*:

- Calcula-se o erro da rede na camada de saída.
  - Calcula-se o termo de correção dos pesos da camada de saída.
- Envia-se o erro da camada de saída para a camada intermediária  $i + 1$ .
  - Calcula-se o erro da camada intermediária  $i + 1$ .
  - Calcula-se o termo de correção dos pesos.
- Envia-se o erro da camada intermediária  $i + 1$  para a camada intermediária  $i$ .
  - Calcula-se o seu peso da camada intermediária  $i$ .
  - Calcula-se o termo de correção dos pesos.
- Atualiza-se o peso de cada unidade das camadas.

Para compreender o processo de aprendizado do algoritmo, supõe-se que cada combinação de pesos e limiares corresponda a um ponto na superfície de solução. Considerando que a altura de um ponto é diretamente proporcional ao erro associado a este ponto, a solução está nos pontos mais baixos da superfície. Para minimizar o erro obtido pela rede o algoritmo *backpropagation* ajusta os pesos e limiares, fazendo-se com que eles correspondam às coordenadas dos pontos mais baixos da superfície de erro (BRAGA et al., 2000).

Para que os pontos mais baixos da superfície de erro sejam atingidos é utilizado o método de gradiente descendente. Este método consegue encontrar a solução para superfícies mais complexas, pois o algoritmo pode ser levado a convergir para mínimos globais. Porém, muitas vezes são encontradas dificuldades no treinamento do *backpropagation*, isso acontece devido à lentidão do algoritmo na solução de problemas complexos e sua tendência a convergir para mínimos locais.

O *backpropagation* possui, basicamente, três parâmetros que determinam seu desempenho e sua capacidade de aprendizagem: taxa de aprendizagem, *momentum* e o número de épocas. A taxa de aprendizagem é um parâmetro muito importante no processo de aprendizagem, pois interfere no tempo de aprendizado e na convergência da solução para um mínimo local ou global. O termo *momentum* é uma constante que determina o efeito de mudanças passadas dos pesos na direção atual do treinamento, que tem por objetivo permitir o aumento na velocidade de convergência.

O número de épocas fornece a quantidade de vezes que o conjunto de treinamento será apresentado à rede a fim de que a mesma realize o ajuste de seus parâmetros livres.

O método de treinamento descrito acima é o método do gradiente descendente, porém, há diversos métodos disponíveis na literatura, sendo que dois deles, Levenberg-Marquardt e Levenberg-Marquardt com regularização Bayesiana, serão detalhados no Apêndice B.

#### **4.4.2 - Perceptron Multicamadas como aproximador universal**

Uma rede PMC treinada com o algoritmo *backpropagation* é capaz de implementar qualquer mapeamento não-linear que relacione as entradas com as saídas.

A rede PMC poderá ser utilizada como aproximador universal de funções, após seu treinamento, somente se os novos dados estiverem inseridos dentro do domínio de definição de cada variável.

$$x_i^{novo} = [x_i^{\min}, x_i^{\max}] \quad (4.7)$$

#### 4.4.3 - Redes com atraso de tempo

A rede neural com atraso de tempo (*Time-Delay Neural Network – TDNN*) é uma rede PMC cujas entradas são escalonadas ao longo do tempo.

Dado um conjunto de medidas observáveis  $x(1), x(2), \dots, x(n)$ , onde  $n$  é o comprimento total da série, uma *TDNN* pode ser utilizada para mapear o comportamento físico associado ao sistema conforme a ilustração seguinte.

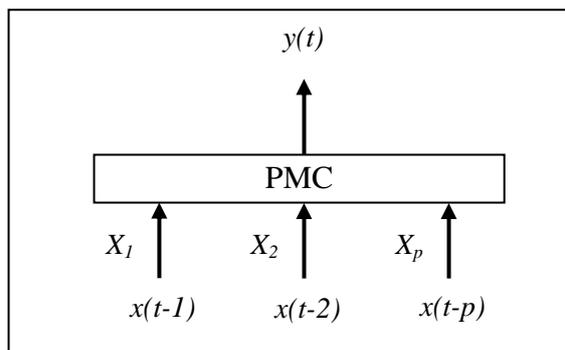


Figura 4.9: Modelo de *TDNN*.

Desta forma, a rede funciona como um preditor de um passo a frente de ordem  $p$ , isto é, prediz/estima a saída  $x(t)$  considerando  $p$  saídas anteriores, onde o valor atual  $x(t)$  atua como a saída desejada da função desconhecida  $f$ . Então, a *TDNN* é treinada para minimizar o valor quadrático do erro de predição dado por:

$$E(t) = d(t) - y(t), \quad p + 1 \leq t \leq n$$

Seleciona-se esta topologia caso o processo requeira uma ordem de predição pequena ou se o processo não requerer referências a valores prévios da série.

#### 4.4.3.1 - Perceptron Multicamadas recorrente

Nesta topologia a saída da rede é retro-alimentada às entradas. A rede é capaz de modelar qualquer sistema dinâmico que possa ser expresso por:

$$x(t) = f(x(t-1), x(t-2), \dots, x(t-p), y(t-1), y(t-2), \dots, y(t-m))$$

Ilustrando, tem-se:

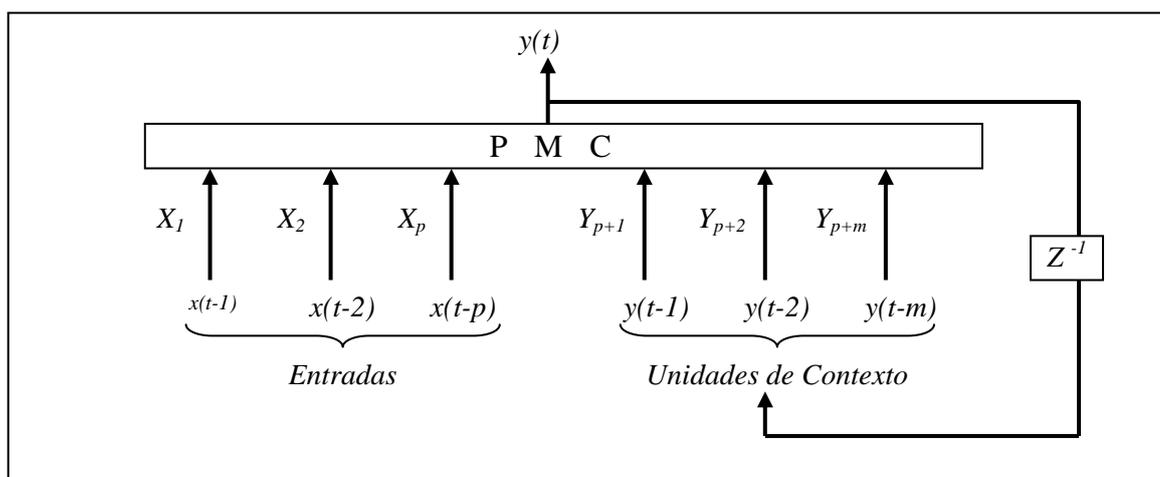


Figura 4.10: Modelo da PMC recorrente.

Esta topologia pode ser usada quando houver necessidade de se referenciar valores prévios da série ou se o sistema for considerado extremamente não-linear e se não tiver qualquer idéia de seu comportamento.

#### 4.5 - Aplicações de Redes Neurais em Sensoriamento Virtual

Diversos trabalhos abordam o tema de sensoriamento virtual na literatura, em que estas redes neurais fazem o *forecasting* (previsão) dos resultados (HOWLETT, 1998; ATKINSON et al., 1998). Alguns dos trabalhos relacionados apenas ao sensoriamento virtual de oxigênio, que é a proposta desta dissertação, são apresentados nos parágrafos seguintes.

Amato et al. (2005) pesquisaram a possibilidade de implementar um sensor de oxigênio virtual utilizando sistemas de inferência *Fuzzy* (*FIS – Fuzzy Inference System*) no modelo TSK (Takagi-Sugeno-Kang), emulando o chaveamento (mistura pobre/mistura rica) realizado pelo sensor. A abordagem proposta foi implementada em um motor mono-cilindro de 125cc; porém, houve erro de até 9% na classificação da mistura. Os autores relataram a necessidade de otimizar a distribuição dos dados utilizando técnicas de análises estatísticas a fim de desenvolver um classificador com mais classes para melhorar os resultados.

Kamat et al. (2006) estudaram a aplicação de redes neurais em quatro sensores veiculares, sendo eles o sensor de oxigênio, sensor de torque do motor, de fluxo da massa de ar (*MAF – Mass of Air Flow*) e o sensor de pressão absoluta do coletor de admissão (*MAP – Manifold Absolute Pressure*). Para tanto, utilizaram um motor 5.8L e redes neurais recorrentes com variações no algoritmo de treinamento recorrente em tempo-real (*RTRL – Real-Time Recurrent Learning*).

Os resultados foram considerados excelentes para os sensores MAF e MAP e razoavelmente bons para o sensor de torque; porém, os resultados para o sensor de oxigênio não foram conforme esperavam, obtendo resultados modestos para grandes universos de dados, entretanto, poderia ser implementado com o objetivo de diagnósticos.

## Capítulo 5 - Arquitetura de Sistema Inteligente Desenvolvida

### 5.1 - Introdução

Visando desenvolver um sensor de oxigênio virtual, o sistema inteligente implementado baseia-se nos conceitos de redes neurais artificiais apresentados anteriormente, visto que os dados disponíveis são numéricos e em grande quantidade, sendo assim é indicada a aplicação de RNA.

Como as variáveis são coletadas utilizando-se o *EMS*, as mesmas podem originar-se de sensores físicos, de equações, mapas de dados ou procedimentos contidos no *EMS*. Visto que a calibração de motores, e conseqüentemente, de *EMS*, devam ser realizadas para cada plataforma veicular, modelo ou até características do veículo, este estudo não visa fornecer topologias que possam ser generalizadas para outros veículos.

Estes dados são oriundos de leituras de diversas variáveis disponíveis (neste trabalho o número de variáveis estudadas é 42) nos *EMS*, coletadas de um veículo flex de 1.8 centímetros cúbicos e de 8 válvulas, em dinamômetros do Centro de Tecnologia da Delphi, sem o motor estar em regime especial de funcionamento e com teste final realizado pela área de calibragem de motores. Ou seja, os motores já estarão calibrados e não será fixada nenhuma variável,

tornando-se o teste no dinamômetro o mais próximo possível do que poderia ocorrer com os condutores.

## 5.2 - Aspectos Básicos do Sistema Inteligente

Considerando o problema como um sistema bem complexo e não-linear, onde deve-se relacionar as entradas com as saídas, a topologia escolhida para o desenvolvimento do sensor virtual de oxigênio foi a topologia Perceptron Multicamadas, utilizando-se uma ou duas camadas neurais escondidas (CNE). A Figura 5.1 representa o modelo de rede com camada neural escondida única, onde  $x_i$  representa as variáveis de entrada,  $M$  o número de neurônios na camada neural escondida e  $y$  representa a saída desejada, ou seja, valor normalizado de lambda.

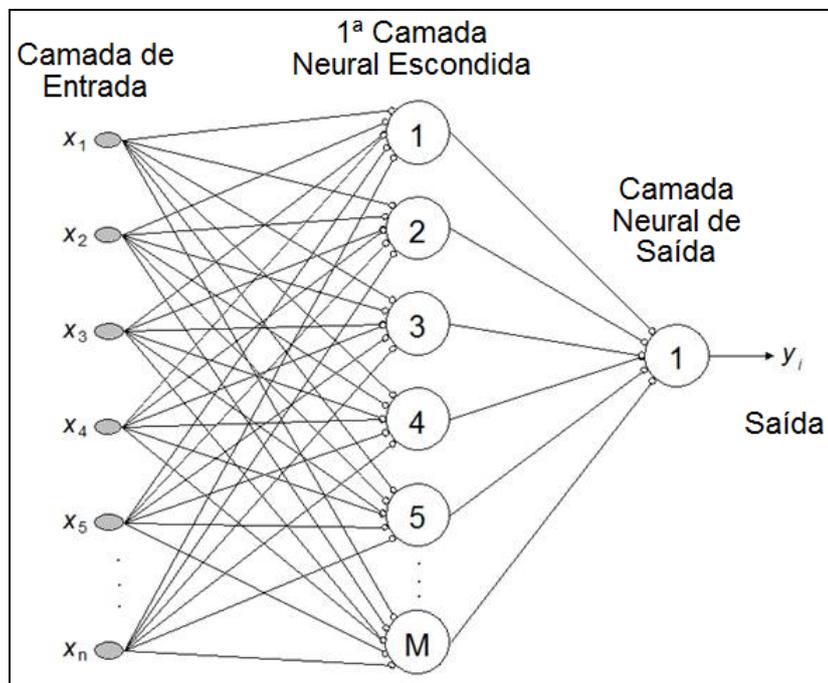


Figura 5.1: Modelo de PMC com uma CNE usado.

A Figura 5.2 representa o modelo com duas camadas neurais escondidas, onde  $x_i$  representa as variáveis de entrada,  $M$  o número de neurônios na primeira camada neural escondida,  $N$  o número de neurônios na segunda camada neural escondida e  $y$  representa a saída desejada, ou seja, valor normalizado de lambda.

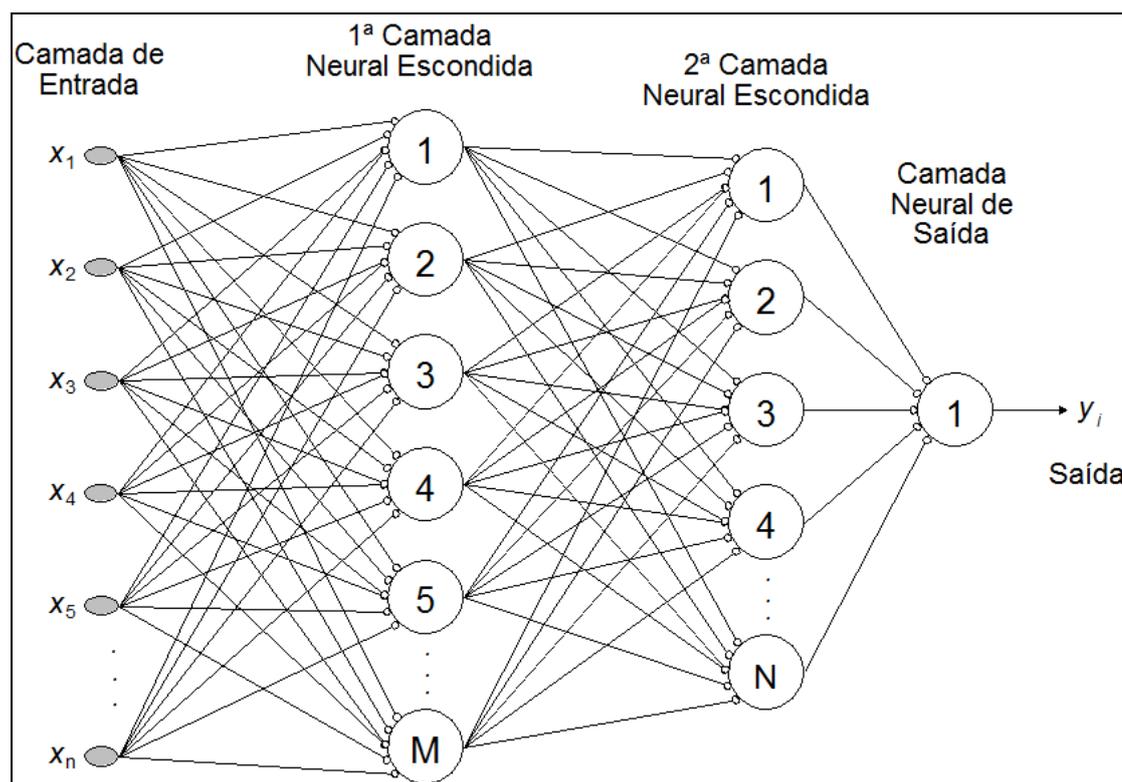


Figura 5.2: Modelo de PMC com duas CNE usado.

A metodologia utilizada para testar o sistema desenvolvido foi a técnica de Validação Cruzada - *Cross Validation* (HAYKIN, 1999), utilizando-se para tanto, 75% dos dados disponíveis para efetuar os treinamentos e 25% para verificar se a rede está generalizando satisfatoriamente. Todas as topologias candidatas foram treinadas com os mesmos conjuntos de dados, assim como todos os testes usaram os mesmos dados.

Como dito anteriormente, efetuaram-se coletas de dados em testes finais de calibração. As coletas foram realizadas com os carros abastecidos com E20, ou seja, 24% de etanol e 76% de gasolina e com E100 (apenas etanol), obtendo-se aproximadamente 900 segundos de dados, que corresponde a aproximadamente 90.000 registros.

A fim de simular o atraso na realimentação do sistema (*EMS*), onde o valor de lambda atual é consequência de todas as outras variáveis, o tempo de lambda em relação às outras variáveis foi deslocado para  $t+1$ .

Nas Figuras 5.3 e 5.4 são apresentados, como exemplo, gráficos de algumas variáveis coletadas e depois usadas nos treinamentos (todas com valores normalizados), sendo que suas legendas em português são descritas na Tabela 5.1.

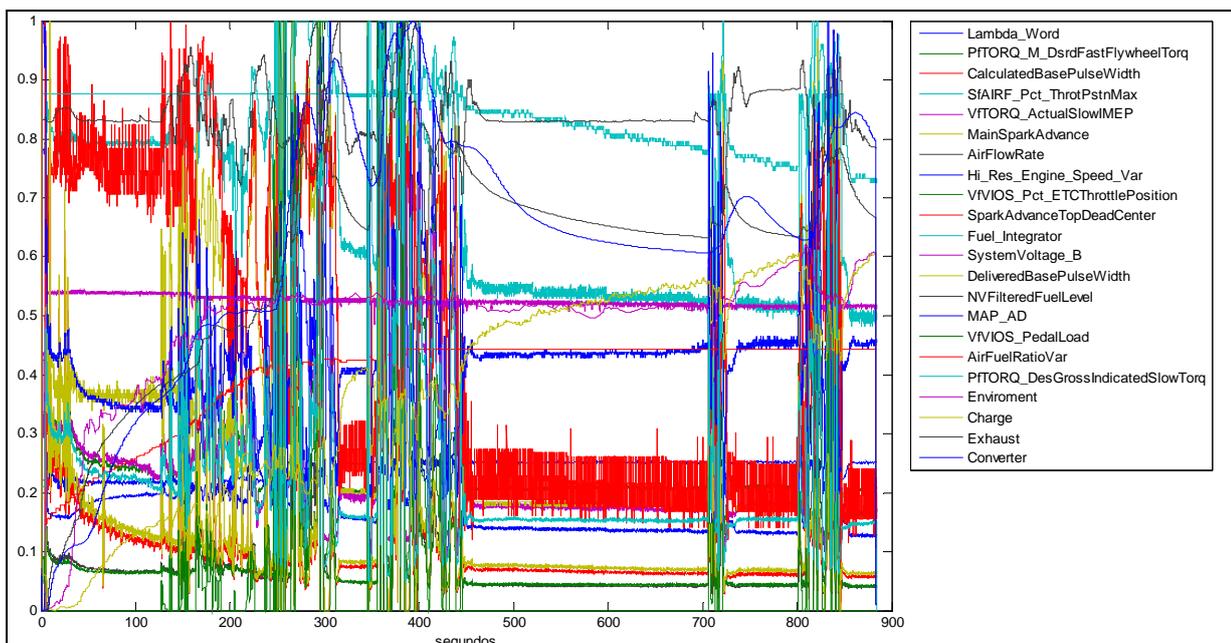


Figura 5.3: Exemplo de variáveis normalizadas usadas no sistema (I).

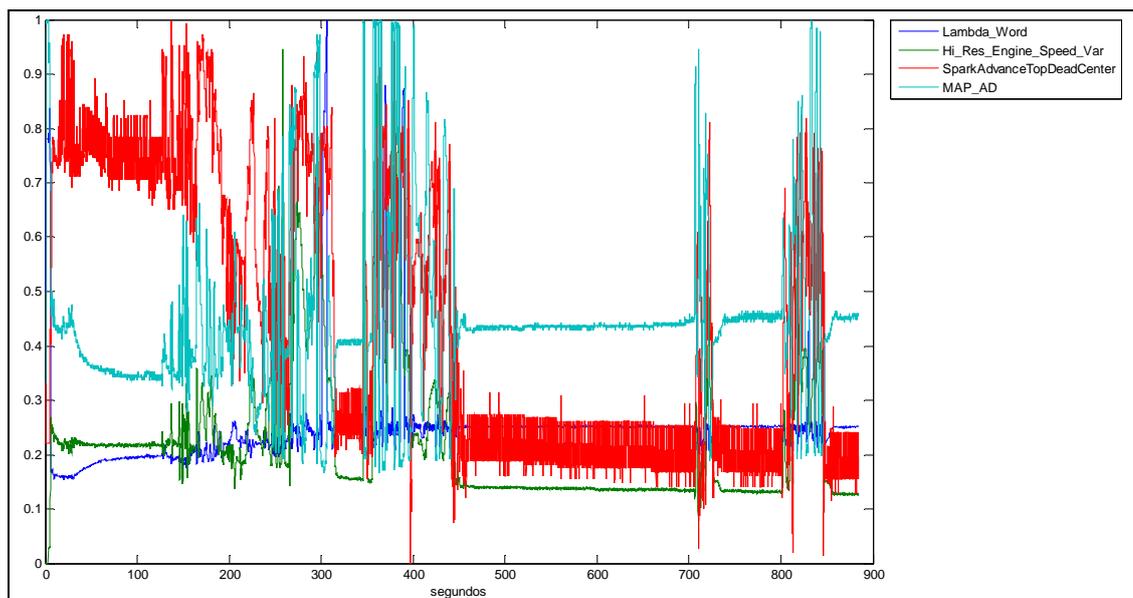


Figura 5.4: Exemplo de variáveis normalizadas usadas no sistema (II).

Em destaque, é apresentado na Figura 5.5 apenas a variável contendo o valor de lambda, sendo este o valor desejado para a saída da rede neural desenvolvida.

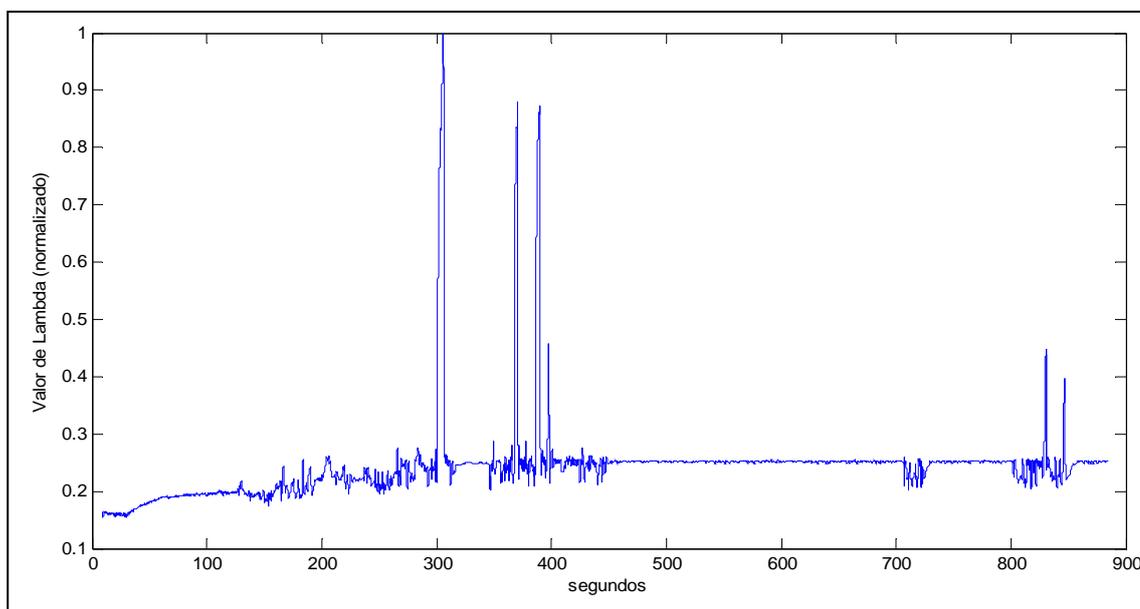


Figura 5.5: Exemplo da variável lambda (saída desejada).

No gráfico da Figura 5.6 é possível observar (a partir de 450s) o comportamento típico do mapa de dados do sensor de oxigênio de um veículo calibrado e em regime estável, onde ocorre apenas a comutação dos valores baseado na constante tentativa do *EMS* em equilibrar a taxa *A/F* na combustão.

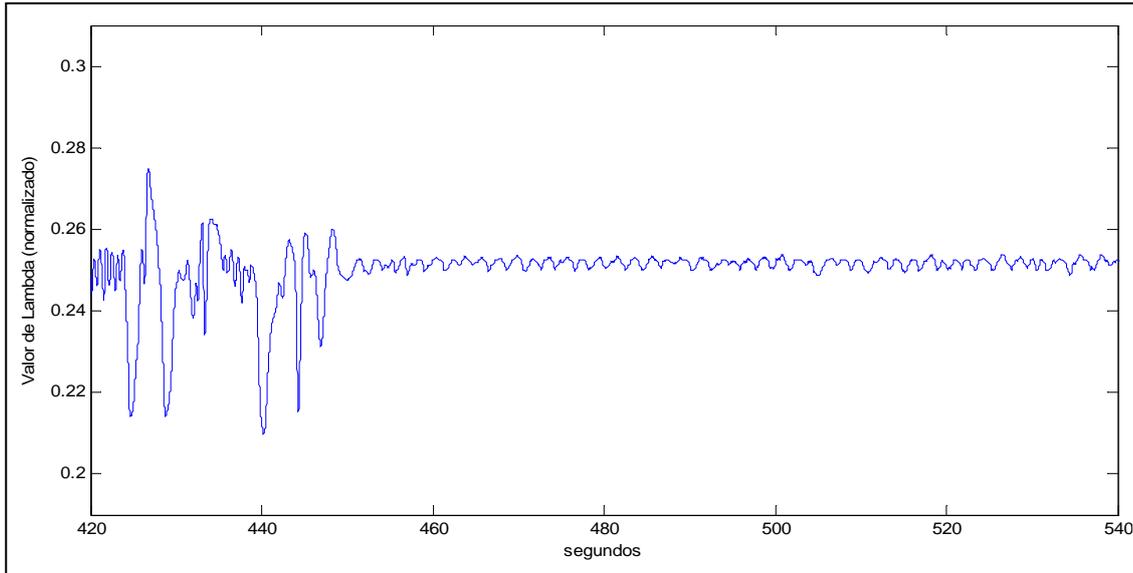


Figura 5.6: Detalhe da saída desejada.

Antes dos 450s nota-se que alguma ação foi executada no motor, onde o lambda oscilou bruscamente.

### 5.3 - Detalhes da Arquitetura do Sistema Inteligente Desenvolvido

Como os mapas de dados são extensos, mais de 85 mil linhas, houve a necessidade de discretiza-los para efetuar os treinamentos e testes. Duas abordagens foram utilizadas, a primeira discretiza todo o universo de dados disponível considerando um percentual de dados desejado. A segunda discretização é realizada em conjuntos reduzidos do universo de dados,

determinados pelo usuário, e utilizando-se também de um percentual de registros em relação a este conjunto.

As variáveis utilizadas como entradas das RNA nos treinamentos e testes foram divididas em quatro conjuntos. O primeiro conjunto utilizou as 21 variáveis menos lineares (N<sup>os</sup>: 2 6 7 8 9 10 12 13 14 15 26 27 28 30 31 32 34 37 40 41 42), observando-se os gráficos (Figura 5.2); o segundo conjunto baseou-se no conhecimento especialista de engenheiros da Delphi Automotive quanto à calibração dos veículos, sendo indicado 6 variáveis (N<sup>os</sup>: 6 12 13 14 30 32); o terceiro conjunto utilizou variáveis indicadas por Kamat et al. (2006) (N<sup>os</sup>: 6 10 12 13 30), sendo que o quarto conjunto utilizou todas as variáveis disponíveis (41 variáveis).

As 42 variáveis usadas no trabalho, sendo a primeira delas a saída desejada, são apresentadas na Tabela 5.1, assim como seus valores mínimos e máximos, e suas Unidades de Medidas (UM). A primeira variável representa a saída desejada e as demais são as entradas disponíveis.

