

UNIVERSIDADE DE BRASÍLIA
FACULDADE DE TECNOLOGIA
DEPARTAMENTO DE ENGENHARIA CIVIL

MODELAGEM CONSTITUTIVA DE AREIAS
USANDO REDES NEURAIS ARTIFICIAIS

HERNÁN EDUARDO MARTÍNEZ CARVAJAL

ORIENTADOR: MÁRCIO MUNIZ DE FARIAS, Ph.D.

TESE DE DOUTORADO EM GEOTECNIA

PUBLICAÇÃO: G.TD - 035/06

BRASÍLIA / DF: MARÇO DE 2006

**UNIVERSIDADE DE BRASÍLIA
FACULDADE DE TECNOLOGIA
DEPARTAMENTO DE ENGENHARIA CIVIL**

**MODELAGEM CONSTITUTIVA DE AREIAS USANDO
REDES NEURAIS ARTIFICIAIS**

HERNÁN EDUARDO MARTÍNEZ CARVAJAL

TESE DE DOUTORADO SUBMETIDA AO DEPARTAMENTO DE ENGENHARIA CIVIL E AMBIENTAL DA UNIVERSIDADE DE BRASÍLIA COMO PARTE DOS REQUISITOS NECESSÁRIOS PARA A OBTENÇÃO DO GRAU DE DOUTOR.

APROVADA POR:

**MÁRCIO MUNIZ DE FARIAS, Ph.D. (UnB)
(ORIENTADOR)**

**ENNIO MARQUES PALMEIRA, Ph.D. (UnB)
(EXAMINADOR INTERNO)**

**LUIS FERNANDO MARTINS RIBEIRO, Ph.D. (UnB)
(EXAMINADOR INTERNO)**

**CHRISTIANNE DE LYRA NOGUEIRA, Ph.D. (UFOP)
(EXAMINADOR EXTERNO)**

**CELSO ROMANEL, Ph.D. (PUC-RIO)
(EXAMINADOR EXTERNO)**

DATA: BRASÍLIA/DF, 28 DE MARÇO DE 2006.

FICHA CATALOGRÁFICA

MARTÍNEZ-CARVAJAL, HERNÁN EDUARDO

Modelagem Constitutiva de Areias Usando Redes Neurais Artificiais [Distrito Federal] 2006

xxv, 250 p., 297 mm (ENC/FT/UnB, Doutor, Geotecnia, 2006)

Tese de Doutorado - Universidade de Brasília. Faculdade de Tecnologia.

Departamento de Engenharia Civil e Ambiental

1. Modelagem Constitutiva 2. Redes Neurais Artificiais

3. Areia 4. Ensaios Triaxiais

I. ENC/FT/UnB II. Título (série)

REFERÊNCIA BIBLIOGRÁFICA

MARTÍNEZ-CARVAJAL, H.E. (2006). Modelagem Constitutiva de Areias Usando Redes Neurais Artificiais. Tese de Doutorado, Publicação G.TD-035/06, Departamento de Engenharia Civil e Ambiental, Universidade de Brasília, Brasília, DF, 247 p.

CESSÃO DE DIREITOS

NOME DO AUTOR: Hernán Eduardo Martínez Carvajal

TÍTULO DA TESE DE DOUTORADO: Modelagem Constitutiva de Areias Usando Redes Neurais Artificiais

GRAU / ANO: Doutor / 2006

É concedida à Universidade de Brasília a permissão para reproduzir cópias desta tese de doutorado e para emprestar ou vender tais cópias somente para propósitos acadêmicos e científicos. O autor reserva outros direitos de publicação e nenhuma parte desta tese de doutorado pode ser reproduzida sem a autorização por escrito do autor.

Hernán Eduardo Martínez Carvajal

Escuela de Ingeniería Civil. Facultad de Minas.

Universidad Nacional de Colombia – Sede Medellín

Carrera 65 x Calle 65. Ciudad Universitaria

Medellín/Antioquia - Colombia

DEDICATÓRIA



Ao meu irmão Alejandro que todos os dias me dá exemplo de liberdade!!!!!!!!!!

“Devemos ser livres, não para fazer a nossa vontade, mas sim para entender muito profundamente os nossos instintos e impulsos”.

“Há duas formas de solidão: uma que é a solidão do isolamento, com desesperação, tristeza e separação de tudo e de todos, e a outra que é a solidão madura do homem livre, que não dependendo de nada nem de ninguém, está relacionado com tudo”. Krishnamurti.

AGRADECIMENTOS

Sendo que na hora de escrever esta página me vi de repente pensando excessivamente, então decidi não agradecer especificamente a ninguém mais do que, de maneira especial, ao meu orientador o professor **Márcio Muniz de Farias** e, de maneira geral, ao Programa de Pós-Graduação em Geotecnia da Universidade de Brasília.

Adicionalmente, em cumprimento dos compromissos adquiridos, agradeço ao CNPq pelo apoio financeiro oferecido para minha permanência no Brasil e à FINATEC pelas verbas recebidas para a apresentação parcial do trabalho no ECCOMAS2004.

MARTINEZ-CARVAJAL, H. E. (2006). Modelagem Constitutiva de Areias Usando Redes Neurais Artificiais. Tese de Doutorado, Publicação G. TD-035/06, Departamento de Engenharia Civil e Ambiental, Universidade de Brasília, Brasília D.F., 247p.

RESUMO

Tradicionalmente, a relação tensão-deformação tem sido representada por uma série de equações matemáticas que tenta descrever o comportamento do solo com base em um conjunto de parâmetros. O problema principal dessas aproximações paramétricas é a alta complexidade matemática envolvida principalmente quando se incluem efeitos não lineares ou quando diferentes tipos de solo são considerados. Uma aproximação diferente consiste na utilização de redes neurais artificiais (RNA) treinadas com informação experimental para capturar diretamente o comportamento constitutivo do solo e eventualmente atuar como um operador constitutivo dentro de um código de elementos finitos para solucionar problemas de contorno.