Tabela 5.1: Variáveis utilizadas nos treinamentos

Nro	Descrição Original (Inglês)	Descrição	Min	Máx	UM
1	Lambda_Word	Sensor de oxigênio	0,6	4,0	$\lambda$
2	PfTORQ_M_DsrdFast FlywheelTorq	Desempenho do torque no volante do motor	-31,5	168,5	Nm
3	PerfmTORQ_EthWarm UpRawTorqLimDecay	Desempenho do torque na variação de temperatura do combustível	0,0	106,9	Nm
4	PrimePulseBPWMultiplier	Multiplicador da largura de pulso	0,0	3,5	escalar
5	PrimePulseElapsedTime_NV	Atraso do pulso	0,0	1165,0	ms
6	CalculatedBasePulseWidth	Largura de pulso da injeção calculada	0,0	71120,0	ms
7	SfAIRF_Pct_ThrotPstnMax	Máxima posição do <i>throttle body</i> - fluxo de ar	0,0	99,6	%
8	VfTORQ_ActualSlowIMEP	Indicador de pressão média efetiva	0,0	1335,2	Kpa
9	MainSparkAdvance	Avanço da fagulha principal	0,0	47,8	graus
10	AirFlowRate	Fluxo da massa ar	0,0	72,5	g/s
11	PfEOT_EstimatedOil Temperature	Temperatura estimada do óleo	-10,0	66,7	°C
12	Hi_Res_Engine_Speed_Var	Velocidade do motor	0,0	6229,0	rpm
13	VfVIOS_Pct_ETCThrottle Position	Posição do <i>throttle body</i>	0,0	100,0	%
14	SparkAdvanceTopDead Center	Avanço da fagulha no ponto morto	-2,5	52,5	graus
15	Fuel_Integrator	Integrador de combustível	0,0	146,0	escalar
16	NfEBOF_AmountOfEthInOil	Quantidade de etanol no óleo	0,0	2000,0	g
17	EstEnabled	Ativação do estimador de combustível	0,0	1,0	Lig/Des
18	PerfmTORQ_Limit MultiplierWasApplied	Aplicação do multiplicador de limite	0,0	1,0	Lig/Des
19	VsEBOF_EthBoilOffFlags. b_EthBoilOffActive	Queima de etanol ativado	0,0	1,0	Lig/Des
20	VsEBOF_EthBoilOffFlags. b_EthBoilOffCondMet	Condição de queima de etanol atingida	0,0	1,0	Lig/Des
21	PerfmTORQ_StopToExecute	Torque inicial para saída	0,0	1,0	Lig/Des
22	Engine_Status.RunAirFuel	Estado do motor – ar/combustível	0,0	1,0	Lig/Des
23	Engine_Status.RunFuel	Estado do motor – combustível	0,0	1,0	Lig/Des
24	NumberOfCylAtBoundary	Número de ciclos no limite	0,0	3,0	escalar
25	ColdStartPrimePulseLength	Largura de pulso para partida à frio	0,0	0,05	s
26	SystemVoltage_B	Voltagem do sistema	0,0	15,0	V
27	DeliveredBasePulseWidth	Largura de pulso distribuído	0,0	71120,0	ms
28	NVFilteredFuelLevel	Nível de combustível filtrado	0,0	98,6	L
29	Coolant_Temperature	Temperatura do líquido refrigerante	-9,0	83,5	°C
30	MAP_AD	Pressão absoluta do coletor de admissão	0,0	93,5	Kpa
31	VfVIOS_PedalLoad	Carga do pedal	0,0	99,9	%
32	AirFuelRatioVar	Taxa de A/F	0,0	15,0	g/g (A/F)
33	CAMSynchA	Sincronização do comando de válvula	0,0	1,0	Lig/Des
34	PfTORQ_DesGrossIndicated SlowTorq	Indicador de Torque baixo	0,0	165,9	Nm
35	Manifold_Air_Temperature_W	Temperatura do ar no coletor de admissão	-10,0	46,0	°C
36	Oil	Temperatura do óleo	-9,0	75,8	°C
37	Enviroment	Temperatura ambiente	-4,6	42,6	°C
38	Fuel rail	Temperatura da galeria de combustível	-8,9	47,9	°C
39	Intake	Temperatura do ar de admissão	-11,5	51,5	°C
40	Charge	Temperatura do acumulador elétrico	-12,1	53,1	°C
41	Exhaust	Temperatura de exaustão	-8,5	692,7	°C
42	Converter	Temperatura do conversor catalítico	-8,9	853,1	°C

O sistema desenvolvido permite criar um laço de treinamento e testes, utilizando-se de variáveis que permitem escolher as características da rede. As configurações permitidas são: definição de quais mapas de dados deve-se utilizar, tipo de discretização, percentual para discretização, escolha das variáveis de entrada, meta da rede (gradiente), quantidade de neurônios nas camadas neurais escondidas, limite de épocas, funções de ativação e métodos de treinamento (cujo detalhamento pode ser observado no Apêndice B). Uma descrição de todos os valores usados como características são apresentadas na Tabela 5.2, onde Mapa de Dados “1” denota o mapa E20, Mapa de Dados “2” significa E100 e Mapa de Dados “1+2” indica a utilização de ambos.

Tabela 5.2: Características utilizadas nas topologias desenvolvidas.

<b>Características</b>	<b>Valores Possíveis</b>
Mapa de Dados	[1; 2; 1+2]
Tipo de Discretização	Segmento; Completo
Fator de Discret.	[0,05; 0,03; 0,025; 0,02; 0,01; 0,005]
Objetivo	[0,0001; 1e-05; 1e-06]
CNE1	[5 7 9 10 12 13 15 20 25 30 35 40 45 50 55 60]
CNE2	[0 2 3 4 5 7 10 15 20]
Variáveis de Entrada	[6 10 12 13 30]
	[6 12 13 14 30 32]
	[2 6 7 8 9 10 12 13 14 15 26 27 28 30 31 32 34 37 40 41 42]
	Todas

A análise dos resultados levou em consideração, primeiramente, o erro relativo médio ( $\bar{e}_r$ ) dos testes realizados e, depois, visando evitar os efeitos *overfitting* e *underfitting*, observaram-se os resultados gráficos dos testes, permitindo validar os melhores modelos. Ao todo foram geradas 4.848 topologias de redes neurais.



## Capítulo 6 - Resultados Experimentais

### 6.1 - Introdução

A seguir serão apresentados diversos resultados das topologias testadas. Conforme pode ser observado na Figura 6.1, que representa o desenvolvimento dos testes, primeiramente, realizaram-se treinamentos e testes usando os mesmos tipos de combustível; após topologias treinadas com um mapa de dados (tipo de combustível) foram testadas com outro mapa; em seguida, treinamentos com discretizações de segmentos foram testadas com mapas completos; topologias treinadas com ambos os mapas foram testadas com um dos mapas e, finalmente, foram testadas e analisadas as melhores topologias aplicáveis para ambos os tipos de combustível. Ao todo, um total de 15.096 testes foi realizado.

Pode-se observar na Figura 6.1, os diversos agrupamentos “treino-teste” realizados, destacando-se, na cor cinza, os treinamentos ou testes tidos como bicombustíveis, onde “Trein.” significa que as informações seguintes referem-se aos treinamentos, já “Teste” denota testes; “Segm.” indica que os dados usados foram discretizações usando-se segmentos do conjunto universo, ao passo que o termo “Comp.” significa que na discretização foi utilizado todo o mapa.

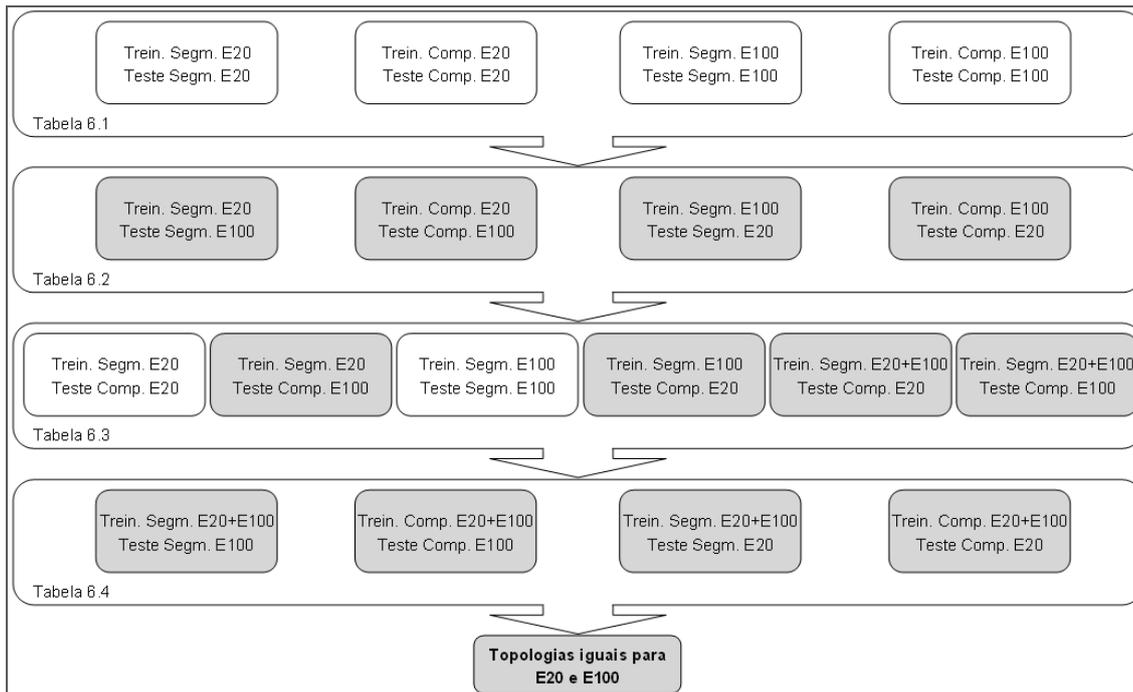


Figura 6.1: Representação do desenvolvimento dos testes.

Nas Tabelas 6.1, 6.2, 6.3 e 6.4 são apresentados os melhores desempenhos das diversas topologias e testes ilustrados na Figura 6.1. A Tabela 6.1 apresenta resultados onde foram utilizados mapas de dados com o mesmo tipo de combustível; na Tabela 6.2 são apresentados resultados onde o tipo de combustível (mapas de dados) utilizado nos treinamentos é diferente dos mapas usados nos testes; na Tabela 6.3 é apresentado resultados onde os tipos de discretização ou mapas de dados são diferentes entre os treinamentos e testes e a Tabela 6.4 apresenta os resultados onde os treinamentos foram feitos com ambos os tipos de combustíveis (mapas de dados) e os testes com mapas únicos. Em referência a esta tabela, a coluna “Nº Top.” indica o índice associado ao referido treinamento; “Mapa Trein.” fornece o tipo de mapa selecionado para o treinamento; “Mapa Teste” fornece o tipo de mapa selecionado para o teste após realizado treinamento; “Tipo [Faixa]” indica o tipo de discretização utilizado e

seu início e fim; “Trein.” apresenta o método de treinamento utilizado, onde *trainlm* é o método de Levenberg-Marquardt e *trainbr* o método Levenberg-Marquardt com regularização Bayesiana; “CNE” significa o número de neurônios contidos na camada neural escondida; “Var.” representa o número de entradas fornecidas à RNA; “Erro absol.” indica o erro absoluto e “Erro relat.” o erro relativo.

Os melhores desempenhos para cada mapa de dados, tipo de discretização, faixa de dados e utilizando os mesmos mapas de dados nos treinamentos e testes são mostrados na Tabela 6.1, sendo que nos treinamentos com todo o universo de dados, foram utilizados os quatro conjuntos de variáveis e nos treinamentos segmentados foram utilizados apenas os três primeiros. Permite-se observar resultados como se o veículo estivesse funcionando com apenas um combustível.

Tabela 6.1: Melhores desempenhos – Treinamentos e Testes utilizando mesmos mapas de dados.

Nro. Top.	Mapa Trein.	Mapa Teste	Tipo [Faixa] Trein.	Tipo [Faixa] Teste	Trein.	CNE	CNE2	Var	Erro Absol.	Erro Relat.
6-Nr229	1	1	Comp. [009 884]	Comp. [009 884]	trainbr	7	7	41	0.00466	1.71110%
6-Nr167	1	1	Comp. [009 884]	Comp. [009 884]	trainbr	10	15	21	0.00472	1.72880%
6-Nr528	2	2	Comp. [009 884]	Comp. [009 884]	trainbr	10	20	41	0.00157	0.32972%
6-Nr456	2	2	Comp. [009 884]	Comp. [009 884]	trainbr	10	20	21	0.00158	0.33015%
2-Nr25	1	1	Segm. [009 096]	Segm. [009 096]	trainlm	20	0	21	0.00039	0.22082%
2-Nr66	1	1	Segm. [097 183]	Segm. [097 183]	trainlm	60	0	21	0.00089	0.44691%
2-Nr92	1	1	Segm. [184 271]	Segm. [184 271]	trainlm	25	0	21	0.00178	0.80673%
2-Nr126	1	1	Segm. [272 359]	Segm. [272 359]	trainlm	30	0	21	0.00173	0.62255%
3-Nr126	1	1	Segm. [360 446]	Segm. [360 446]	trainlm	15	0	21	0.00382	1.33280%
2-Nr194	1	1	Segm. [447 533]	Segm. [447 533]	trainlm	40	0	21	0.00052	0.20619%
2-Nr223	1	1	Segm. [534 622]	Segm. [534 622]	trainlm	20	0	21	0.00056	0.22283%
3-Nr184	1	1	Segm. [623 709]	Segm. [623 709]	trainlm	7	0	21	0.00069	0.27514%
2-Nr296	1	1	Segm. [710 796]	Segm. [710 796]	trainlm	55	0	21	0.00075	0.30688%
2-Nr323	1	1	Segm. [797 884]	Segm. [797 884]	trainlm	25	0	21	0.00123	0.49763%
3-Nr273	2	2	Segm. [009 096]	Segm. [009 096]	trainlm	15	0	21	0.00101	0.30093%
2-Nr396	2	2	Segm. [097 183]	Segm. [097 183]	trainlm	60	0	21	0.00106	0.26261%
2-Nr427	2	2	Segm. [184 271]	Segm. [184 271]	trainlm	50	0	21	0.00119	0.25627%
2-Nr455	2	2	Segm. [272 359]	Segm. [272 359]	trainlm	25	0	21	0.00075	0.15184%
3-Nr354	2	2	Segm. [360 446]	Segm. [360 446]	trainlm	10	0	21	0.00093	0.18484%
2-Nr526	2	2	Segm. [447 533]	Segm. [447 533]	trainlm	50	0	21	0.00123	0.24450%
2-Nr555	2	2	Segm. [534 622]	Segm. [534 622]	trainlm	30	0	21	0.00144	0.28630%
2-Nr588	2	2	Segm. [623 709]	Segm. [623 709]	trainlm	30	0	21	0.00144	0.28552%
3-Nr438	2	2	Segm. [710 796]	Segm. [710 796]	trainlm	10	0	21	0.00133	0.26342%
3-Nr460	2	2	Segm. [797 884]	Segm. [797 884]	trainlm	12	0	21	0.00111	0.22006%

Objetivando verificar a possibilidade de aplicar apenas uma topologia para qualquer tipo de combustível, a Tabela 6.2 mostra os melhores desempenhos onde os mapas de dados utilizados nos treinamentos são diferentes dos mapas usados nos testes para os mesmos tipos de discretizações.

Tabela 6.2: Melhores desempenhos – Treinamentos e Testes utilizando mapas de dados distintos.

Nro. Top.	Mapa Trein.	Mapa Teste	Tipo [Faixa] Trein.	Tipo [Faixa] Teste	Trein.	CNE	CNE2	Var	Erro Absol.	Erro Relat.
4-Nr19	1	2	Comp. [009 884]	Comp. [009 884]	trainbr	12	0	21	0.04481	9.32100%
6-Nr223	1	2	Comp. [009 884]	Comp. [009 884]	trainbr	5	15	41	0.06830	15.51000%
3-Nr232	2	1	Comp. [009 884]	Comp. [009 884]	trainlm	5	0	6	0.09167	38.17700%
6-Nr337	2	1	Comp. [009 884]	Comp. [009 884]	trainbr	25	2	6	0.11796	51.07200%
4-Nr33	1	2	Segm. [009 096]	Segm. [009 096]	trainbr	12	0	6	0.06332	18.23300%
3-Nr53	1	2	Segm. [097 183]	Segm. [097 183]	trainlm	10	0	6	0.10456	26.26000%
4-Nr83	1	2	Segm. [184 271]	Segm. [184 271]	trainbr	13	0	21	0.05266	11.18100%
4-Nr104	1	2	Segm. [272 359]	Segm. [272 359]	trainbr	13	0	21	0.08007	16.21000%
4-Nr107	1	2	Segm. [360 446]	Segm. [360 446]	trainbr	7	0	6	0.06062	12.13200%
4-Nr143	1	2	Segm. [447 533]	Segm. [447 533]	trainbr	9	0	21	0.05804	11.49500%
3-Nr151	1	2	Segm. [534 622]	Segm. [534 622]	trainlm	10	0	6	0.03198	6.33370%
4-Nr177	1	2	Segm. [623 709]	Segm. [623 709]	trainbr	7	0	6	0.03247	6.43700%
2-Nr296	1	2	Segm. [710 796]	Segm. [710 796]	trainlm	55	0	21	0.03731	7.39950%
2-Nr326	1	2	Segm. [797 884]	Segm. [797 884]	trainlm	40	0	21	0.07073	14.03400%
3-Nr267	2	1	Segm. [009 096]	Segm. [009 096]	trainlm	5	0	21	0.11087	62.86900%
2-Nr387	2	1	Segm. [097 183]	Segm. [097 183]	trainlm	15	0	21	0.71363	359.52000%
2-Nr400	2	1	Segm. [184 271]	Segm. [184 271]	trainlm	25	0	6	0.12333	56.27100%
3-Nr330	2	1	Segm. [272 359]	Segm. [272 359]	trainlm	5	0	21	0.14912	53.18800%
3-Nr337	2	1	Segm. [360 446]	Segm. [360 446]	trainlm	5	0	6	0.22747	79.51200%
2-Nr500	2	1	Segm. [447 533]	Segm. [447 533]	trainlm	30	0	6	0.05417	21.56300%
3-Nr382	2	1	Segm. [534 622]	Segm. [534 622]	trainlm	10	0	6	0.01699	6.74870%
3-Nr419	2	1	Segm. [623 709]	Segm. [623 709]	trainlm	13	0	21	0.06815	27.45400%
2-Nr596	2	1	Segm. [710 796]	Segm. [710 796]	trainlm	15	0	6	0.06324	26.32000%
3-Nr452	2	1	Segm. [797 884]	Segm. [797 884]	trainlm	10	0	6	0.11511	47.61600%

Os desempenhos apresentados na Tabela 6.2 estão aquém de resultados obtidos em outras topologias. Como a dinâmica dos motores é alterada com diferentes proporções de combustíveis, torna-se mais difícil convergir a rede neural quando treinada exclusivamente por um conjunto de dados e, posteriormente, é testada com outro conjunto que altera a dinâmica do motor.

Uma amostra das topologias, onde os treinamentos foram realizados de forma segmentada (discretização) e os testes usaram todo o universo de dados, é apresentada na Tabela 6.3, ordenada pelo erro relativo. Permite-se analisar se a rede neural artificial converge para todo o conjunto disponível, sendo treinada usando-se apenas subconjuntos de mapas iguais ou distintos.

Tabela 6.3: Melhores desempenhos – Treinamentos e Testes utilizando tipos de discretização ou mapas de dados distintos.

Nro Top.	Mapa Trein.	Mapa Teste	Tipo [Faixa] Trein.	Tipo [Faixa] Teste	Trein.	CNE	CNE2	Var	Erro Absol.	Erro Relat.
3-Nr51	1	1	Segm. [097 183]	Comp. [009 884]	trainlm	7	0	6	0.02472	8.51890%
7-Nr333	1+2	1	Segm. [360 446]	Comp. [009 884]	trainlm	5	15	21	0.02444	9.84490%
4-Nr170	1	1	Segm. [623 709]	Comp. [009 884]	trainbr	7	0	6	0.02829	10.97600%
7-Nr106	1+2	1	Segm. [009 096]	Comp. [009 884]	trainlm	5	5	41	0.03247	11.12700%
4-Nr148	1	1	Segm. [534 622]	Comp. [009 884]	trainbr	5	0	6	0.02816	11.35400%
4-Nr211	1	1	Segm. [797 884]	Comp. [009 884]	trainbr	5	0	6	0.02913	11.52500%
2-Nr155	1	1	Segm. [360 446]	Comp. [009 884]	trainlm	10	0	21	0.02699	12.23800%
7-Nr649	1+2	1	Segm. [797 884]	Comp. [009 884]	trainlm	7	5	41	0.03576	13.41800%
4-Nr72	1	1	Segm. [184 271]	Comp. [009 884]	trainbr	7	0	6	0.03650	13.42300%
4-Nr127	1	1	Segm. [447 533]	Comp. [009 884]	trainbr	5	0	6	0.03456	13.93600%
8-Nr256	1+2	2	Segm. [272 359]	Comp. [009 884]	trainbr	5	5	6	0.00809	1.91540%
3-Nr446	2	2	Segm. [797 884]	Comp. [009 884]	trainlm	12	0	6	0.00974	2.34580%
8-Nr432	1+2	2	Segm. [534 622]	Comp. [009 884]	trainbr	15	15	6	0.01191	2.79220%
8-Nr487	1+2	2	Segm. [623 709]	Comp. [009 884]	trainbr	10	5	6	0.01384	3.25750%
7-Nr552	1+2	2	Segm. [710 796]	Comp. [009 884]	trainlm	15	15	6	0.01513	3.66980%
8-Nr606	1+2	2	Segm. [797 884]	Comp. [009 884]	trainbr	7	15	6	0.01640	4.00890%
8-Nr390	1+2	2	Segm. [447 533]	Comp. [009 884]	trainbr	20	15	6	0.02012	5.29030%
8-Nr336	1+2	2	Segm. [360 446]	Comp. [009 884]	trainbr	7	15	21	0.02383	5.92080%
4-Nr267	2	2	Segm. [009 096]	Comp. [009 884]	trainbr	5	0	21	0.02985	6.05970%
7-Nr74	1+2	2	Segm. [009 096]	Comp. [009 884]	trainlm	20	10	6	0.02989	6.19620%

Na Tabela 6.4 são apresentados os melhores desempenhos onde os dois mapas de dados utilizados nos treinamentos foram unidos, objetivando melhorar a generalização para qualquer tipo de combustível. Em comparação com a Tabela 6.2, os resultados obtidos pelas topologias contidas na Tabela 6.4 mostram-se melhores. Motivado pela utilização dos dois conjuntos no treinamento, as

topologias têm boa convergência para ambos os mapas de dados, ou seja, bicombustíveis.

Tabela 6.4: Melhores desempenhos – Treinamentos utilizando ambos os mapas de dados e testes utilizando apenas um dos mapas.

Nro Top.	Mapa Trein.	Mapa Teste	Tipo [Faixa] Trein.	Tipo [Faixa] Teste	Trein.	CNE	CNE2	Var	Erro Absol.	Erro Relat.
7-Nr48	1+2	1	Comp. [009 884]	Comp. [009 884]	trainlm	5	15	41	0.00650	2.39940%
7-Nr50	1+2	1	Comp. [009 884]	Comp. [009 884]	trainlm	7	10	41	0.00640	2.57050%
7-Nr57	1+2	2	Comp. [009 884]	Comp. [009 884]	trainlm	15	15	41	0.00181	0.39128%
7-Nr60	1+2	2	Comp. [009 884]	Comp. [009 884]	trainlm	20	15	41	0.00183	0.39732%
8-Nr119	1+2	1	Segm. [009 096]	Segm. [009 096]	trainbr	20	10	41	0.00028	0.15598%
8-Nr162	1+2	1	Segm. [097 183]	Segm. [097 183]	trainbr	15	15	21	0.00083	0.42536%
8-Nr225	1+2	1	Segm. [184 271]	Segm. [184 271]	trainbr	20	15	21	0.00092	0.41035%
8-Nr285	1+2	1	Segm. [272 359]	Segm. [272 359]	trainbr	20	15	21	0.00067	0.21339%
8-Nr345	1+2	1	Segm. [360 446]	Segm. [360 446]	trainbr	20	15	21	0.00108	0.37373%
8-Nr420	1+2	1	Segm. [447 533]	Segm. [447 533]	trainbr	20	15	41	0.00045	0.17864%
8-Nr476	1+2	1	Segm. [534 622]	Segm. [534 622]	trainbr	15	10	41	0.00059	0.23611%
7-Nr540	1+2	1	Segm. [623 709]	Segm. [623 709]	trainlm	20	15	41	0.00061	0.24170%
8-Nr594	1+2	1	Segm. [710 796]	Segm. [710 796]	trainbr	10	15	41	0.00058	0.23128%
8-Nr643	1+2	1	Segm. [797 884]	Segm. [797 884]	trainbr	20	5	21	0.00095	0.37373%
8-Nr116	1+2	2	Segm. [009 096]	Segm. [009 096]	trainbr	15	10	41	0.00052	0.14663%
8-Nr175	1+2	2	Segm. [097 183]	Segm. [097 183]	trainbr	15	5	41	0.00105	0.26128%
8-Nr237	1+2	2	Segm. [184 271]	Segm. [184 271]	trainbr	15	15	41	0.00083	0.18021%
8-Nr298	1+2	2	Segm. [272 359]	Segm. [272 359]	trainbr	20	5	41	0.00060	0.12252%
8-Nr360	1+2	2	Segm. [360 446]	Segm. [360 446]	trainbr	20	15	41	0.00083	0.16537%
8-Nr420	1+2	2	Segm. [447 533]	Segm. [447 533]	trainbr	20	15	41	0.00114	0.22607%
8-Nr476	1+2	2	Segm. [534 622]	Segm. [534 622]	trainbr	15	10	41	0.00110	0.21889%
8-Nr526	1+2	2	Segm. [623 709]	Segm. [623 709]	trainbr	5	5	41	0.00125	0.24771%
8-Nr589	1+2	2	Segm. [710 796]	Segm. [710 796]	trainbr	7	5	41	0.00119	0.23624%
7-Nr658	1+2	2	Segm. [797 884]	Segm. [797 884]	trainlm	20	5	41	0.00088	0.17441%

Os resultados apresentados neste capítulo foram obtidos utilizando o toolbox Neural Networks do Matlab e foram agrupados em função dos mapas de dados utilizados nos treinamentos e testes e dos tipos de discretizações.

## 6.2 - Resultados de Testes como Monocombustíveis

### 6.2.1 - Treinamentos e Testes com E20

#### 6.2.1.1 - Mapa Segmentado

Na sequência são apresentados alguns resultados dos testes em que foi utilizado treinamento segmentado para combustível do tipo E20.

Dentre as várias configurações que não foram satisfatórias, na Figura 6.2 é exemplificado uma topologia invalidada, em que foi utilizado 35 neurônios na primeira camada neural escondida, 5 variáveis como entrada e 5% dos dados do conjunto foram usados (Qt. Discr.) no treinamento e teste, podendo-se observar primeiramente o alto  $\bar{e}_r$  e, em seguida, utilizando a figura, o efeito *overfitting*.