No caso das areias, o seu comportamento mecânico depende basicamente da sua natureza física e do seu estado. A primeira está relacionada com a composição da areia como material granular e tem sido descrita por meio da distribuição granulométrica, teor de finos, forma e mineralogia dos grãos. Por outra parte, o estado físico tem sido descrito por meio de variáveis como a densidade relativa, o estado de tensão efetiva e o tipo de arranjo entre os grãos. Conseqüentemente, numa aproximação com RNA o conjunto de variáveis de treinamento deve conter informação sobre o estado tensão-deformação do solo e sobre a sua natureza física.

Nesta tese foram usados o fator de forma, n_s , o índice de vazios inicial, e_0 , e o $e_{\max}-e_{\min}$ como variáveis físicas para treinar um PMC (Perceptron de Múltiplas Camadas) de tal forma que se pudesse estimar os acréscimos de tensão dados os correspondentes acréscimos de deformação além dos estados tensão-deformação atual e anterior. Isto com o intuito de usar a RNA treinada, como operador constitutivo dentro de um código de elementos finitos (programa ALLFINE) agindo ao nível de ponto de Gauss em substituição de uma relação elasto-plástica convencional.

A base de dados para o treinamento da RNA está conformada por uma série de ensaios de laboratório que incluem uma ampla gama de trajetórias de tensão tanto do tipo convencional como de compressão isotrópica e trajetórias no plano octaédrico. A informação dos ensaios foi majoritariamente obtida de publicações de trabalhos anteriores, incluindo 13 tipos diferentes de areia.

Após finalizada a fase de treinamento, a RNA constitutiva foi implementada no programa ALLFINE, e algumas simulações foram realizadas para a validação do comportamento do sistema conjunto. Inicialmente, foram realizadas simulações do comportamento das areias ao nível de ponto constitutivo. A seguir, o grau de complexidade das análises aumentou ao considerar a amostra de solo como um volume discretizado com uma malha de elementos finitos. Finalmente foi analisado o comportamento de uma sapata retangular assente num depósito de areia homogênea. Não tendo informação experimental para fazer as devidas comparações, considerou-se pertinente comparar com os resultados obtidos por meio do modelo Cam-Clay.

O comportamento do sistema híbrido (SH) RNA-Elementos Finitos nas simulações realizadas se mostrou satisfatório, o que é um alento para seguir pesquisando nesta área da geotecnia que se mostra bastante promissora.

MARTINEZ-CARVAJAL, H. E. (2006). Constitutive Modelling of Sands Using Artificial Neural Networks. Tese de Doutorado, Publicação G. TD-035/06, Departamento de Engenharia Civil e Ambiental, Universidade de Brasília, Brasília D.F., 247p.

ABSTRACT

In a traditional geotechnical approach, the relationship between stresses and strains has been represented by a series of mathematical equations that tries to describe the soil behavior, based on a set of fundamental parameters. The main problem with this parametric approximation is the high mathematical complexity involved mainly when non-linear effects have to be included and when different types of soils are considered. In addition, many of the traditional mathematical models were deduced based exclusively on conventional axis-symmetric triaxial stress paths. A fundamentally different approach consists in the use of Artificial Neural Networks (ANN) to model the material behavior directly from laboratory tests results. In this non-parametric approach, the soil constitutive behavior will be directly captured by the ANN, and stored for later generalization. The interest of such an application of ANN in the case when the model is built directly from available experimental data is obvious. In such a case a suitably trained ANN can replace an unknown conventional analytical description, and perform as a constitutive operator embedded into a finite element code. A source of knowledge for ANN is not a symbolic formula but the set of experimental data.

In the case of sandy soils, their mechanical behavior depends fundamentally on their physical nature and state. The first, is related with the sand composition as a granular material, and has been frequently described with the use of properties such a grain size distribution, fines content, grain shape and mineralogy, among others. On the other hand, the state of the sand is related with the “in situ” physical conditions under which the soil exists, and it has been traditionally described with the aid of variables such a relative density (D_r), the effective stress state, and the fabric. Consequently, in a knowledge-based approach, where an ANN acts as a constitutive operator, the set of input variables must contain not only information about the stress-strain state, but also about the sand physical nature and some other variables.

In this thesis, the use of the void ratio range ($e_{max}-e_{min}$), the shape factor (n_s) and the initial void ratio (e_0) were used as physical input variables for training a multilayered perceptron ANN, to simulate the stress path response for given strain paths imposed during 3D triaxial tests on sand. In this case the ANN is trained to compute the stress increment, given a known strain increment and the current and previous stress-strain states.

The laboratory tests correspond to axis-symmetric stress paths as well as paths defined in the π -plane, and some hydrostatic compression tests. Almost all of them were obtained from published literature on the area, and covered 13 different types of well known sands.

Once trained, the ANN was implemented into a Finite Element Code (ALLFINE) and some simulations were carried out for validating the behavior of the system as a whole. First, the behavior of the model at a level of gauss point were tested, after that, cubical soil samples were simulated and finally a rectangular slab resting on a mass of soft sand were simulated as a final validation of the hybrid system (ALLFINE-ANN). The results were compared against those obtained with the use of the Cam-Clay model, showing that the neural model is a very promising technique, able to predict complete and complex stress paths.

ÍNDICE

CAPÍTULO 1	1
INTRODUÇÃO	1
1.1 – DEFINIÇÃO DO PROBLEMA.....	1
1.2 – OBJETIVO.....	5
1.3 – METODOLOGIA	6
1.3.1 – ESCOLHA DO CONJUNTO DE DADOS	6
1.3.2 – ESCOLHA DA ARQUITETURA DA REDE.....	9
1.3.3 – TREINAMENTO E TESTE DA REDE	10
1.3.4 - SIMULAÇÃO FINAL E IMPLEMENTAÇÃO NO PROGRAMA DE ELEMENTOS FINITOS.....	10
1.4 – ESCOPO DA TESE	11
CAPÍTULO 2	13
REVISÃO BIBLIOGRÁFICA.....	13
2.1 – BREVE HISTÓRICO SOBRE AS REDES NEURAIS	13
2.2 – AS REDES NEURAIS EM ENGENHARIA CIVIL.....	16
2.3 – APLICAÇÕES EM ENGENHARIA GEOTÉCNICA	19
2.3.1 – CLASSIFICAÇÃO E CARACTERIZAÇÃO DE SOLOS E ROCHAS.....	20
2.3.2 – ESTUDO DO POTENCIAL DE LIQUEFAÇÃO DE AREIAS	20
2.3.3 – APLICAÇÕES EM PAVIMENTAÇÃO.....	21
2.3.4 – MODELAGEM CONSTITUTIVA.....	22
2.3.5 – MODELAGEM “IN SITU” DE GEOMATERIAIS.....	24
CAPÍTULO 3	26
FUNDAMENTOS TEÓRICOS	26
3.1 – O NEURÔNIO ARTIFICIAL.....	26
3.2 – AS REDES NEURAIS.....	29
3.3 – O PERCEPTRON E O APRENDIZADO SUPERVISIONADO.....	32
3.3.1 – MÉTODO DA DESCIDA MAIS ÍNGREME	35
3.3.2- MÉTODO DE NEWTON.....	36

3.3.3 – MÉTODO DE GAUSS-NEWTON.....	37
3.4 – PERCEPTRONS DE MÚLTIPLAS CAMADAS	39
3.4.1 - INTRODUÇÃO	39
3.4.2 – O ALGORITMO DE RETROPROPAGAÇÃO.....	41
3.4.3 – TREINAMENTO	47
3.4.4 – GENERALIZAÇÃO	53
3.4.5 – APROXIMAÇÃO DE FUNÇÕES.....	56
3.5 – AS REDES COMPETITIVAS.....	57
CAPÍTULO 4.....	61
APRESENTAÇÃO DOS MODELOS CONSTITUTIVOS NEURAI PARA AREIAS	61
4.1 – INTRODUÇÃO	62
4.2 – MODELOS PARAMÉTRICOS.....	62
4.3 – MODELAGEM USANDO REDES NEURAI ARTIFICIAIS.....	65
4.4 – MODELO NEURAL 1.....	67
4.4.1 – A BASE DE DADOS.....	68
4.4.2 - ARQUITETURA E TREINAMENTO.....	71
4.4.3 – RESULTADOS DA SIMULAÇÃO COM O MODELO NEURAL 1	75
4.4.4 – DISCUSSÃO.....	79
4.5 – MODELO NEURAL 2.....	80
4.5.1 – A BASE DE DADOS.....	81
4.5.2 – ARQUITETURA E TREINAMENTO	83
4.5.3 – RESULTADOS DA SIMULAÇÃO	86
4.5.4 – DISCUSSÃO.....	89
4.6 – MODELO NEURAL 3.....	90
4.6.1 – A BASE DE DADOS.....	91
4.6.2 – ARQUITETURA E TREINAMENTO	92
4.7 – MODELO NEURAL 4.....	99
4.7.1 – PRÉ-CLASSIFICAÇÃO DOS TESTES DE TREINAMENTO	100
4.7.2 – PRÉ-PROCESSAMENTO DA BASE DE DADOS.....	101
4.7.3 – ARQUITETURA E TREINAMENTO	102
4.7.4 – RESULTADOS DA SIMULAÇÃO	106
4.7.5 – DISCUSSÃO.....	117

4.8 – MODELO NEURAL 5.....	123
4.8.1 - PRÉ-PROCESSAMENTO DA BASE DE DADOS	124
4.8.2 – ARQUITETURA E TREINAMENTO	126
4.8.3 – RESULTADOS DA SIMULAÇÃO	126
4.8.4 – DISCUSSÃO.....	148
4.9 – MODELO NEURAL 6.....	149
4.9.1 – PRÉ-PROCESSAMENTO DA BASE DE DADOS.....	150
4.9.2 – ARQUITETURA E TREINAMENTO	154
4.9.3 – RESULTADOS DA SIMULAÇÃO	156
4.9.4 – DISCUSSÃO.....	162
CAPÍTULO 5	164
IMPLEMENTAÇÃO	164
5.1 – INTRODUÇÃO	164
5.2 – IMPLEMENTAÇÃO NUMÉRICA EM UM CÓDIGO DE ELEMENTOS FINITOS	165
5.3 – CASOS DE APLICAÇÃO.....	172
5.3.1 – SIMULAÇÃO DE ENSAIOS DE LABORATÓRIO	172
5.3.2 – SIMULAÇÃO DE UMA AMOSTRA CÚBICA.....	195
5.3.3 – ANÁLISE DE UMA SAPATA RETANGULAR	197
5.4 – DISCUSSÃO.....	205
CAPÍTULO 6	206
CONCLUSÕES.....	206
6.1 – SUGESTÕES PARA FUTURAS PESQUISAS.....	209
REFERÊNCIAS BIBLIOGRÁFICAS.....	210
APÊNDICE A	215
BASE DE DADOS.....	215
APÊNDICE B	222
CLASSIFICAÇÃO DOS TESTES	222
APÊNDICE C	224
BASE DE DADOS – MODELO NEURAL 4	224