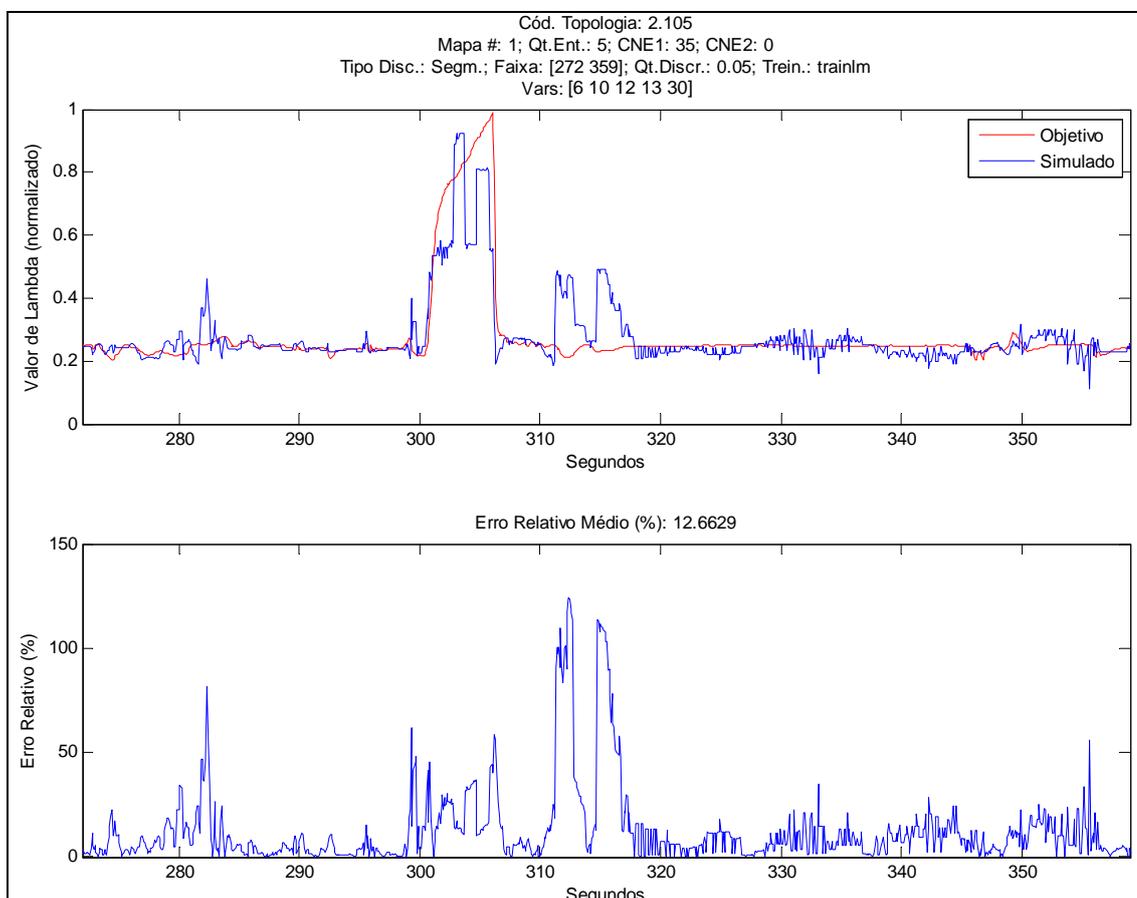


Figura 6.2: Exemplo de resultado invalidado de treinamento e teste com E20-Segmentado.

Em relação à próxima figura (Figura 6.3), pode-se notar um resultado satisfatório, em que o valor real de lambda é bem irregular e o sistema conseguiu aproximar o resultado alcançado daquele desejado. Neste caso, a rede PMC foi composta de 60 neurônios na primeira camada intermediária, tendo a mesma 21 entradas e treinada usando-se o método *trainlm* (método de Levenberg-Marquardt).

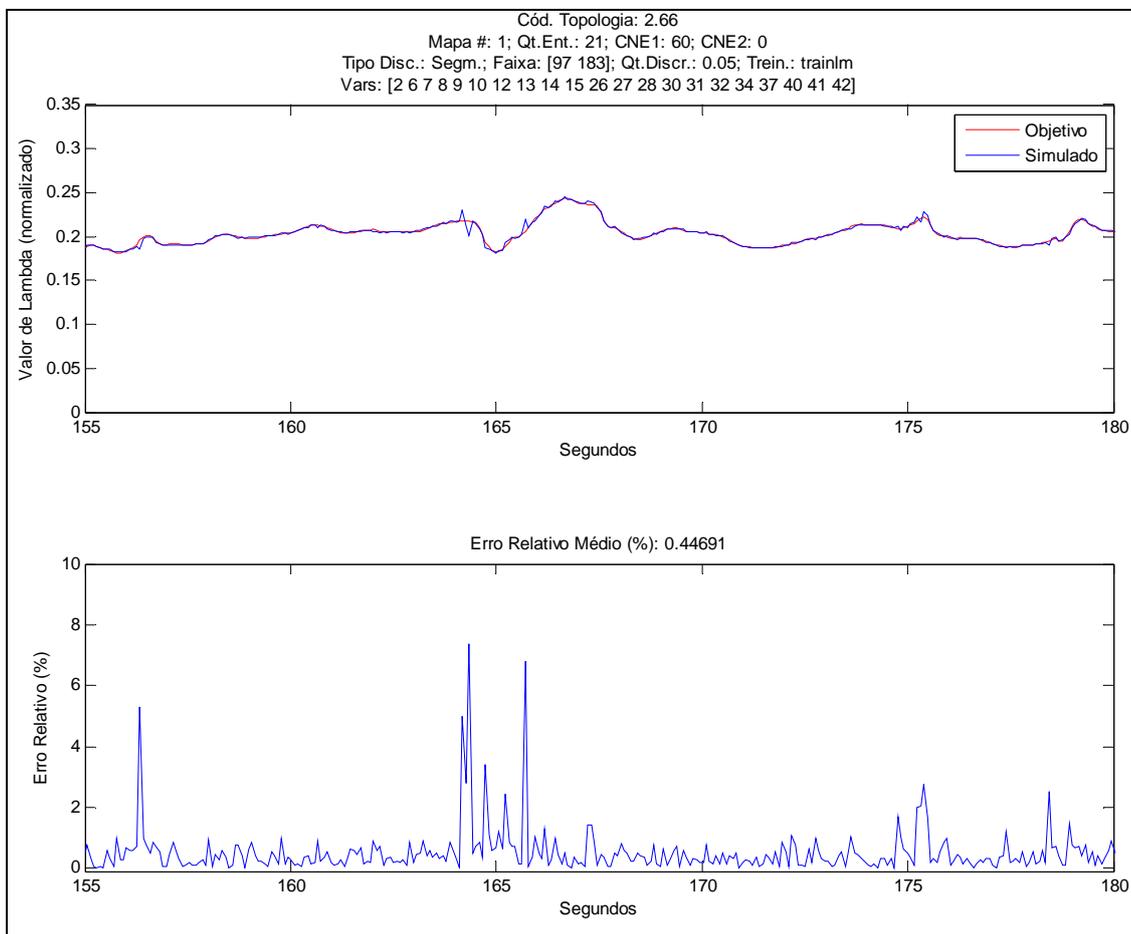


Figura 6.3: Exemplo de resultado de treinamento e teste com E20-Segmentado (Situação I).

A seguir (Figura 6.4), outro bom exemplo de topologia em que ocorre um grande pico após 300s. Esta configuração usando 30 neurônios na primeira camada neural e 21 entradas, embora tenha tido um erro pontual grande, teve um resultado satisfatório, onde o  $\bar{e}_r$  foi igual a 0,62255%.

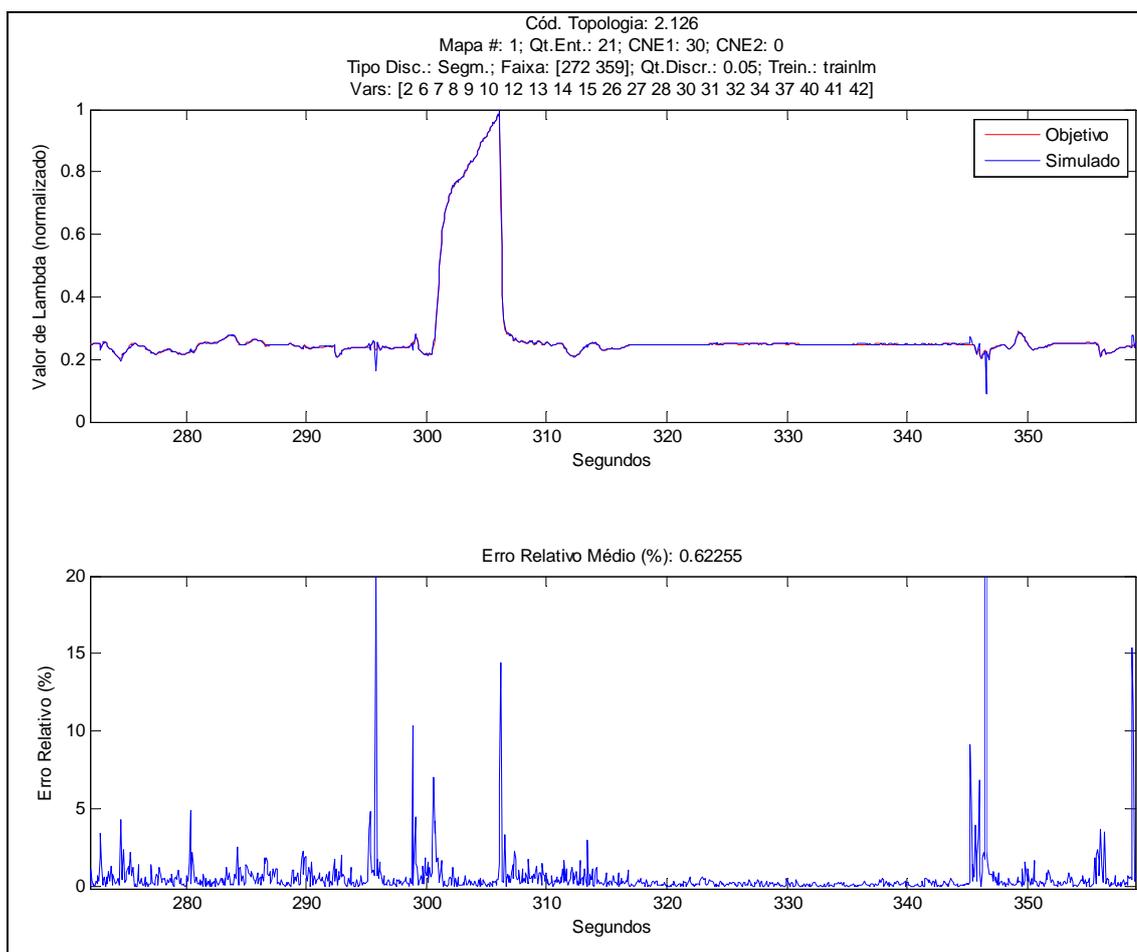


Figura 6.4: Exemplo de resultado de treinamento e teste com E20-Segmentado (Situação II).

O próximo exemplo (Figura 6.5) ilustra o típico comportamento do sensor de oxigênio, oscilante, em que se observa uma boa precisão alcançada pelo SI na estimação dos valores de lambda, tendo a RNA um total de 40 neurônios na primeira camada, 21 entradas e utilizando-se o método de treinamento *trainlm*.

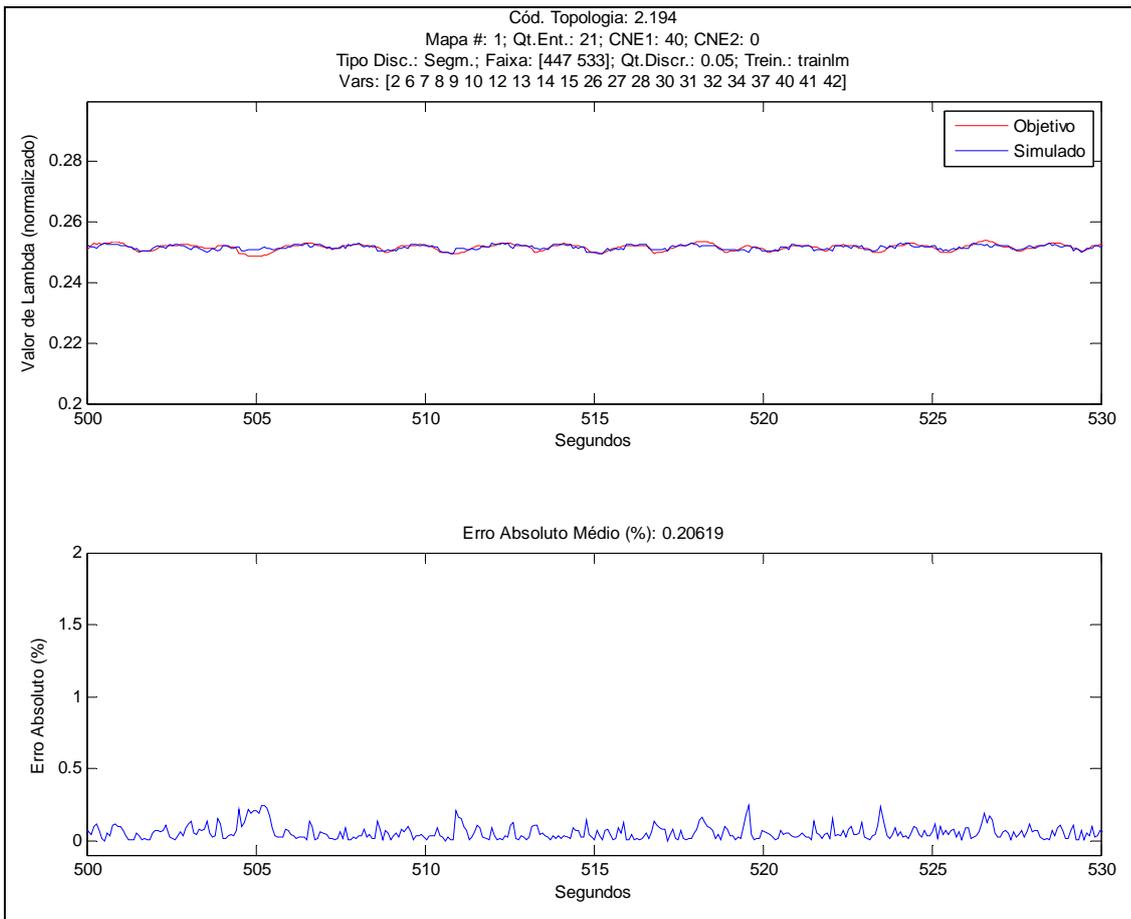


Figura 6.5: Exemplo de resultado de treinamento e teste com E20-Segmentado (Situação III).

Na Figura 6.6 se manteve as mesmas entradas usadas na Figura 6.5, alterando-se a quantidade de neurônios para 10, a amplitude e o tempo, porém, obteve erro relativo médio de 0,2447%.

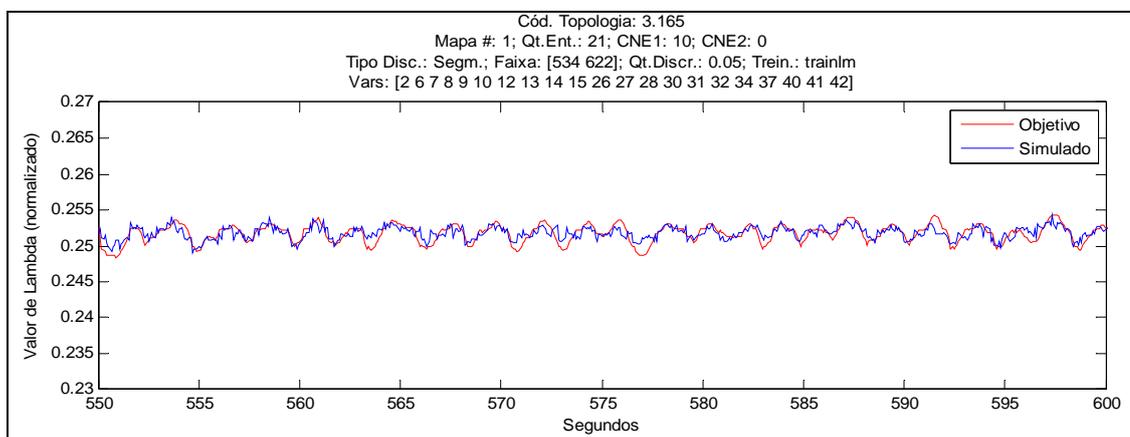


Figura 6.6: Exemplo de resultado de treinamento e teste com E20-Segmentado (Situação IV).

Na Figura 6.7 é apresentado o melhor resultado para a décima faixa de dados discretizada (710 a 796s). Neste caso, a rede PMC foi composta de 55 neurônios na primeira camada e 21 variáveis.

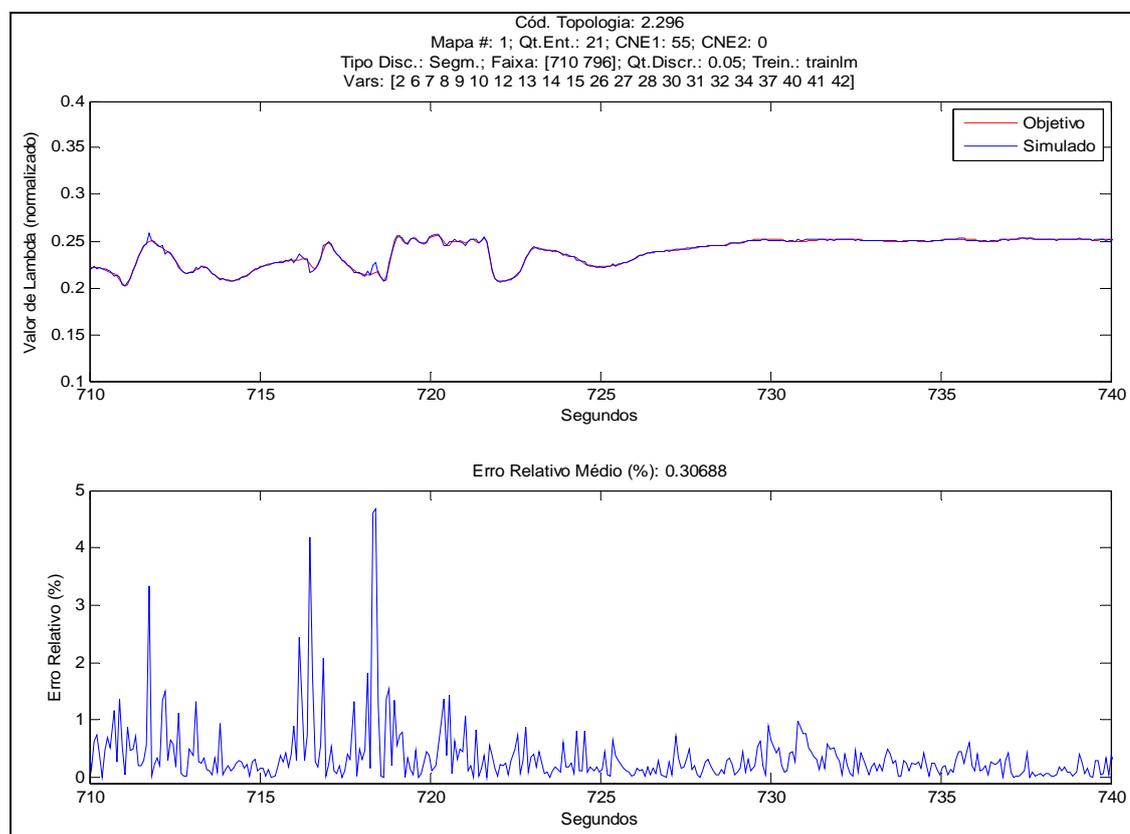


Figura 6.7: Exemplo de resultado de treinamento e teste com E20-Segmentado (Situação V).

Na figura seguinte (Figura 6.8) apresenta-se o resultado do teste em outro subconjunto de dados, sendo a rede composta por 25 neurônios e 21 entradas.

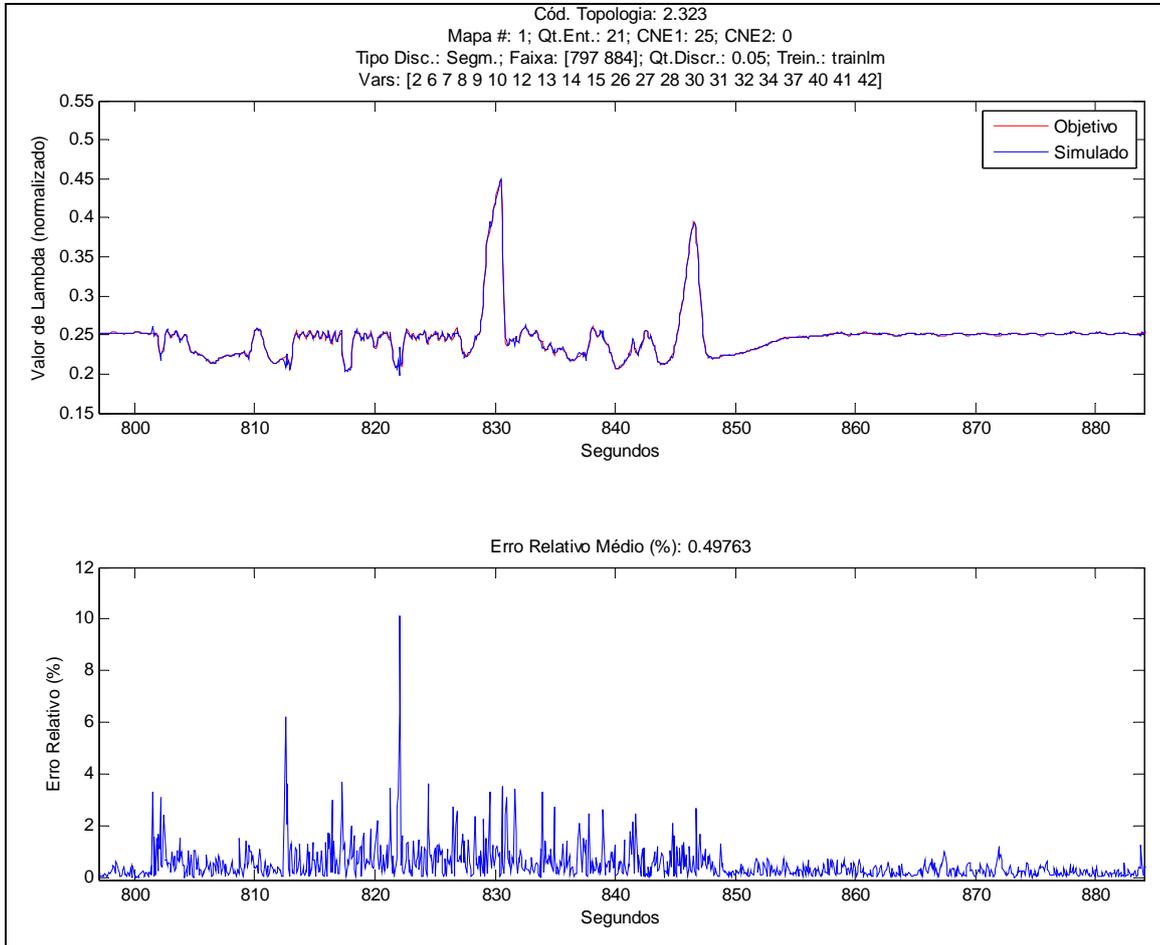


Figura 6.8: Exemplo de resultado de treinamento e teste com E20-Segmentado (Situação VI).

### 6.2.1.2 - Mapa Completo

Baseado nos bons resultados dos testes segmentados, treinamentos e testes com todo o espectro de dados disponível foram também realizados. Nestes treinamentos foram utilizados todos os quatro conjuntos de variáveis de entrada.

Como apresentado nos tópicos anteriores, é exemplificado na Figura 6.9 as configurações invalidadas. Neste caso, utilizou-se 6 entradas e 40 neurônios na

primeira camada neural. Como o número de neurônios é muito grande em relação às entradas, ocorreu-se então o *overfitting*.

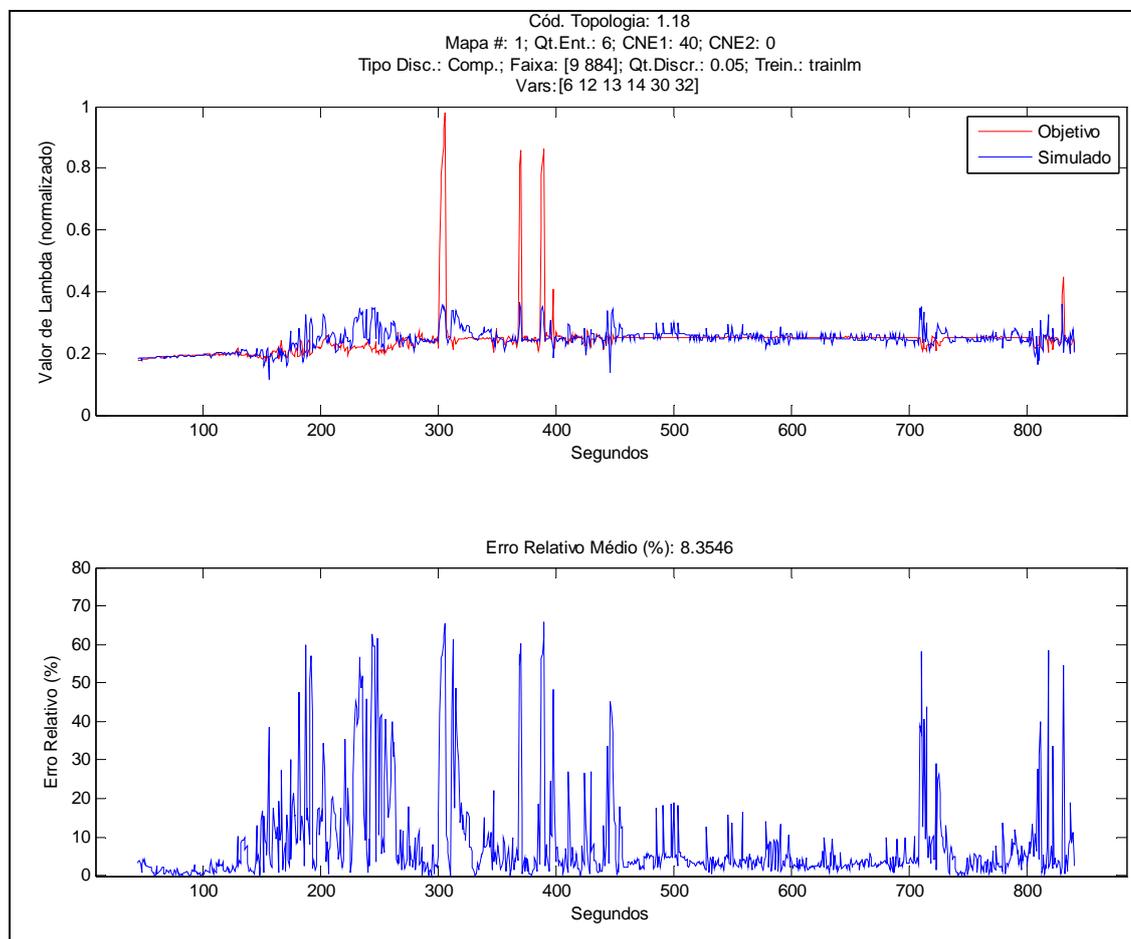


Figura 6.9: Exemplo de resultado invalidado de treinamento e teste com E20-Completo.

Na sequência (Figura 6.10) é apresentado um dos melhores resultados para os treinamentos com todo o universo de dados. Neste caso, a rede treinada com o método *trainbr* (método de Levenberg-Marquardt com regularização Bayesiana) foi composta por 10 neurônios na primeira camada neural escondida, 15 na segunda camada neural escondida e 21 variáveis de entrada.

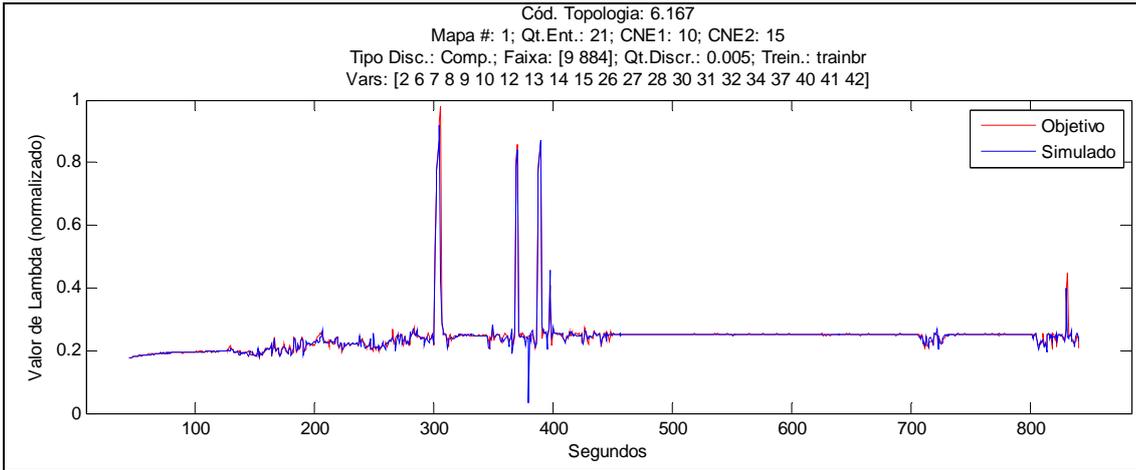


Figura 6.10: Exemplo de resultado de treinamento e teste com E20-Completo (Situação I).

Na Figura 6.11 é realizada uma ampliação da figura anterior, permitindo-se visualizar maiores detalhes.

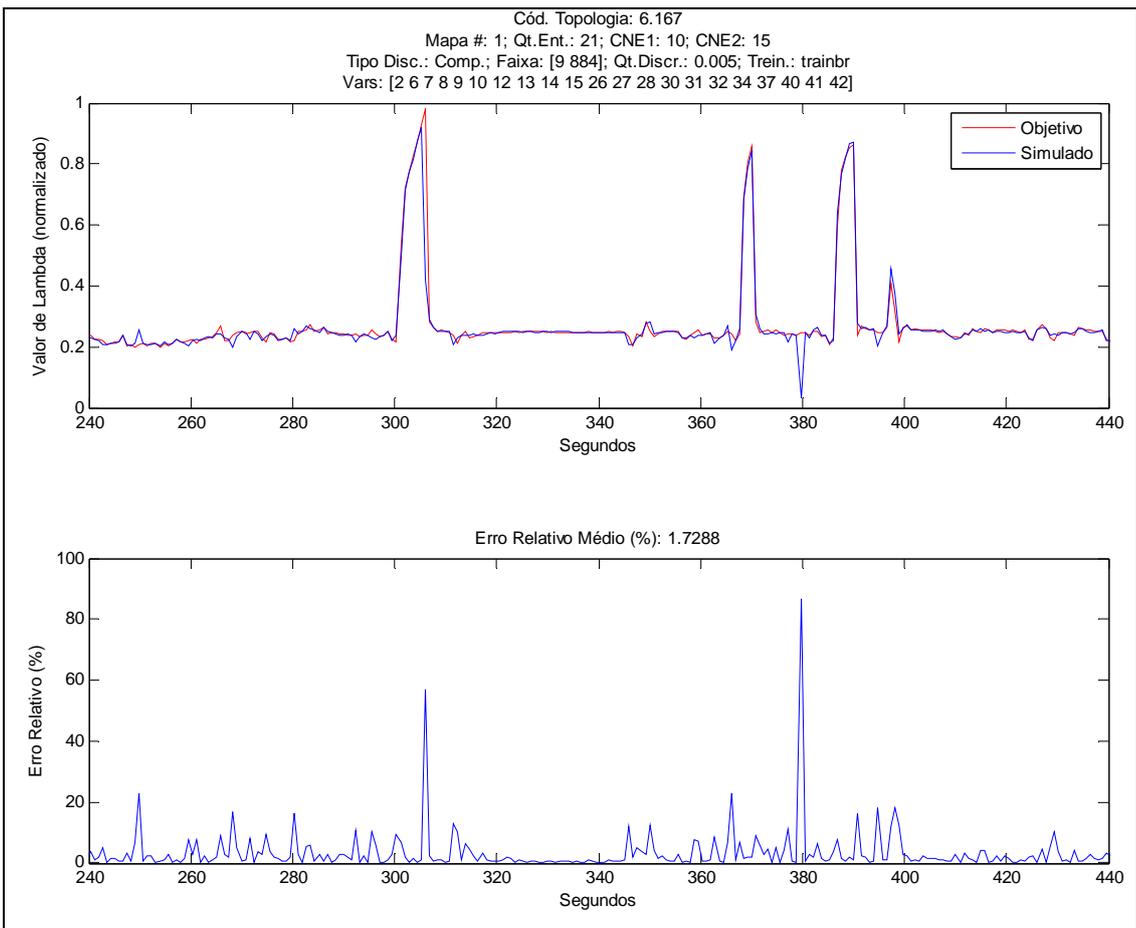


Figura 6.11: Detalhes dos resultados apresentados na Figura 6.10.

É apresentado na Figura 6.12 resultados melhorados em comparação àqueles apresentados na Figura 6.11, utilizando-se para tanto todas as variáveis, 7 neurônios em cada camada neural e o método *trainbr*.

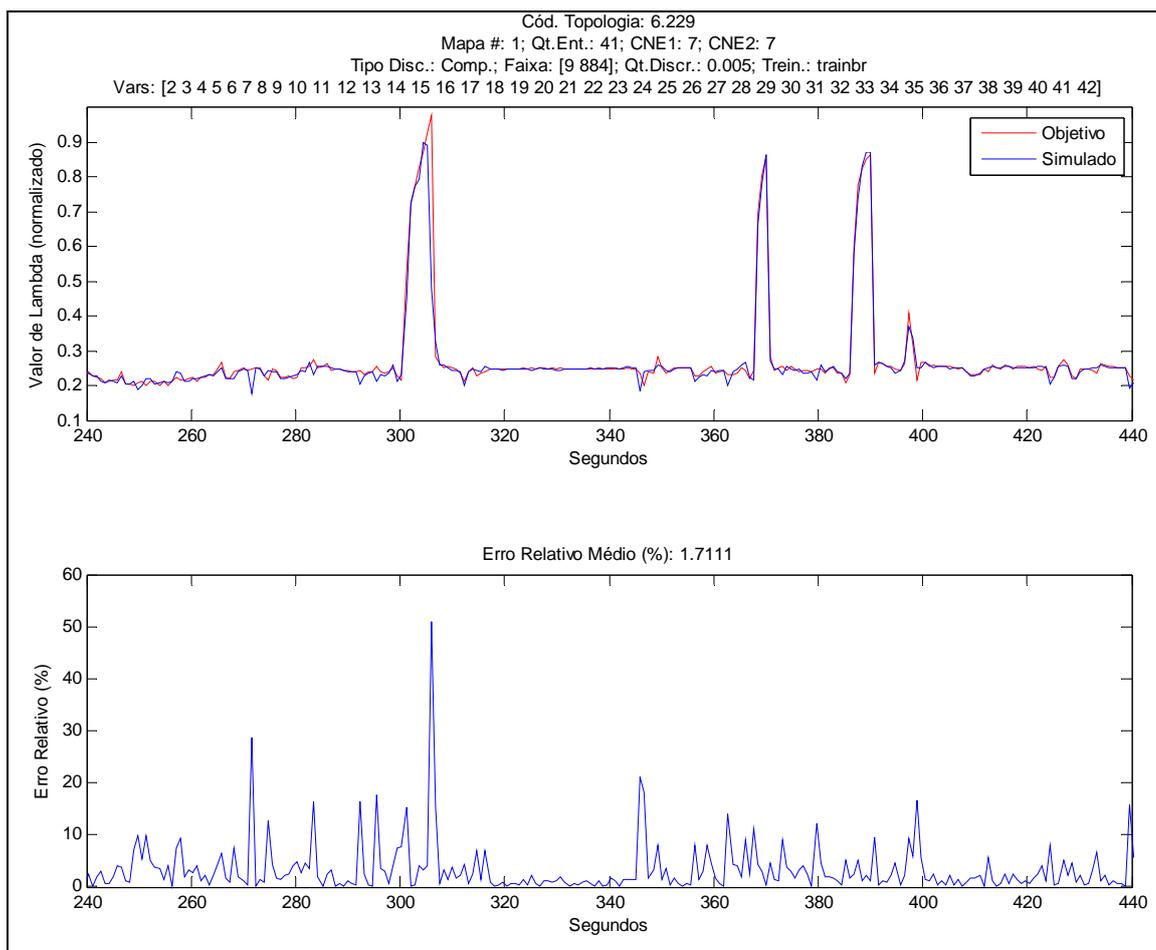


Figura 6.12: Exemplo de resultado de treinamento e teste com E20-Completo (Situação II).

## 6.2.2 - Treinamentos e Testes com E100

### 6.2.2.1 - Mapa Segmentado

Os resultados a seguir pertencem aos testes do treinamento segmentado para combustível do tipo E100. A Figura 6.13 detalha um resultado validado, em que nota-se que a amplitude está aumentada, tendo a rede PMC 60 neurônios na primeira camada e 21 entradas.

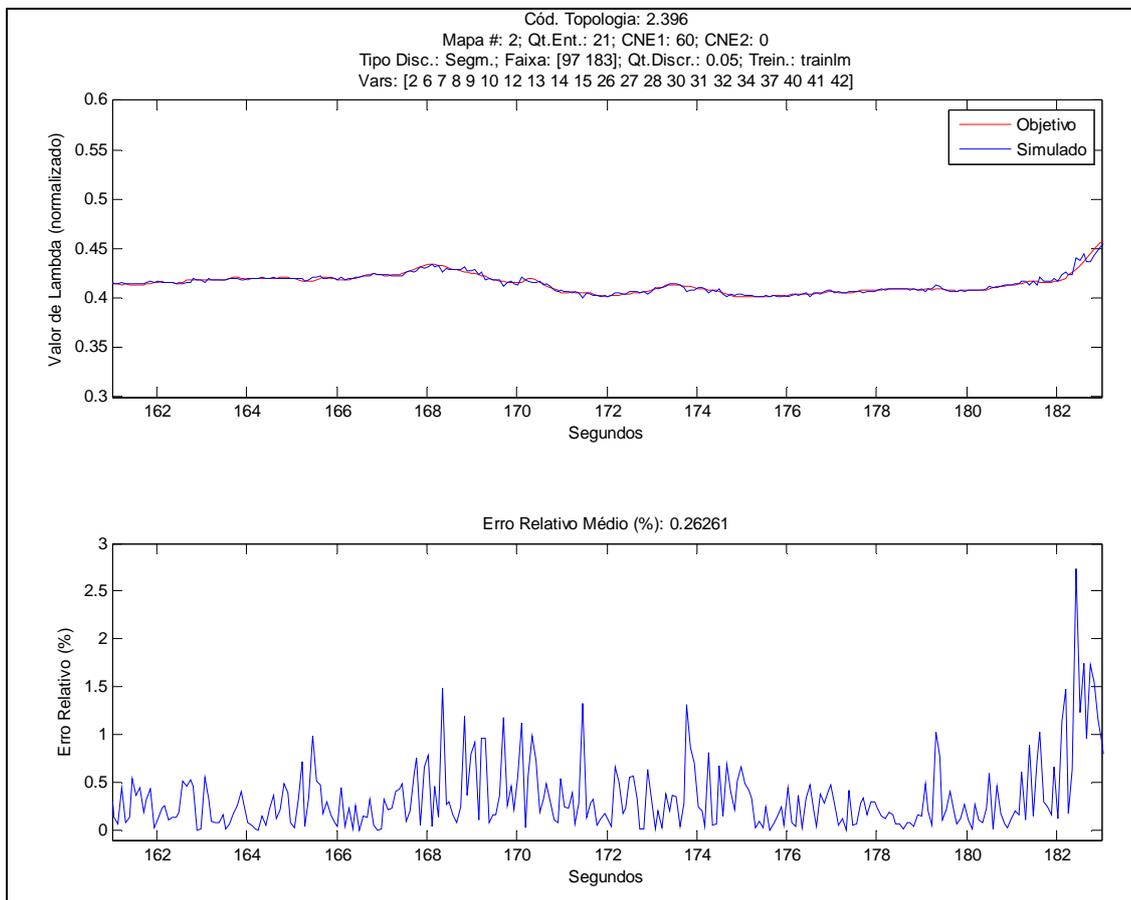


Figura 6.13: Exemplo de resultado de treinamento e teste com E100-Segmentado (Situação I).

A seguir (Figura 6.14) é apresentado um resultado de uma configuração invalidada usando 12 neurônios, 6 entradas e função de treinamento *trainbr*. Nota-se que o  $\overline{e_r}$  foi alto, em comparação com configurações parecidas, e ocorreu o efeito *underfitting*, implicando em sua não convergência.

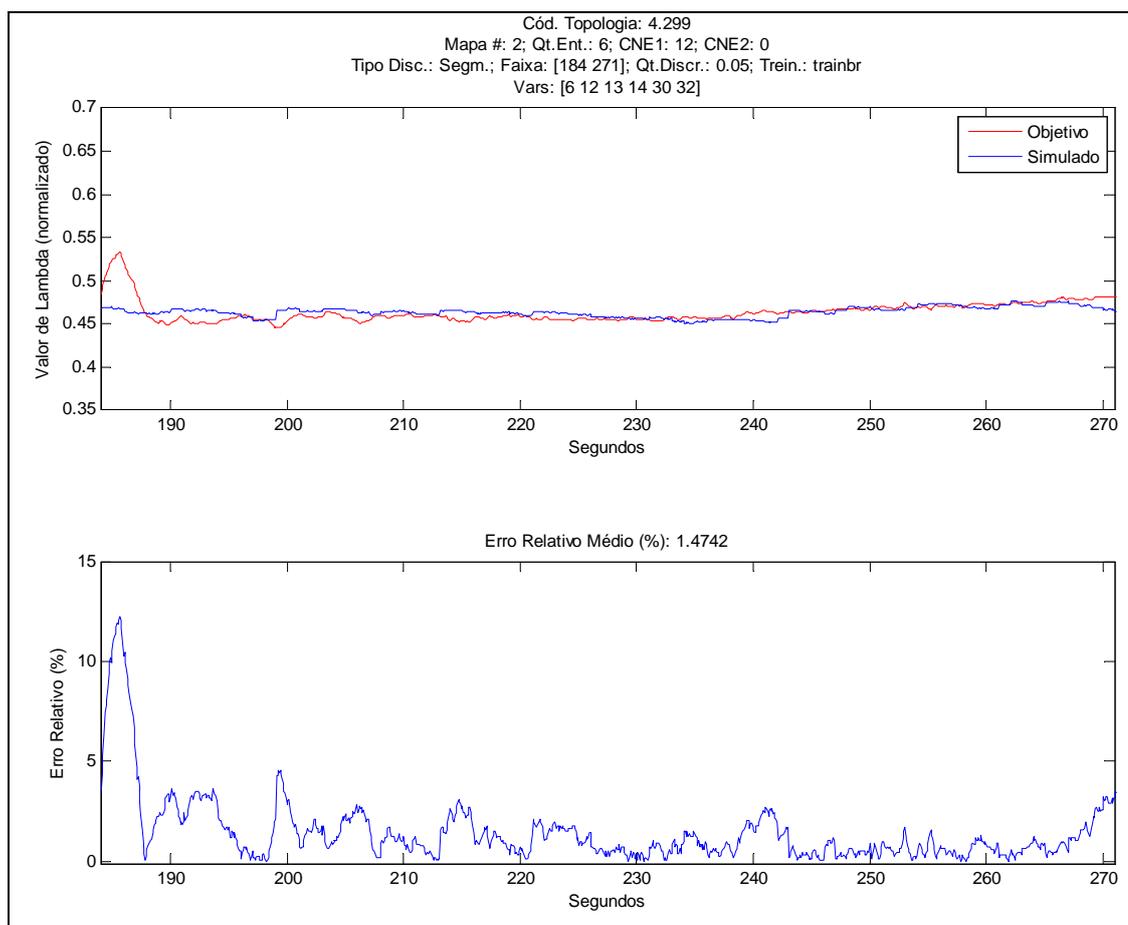


Figura 6.14: Exemplo de resultado invalidado de treinamento e teste com E100-Segmentado.

Este próximo exemplo (Figura 6.15) ilustra novamente o comportamento típico do sensor de oxigênio, observando-se que o comportamento foi mantido e com uma boa precisão para esta rede constituída de 50 neurônios e 21 entradas.

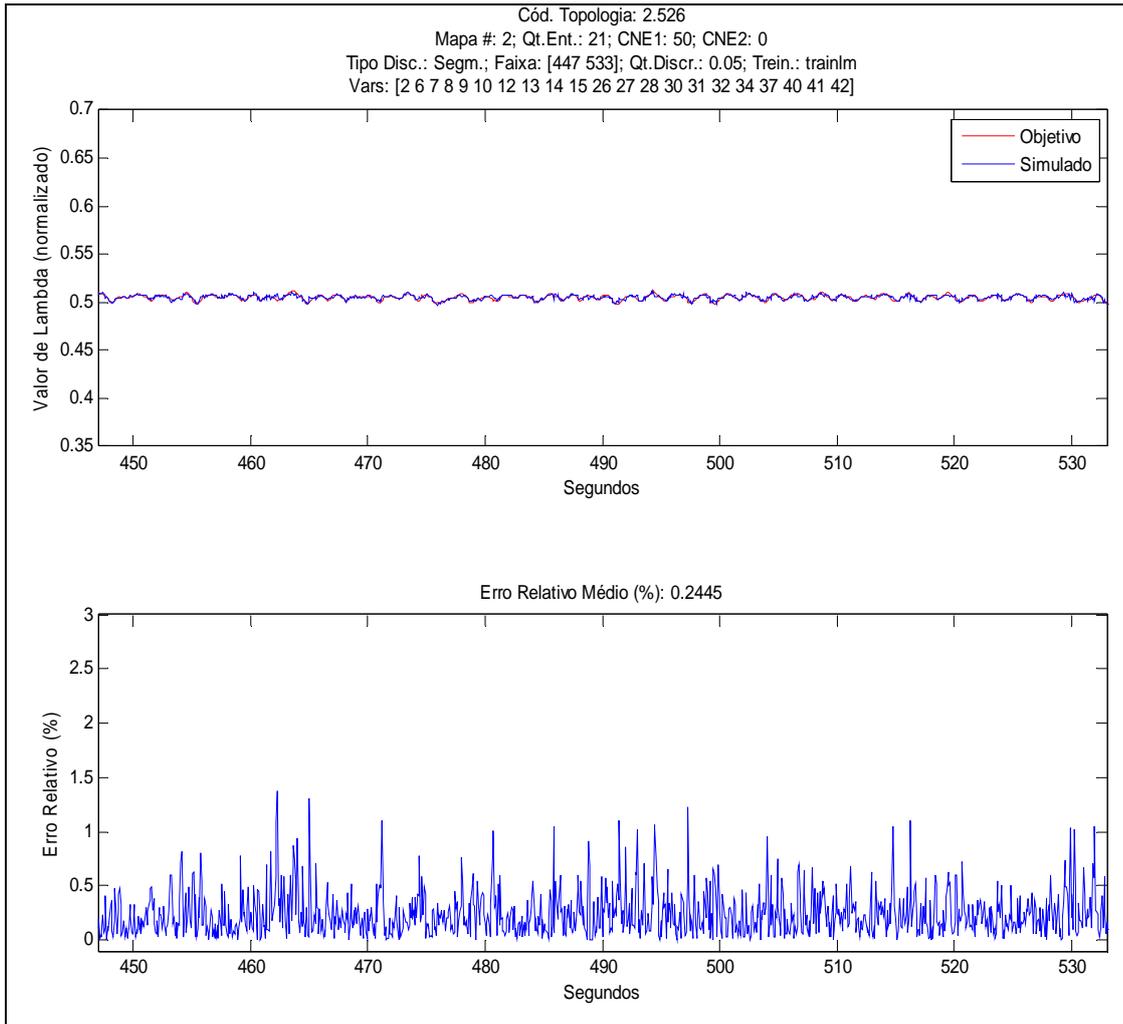


Figura 6.15: Exemplo de resultado de treinamento e teste com E100-Segmentado (Situação II).

Na Figura 6.16 é feito uma ampliação, visando-se apresentar maiores detalhes do resultado apresentado na Figura 6.15.

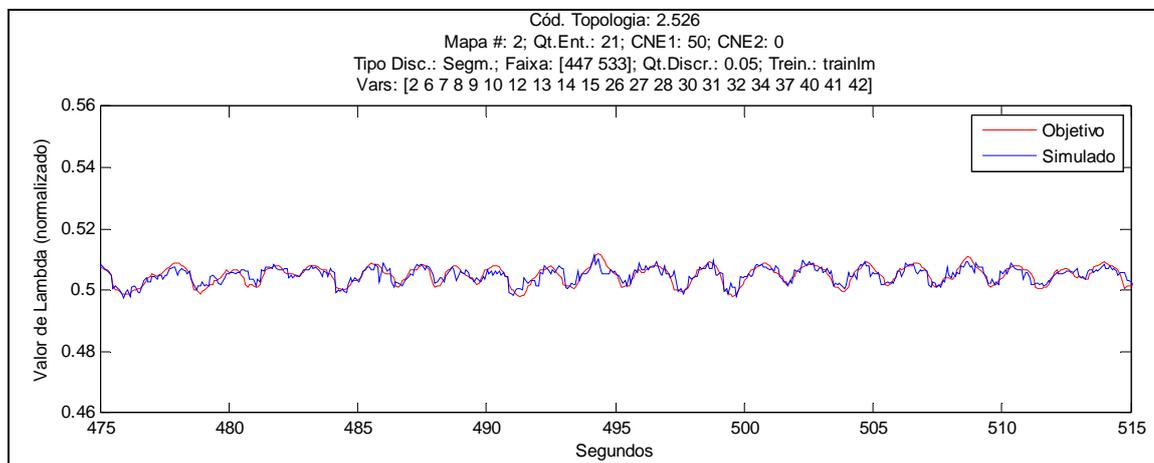


Figura 6.16: Detalhes dos resultados apresentados na Figura 6.14.

#### 6.2.2.2 - Mapa Completo

Nos gráficos seguintes são apresentados alguns resultados dos testes em que foi utilizado treinamento completo para combustível do tipo E100. Os resultados para este mapa de dados, com discretização de todo o universo de discurso, foram significativamente melhores comparados aos resultados obtidos para o mapa E20 (Tabela 6.1).

Um resultado validado é apresentado na Figura 6.17, onde  $\overline{e_r}$  foi 0,46373% e a rede foi configurada para ter 6 entradas e 55 neurônios na primeira camada neural escondida.

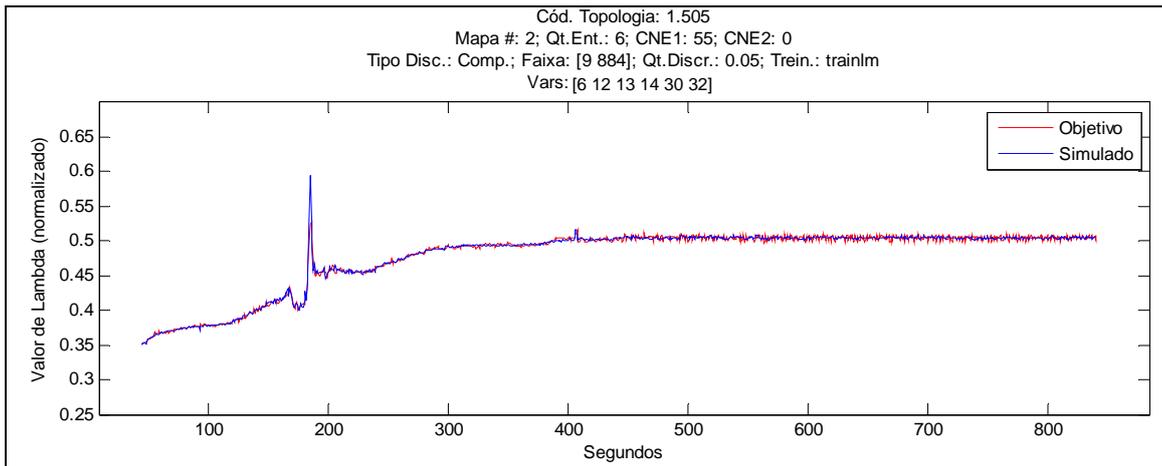


Figura 6.17: Exemplo de resultado de treinamento e teste com E100-Completo (Situação I).

Por meio de ampliação, é apresentado na próxima figura (Figura 6.18), maiores detalhes da Figura 6.17.

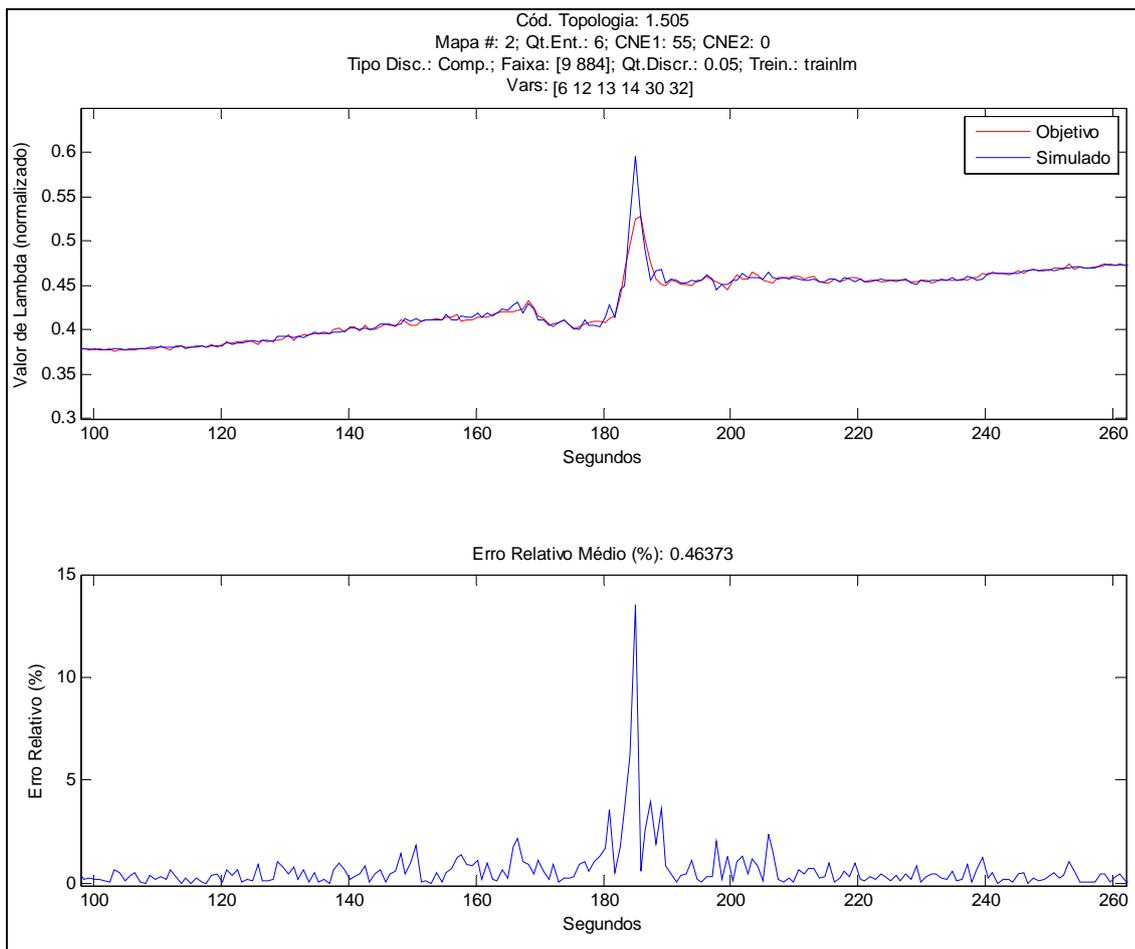


Figura 6.18: Detalhes dos resultados apresentados na Figura 6.17.

Resultados melhorados em comparação aos resultados apresentados na Figura 6.18 são mostrados na Figura 6.19, utilizando-se para tanto um número menor de neurônios nas camadas intermediárias, 10 na primeira e 20 na segunda, e mais variáveis de entrada (41).