APÊNDICE D	226
VALORES DOS PARÂMETROS LIVRES DA REDE <i>MAGOLA11</i>	226

ÍNDICE DE TABELAS

TABELA	PÁGINA
Tabela 1.1. Relação das areias consideradas na base de dados.....	7
Tabela 5.1. Análise dos gráficos da Figura 5.6.	179
Tabela 5.2. Ensaios usados para simular o comportamento de uma amostra cúbica.	196
Tabela A.1. Resumo das características gerais dos testes usados na base de dados.	215
Tabela B.1. Resultado da classificação realizada com três redes tipo Kohonen. Em preto e azul os testes agrupados na classe 1 usados para treinamento e generalização respectivamente, em vermelho os testes pertencentes às outras classes.	222
Tabela C.1. Intervalos de valores da base de dados usada para o treinamento dos modelos neurais PMC.	224
Tabela C.2. Síntese das diferentes trajetórias usadas no treinamento e validação da rede <i>Magola9</i> , para cada tipo de areia.	225
Tabela D.1. Valores dos pesos das conexões entre as 10 primeiras entradas da camada de entrada e a primeira camada oculta da rede <i>Magola11</i> . Matriz de pesos IW.	226
Tabela D.2. Valores dos pesos das conexões entre as 10 últimas entradas da camada de entrada e a primeira camada oculta da rede <i>Magola11</i> . Matriz de pesos IW.	228
Tabela D.3. Valores dos pesos das conexões entre a primeira camada oculta (120 neurônios) e a segunda camada oculta (100 neurônios) da rede <i>Magola11</i> . Coluna 1 a 15. Matriz de Pesos LW1.....	230
Tabela D.4. Valores dos pesos das conexões entre a primeira camada oculta (120 neurônios) e a segunda camada oculta (100 neurônios) da rede <i>Magola11</i> . Coluna 16 a 30. Matriz de pesos LW1.....	232
Tabela D.5. Valores dos pesos das conexões entre a primeira camada oculta (120 neurônios) e a segunda camada oculta (100 neurônios) da rede <i>Magola11</i> . Coluna 31 a 45. matriz de pesos LW1.....	235

Tabela D.6. Valores dos pesos das conexões entre a primeira camada oculta (120 neurônios) e a segunda camada oculta (100 neurônios) da rede <i>Magola11</i> . Coluna 46 a 60. Matriz de pesos LW1.....	237
Tabela D7. Valores dos pesos das conexões entre a primeira camada oculta (120 neurônios) e a segunda camada oculta (100 neurônios) da rede <i>Magola11</i> . Coluna 61 a 75. Matriz de pesos LW1.....	239
Tabela D8. Valores dos pesos das conexões entre a primeira camada oculta (120 neurônios) e a segunda camada oculta (100 neurônios) da rede <i>Magola11</i> . Coluna 76 a 90. matriz de pesos LW1.....	242
Tabela D9. Valores dos pesos das conexões entre a primeira camada oculta (120 neurônios) e a segunda camada oculta (100 neurônios) da rede <i>Magola11</i> . Coluna 91 a 105. Matriz de pesos LW1.....	244
Tabela D10. Valores dos pesos das conexões entre a primeira camada oculta (120 neurônios) e a segunda camada oculta (100 neurônios) da rede <i>Magola11</i> . Coluna 106 a 120. Matriz de pesos LW1.....	246
Tabela D11. Valores dos pesos das conexões entre a segunda camada oculta (100 neurônios) e a camada de saída da rede <i>Magola11</i> . Matriz de pesos LW2	249
Tabela D12. Valores dos elementos do primeiro vetor de biases, b1, (120 elementos) da rede <i>Magola11</i>	250
Tabela D13. Valores dos elementos do segundo vetor de biases, b2, (100 elementos) da rede <i>Magola11</i>	250
Tabela D14. Valores dos elementos do terceiro vetor de biases, b3, (3 elementos) da rede <i>Magola11</i>	250

ÍNDICE DE FIGURAS

FIGURA	PÁGINA
Figura 1.1. Representação geral do comportamento não linear: (a) carga-deslocamento; (b) tensão–deformação (Modificado de Desai & Siriwardane, 1984).	3
Figura 2.1. Esquema de uma rede neural simples.	14
Figura 2.2. Classes linearmente separáveis A e B que podem ser aprendidas por um perceptron (Kasabov, 1998).	15
Figura 3.1. Modelo de neurônio de McCulloch-Pitts (Modificado de Ghaboussi, 1992).....	26
Figura 3.2. Funções de ativação mais usadas: (a) Degrau, (b-1 e b-2) Limiar lógico, (c) sigmóide: Logística (c-1), tangente hiperbólica (c-2) e Gaussiana (d). (modificada de Kasabov, 1998).....	28
Figura 3.3. Transformação afim produzida pela presença de um bias; note que $v_k=b_k$ em $u_k=0$ (modificado de Kasabov, 1998)	29
Figura 3.4. Uma rede neural simples. Os pesos das conexões são produto de um processo de treinamento. O valor de ativação do nó 5 (u_5) é apresentado (modificado de Kasabov, 1998).....	30
Figura 3.5. Algumas arquiteturas neurais (modificado de Braga et al., 2000).....	31
Figura 3.6. Trajetória do método da descida mais íngreme em um espaço bidimensional para dois valores diferentes de parâmetro de taxa de aprendizagem: (a) η pequeno, (b) η grande. As coordenadas w_1 e w_2 são elementos do vetor de peso W	36
Figura 3.7. Esquema de um perceptron de múltiplas camadas com duas camadas ocultas (Modificado de Haykin, 2002).	40
Figura 3.8. Resumo do algoritmo de treinamento para um perceptron. O perceptron “aprende” quando erra na classificação de um vetor de treinamento (Modificada de Kasabov, 1998).	1
Figura 3.9. Fluxo de sinais ressaltando os detalhes do neurônio de saída j (Haykin, 2001)....	43
Figura 3.10. Efeito da normalização do conjunto de treinamento. (a) domínio de treinamento antes da normalização; (b) domínio de treinamento após a normalização. (modificado de Flood & Kartam, 1994).	52