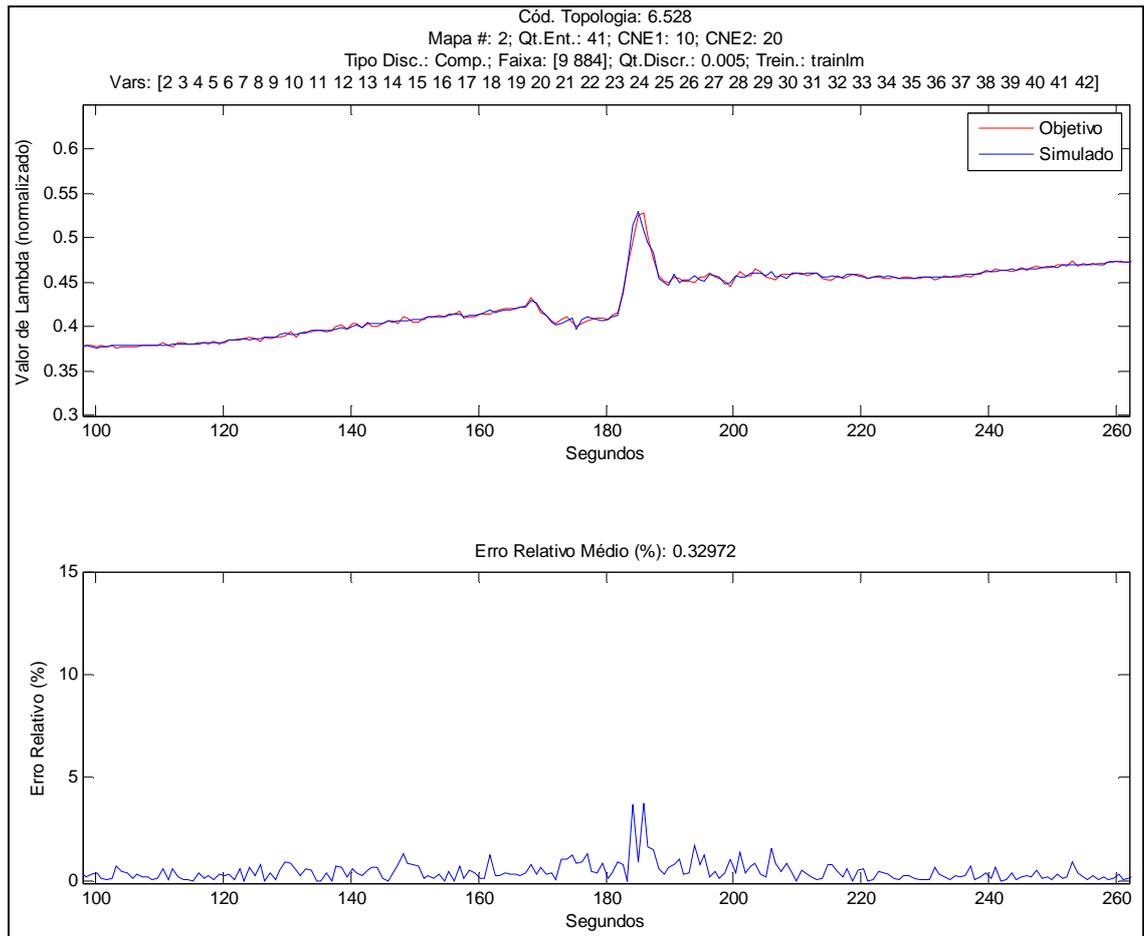


Figura 6.19: Exemplo de resultado de treinamento e teste com E100-Completo (Situação II).

A próxima situação (Figura 6.20) mostra novamente o comportamento irregular do valor desejado; desta vez, para um teste com todo o conjunto universo disponível e considerando combustível E100. Observa-se que a tendência é seguida levando-se em conta a precisão do valor de lambda. A rede é

composta de 35 neurônios na primeira camada, 41 variáveis de entrada e teve 3% de discretização dos dados para realizar os treinamentos e os testes.

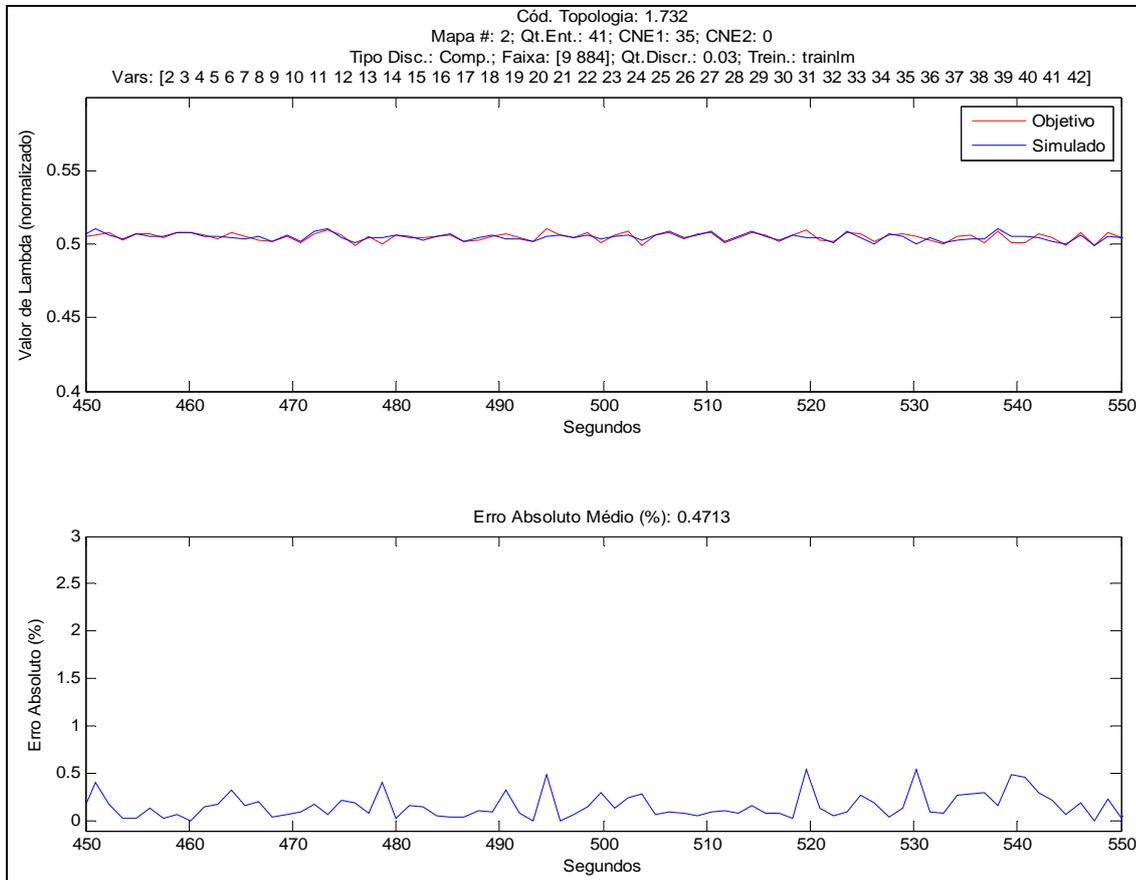


Figura 6.20: Exemplo de resultado de treinamento e teste com E100-Completo (Situação III).

### 6.2.3 - Treinamentos com Mapas Segmentados e Testes com Completos

Os resultados apresentados nesta seção visam analisar a aplicação das RNA, treinadas com segmentos dos mapas, em mapas de dados completos.

#### 6.2.3.1 - Teste com E20

Na Figura 6.21 é apresentado o melhor resultado de um teste com mapa completo utilizando uma rede treinada com a discretização segmentada. A rede foi composta de 6 entradas e 7 neurônios na primeira camada.

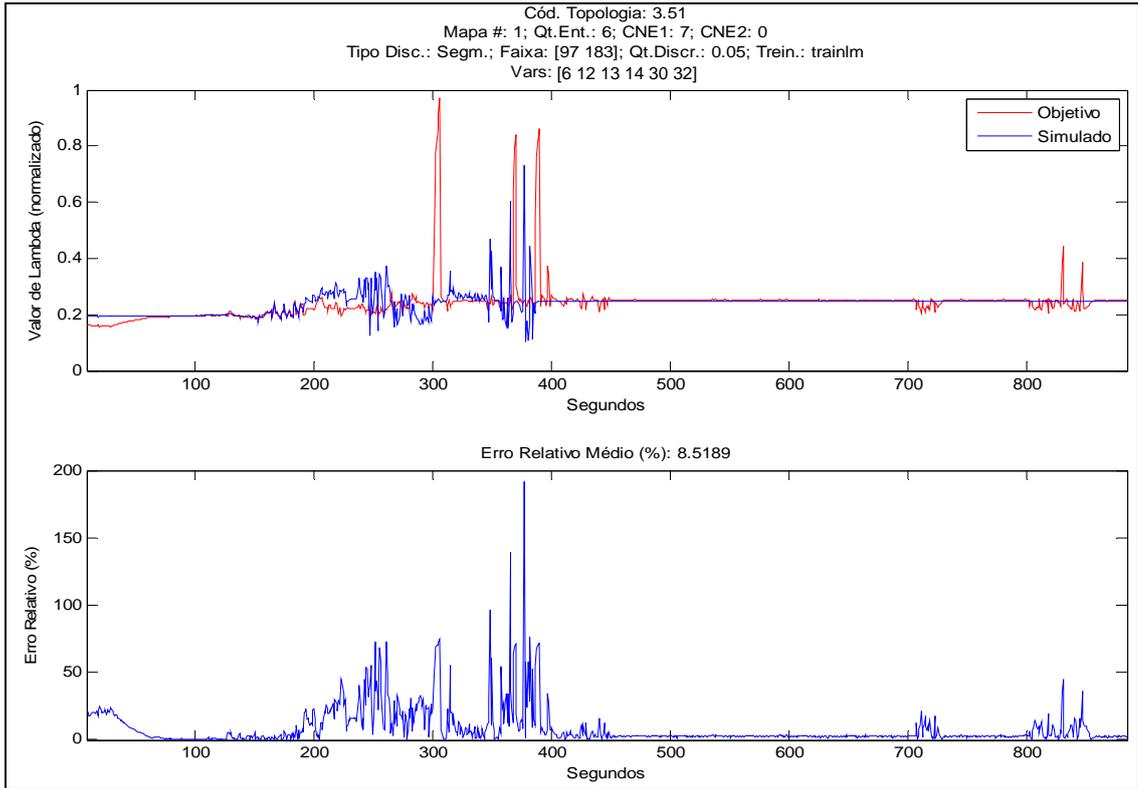


Figura 6.21: Exemplo de resultado invalidado de treinamento com E20-Segmentado e teste com E20-Completo (Situação I).

O treinamento original da rede utilizada no teste acima (Figura 6.21) é apresentado na Figura 6.22, onde foi utilizado o método *trainlm* e o segundo segmento de dados (97 a 183s), tendo-se  $\bar{e}_r$  de 1,454%.

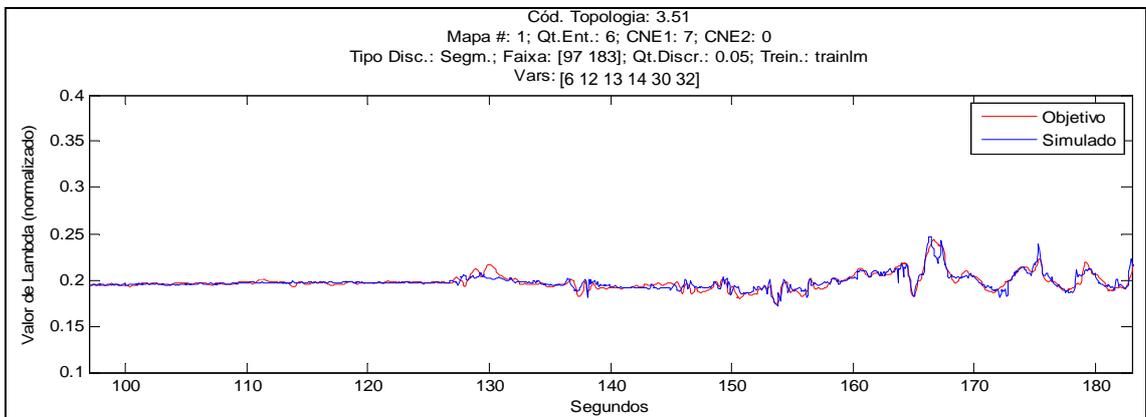


Figura 6.22: Exemplo de resultado invalidado de treinamento com E20-Segmentado e teste com E20-Completo (Situação II).

### 6.2.3.2 - Teste com E100

Para testes com E100, o melhor resultado foi obtido através da rede treinada com o último segmento de dados, tendo 5 entradas e 12 neurônios. O resultado da RNA em seu treinamento é apresentado na Figura 6.23.

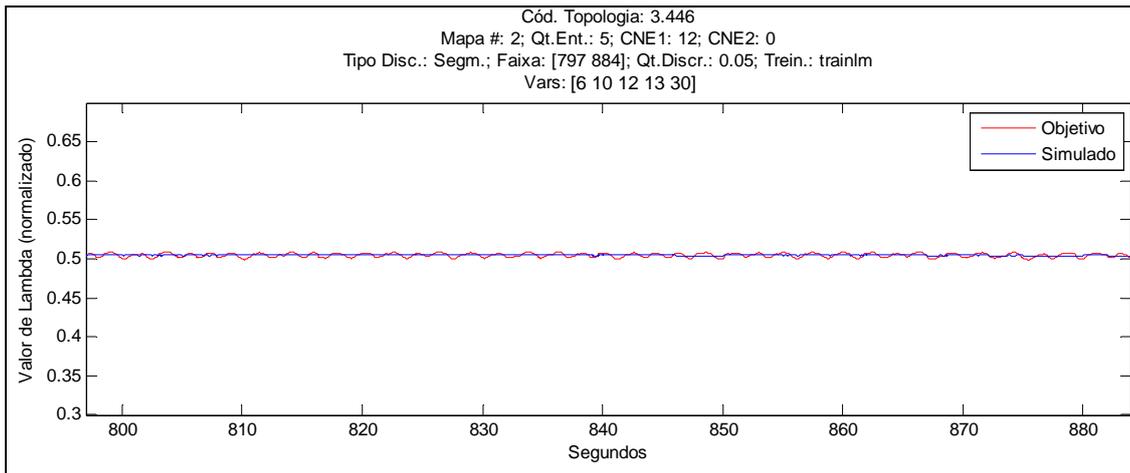


Figura 6.23: Exemplo de resultado invalidado de treinamento com E100-Segmentado e teste com E100-Completo (Situação I).

Na Figura 6.24 pode-se observar o comportamento da rede quando usado todo o mapa de dados, obtendo-se um  $\bar{e}_r$  de 2,3458.

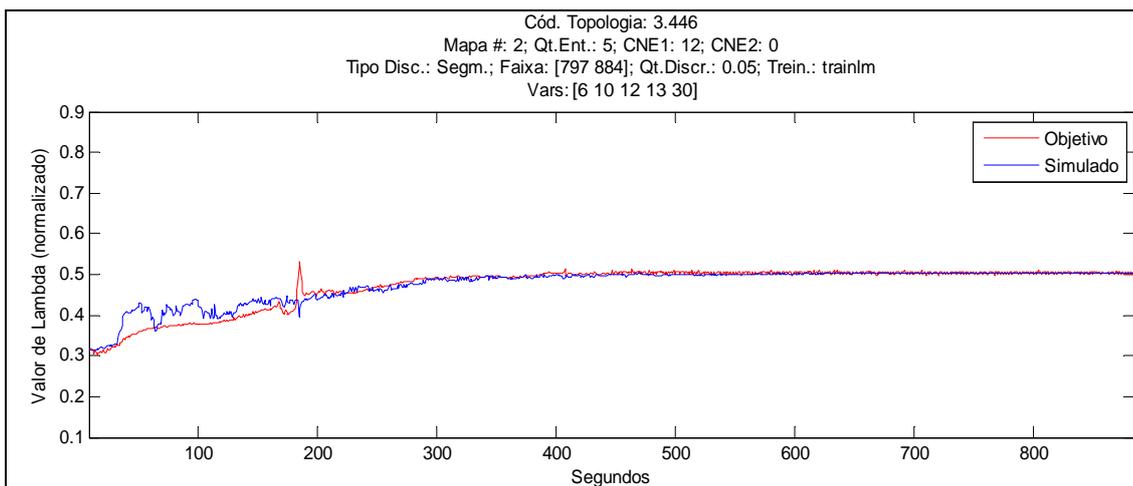


Figura 6.24: Exemplo de resultado invalidado de treinamento com E100-Segmentado e teste com E100-Completo (Situação II).

Nota-se que os resultados apresentados nesta Subseção, 6.2.3, não são tão satisfatórios como em seções anteriores.

### 6.3 - Resultados de Testes como Bicombustíveis

Esta seção tem o objetivo de analisar se alguma RNA treinada com um tipo de combustível pode ser usada por outro tipo de combustível.

#### 6.3.1 - Treinamentos com E20 e Testes com E100

##### 6.3.1.1 - Mapas Segmentados

A seguir (Figura 6.25) é apresentado um resultado de teste onde utilizou o combustível E100 em uma rede neural treinada com E20. Neste caso, a rede foi composta de 5 entradas e 10 neurônios na primeira camada neural. Observa-se que embora o erro relativo médio tenha sido próximo de 6%, a rede não obteve resultados satisfatórios.

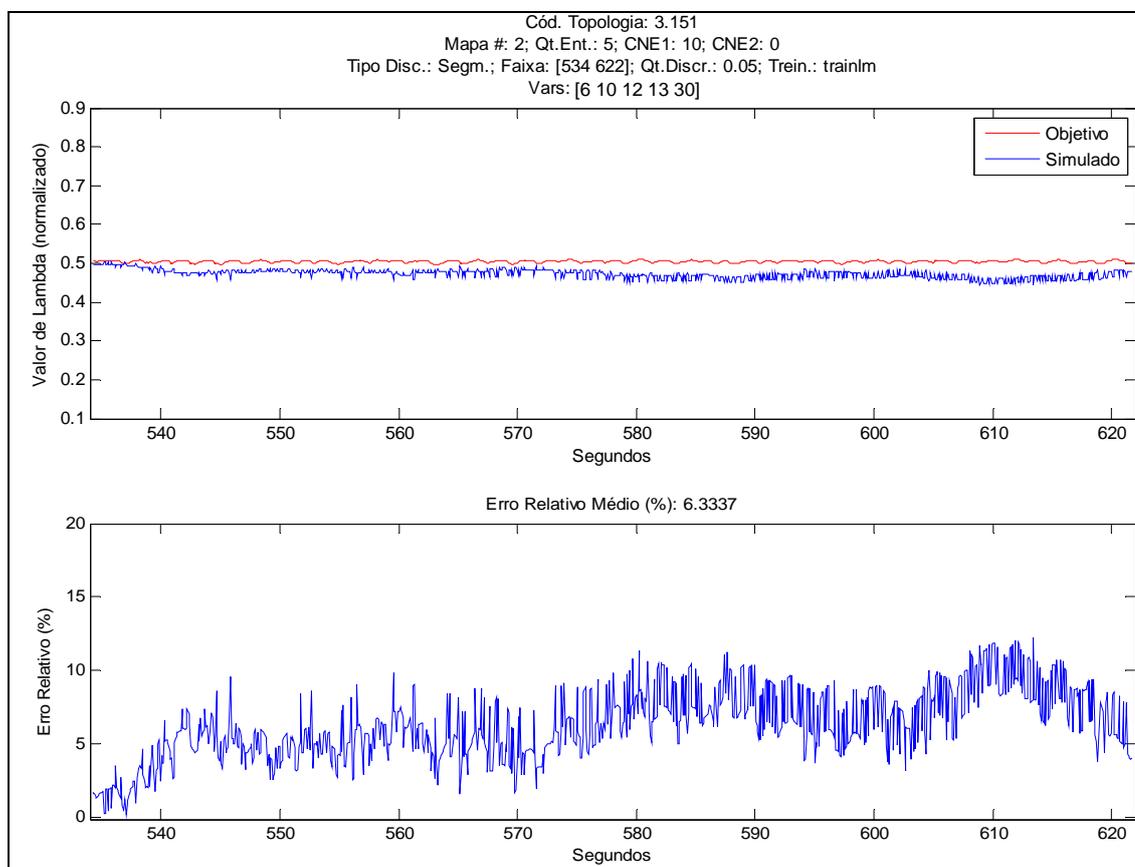


Figura 6.25: Exemplo de resultado invalidado de treinamento com E20-Segmentado e teste com E100-Segmentado.

### 6.3.1.2 - Mapas Completos

Na Figura 6.26 é apresentado um resultado de uma configuração usando 12 neurônios na primeira camada, 21 entradas e função treinamento *trainbr*.

Nota-se que o  $\overline{e_r}$  também foi alto (9,31%).

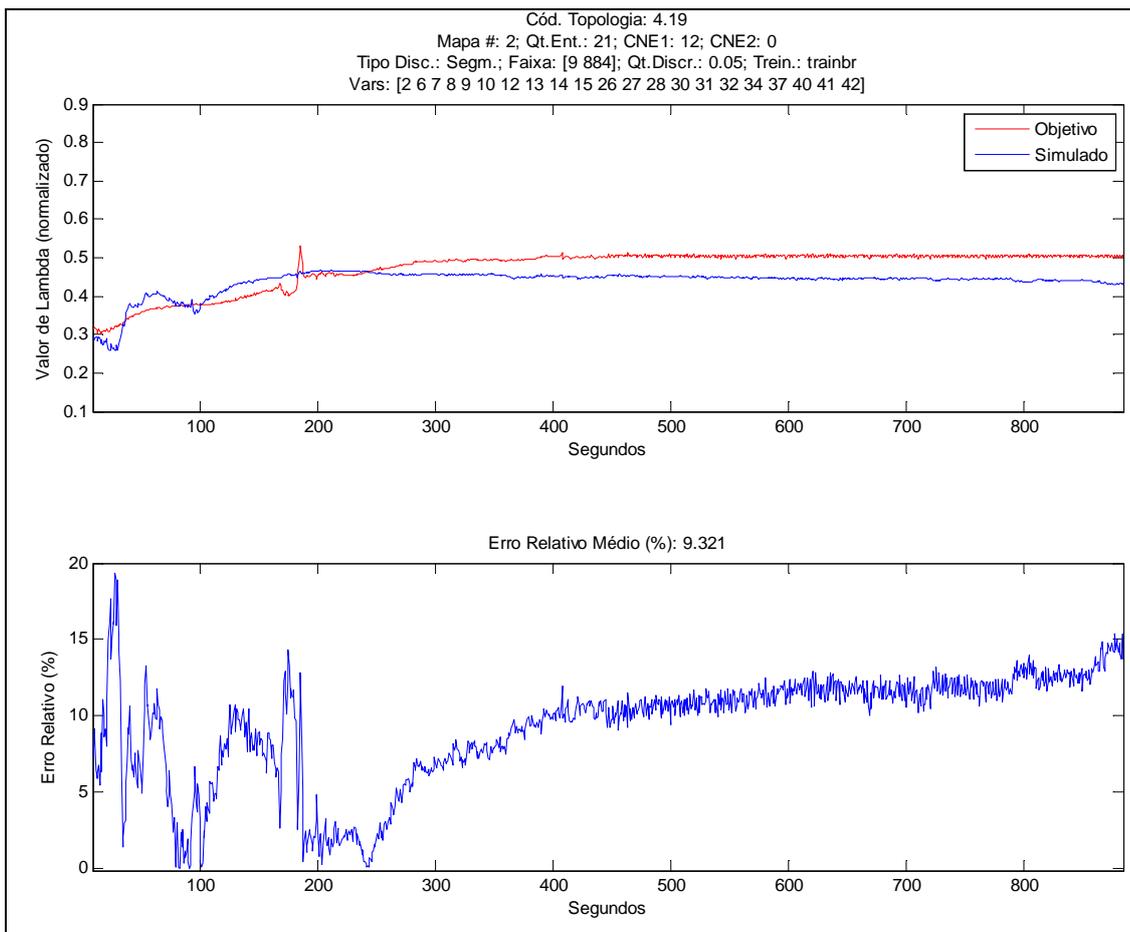


Figura 6.26: Exemplo de resultado invalidado de treinamento com E20-Completo e teste com E100-Completo.

### 6.3.2 - Treinamentos com E100 e Testes com E20

#### 6.3.2.1 - Mapas Segmentados

Na Figura 6.27 é apresentado um resultado de teste para E20 que utilizou RNA treinada com o tipo de discretização segmentada para E100. A RNA tinha 5 entradas, 10 neurônios na primeira camada, tendo sido treinada com o método *trainlm* e seu  $\bar{e}_r$  foi de 6,7487%.

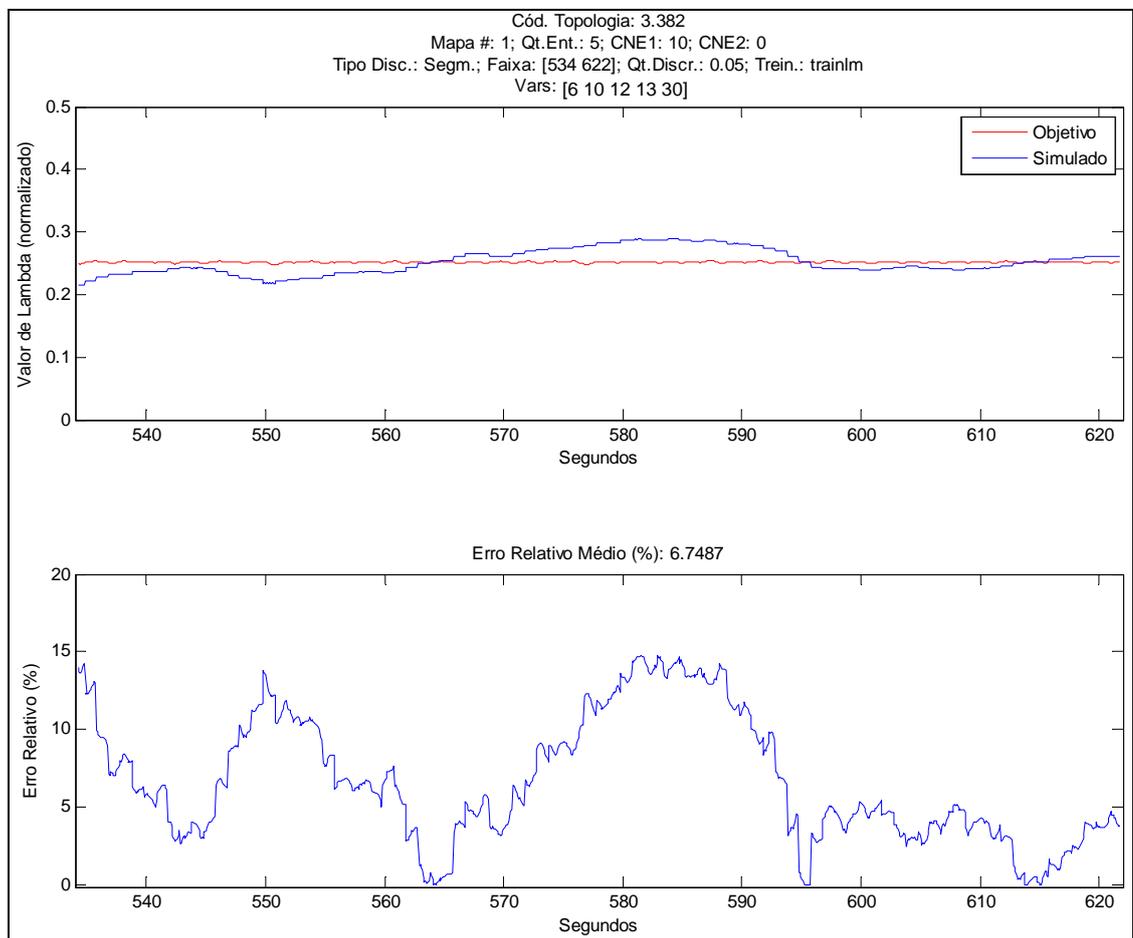


Figura 6.27: Exemplo de resultado inválido de treinamento com E100-Segmentado e teste com E20-Segmentado.

### 6.3.2.2 - Mapas Completos

Os resultados apresentados por redes treinadas com mapas completos do tipo E100 e depois testadas com E20, não apresentaram resultados satisfatórios. Como pode-se observar na Figura 6.28, a RNA composta por 5 neurônios na primeira camada e 5 entradas não conseguiu convergir os resultados para outro tipo de combustível.