Figura 3.11. Localização espacial de pontos para um caso com sete vetores de treinamento (modificado de Flood & Kartam, 1994).....	54
Figura 3.12. Diferentes alternativas de generalização para um conjunto de treinamento: (a) interpolação deficiente; (b) interpretação literal; (c) identificação de tendência. (modificado de Flood & Kartam, 1994).....	55
Figura 3.13. Monitoramento do desempenho da rede durante o processo de treinamento (modificado de Flood & Kartam, 1994).....	56
Figura 3.14. Tipos de estímulos produzidos de acordo com a função chapéu mexicano (modificada de Braga et al., 2000).....	59
Figura 3.15. Resumo do procedimento de treinamento do algoritmo de Kohonen (Kasabov, 1998).....	60
Figura 4.1. Curva típica tensão-deformação para um metal sob condições de carregamento uniaxial simples (modificada de Desai & Siriwardane, 1984).....	63
Figura 4.2. Superfície de plastificação no espaço de tensões principais.....	64
Figura 4.3. Processo de recuperação dos pares de dados tensão-deformação a partir dos gráficos originais dos testes usados na base de dados.....	68
Figura 4.4. Trajetórias de tensão dos ensaios de tensão usados no modelo neural 1.....	69
Figura 4.5. Distribuição da quantidade de pontos por teste. Em amarelo os testes da areia de Cambria, em vermelho os testes de Monterrey, em azul os testes de Toyoura.....	70
Figura 4.6. Histograma de frequência da quantidade de pontos experimentais por teste.	70
Figura 4.7. Representação dos estados de tensão-deformação necessários para o treinamento da RNA.....	72
Figura 4.8. Correlação entre as respostas da rede <i>Magola 3</i> e os valores experimentais (linha contínua). Em tracejado a linha $y=x$	76
Figura 4.9. Resultados da simulação da rede <i>Magola 3</i> para a areia de Cambria. Os pontos são os dados experimentais, as linhas são a resposta da RNA.	77
Figura 4.10. Resultados da simulação da rede <i>Magola 3</i> para a areia de Monterrey. Os pontos são os dados experimentais, as linhas, a resposta da RNA.	78
Figura 4.11. Resultados da simulação da rede <i>Magola 3</i> para a areia de Toyoura. Em tracejado o eixo hidrostático. À direita os dados experimentais, à esquerda a resposta da rede.	79
Figura 4.12. Representação esquemática dos processos de treinamento <i>estático</i> e <i>recorrente</i> , e de simulação “ <i>Ponto-a-Ponto</i> ” e “ <i>Autônoma</i> ”.....	81

Figura 4.13. Estrutura geral de um conjunto de treinamento que inclui o primeiro ponto experimental. Em preto os dados de entrada e em vermelho os dados de saída.	82
Figura 4.14. Comparação dos resultados para a rede <i>Magola 6</i>	87
Figura 4.15. Resultados da simulação incremental da rede <i>Magola 6</i>	88
Figura 4.16. Histogramas de frequência das deformações e dos acréscimos de deformação do teste 21 para os conjuntos de treinamento e generalização.	90
Figura 4.17. Histograma de frequência dos acréscimos de deformação para a base de dados completa (171 testes).	91
Figura 4.18. Estrutura geral de um conjunto de treinamento que inclui tanto o primeiro como o último ponto experimental. Em preto os dados de entrada e em vermelho os dados de saída.	92
Figura 4.19. Evolução do tempo de treinamento com o número de exemplos para uma rede 19x120x90x3 treinada durante uma iteração com o algoritmo de gradiente conjugado modificado.	93
Figura 4.20. Evolução dos erros de treinamento e de validação de seis redes treinadas com a técnica de treinamento seletivo.	95
Figura 4.21. Gráficos de evolução do treinamento de uma rede 90x60 treinada seletivamente com sete estratégias diferentes.	96
Figura 4.22. Erros de treinamento e validação de quatro redes treinadas seletivamente para a primeira componente do vetor de acréscimos de tensão ($\Delta\sigma_1$).	99
Figura 4.23. Comparação dos resultados de classificação de três redes auto-organizáveis do tipo Kohonen para o conjunto completo de testes.	101
Figura 4.24. Evolução dos erros de treinamento e validação para duas redes do tipo PMC treinadas seletivamente com uma base de dados previamente classificada por uma rede de Kohonen.	103
Figura 4.25. Transformação dos valores do e_{\max} - e_{\min} e do n_s	104
Figura 4.26. Evolução do erro de treinamento em função do número de iterações para a rede <i>Magola 9</i>	105
Figura 4.27. Resultados de simulação estática da rede <i>Magola9</i> treinada com a técnica de subdivisão da base de dados, para os acréscimos de tensão e para as tensões correspondentes.	107
Figura 4.28. Resultados de simulação autônoma da rede <i>Magola9</i> treinada com a técnica de subdivisão da base de dados, para os três componentes dos acréscimos de tensão.	107

Figura 4.29. Resultados de simulação incremental ou autônoma da rede MLP 90x90 <i>Magola9</i> treinada com a técnica de subdivisão da base de dados, para os três componentes de tensão.....	108
Figura 4.30. Simulação do modelo <i>Magola 9</i> para areia Karlsruhe. Trajetórias CTC.....	109
Figura 4.31. Simulação do modelo <i>Magola 9</i> para a areia Hostun. Trajetórias CTC.....	110
Figura 4.32. Simulação do modelo <i>Magola 9</i> para a areia Hostun. Trajetórias TC.....	111
Figura 4.33. Resultados de simulação do <i>Modelo 9</i> para a areia Hostun. Trajetórias tipo RTE.	111
Figura 4.34. Resultados de simulação do modelo <i>Magola 9</i> para areia Toyoura. Trajetórias CTC.....	112
Figura 4.35. Resultados de simulação do modelo <i>Magola 9</i> para a areia Toyoura. Trajetórias TC.....	112
Figura 4.36. Resultados de simulação do modelo <i>Magola 9</i> para areia Toyoura. Trajetórias CTE.....	112
Figura 4.37. Resultados de simulação do modelo <i>Magola 9</i> para a areia Toyoura. Trajetórias TE.....	113
Figura 4.38. Resultados de simulação do modelo <i>Magola 9</i> , areia Toyoura. Trajetórias RTC.	113
Figura 4.39. Resultados de simulação do modelo <i>Magola 9</i> para areia Toyoura. Trajetórias RTE.....	113
Figura 4.40. Resultados de simulação do modelo <i>Magola 9</i> para a areia Toyoura. O teste 1 corresponde a uma trajetória HC, os restantes correspondem a trajetórias no plano octaédrico.....	113
Figura 4.41. Resultados de simulação do modelo <i>Magola 9</i> para areia Cambria. Trajetórias CTC.....	114
Figura 4.42. Resultados de simulação do modelo <i>Magola 9</i> para areia Cambria. Trajetórias TC.....	114
Figura 4.43. Resultados de simulação do modelo <i>Magola 9</i> para areia Cambria. Trajetórias HC.....	114
Figura 4.44. Resultados de simulação do modelo <i>Magola 9</i> para areia Cambria. Trajetórias D.P.....	114
Figura 4.45. Resultados de simulação do modelo <i>Magola 9</i> para areia Cambria. Trajetórias CTE.....	115

Figura 4.46. Resultados de simulação do modelo <i>Magola 9</i> para areia Cambria. Trajetórias TE.....	115
Figura 4.47. Resultados de simulação do modelo <i>Magola 9</i> para areia Monterrey. Trajetórias TC.....	115
Figura 4.48. Resultados de simulação do modelo <i>Magola 9</i> para a areia Monterrey. Trajetórias tipo TE.....	115
Figura 4.49. Resultados de simulação do modelo <i>Magola 9</i> para a areia Monterrey. Trajetórias no plano octaédrico.	116
Figura 4.50. Resultados de simulação do modelo <i>Magola 9</i> para a areia Calcárea. Trajetórias tipo CTC.....	116
Figura 4.51. Resultados de simulação do modelo <i>Magola 9</i> para a areia Ottawa. Trajetórias tipo CTC.....	116
Figura 4.52. Resultados de simulação do modelo <i>Magola 9</i> para a areia Antioch. Trajetórias tipo CTC.....	117
Figura 4.53. Resultados de simulação do modelo <i>Magola 9</i> para a areia Chattahoochee. Trajetórias CTC.....	117
Figura 4.54. Resultados de simulação do modelo <i>Magola 9</i> para a areias San Fernando. Trajetórias CTC.....	117
Figura 4.55. Resultados de simulação das areias SLB e Ticino. Trajetórias tipo HC.....	117
Figura 4.56. Histograma de frequência dos erros de simulação incremental das tensões para a rede <i>Magola9</i>	118
Figura 4.57. Valores do e_{max} e do e_{min} para a areia de Toyoura determinados a partir de 13 estudos independentes usando procedimentos padronizados da JGS – <i>Japanese Geotechnical Society</i> (modificada de Cubrinovski & Ishihara, 2002).....	119
Figura 4.58. Histograma de frequência do e_{max} - e_{min} de todos os exemplos de treinamento da rede <i>Magola9</i>	119
Figura 4.59. Diferentes casos para analisar a sensibilidade do modelo <i>Magola9</i> à variações do e_{max} e do e_{min}	120
Figura 4.60. Gráfico de coeficientes de correlação para cada uma das componentes de tensão e para cada um dos casos considerados na análise de sensibilidade do modelo <i>Magola9</i>	121
Figura 4.61. Curvas de frequência acumulada do e_{max} - e_{min} para os casos 3, 4, 5 e 6 da rede <i>Magola9</i>	122

Figura 4.62. Curvas de frequência acumulada do $e_{\max}-e_{\min}$ para os casos 1, 2, 7 e 8 da rede <i>Magola9</i> .	123
Figura 4.63. Transformação do $e_{\max}-e_{\min}$ para o modelo neural 5.	125
Figura 4.64. Curvas de frequência acumulada para os valores das tensões principais e a tensão de confinamento inicial da base de dados do modelo neural 5.	125
Figura 4.65. Evolução do erro de treinamento da rede <i>Magola10</i> .	126
Figura 4.66. Resultados de simulação incremental das tensões para o modelo <i>Magola10</i> .	127
Figura 4.67. Gráficos tensão-deformação para a areia Karlsruhe. Testes CTC	127
Figura 4.68. Gráficos tensão-deformação para a areia Hostun. Testes CTC	132
Figura 4.69. Gráficos tensão-deformação para a areia Hostun. Testes TC.	134
Figura 4.70. Gráficos tensão-deformação para a areia Hostun. Testes RTE.	135
Figura 4.71. Gráficos tensão-deformação para a areia Toyoura. Testes CTC	135
Figura 4.72. Gráficos tensão-deformação para a areia Toyoura. Testes TC, CTE, TE, RTC, RTE e no plano octaédrico.	137
Figura 4.73. Gráficos tensão-deformação para a areia de Cambria. Testes CTC e TC.	139
Figura 4.74. Gráficos tensão-deformação para a areia de Cambria. Testes de deformação plana	140
Figura 4.75. Gráficos tensão-deformação para a areia de Cambria. Testes CTE e TE.	141
Figura 4.76. Gráficos tensão-deformação para a areia de Monterrey. Testes TC, TE e testes no plano octaédrico.	142
Figura 4.77. Gráficos tensão-deformação para a areia calcária. Testes CTC.	144
Figura 4.78. Gráficos tensão-deformação para a areia de Ottawa. Testes CTC.	145
Figura 4.79. Gráficos tensão-deformação para a areia de Antioch. Testes CTC.	146
Figura 4.80. Gráficos tensão-deformação para a areia de Chatahoochee. Testes CTC.	147
Figura 4.81. Gráficos tensão-deformação para a areia de San Fernando. Testes CTC.	147
Figura 4.82. Gráficos tensão-deformação para testes de compressão hidrostática (HC) nas areias de Toyoura, SLB, Ticino e Cambria.	148
Figura 4.83. Histograma de frequência para o erro de simulação incremental da rede <i>Magola10</i> .	148
Figura 4.84. Relação entre a tensão de confinamento inicial (σ_{c0}) e a tensão octaédrica na ruptura (p_{rup}) para todos os testes da base de dados.	151