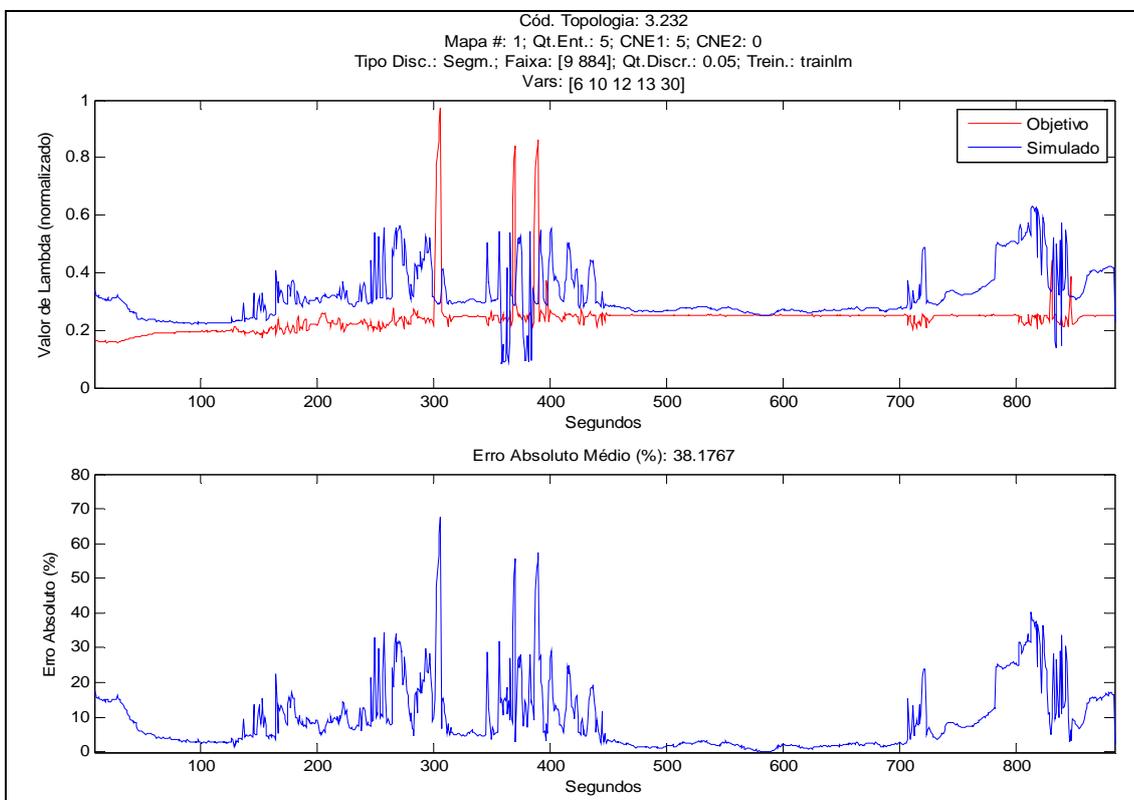


Figura 6.28: Exemplo de resultado invalidado de treinamento com E100-Completo e teste com E20-Completo.

Como pode ser visto nas Figuras 6.25 até 6.28, os treinamentos realizados com apenas um tipo de combustível não foram capazes de ajusta eficazmente as redes para que pudessem ser aplicadas a outro tipo de combustível.

Este comportamento pode ser explicado pela alteração da dinâmica do motor que ocorre quando são utilizadas diferentes proporções de combustível (Capítulos 2 e 3).

### 6.3.3 - Treinamentos com E20 e E100 e Testes com E20

Objetivando realizar um mapeamento mais abrangente e eficaz, as seções seguintes abordam treinamentos realizados com duas proporções de combustíveis (E20 e E100). Esta abordagem baseia-se na metodologia utilizada para calibrar motores (Seção 2.3), onde todas as condições de funcionamento do motor devem ser analisadas, permitindo criar mapeamentos mais completos.

#### 6.3.3.1 - Mapas Segmentados

É apresentado, na Figura 6.29, o melhor dos testes realizados com o último segmento dos dados do mapa 1 (E20). Esta topologia contém 41 entradas, 20 neurônios na primeira camada, 15 na segunda camada, sendo que se utilizou o método *trainlm* e alcançou um  $\bar{e}_r$  de 0,287%.

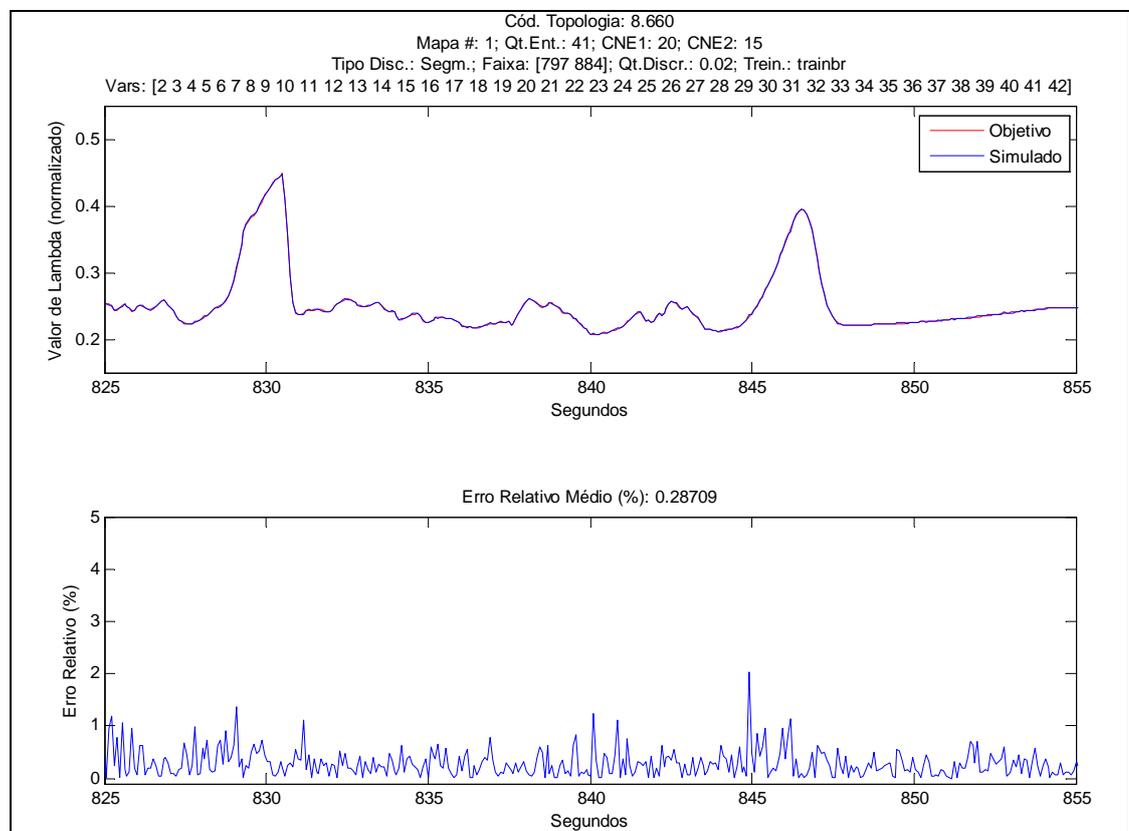


Figura 6.29: Exemplo de resultado de treinamento com E20 e E100-Segmentado e teste com E20-Segmentado (Situação I).

Já na Figura 6.30, é apresentado um exemplo de topologia treinada com ambos os mapas e posteriormente testada com o Mapa 1. Com 21 variáveis de entrada, 20 neurônios na primeira camada, 15 na segunda e treinamento *trainbr*, observou-se que esta topologia obteve resultados melhores que topologias que usaram apenas o mapa de dados 1 em seu treinamento, tendo  $\overline{e_r}$  igual 0,3737% contra  $\overline{e_r}$  de 1,3328 (Tabela 6.1, topologia 3-Nr126).

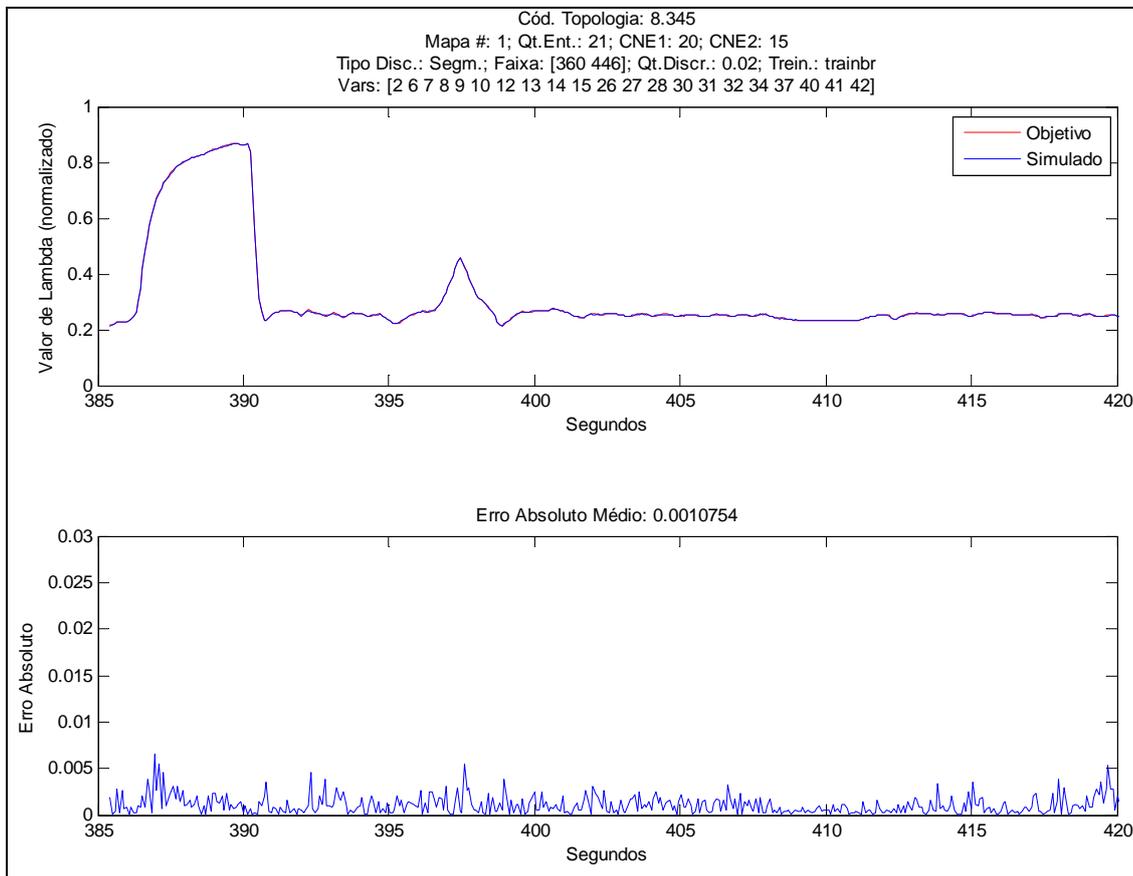


Figura 6.30: Exemplo de resultado de treinamento com E20 e E100-Segmentado e teste com E20-Segmentado (Situação II).

### 6.3.3.2 - Mapas Completos

Um exemplo de resultado validado, com  $\overline{e_r}$  de 2,8442%, é apresentado na Figura 6.31, tendo a RNA 21 entradas e 10 neurônios em cada camada.

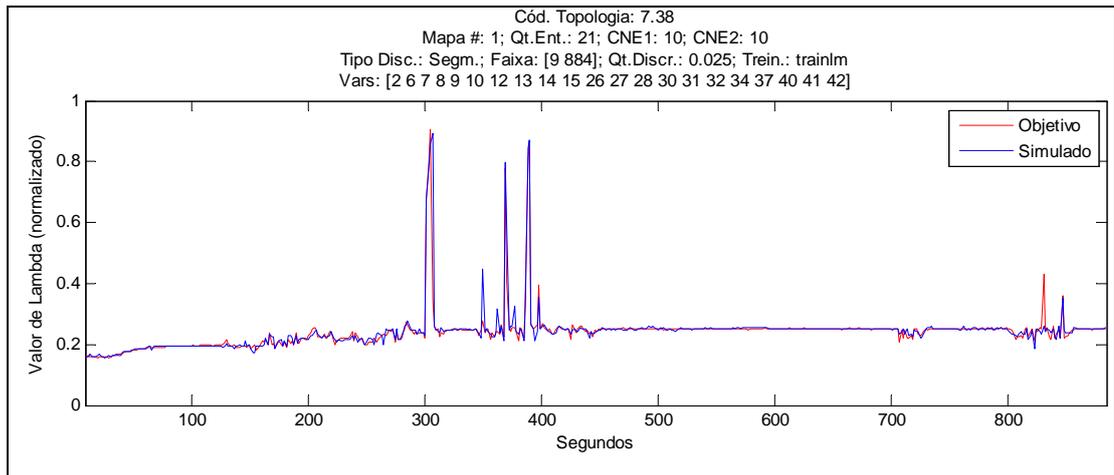


Figura 6.31: Exemplo de resultado de treinamento com E20 e E100-Completo e teste com E20-Completo (Situação I).

Na Figura 6.32 é apresentado o melhor resultado para testes de E20 usando RNA com treinamentos mistos (ambos os mapas). Nesta topologia a rede foi composta de 41 entradas, 5 neurônios na primeira camada neural e 15 na segunda, obtendo  $\overline{e_r}$  de 2,3994%.

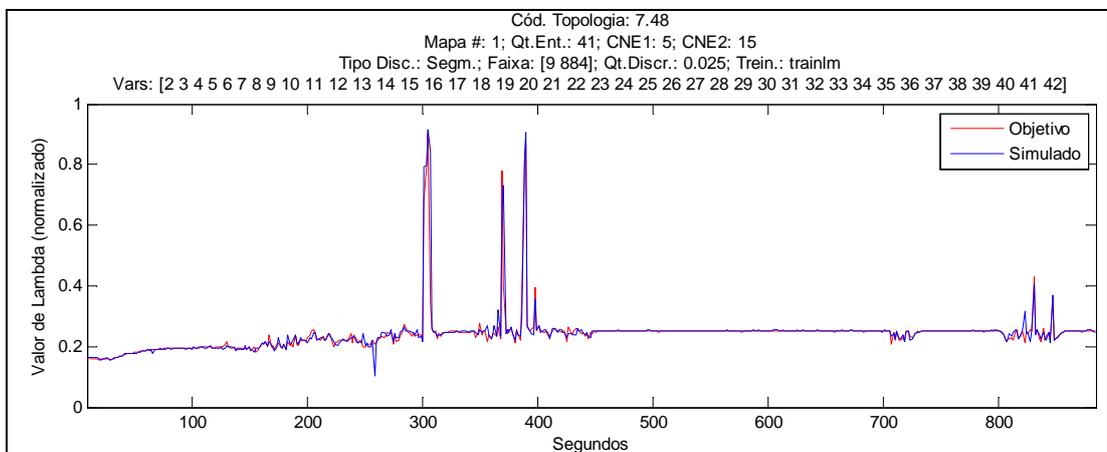


Figura 6.32: Exemplo de resultado de treinamento com E20 e E100-Completo e teste com E20-Completo (Situação II).

### 6.3.4 - Treinamentos com E20 e E100 e Testes com E100

#### 6.3.4.1 - Mapas Segmentados

Como ilustrado anteriormente, o comportamento típico do sensor de oxigênio deve ser oscilante e este exemplo (Figura 6.33) ilustra que o comportamento foi conservado para esta topologia treinada com ambos os mapas e sendo testada com o mapa E100.

A RNA teve  $\bar{e}_r$  de 0,17441% neste segmento de dados, tendo 41 variáveis de entrada, 20 neurônios na primeira camada neural escondida, 5 na segunda camada neural, sendo treinada com o método *trainlm*.

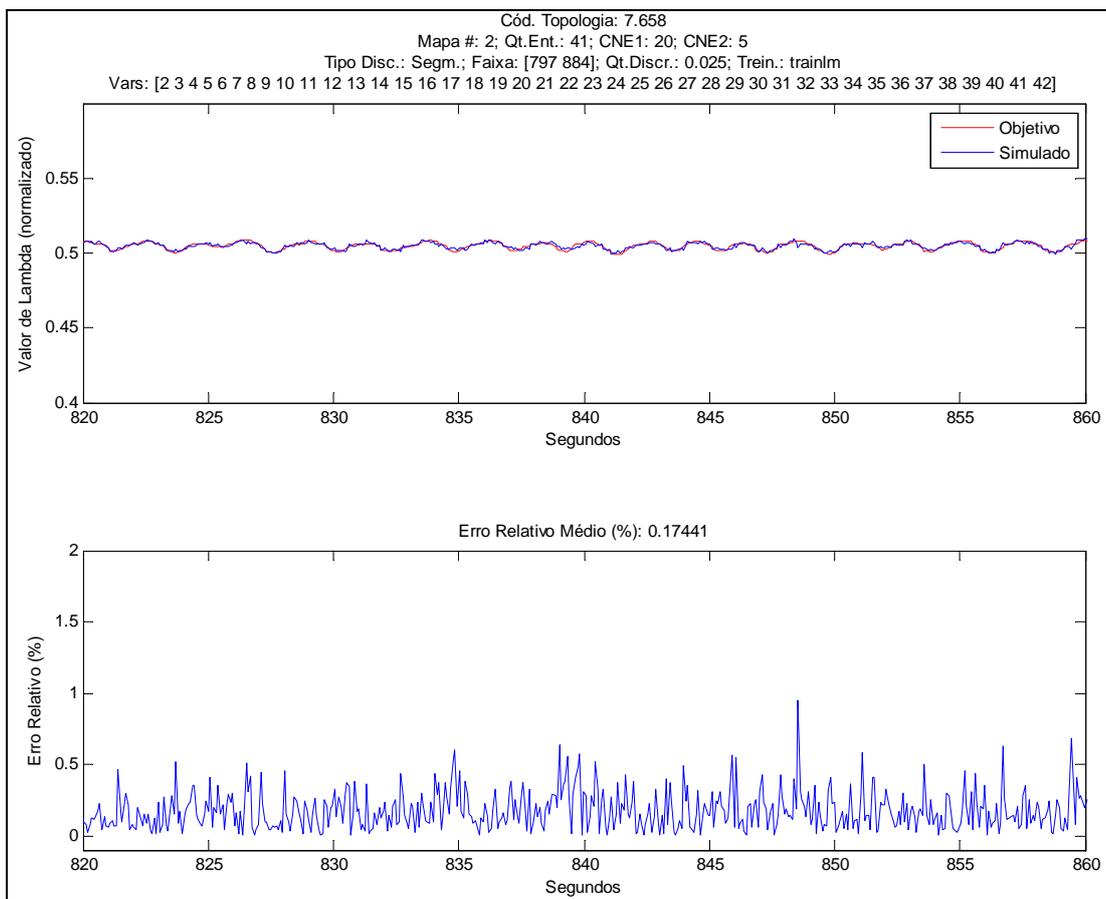


Figura 6.33: Exemplo de resultado de treinamento com E20 e E100-Segmentado e teste com E100-Segmentado (Situação I).

No gráfico seguinte (Figura 6.34) é apresentado um exemplo de resultado para o quarto segmento do conjunto de dados (272 até 359s), onde havia na rede 20 e 5 neurônios, na primeira e segunda camadas, respectivamente, sendo treinada pelo método *trainbr*, com 41 entradas e tendo  $\bar{e}_r$  de 0,12252%.

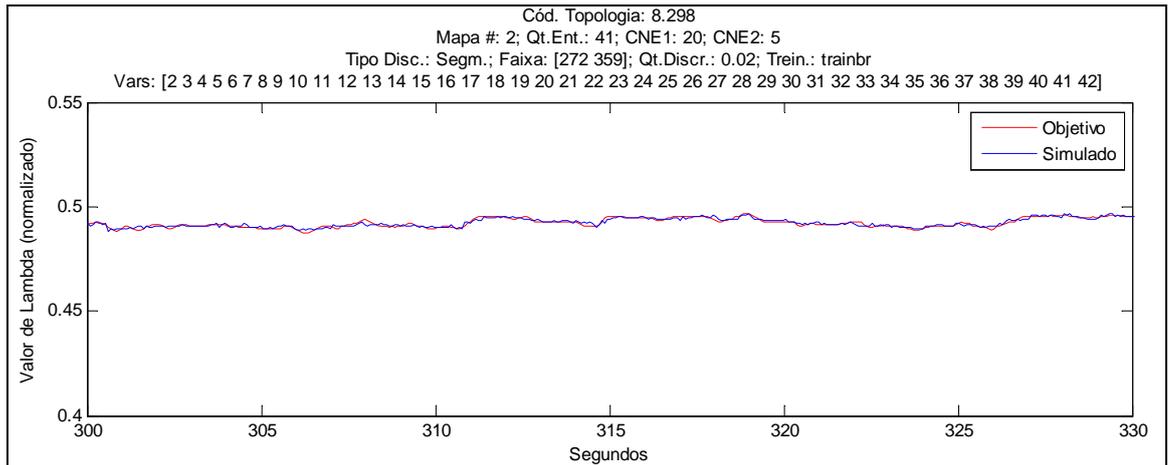


Figura 6.34: Exemplo de resultado de treinamento com E20 e E100-Segmentado e teste com E100-Segmentado (Situação II).

#### 6.3.4.2 - Mapas Completos

Um resultado validado para testes do mapa 2 usando topologias que foram treinadas com ambos os mapas é apresentado na Figura 6.35. Neste exemplo a RNA foi composta de 6 variáveis de entrada, 20 neurônios na primeira camada e 15 na segunda, obtendo-se  $\bar{e}_r$  de 0,51907% com o método *trainbr*.

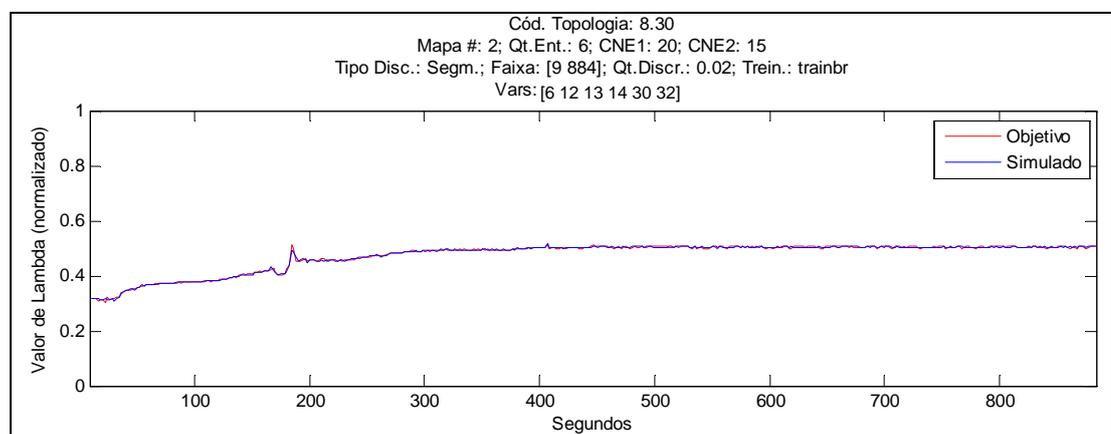


Figura 6.35: Exemplo de resultado de treinamento com E20 e E100-Completo e teste com E100-Completo (Situação I).

Na Figura 6.36 é apresentado o melhor resultado para testes de E100 com topologias de treinamento misto (utilizando ambos os mapas), obtendo erros relativos médios muito próximos dos melhores resultados para este mapa de dados, com uma diferença de 0,06% no erro relativo médio.

Neste caso, a rede PMC foi composta de 15 neurônios em cada camada, 41 variáveis, e usando-se o método *trainlm* como algoritmo de treinamento.

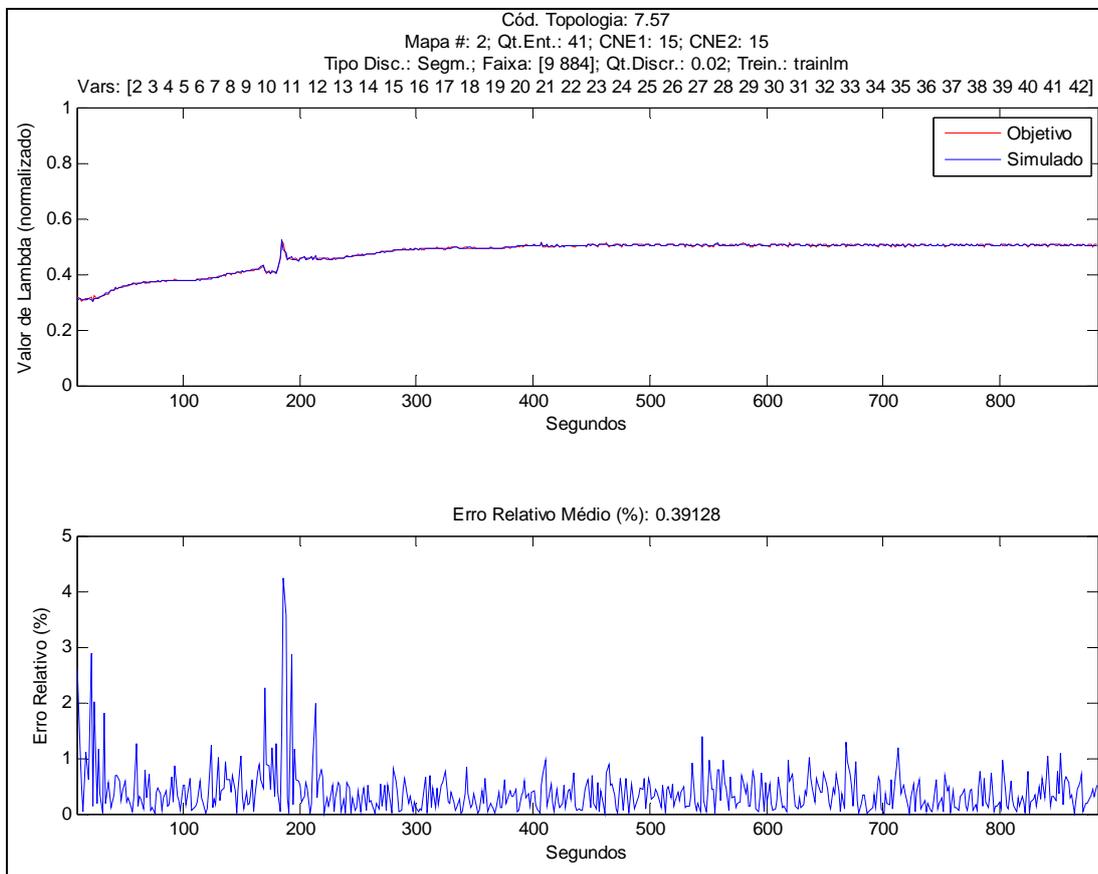


Figura 6.36: Exemplo de resultado de treinamento com E20 e E100-Completo e teste com E100-Completo (Situação II).

### 6.3.5 - Treinamentos com Mapas Segmentados e Testes com Completos

Esta seção visa analisar a aplicação das RNA, treinadas com segmentos de ambos os mapas, em mapas de dados completos.

#### 6.3.5.1 - Teste com E20

No gráfico seguinte (Figura 6.37) é apresentado o melhor resultado, invalidado, de um teste com mapa completo utilizando uma rede treinada com a discretização segmentada de ambos os mapas disponíveis (E20 e E100). A rede foi composta de 21 variáveis de entrada, 5 neurônios na primeira camada, 15 na segunda camada e obteve  $\bar{e}_r$  de 9,8449%.