Figura 4.85. Evolução do invariante $X = \frac{I1 \times I2}{I3}$ para o teste 70.	152
Figura 4.86. Relação entre a tensão octaédrica na ruptura (p_{rup}) e o invariante X na ruptura para areias na condição densa (a) e na condição fofa (b).....	153
Figura 4.87. Histogramas de freqüência das primeiras quatro entradas da base de dados do modelo neural 6.	154
Figura 4.88. Evolução do erro de treinamento para cinco modelos neurais.	155
Figura 4.89. Evolução do erro de treinamento da rede <i>Magola11</i>	156
Figura 4.90. Simulação incremental das tensões para o modelo <i>Magola11</i>	157
Figura 4.91. Gráficos tensão-deformação para cinco testes que não foram adequadamente simulados pelo modelo <i>Magola11</i>	157
Figura 4.92. Curvas tensão-deformação dos testes para os quais a rede <i>Magola11</i> apresentou melhora significativa do desempenho em comparação com a rede <i>Magola10</i>	159
Figura 4.93. Histograma de freqüência do erro de simulação autônoma da rede <i>Magola11</i>	163
Figura 5.1. Representação gráfica do processo iterativo para solucionar o n-ésimo acréscimo de carga usando um modelo constitutivo neural dentro de um código de elementos finitos.....	166
Figura 5.2. Arquitetura da rede <i>Magola11</i>	169
Figura 5.3. Algoritmo de implementação da RN constitutiva <i>Magola11</i> no programa ALLFINE.	171
Figura 5.4. Histogramas de freqüência das três componentes dos acréscimos de deformação usados no treinamento da rede <i>Magola11</i>	172
Figura 5.5. Diagrama mostrando um elemento cúbico de 8 nós.	173
Figura 5.6. Comparação de simulações do sistema híbrido ALLFINE-RNA para diferentes incrementos de carga.	175
Figura 5.7. Simulações do modelo híbrido ALLFINE-NN. Areia Karlsruhe, trajetórias CTC	181
Figura 5.8. Simulações do modelo híbrido ALLFINE-NN. Areia Hostun, trajetórias CTC .	183
Figura 5.9. Simulações do modelo híbrido ALLFINE-NN. Areia Hostun, trajetórias TC	185
Figura 5.10. Simulações do modelo híbrido ALLFINE-NN. Areia Hostun, trajetórias RTE	185
Figura 5.11. Simulações do modelo híbrido ALLFINE-NN. Areia Toyoura, trajetórias CTC	186

Figura 5.12. Simulações do modelo híbrido ALLFINE-NN. Areia Toyoura, trajetórias TC	186
Figura 5.13. Simulações do modelo híbrido ALLFINE-NN. Areia Toyoura, trajetórias CTE	187
Figura 5.14. Simulações do modelo híbrido ALLFINE-NN. Areia Toyoura, trajetórias TE	187
Figura 5.15. Simulações do modelo híbrido ALLFINE-NN. Areia Toyoura, trajetórias RTC	187
Figura 5.16. Simulações do modelo híbrido ALLFINE-NN. Areia Toyoura, trajetórias RTE	187
Figura 5.17. Simulações do modelo híbrido ALLFINE-NN. Areia Toyoura, trajetórias no plano octaédrico.	188
Figura 5.18. Simulações do modelo híbrido ALLFINE-NN. Areia Cambria TC	189
Figura 5.19. Simulações do modelo híbrido ALLFINE-NN. Areia Cambria deformação plana	189
Figura 5.20. Simulações do modelo híbrido ALLFINE-NN. Areia Cambria CTE	190
Figura 5.21. Simulações do modelo híbrido ALLFINE-NN. Areia Cambria TE	190
Figura 5.22. Simulações do modelo híbrido ALLFINE-NN. Areia Monterrey TC	191
Figura 5.23. Simulações do modelo híbrido ALLFINE-NN. Areia Monterrey TE	191
Figura 5.24. Simulações do modelo híbrido ALLFINE-NN. Areia Monterrey plano octaédrico	191
Figura 5.25. Simulações do modelo híbrido ALLFINE-NN. Areia Calcária CTC	192
Figura 5.26. Simulações do modelo híbrido ALLFINE-NN. Areia Ottawa CTC	193
Figura 5.27. Simulações do modelo híbrido ALLFINE-NN. Areia Antioch CTC	194
Figura 5.28. Simulações do modelo híbrido ALLFINE-NN. Areia Chattahoochee CTC	194
Figura 5.29. Simulações do modelo híbrido ALLFINE-NN. Areia San Fernando CTC	194
Figura 5.30. Simulações do modelo híbrido ALLFINE-NN. Ensaio HC	195
Figura 5.31. Esquema da geometria e condições de carregamento para a simulação de uma amostra cúbica num ensaio triaxial.	196
Figura 5.32. Resposta tensão – deformação de uma amostra cúbica simulada com o sistema híbrido ALLFINE-RN.	197
Figura 5.33. Malha de elementos finitos usada para a modelagem do comportamento de uma sapata assente em areia densa.	198
Figura 5.34. Gráficos de acréscimos de tensão vertical e de deslocamentos verticais para uma sapata com deslocamento pequeno imposto, analisada com o SH.	199