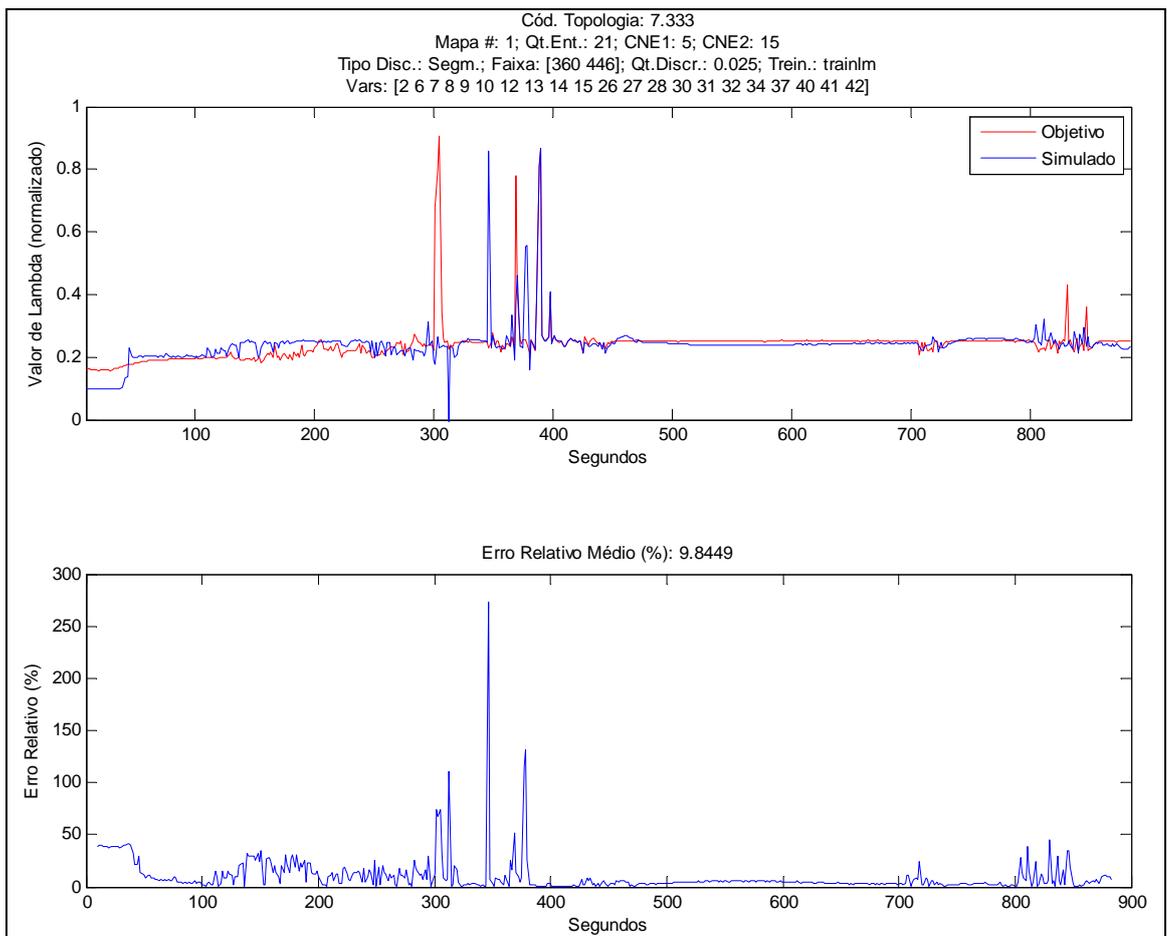


Figura 6.37: Exemplo de resultado invalidado de treinamento com E20 e E100-Segmentado e teste com E20-Completo.

## 6.3.5.2 - Teste com E100

Na Figura 6.38 é mostrado o melhor resultado, invalidado, de um teste com mapa completo utilizando uma rede treinada com a discretização segmentada de ambos os mapas disponíveis (E20 e E100). A rede foi composta de 21 variáveis de entrada, 5 neurônios na primeira camada, 15 na segunda camada e obteve  $\overline{e_r}$  de 1,9154%.

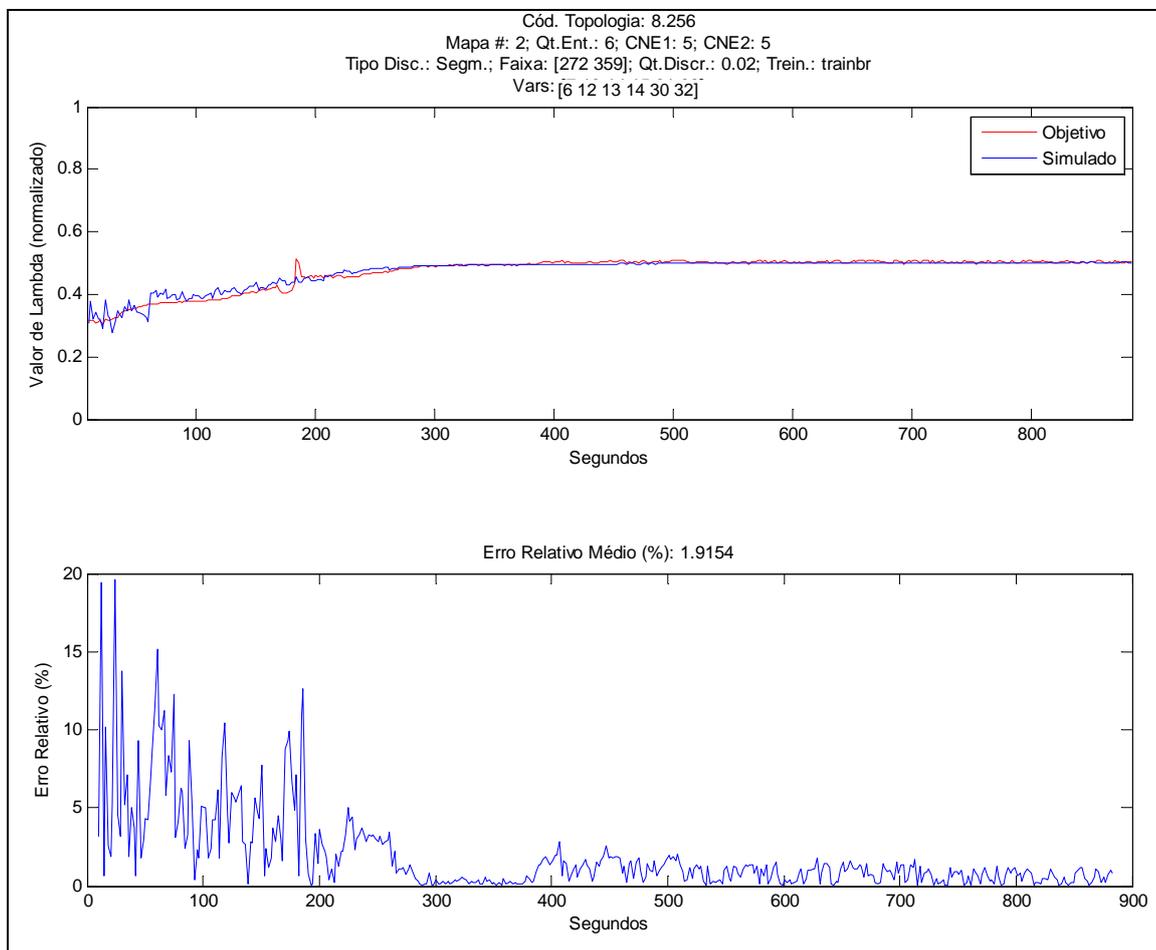


Figura 6.38: Exemplo de resultado invalidado de treinamento com E20 e E100-Segmentado e teste com E100-Completo.

Pôde-se observar que, assim como na Subseção 6.2.3, os resultados apresentados aqui não são também satisfatórios.

### 6.3.6 - Testes de redes com topologias iguais para E20 e E100

Esta seção apresenta resultados de topologias iguais testadas com ambos os tipos de combustíveis (E20 e E100), objetivando identificar topologias aplicáveis a ambos. Ao todo foram identificadas 58 topologias que obtiveram  $\overline{e_r}$  menores que 5% em ambos os mapas de dados e para todo o conjunto de dados.

Nas Figuras 6.39 é apresentado o melhor resultado alcançado por uma topologia para o primeiro mapa de dados, já na Figura 6.40 é apresentado o resultado correspondente à topologia da Figura 6.39 aplicado ao segundo mapa de dados. A Figura 6.39 teve como configuração em sua topologia, 41 entradas, 5 neurônios na primeira camada, 15 na segunda camada neural escondida e método de treinamento *trainlm*; como resultado, teve-se  $\overline{e_r}$  igual a 2,3994%.

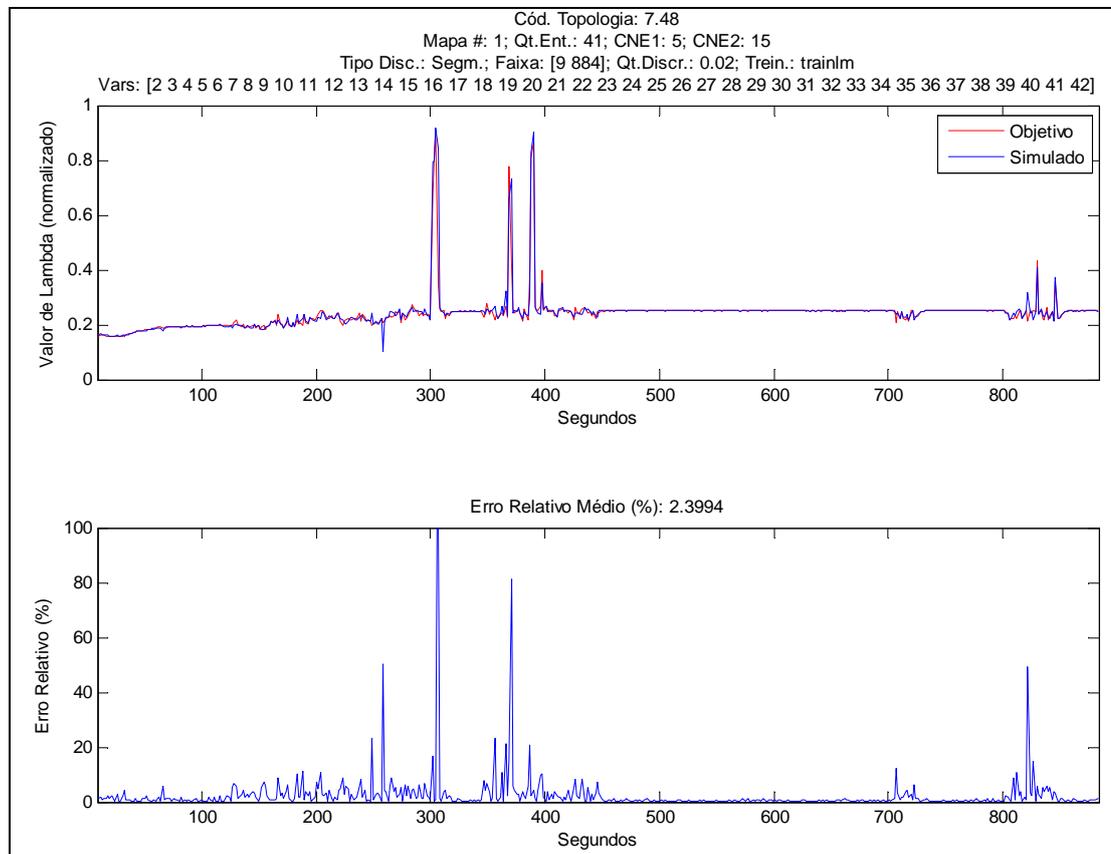


Figura 6.38: Exemplo de resultado de teste com topologias iguais - E20-Completo (Situação I).

A Figura 6.40 teve as mesmas configurações que o resultado anterior (Figura 6.38); porém, utilizou-se o mapa de dados E100 e seu  $\overline{e_r}$  foi de 0,54709%.

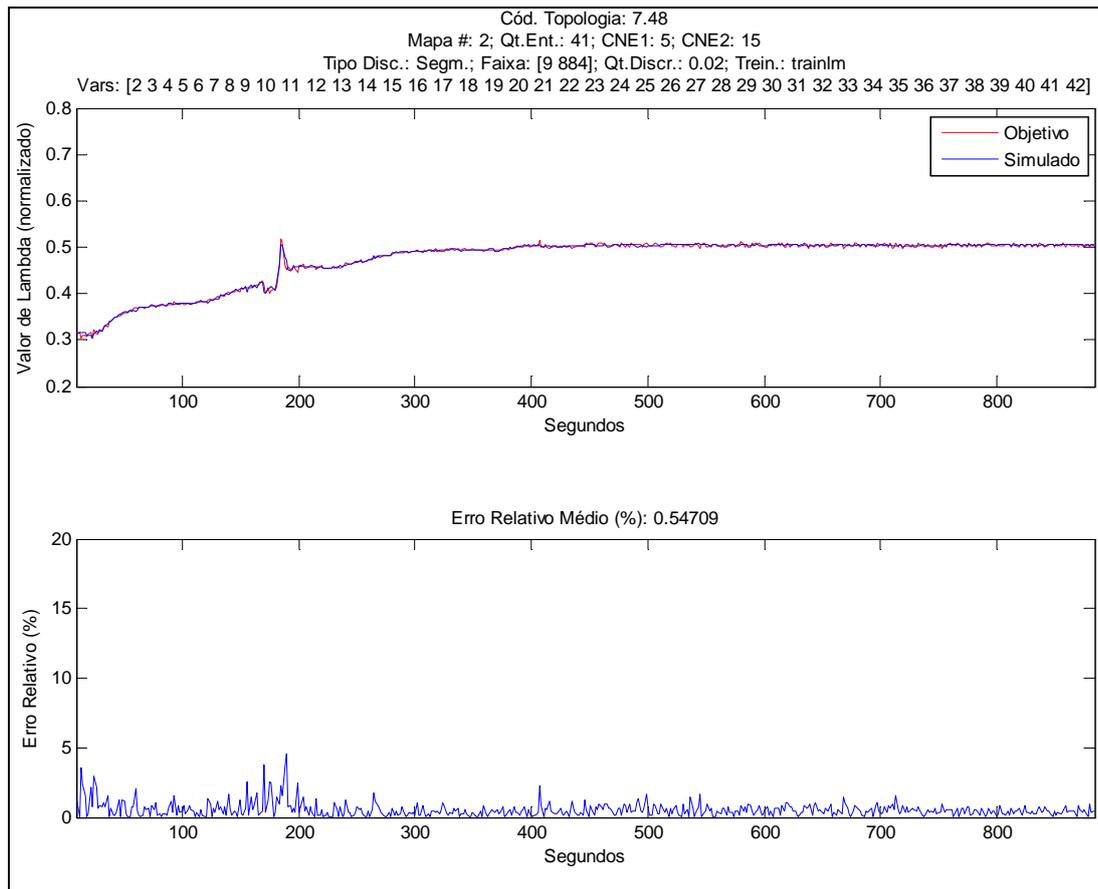


Figura 6.40: Exemplo de resultado de teste com topologias iguais – E100-Completo (Situação I).

Os exemplos apresentados nas Figuras 6.41 e 6.42 representam, respectivamente, a melhor topologia para o mapa 2 (Figura 6.42) e o resultado da topologia aplicada ao mapa 1 (Figura 6.41).

A topologia foi composta de 15 neurônios na primeira camada neural, 10 na segunda camada, 21 entradas e método de treinamento *trainbr*. A Figura 6.41, utilizando o mapa E20, obteve  $\overline{e_r}$  de 4,2301%.

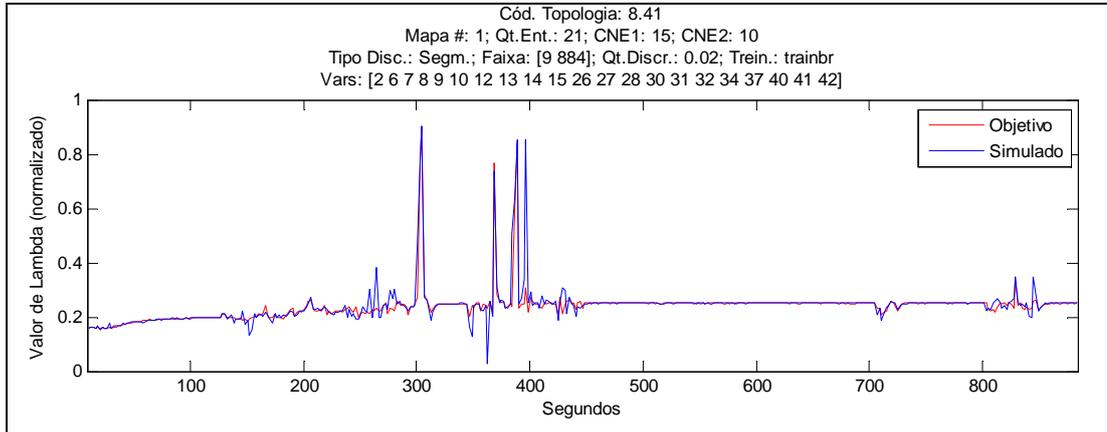


Figura 6.41: Exemplo de resultado de teste com topologias iguais - E20-Completo (Situação II).

Já na Figura 6.42, utilizando o segundo mapa de dados (E100), resultou-se em  $\overline{e_r}$  igual a 0,45096%.

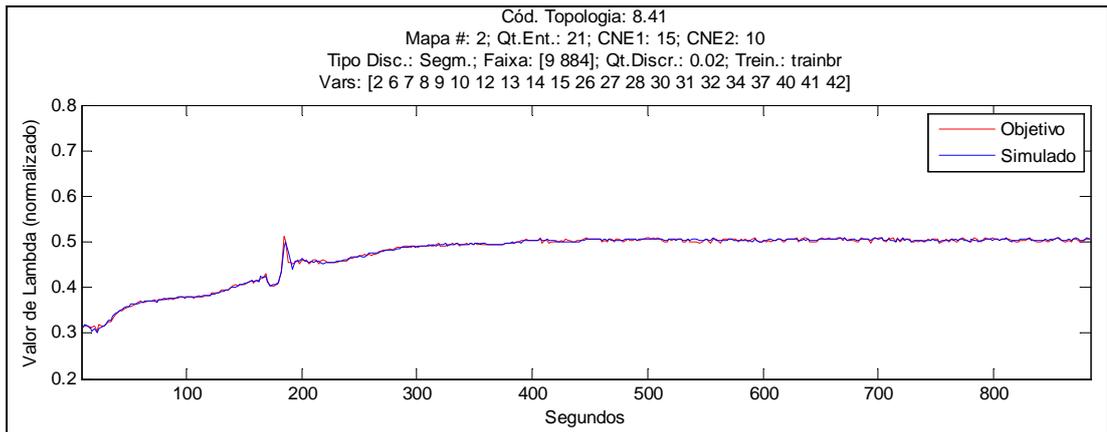


Figura 6.42: Exemplo de resultado de teste com topologias iguais – E100-Completo (Situação II).

## 6.4 - Síntese dos Resultados de Testes

A seguir serão apresentados, em forma de tabelas, resumos dos desempenhos dos testes realizados, considerando-se seus erros relativos médios.

A Tabela 6.5 apresenta a quantidade de RNA, resultante dos testes realizados, em cada faixa de erro relativo médio para as topologias que tiveram o mesmo mapa de dados no treinamento e testes. Torna-se possível observar a elevada concentração de topologias com erros menores que 5%, em especial dos testes com discretização segmentada.

Tabela 6.5: Resumo dos resultados de testes – Treinamentos e Testes utilizando mesmos mapas de dados.

Treinamento		Teste		Erros Relativos Médios (Qtde RNA)								Total
Nro Mapa	Tipo Discr.	Nro Mapa	Tipo Discr.	<0,5%	<1%	<3%	<5%	<10%	<25%	<50%	>=50%	
1	Comp.	1	Comp.	-	-	302	192	346	147	23	4	1.014
2	Comp.	2	Comp.	570	348	95	-	1	-	-	-	1.014
1	Segm.	1	Segm.	272	162	190	69	37	20	-	-	750
2	Segm.	2	Segm.	564	158	28	-	-	-	-	-	750

Já na Tabela 6.6, pode-se observar que não há resultados satisfatórios, menores que 5%. Esta tabela apresenta testes realizados com RNA treinadas com mapas distintos, resultando-se em mapeamentos insatisfatórios.

Tabela 6.6: Resumo dos resultados de testes – Treinamentos e Testes utilizando mapas de dados distintos.

Treinamento		Teste		Erros Relativos Médios (Qtde RNA)								Total
Nro Mapa	Tipo Discr.	Nro Mapa	Tipo Discr.	<0,5%	<1%	<3%	<5%	<10%	<25%	<50%	>=50%	
1	Comp.	2	Comp.	-	-	-	-	1	6	97	910	1.014
1	Segm.	2	Segm.	-	-	-	-	8	33	75	634	750
2	Comp.	1	Comp.	-	-	-	-	-	-	1	1.013	1.014
2	Segm.	1	Segm.	-	-	-	-	2	4	18	726	750

Na Tabela 6.7, verifica-se que a utilização de treinamentos com discretização segmentada não é recomendada para testes com todo o conjunto de dados, tanto para conjuntos de treinamentos e testes iguais, como para distintos.

Tabela 6.7: Resumo dos resultados de testes – Treinamentos e Testes utilizando tipos de discretização ou mapas de dados distintos.

Treinamento		Teste		Erros Relativos Médios (Qtde RNA)								Total
Nro Mapa	Tipo Discr.	Nro Mapa	Tipo Discr.	<0,5%	<1%	<3%	<5%	<10%	<25%	<50%	>=50%	
1	Segm.	1	Comp.	-	-	-	-	1	52	34	663	750
1	Segm.	2	Comp.	-	-	-	-	-	10	39	701	750
2	Segm.	1	Comp.	-	-	-	-	-	-	-	750	750
2	Segm.	2	Comp.	-	-	3	-	52	92	81	522	750
1+2	Segm.	1	Comp.	-	-	-	-	1	9	117	1.073	1.200
1+2	Segm.	2	Comp.	-	-	2	5	59	144	171	819	1.200

A Tabela 6.8 mostra que há a possibilidade de utilizar apenas uma topologia de RNA para ambos os tipo de combustíveis, E20 ou E100, respectivamente, Mapa 1 e Mapa 2 (coluna Teste). Porém, o treinamento deve envolver ambos os mapas, a fim de generalizar mais eficazmente. Nota-se nas topologias com discretização segmentada, um aumento de 16% no número de RNA do mapa 1 e 2% do mapa 2, que estão com  $\overline{e_r}$  menores que 1%.

Tabela 6.8: Resumo dos resultados de testes – Treinamentos utilizando ambos os mapas de dados e testes utilizando apenas um dos mapas.

Treinamento		Teste		Erros Relativos Médios (Qtde RNA)								Total
Nro Mapa	Tipo Discr.	Nro Mapa	Tipo Discr.	<0,5%	<1%	<3%	<5%	<10%	<25%	<50%	>=50%	
1+2	Comp.	1	Comp.	-	-	6	52	56	5	-	1	120
1+2	Comp.	2	Comp.	19	71	29	1	-	-	-	-	120
1+2	Segm.	1	Segm.	601	205	195	62	51	58	21	7	1.200
1+2	Segm.	2	Segm.	1012	170	15	1	1	-	1	-	1.200



## Capítulo 7 - Conclusões Gerais e Trabalhos Futuros

### 7.1 - Conclusões Gerais

O estudo do sensor virtual de oxigênio usando redes neurais PMC forneceu um resultado satisfatório, dado a complexidade de um sistema motor, atingindo-se erros relativos médios menores que 1% em centenas de topologias testadas.

A respeito de arquiteturas utilizadas como monocombustíveis, ou seja, utilizando o mesmo mapa de dados para treinamentos e testes, verificou-se que há a possibilidade de utilizar topologias de redes únicas para todo o conjunto de dados, embora utilizar topologias distintas para cada segmento no conjunto de dados obteve melhores resultados. Utilizando treinamentos completos obteve-se mais de 300 topologias que atingiram  $\bar{e}_r$  menores que 3% para o mapa de dados 1 (E20), e 570 topologias com  $\bar{e}_r$  menor que 1% para o mapa de dados 2 (E100), sendo 1.014 testes para cada mapa. Já nos treinamentos segmentados, mais de 400 topologias de ambos os mapas atingiram  $\bar{e}_r$  menor que 0,5%, tendo-se 1.500 testes ao todo. Comparando-se, único e simplesmente, os resultados apresentados por Kamat et al. (2006) com aqueles deste trabalho, pode-se concluir que os resultados aqui apresentados foram significativamente melhores. Kamat et al. (2006) apresentaram erros absolutos médios entre 0,0331 e 0,0519, enquanto os

melhores resultados neste trabalho foram de erros absolutos médios entre 0,0015 e 0,0065, para testes com todo o universo de dados.

Pode-se concluir que a utilização de pequenos segmentos de dados em treinamentos não é eficaz no mapeamento dos pesos sinápticos da rede neural artificial quando testados com conjuntos maiores de dados. Assim como não é também eficaz o treinamento da rede com apenas um mapa de dados e, posteriormente, utilizar com outro mapa de dados, corroborando a alteração da dinâmica no motor quando submetido a diferentes proporções de combustíveis.

Já as arquiteturas que foram submetidas a treinamentos com ambos os tipos de combustíveis e, posteriormente, testadas com cada mapa, ou seja, como se fossem bicombustíveis, obtiveram resultados próximos as das arquiteturas monocombustíveis. De fato, aumentou-se em 16% o número de RNA, com discretização segmentada do mapa 1, na faixa de erros relativos médios menores que 1%.

Utilizando treinamentos segmentados, mais de 600 topologias do mapa 1 (E20) e mais de 1.000 do mapa 2 (E100) atingiram  $\bar{e}_r$  menores que 0,5%, de um total de 2.400 testes. Já nos treinamentos completos, o número de topologias que atingiram  $\bar{e}_r$  menores que 5% para o mapa de dados 1 foi de apenas 58, de um total de 120; e para o mapa 2, o número de RNA que atingiram  $\bar{e}_r$  menores que 1% foi de 90, sendo 120 ao todo. Dentre as 1.320 topologias “bicombustíveis” testadas em ambos os mapas completos, quase 4,5% delas tiveram  $\bar{e}_r$  menores que 5% em ambos os mapas.

Deste modo, pode-se concluir que para o sensoriamento virtual de oxigênio em veículos bicomustíveis, a abordagem de realizar treinamentos com todos os tipos de combustíveis, segmentando conjuntos de todo o universo de dados, mostra-se a mais adequada e possível.

## 7.2 - Trabalhos Futuros

Como sugestão para trabalhos futuros, pode-se ampliar a gama de mapas de dados, incluindo situações intermediárias referentes à utilização do veículo (cidade e rodovia), e à situação do motor (novo ou usado). Outra frente de investigação é identificar quais critérios utilizar para segmentar os mapas de dados.

Pesquisar a aplicação de redes neuro-fuzzy do tipo ANFIS é também uma linha de atuação promissora. Para tanto, deve-se identificar quais seriam as melhores variáveis de entrada a fim de reduzir a dimensionalidade.

Finalmente, a implementação em hardware das melhores arquiteturas de redes neurais treinadas dentro do sistema de gerenciamento de motor do veículo pesquisado é também uma das frentes de pesquisa a ser desenvolvida.



## Referências Bibliográficas

- AMATO, P.; CESARIO, N.; MEGLIO, M. Di; PIROZZI, F. **Realization of a Virtual Lambda Sensor on a Fixed Precision System**. Proceedings of the Design, Automation and Test in Europe, IEEE, 2005.
- AMEY, R. G. Automotive component innovation: development and diffusion of engine management technologies. Technovation, Elsevier, 1995.
- ANFAVEA. **Carta da ANFAVEA - 272**. Associação Nacional dos Fabricantes de Veículos Automotores, São Paulo, 2009.
- ANFAVEA. **Anuário da Indústria Automobilística Brasileira – 2008**. Associação Nacional dos Fabricantes de Veículos Automotores, São Paulo, 2008.
- ATKINSON, C. M.; LONG, T. W.; HANZEVACK, E. L. Virtual Sensing: A Neural Network-based Intelligent Performance and Emissions Prediction System for On-Board Diagnostics and Engine Control. Diagnostics and Controls, SAE, Michigan, 1998.
- BOSCH. **Press Release: Desenvolvimento de “tecnologia limpa” é objetivo da Bosch**, 2006. Disponível em: <[http://www.bosch.com.br/Imprensa/Shared/Documents/Release\\_1282/20%20anos%20do%20Proconve%20-%20tecnologia%20limpa.doc](http://www.bosch.com.br/Imprensa/Shared/Documents/Release_1282/20%20anos%20do%20Proconve%20-%20tecnologia%20limpa.doc)>. Acesso em: 30 jun 2008.
- BOSCH. Manual de Tecnologia Automotiva. Edgard Blucher, 2005.
- BOSCH. Gasoline-engine management. SAE, 1999.
- BRAGA, A. P.; CARVALHO, A. P. L.; LUDERMIR, T. B. **Redes Neurais Artificiais: teoria e aplicações**. Rio de Janeiro: Livros Técnicos e Científicos, 2000.
- BRASIL. Lei nº 6.938, de 31 de Agosto de 1981. Dispõe sobre a Política Nacional do Meio Ambiente. **Diário Oficial da União**. Brasília, 31 ago. 1981. Art. 3.
- BRASIL. Resolução nº 18, de 6 de maio de 1986. Dispõe sobre Programa de controle da poluição do ar por veículos automotores – PROCONVE. **Diário Oficial da União**. Brasília, 17 jun. 1986.

- CETESB. **Relatório de qualidade do ar no estado de São Paulo 2005**. São Paulo: Companhia de Tecnologia de Saneamento Ambiental, 2006.
- DELPHI. **Delphi wide range oxygen sensor**. Delphi Automotive Systems, Troy, EUA, 2007.
- DELPHI. **Delphi high function engine management system**. Delphi Automotive Systems, Troy, EUA, 2006.
- DELPHI. **Planar oxygen sensor: application manual**. Delphi Automotive Systems, Piracicaba, Brasil, 2005.
- DELPHI. **EMS Mechanization Drawings**, 2002. Disponível em: <<http://tcb.delphiauto.net>>. Acesso em: 15 jul 2008.
- DELPHI. **Oxygen sensor application manual**. Delphi Automotive Systems, Flint, EUA, 1995.
- DEMUTH, H; BEALE, M.; HAGAN, M. **Neural Network Toolbox™ 6: User's Guide**. The Math Works, Inc, EUA, 2009.
- DRI-WEFA INC; ARTHUR D. LITTLE, INC. **Future Powertrain Technologies: The Next Generation: Final Report**. 2001.
- EMBAIXADA AMERICANA. **Ford Brasil**. 01 ago 2007. Disponível em <<http://embaixadaamericana.org.br/index.php?action=materia&id=5990&submenu=press.inc.php&itemmenu=21>>. Acesso em: 30 jun 2008.
- ESALQ. **Institucional: Pólo Nacional de Biocombustíveis – ESALQ/USP**. 2004. Disponível em: <<http://www.polobio.esalq.usp.br/institucional.html>>. Acesso em: 30 jun 2008.
- FORESEE, F.D.; HAGAN, M.T. **Gauss-Newton approximation to Bayesian regularization**, Proceedings of the 1997 International Joint Conference on Neural Networks, 1997.
- GIVENS , L. **Cadillac's electronic fuel injection**. Automotive Engineering, 1976.
- GNANAM, G.; HABIBI, S.R.; BURTON, R.T.; SULATISKY, M.T. **Neural Network Control of Air-to-Fuel Ratio in Bi-Fuel Engine**. 2003 IEEE International Symposium on Intelligent Control, 2003.
- HAGAN, M.T.; MENHAJ, M. **Training feed-forward networks with the Marquardt algorithm**, IEEE Transactions on Neural Networks, 1994.
- HAYKIN, S. **Neural Networks**. Prentice-Hall, 1999.

- HOWLETT, R. J. Monitoring and Control of an Internal Combustion Engine Air-Fuel Ratio using Neural and Fuzzy Techniques. Department of Electrical and Electronic Engineering, University of Brighton, UK, 1998.
- KAMAT, S.; DIWANJI, V.; SMITH, J.; JAVAHERIAN, H.; MADHAVAN, K.P. **Virtual Sensing of SI Engines Using Recurrent Neural Networks**. 2006 SAE World Congress, Electronic Engine Controls, SAE, 2006.
- KOVÁCS, Z. L. *Redes Neurais Artificiais: Fundamentos e Aplicações*. Edição Acadêmica, 1996.
- LNCC. **Tutorial de Redes Neurais: Aplicações em Bioinformática**. 2008. Disponível em: <[http://www.lncc.br/~labinfo/tutorialRN/frm1\\_inspiracao\\_Biologica.htm](http://www.lncc.br/~labinfo/tutorialRN/frm1_inspiracao_Biologica.htm)>. Acesso em 15 jul 2008.
- MARELLI, MAGNETI. Escrito por MONTANARI, G. **Divulgação: O segredo dos carros flex está na tecnologia dos bits e bytes**. 2005. Disponível em: <<http://www.prodemge.gov.br/images/stories/volumes/volume4/ginomontanari.pdf>>. Acesso em: 30 jun 2008.
- MILHOR, C. E. **Sistema de desenvolvimento para controle eletrônico dos motores de combustão interna ciclo Otto**. São Carlos. 101p. Dissertação (Mestrado) – Escola de Engenharia de São Carlos, Universidade de São Paulo, São Carlos, 2002.
- OICA. **2008 Production Statistics**. Organisation Internationale des Constructeurs d'Automobiles. 2009. Disponível em: <<http://oica.net/category/production-statistics/>>. Acesso em 29 jun 2009.
- PARLAK, A.; ISLAMOGLU, Y.; YASAR, H.; EGRISOGUT, A. **Application of artificial neural network to predict specific fuel consumption and exhaust temperature for a Diesel engine**. Applied Thermal Engineering, Elsevier, Oxford, 2006, vol. 26, no. 8-9, p. 824-828.
- PASSARINI, L. C. **Projeto e análise de válvulas eletromagnéticas injetoras de combustível: uma nova proposta**. São Carlos. 388p. Tese (Doutorado) – Escola de Engenharia de São Carlos, Universidade de São Paulo, São Carlos, 1993.
- RIBBENS, W. B. **Understanding Automotive Electronics**. USA. Elsevier Science, 6<sup>th</sup>. Ed., 2003.
- SHAYLER, P.J.; GOODMAN, M.; MA, T. **The exploitation of neural networks in automotive engine management systems**. Engineering Applications of Artificial Intelligence, Elsevier, 2000, vol. 13, nro. 2, p. 147-157.

- SILVA, J. A.; SILVA, R. M. U.; OLIVEIRA, T. O. **Eletrônica Embarcada – Redução da Emissão de Gases Poluentes**. UNISAL, Americana, 2007.
- TEIXEIRA, E. C. **O desenvolvimento da tecnologia Flex-fuel no Brasil**. São Paulo: Instituto DNA Brasil, 2005.
- UEDA, T. **Trends of Future Powertrain Development and the Evolution of Powertrain Control Systems**. Convergence International Congress & Exposition On Transportation Electronics, SAE, Detroit, 2004, paper nro. 2004-01-0063.
- UNICA. **Cana, Açúcar e Álcool: Pressão baixista persistirá no setor**. Mai 2008. Disponível em: <[http://www.deere.com/pt\\_BR/ag/veja\\_mais/info\\_mercado/sugar\\_cane.html](http://www.deere.com/pt_BR/ag/veja_mais/info_mercado/sugar_cane.html)>. Acesso em: 30 jun 2008.
- VICENTE, B. G. G. L. Z.; CEZARE, M. J.; SILVA, I. N. **Controlador Neural de Marcha Lenta Para Motores de Combustão Interna**. In: VIII Simpósio Brasileiro de Automação Inteligente, 2007, Florianópolis. Anais do VIII Simpósio Brasileiro de Automação Inteligente, 2007.
- VOLPATO FILHO, O.; THEUNISSEN, F.; COLLI, G. B.; LIU, X.; SHI, L.; PRIESTLEY, J. **Flex Fuel Engine Management for Small Motorcycles**. Congresso SAE Brasil 2007, Delphi Automotive Systems, São Paulo, 2007, paper nro. 2007-01-2729.
- WARK, K.; WARNER, C. F.; DAVIS, W. T. **Air pollution: its origin and control**. California: Addison-Wesley, 1998.
- ZHAO, H.; LADAMMATOS, N. **Engine Combustion Instrumentation and Diagnostics**. SAE, 2001.

## **Apêndice A - Veículos Bicombustíveis**

### **A.1 – Introdução**

Segundo a ANFAVEA (2008), em 2007, mais de 90% dos veículos licenciados no Brasil foram de veículos equipados com a tecnologia bi-combustível, capazes de operar com qualquer mistura de etanol (álcool) e gasolina. Como será explicado neste apêndice, tal tecnologia possibilita ao condutor do veículo escolher a quantidade de gasolina ou álcool que deverá usar, acarretando numa possível economia nos gastos com combustíveis. Além de ser um combustível renovável e pouco danoso ao meio ambiente, atualmente o etanol é a opção mais econômica; portanto, a demanda por este combustível tende a aumentar no Brasil e em outros países. Conforme previsões da UNICA (2008), a safra 2008/2009 deverá produzir 19% mais álcool que a safra 2007/2008.

Nos últimos anos, o Brasil vem se tornando um centro de referência mundial em tecnologias multicombustível, detendo o conhecimento da fabricação de motores bicombustíveis, multicombustíveis ou biocombustíveis e da produção de etanol a partir da cana de açúcar. Em 2004, foi lançado na Escola Superior de Agricultura “Luiz de Queiroz” (ESALQ/USP), em Piracicaba, o Pólo Nacional de Biocombustíveis (ESALQ, 2004). A seguir é apresentado o desenvolvimento da tecnologia bicombustível.

## A.2 – Desenvolvimento da Tecnologia

O desenvolvimento da tecnologia bicomcombustível no Brasil teve início com a estagnação do Proálcool, programa do governo federal que visava reduzir a dependência brasileira do petróleo importado, iniciada em 1987 (MARELLI, 2005), também conhecida como a crise do abastecimento do álcool. Nesta época, o governo dos Estados Unidos da América também iniciou o incentivo do desenvolvimento de tecnologias alternativas ao uso de chumbo na gasolina como aditivo (TEIXEIRA, 2005). Este incentivo, aliado à crise brasileira, instigaram as montadoras e sistemistas brasileiras e norte americanas a investirem em pesquisas sobre bicomcombustíveis, tendo como precursora a filial norte americana da Bosch, a qual patenteou, em 1988 uma técnica de detecção de combustível através do uso do sensor físico e do uso de um sistema de gerenciamento via software. A partir deste ponto todos os projetos contariam com os sistemas de gerenciamento, hoje chamados EMS, conforme apresentado no Capítulo 2.

A filial brasileira da Bosch iniciou, em 1991, os estudos sobre miscibilidade dos combustíveis e, em 1992, a equipe começou a projetar o automóvel Omega 2.0 com a primeira proposta de tecnologia flex (BOSCH, 2006), como também é chamada a tecnologia bicomcombustível, derivando-se da palavra flexível. Em 1994, a Bosch e a Chevrolet, lançaram no Brasil o veículo-conceito Omega 2.0.

Segundo relatos da comunidade técnica, o protótipo da Robert Bosch foi desenvolvido usando um veículo movido a gasolina, sendo necessário alterar

algumas peças no veículo para suportar mudanças ocorridas com o uso do etanol, como pressão e desgastes.

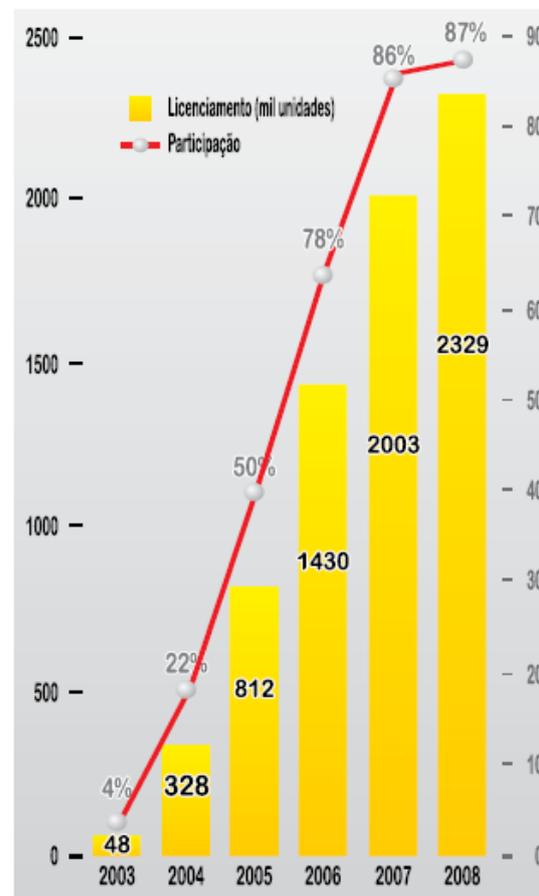
Os primeiros protótipos dos veículos bicom bustíveis necessitavam de um sensor físico para detectar a quantidade de cada combustível. Esta tecnologia, que inclusive foi aplicada nos Estados Unidos, tornava o sistema flex inviável no Brasil por causa do custo. Em 1997, a Magneti Marelli apresentou uma solução para este problema, em que o trabalho de calibração de motores a álcool eliminava este sensor extra, utilizando apenas os sensores já existentes no veículo (Teixeira, 2005).

De acordo com Teixeira (2005), passados dez anos do registro de sua primeira patente referente à tecnologia flex (1988), a Bosch criou seu primeiro protótipo sem o sensor de detecção de combustível e, no mesmo ano, o IPT (Instituto de Pesquisas Tecnológicas de São Paulo) promoveu um seminário sobre as tecnologias, resultando-se num artigo que precedeu o avanço mercadológico que a tecnologia flex teria nos anos seguintes.

Com a evolução da eletrônica móvel, com os incentivos fiscais dados pelo governo e com a maturidade tecnológica, nos anos 2000, ocorreu uma explosão na produção de veículos leves (automóveis e comerciais leves) equipados com motores flex. Conforme pode ser observado na Figura A.1, em cinco anos as vendas de veículos flex saltaram de 4%, em 2003, para 86% em 2007. Em 2008 este percentual foi de 87% (ANFAVEA, 2009).

## Autoveículos flex fuel

### Participação no licenciamento de veículos leves



Total do licenciamento acumulado de autoveículos flex fuel de março de 2003 (lançamento do produto) a dezembro de 2008: **6,95 milhões de unidades**.

Figura A.1: Participação dos veículos flex nas vendas de veículos leves.

Em Abril de 2003, o sistema desenvolvido pela Magneti Marelli é usado no primeiro veículo flex a entrar no mercado, o Gol Total Flex 1.6 da Volkswagen.

Em Junho do mesmo ano, a ex-subsidiária da GM (General Motors), Delphi Automotive Systems, tem seu sistema usado no Chevrolet Corsa Flexpower. Até a separação da GM, ocorrida no final dos anos 90, a Delphi vinha participando de pesquisas e das produções como uma empresa subsidiária da

GM. Ainda em Junho, o terceiro automóvel a entrar no mercado e com tecnologia Magneti Marelli é o Ford Fiesta Flex-fuel e, em Outubro, a Volkswagen lança seu segundo automóvel flex, o Fox 1.6, também usando Magneti Marelli.

Recentemente, foi apresentada pela Delphi Automotive mais uma aplicação da tecnologia flex, em Novembro de 2007, no Congresso SAE Brasil, para aplicação em motocicletas leves (Volpato et al., 2007). Em Março de 2009 foi lançada, pela Honda, a primeira moto leve com tecnologia flex (Honda Titan Mix).

A diversificação das fontes de energia usada nos veículos tende a um crescimento, conforme é a proposta do relatório final *Future Powertrain Technologies: The Next Generation* (DRI-WEFA, 2001) e também abordado em Ueda (2004). Hoje, sistemas usando bio-combustíveis, multi-combustíveis ou híbridos, já estão disponíveis no mercado.

### A.3 – Poluição e Regulamentação

#### A.3.1 – Introdução

Além da motivação pela escolha do combustível desejado, o uso dessa tecnologia bicomcombustível por parte da população também foi motivada pela questão ambiental, ou seja, reduzir a emissão de gases poluentes usando combustíveis renováveis e de menor impacto ao meio ambiente.

Os sensores de oxigênio, os sistemas de gerenciamento de motores, a tecnologia bicomustível, entre outras inovações, foram importantíssimos instrumentos que colaboraram para a redução cada vez maior da emissão de poluentes.

No tópico seguinte será apresentada a razão da qual se necessita de regulamentações e conceitos básicos de Poluição. Depois serão apresentados, resumidamente, alguns itens de Regulamentações nacionais e estrangeiras.

### A.3.2 – Poluição

Pela lei brasileira, poluição é definida no art. 3, III, da Lei 6.938/81 (BRASIL, 1981), como:

- III - poluição, a degradação da qualidade ambiental resultante de atividades que direta ou indiretamente:
- a) prejudiquem a saúde, a segurança e o bem-estar da população;
  - b) criem condições adversas às atividades sociais e econômicas;
  - c) afetem desfavoravelmente a biota;
  - d) afetem as condições estéticas ou sanitárias do meio ambiente;
  - e) lancem matérias ou energia em desacordo com os padrões ambientais estabelecidos.

Elemento indispensável à sobrevivência humana, o ar que respiramos é composto por 78% de nitrogênio, 21% de oxigênio e 1% de outros gases, como 0,9% de gases nobres, 0,03% de dióxido de carbono (CO<sub>2</sub>), vapor d'água, etc. (Wark et al., 1998).

Segundo a CETESB (2006), os gases nocivos mais comumente encontrados nas cidades são o material particulado (conjunto de poeira e fumaça preta emitido principalmente por caminhões), o CO (monóxido de carbono, gerado pelos automóveis), o SO<sub>2</sub> (dióxido de enxofre, que sai das chaminés das fábricas) e o O<sub>3</sub> (ozônio, resultado de uma reação química de outros poluentes com a luz solar).

A inalação destas substâncias pode causar problemas como: asma e bronquite; redução da capacidade de trabalho e de praticar exercícios físicos e até mesmo agravar o quadro de portadores de doenças respiratórias e cardíacas; prejuízo dos reflexos, da capacidade de estimar intervalos de tempo, visual e do raciocínio; irritação nos olhos e vias respiratórias; envelhecimento precoce da pele, náusea, dor de cabeça, tosse, fadiga, aumento do muco, diminuição da resistência orgânica as infecções e agravamento de doenças respiratórias e até a morte. Além de consequências ao meio ambiente como a contaminação do solo e a chuva ácida (CETESB, 2006).

Como exemplo da periculosidade, tem-se:

Se um carro ficar ligado em uma garagem fechada de 4 metros de comprimento, 4 metros de largura e 2,5 metros de altura (formando um ambiente de 40 mil litros), durante aproximadamente dez minutos, o volume de CO produzido terá atingido a quantidade letal. Nesse caso, a morte de um homem que estivesse nessa garagem ocorreria por asfixia. (SCHIVARTCHE, 2005 apud SILVA et al., 2007).

Mais de 70% das emissões de CO e mais de 80% das de HC (hidróxido de carbono) em 2004, na Região Metropolitana de São Paulo, foram originadas por veículos leves segundo a CETESB (2006)

### A.3.3 – Legislação de Controle de Emissões

Com o volume crescente do tráfego de veículos em grandes cidades a poluição tornou-se um grande problema e, para diminuir os problemas, teve início as definições de limites de emissões. Segundo BOSCH (2005), o pioneiro na tentativa de aplicar o controle de emissões veiculares foi o estado da Califórnia (EUA), na década de 1950 (SILVA et al., 2007). Desde então, o controle de emissões veiculares está sendo implantado em diversos países do mundo, obrigando as indústrias automobilísticas e as petrolíferas a pesquisarem e produzir veículos e combustíveis menos poluentes.

Os norte-americanos implantaram desafios gradativos na redução dos limites de emissões (denominados Tier 0, Tier 1 e Tier 2); deste modo, possibilitaram o desenvolvimento tecnológico dos veículos e dos testes de emissão antes dos demais países. O processo europeu de regulamentação de emissões foi consolidado junto com a formação da União Européia e seguiu o aumento gradativo dos limites implantado pelos EUA.

Os testes de emissão europeus, realizados para assegurar o atendimento dos limites de emissão (Euro I até Euro VI), diferem dos testes norte-americanos pelo fato de ser necessário realizar dois testes, com diferentes ciclos de direção, um representando as condições de direção urbanas de baixa velocidade, chamado

de ECE, e outro representando condições com velocidades altas, denominado EUDC.

Na América do Sul, o Brasil foi o pioneiro, controlando as emissões de gases e vapores do cárter a partir de 1976 (SILVA et al., 2007); foi também o pioneiro no mundo, a controlar a emissão de aldeídos, devido a ter frota movida a etanol. Em 1986, o governo federal instituiu através da resolução 18/1986 o PROCONVE - Programa de Controle da Poluição do Ar por Veículos Automotores (BRASIL, 1986), onde inicialmente foram usados como modelo os procedimentos utilizados pelos EUA.

Os limites máximos de emissão de poluentes foram fixados usando-se cronograma específico para três categorias distintas de veículos: veículos leves de passageiros (automóveis); veículos leves comerciais (pick-up, van, utilitários, etc) e veículos pesados (ônibus e caminhões).

Em 1º de Abril de 2009, entra em vigor a fase 6 do PROCONVE para veículos pesados e a fase 5 para veículos leves, sendo estas fases equivalentes a Euro IV, que vigoram na Europa a partir de 2006. Atualmente, a Europa segue o Euro V. Desde 1º de Janeiro de 2009, está em vigor no Brasil, o Promot (Programa de controle da poluição do ar por Motociclos e veículos similares), que regulamenta as emissões para motocicletas. Na Tabela A.1 são apresentadas as metas especificadas no PROCONVE para veículos leves novos (CETESB, 2006).

Tabela A.1: PROCONVE – Limites de emissão para veículos leves novos.

Ano	CO (g/Km)	HC (g/Km)	NOx (g/Km)	RCHO <sup>2</sup> (g/Km)	MP <sup>3</sup> (g/Km)	EVAP. <sup>4</sup> (g/teste)	CO-ML (% vol)
89 – 91	24	2,10	2,0	--	--	6	3
92 – 96 <sup>6</sup>	24	2,10	2,0	0,15	--	6	3
92 – 93	12	1,20	1,4	0,15	--	6	2,5
Mar/94	12	1,20	1,4	0,15	0,05	6	2,5
Jan/97	2	0,30	0,6	0,03	0,05	6	0,5
Mai/03	2	0,30	0,6	0,03	0,05	2	0,5
Jan/05 (40%)	2	0,16 <sup>5</sup>	0,25 <sup>7</sup>	0,03	0,05	2	0,57
Jan/06 (70%)	2	ou	ou	0,03	0,05	2	0,57
Jan/07 (100%)	2	0,30 <sup>6</sup>	0,60 <sup>3</sup>	0,03	0,05	2	0,57
Jan/09	2	0,05 <sup>5</sup> ou	0,12 <sup>7</sup> ou	0,02	0,05	2	0,57
Jan/09	2	0,30 <sup>6</sup>	0,2 <sup>5</sup>	0,02	0,05	2	0,57

1 – Medições de acordo com a NBR6601 (US-FTP75), e conforme as Resoluções CONAMA n° 15/95 e n° 315/02.

2 – Apenas para veículos do ciclo Otto. Aldeídos totais de acordo com a NBR 12026.

3 – Apenas para veículos do ciclo diesel.

4 – Apenas para veículos do ciclo Otto, exceto a GNV.

5 – Hidrocarbonetos não metano (NMHC)

6 – Hidrocarbonetos totais somente para veículos a GNV, que também atendam ao item 5.

7 – Apenas para veículos do ciclo Otto, inclusive a GNV.

## **Apêndice B - Métodos Levenberg-Marquardt e Levenberg-Marquardt com Regularização Bayesiana**

### **B.1 – Introdução**

Na Seção 4.3 foi apresentado o método do gradiente descendente, um dos métodos que podem ser usados para treinamentos de redes PMC. Entretanto, a utilização deste método exige elevado esforço computacional. Neste apêndice serão apresentados os métodos utilizados no desenvolvimento deste trabalho: Levenberg-Marquardt e Levenberg-Marquardt com regularização Bayesiana, que são técnicas incorporadas ao algoritmo *backpropagation* e podem convergir de dez a cem vezes mais rápido que o método do gradiente descendente (DEMUTH et. al., 2009).

### **B.2 – Levenberg-Marquardt (*trainlm*)**

O método Levenberg-Marquardt é uma técnica baseada no método dos mínimos quadrados para modelos não-lineares, que pode ser incorporada ao algoritmo *backpropagation* a fim de aumentar a eficiência do processo de treinamento. Este algoritmo é do tipo quase-Newton, pois utiliza as segundas derivadas (matriz Hessiana) para minimizar o erro. Porém, o algoritmo não calcula a matriz Hessiana, pois a matriz pode ser aproximada, ou seja:

$$H = J^T J \quad (\text{B.1})$$

onde:

$J \rightarrow$  Matriz Jacobiana que contém as primeiras derivadas dos erros.

O gradiente pode ser aproximado como:

$$g = J^T e \quad (\text{B.2})$$

onde:

$e \rightarrow$  Vetor de erros da rede.

Portanto, a característica principal deste algoritmo é a computação da matriz Jacobiana, que pode ser computada através de uma técnica de retro-propagação, que é muito menos complexo do que o cálculo da matriz Hessiana (HAGAN; MENHAJ, 1994). O método modifica o método Gauss-Newton, incluindo um fator  $\mu$  responsável pelo ajuste do tamanho do incremento realizado pelo algoritmo. A equação de atualização dos pesos é descrita como:

$$x_{k+1} = x_k - [J^T J + \mu I]^{-1} J^T e \quad (\text{B.3})$$

Quando o escalar  $\mu$  é zero, o método de Levenberg-Marquardt funciona apenas como o método de Newton, usando a matriz Hessiana aproximada. Quando  $\mu$  é grande, o método torna-se o de gradiente descendente. O método de Newton é mais rápido e exato, próximo a um erro mínimo; sendo assim, o objetivo é deslocar para o método de Newton o mais rápido possível. Com estas modificações, o algoritmo Levenberg-Marquardt pode convergir de dez a cem vezes mais rápido que o método do gradiente descendente (DEMUTH et. al., 2009).

### B.3 – Levenberg-Marquardt com regularização Bayesiana (*trainbr*)

Um dos problemas que ocorrem durante o treinamento de redes do tipo Perceptron é denominado *overfitting*. Este problema ocorre quando a rede converge em seu treinamento, mas quando é submetida a novos padrões, não pertencentes ao conjunto de treinamento, o erro torna-se grande. Isto ocorre porque a rede memorizou apenas os padrões de treinamento, não conseguindo generalizar novas situações.

Uma abordagem para melhorar a generalização, evitando o *overfitting*, é a regularização, modificando a função-objetivo, que normalmente utiliza a média dos erros quadrados. No algoritmo de Regularização Bayesiana, a função-objetivo assume a forma descrita na Equação B.4:

$$F = \alpha \cdot SSE + \beta \cdot SSW \quad (\text{B.4})$$

Onde:

$SSE \rightarrow$  Somatório dos erros quadrados;

$SSW \rightarrow$  Somatórios do quadrado dos pesos e limiar de ativação;

$\alpha$  e  $\beta \rightarrow$  Parâmetros da função-objetivo.

Os parâmetros de regularização são obtidos com técnicas estatísticas (Bayes) e, como a técnica de regularização bayesiana requer o cálculo da matriz Hessiana, o algoritmo de treinamento associado à regularização é o de Levenberg-Marquardt (FORESEE; HAGAN, 1997).

# Livros Grátis

( <http://www.livrosgratis.com.br> )

Milhares de Livros para Download:

[Baixar livros de Administração](#)

[Baixar livros de Agronomia](#)

[Baixar livros de Arquitetura](#)

[Baixar livros de Artes](#)

[Baixar livros de Astronomia](#)

[Baixar livros de Biologia Geral](#)

[Baixar livros de Ciência da Computação](#)

[Baixar livros de Ciência da Informação](#)

[Baixar livros de Ciência Política](#)

[Baixar livros de Ciências da Saúde](#)

[Baixar livros de Comunicação](#)

[Baixar livros do Conselho Nacional de Educação - CNE](#)

[Baixar livros de Defesa civil](#)

[Baixar livros de Direito](#)

[Baixar livros de Direitos humanos](#)

[Baixar livros de Economia](#)

[Baixar livros de Economia Doméstica](#)

[Baixar livros de Educação](#)

[Baixar livros de Educação - Trânsito](#)

[Baixar livros de Educação Física](#)

[Baixar livros de Engenharia Aeroespacial](#)

[Baixar livros de Farmácia](#)

[Baixar livros de Filosofia](#)

[Baixar livros de Física](#)

[Baixar livros de Geociências](#)

[Baixar livros de Geografia](#)

[Baixar livros de História](#)

[Baixar livros de Línguas](#)

[Baixar livros de Literatura](#)  
[Baixar livros de Literatura de Cordel](#)  
[Baixar livros de Literatura Infantil](#)  
[Baixar livros de Matemática](#)  
[Baixar livros de Medicina](#)  
[Baixar livros de Medicina Veterinária](#)  
[Baixar livros de Meio Ambiente](#)  
[Baixar livros de Meteorologia](#)  
[Baixar Monografias e TCC](#)  
[Baixar livros Multidisciplinar](#)  
[Baixar livros de Música](#)  
[Baixar livros de Psicologia](#)  
[Baixar livros de Química](#)  
[Baixar livros de Saúde Coletiva](#)  
[Baixar livros de Serviço Social](#)  
[Baixar livros de Sociologia](#)  
[Baixar livros de Teologia](#)  
[Baixar livros de Trabalho](#)  
[Baixar livros de Turismo](#)