Figura 5.35. Gráficos de acréscimos de tensão vertical e de deslocamentos verticais para uma sapata com deslocamento grande imposto, analisada com o SH.	200
Figura 5.36. Simulação do teste 26 (Toyoura, CTC) realizada com o modelo Cam-Clay para calibração de parâmetros.	201
Figura 5.37. Gráficos de acréscimos de tensão vertical e de deslocamentos verticais para uma sapata com deslocamento pequeno imposto, analisada com o modelo Cam-Clay.	201
Figura 5.38. Gráficos de acréscimos de tensão vertical e de deslocamentos verticais para uma sapata com deslocamento grande imposto, analisada com o modelo Cam-Clay implementado no programa ALLFINE.	202
Figura 5.39. Gráficos de acréscimos de tensão vertical e de deslocamentos verticais para uma sapata com um carregamento prescrito de pequena magnitude, analisada com o SH. ..	203
Figura 5.40. Gráficos de acréscimos de tensão vertical e de deslocamentos verticais para uma sapata com um carregamento prescrito de magnitude média, analisada com o SH.	203
Figura 5.41. Gráfico carga-deslocamento para uma sapata simples analisada com o SH e o modelo Cam-Clay.	204

LISTA DE ABREVIATURAS E SÍMBOLOS

ABREVIATURAS

BVP	Boundary Value Problem (Problema de Contorno)
CTC	Teste Triaxial Convencional de Compressão
CTE	Teste triaxial Convencional de Extensão
DP	Deformação Plana
HC	Teste de Compressão Hidrostática
MEF	Método dos Elementos Finitos
MSE	Erro Quadrático Médio
PMC	Perceptron de Múltiplas Camadas
RNA	Rede Neural Artificial
RTC	Teste Triaxial de Compressão por Redução
RTE	Teste Triaxial de Extensão por Redução
SH	Sistema Híbrido ALLFINE-Rede Neural
SOM	Mapas Auto-Organizáveis (Self Organizing Maps)
TC	Teste Triaxial de Compressão com p constante
TE	Teste Triaxial de Extensão com p constante

SÍMBOLOS

Letras Arábicas

[B]	Matriz de transformação deformação-deslocamento
b_k	Bias associado à camada k
C_c	Coefficiente de curvatura
C_u	Coefficiente de uniformidade
[D]	Matriz constitutiva tensão-deformação
D_r	Densidade relativa do solo
D_{50}	Diâmetro médio de partícula
E	Módulo de Young
e_0	Índice de vazios inicial
e_{max}	Índice de vazios máximo
e_{min}	Índice de vazios mínimo
$F(\sigma)$	Função de plastificação

$H(n)$	Matriz hessiana
I_1	Primeiro invariante de tensão
I_2	Segundo invariante de tensão
I_3	Terceiro invariante de tensão
$J(n)$	Matriz jacobiana
k_N	Quilo Newton
k_{Pa}	Quilo Pascal
$[k_t]$	Matriz de rigidez
NN	Operador Neural
n_s	Fator de forma dos grãos da areia
p	Tensão normal octaédrica
$[Q]$	Matriz de rotação
$Q(\sigma)$	Superfície de potencial plástico
v_k	Nível de ativação do neurônio k
w_{km}	Peso de conexão entre as camadas k e m
X	Invariante de tensão do critério de ruptura de Matsuoka-Nakai
Y_0	Tensão de plastificação

Letras Gregas

$d\sigma$	Acréscimo de tensão
$d\varepsilon$	Acréscimo de deformação
$\Delta\varepsilon_{ot}$	Tamanho ótimo de acréscimo de deformação
δ	Gradiente
ε	Deformação
ε_{max}	Deformação principal máxima
ε_1	Deformação principal na direção 1
ε_2	Deformação principal na direção 2
ε_3	Deformação principal na direção 3
ε_{vol}	Deformação volumétrica
$\mathcal{E}(w)$	Função de custo ou índice de desempenho.
$\phi(.)$	Função de transferência
ν	Coefficiente de Poisson

κ	Coeficiente de inchamento
λ	Coeficiente de inclinação virgem
η	Relação entre a tensão octaédrica cisalhante e a tensão octaédrica média (no capítulo 4 e subseqüentes)
η	Taxa de aprendizagem (no capítulo 3)
θ	Ângulo de Lade
σ	Tensão
σ_{co}	Tensão de confinamento inicial
σ_1	Tensão principal na direção 1
σ_2	Tensão principal na direção 2
σ_3	Tensão principal na direção 3
τ_{oct}	Tensão octaédrica cisalhante
∇	Operador gradiente

CAPÍTULO 1

INTRODUÇÃO

Em termos de engenharia, trabalhar com materiais de origem natural sempre foi um grande desafio. Devido a fatores que ocorrem desde a sua gênese até as condições do meio ambiente em que se encontram atualmente, faz com que a complexidade destes materiais seja maior que a dos que tiveram seu processo de transformação controlado pelo homem. Quanto mais se conhece sobre estes materiais, maior controle se tem sobre o projeto e a obra e, conseqüentemente, soluções de melhor qualidade técnica e viabilidade econômica podem ser adotadas. Por se tratar de materiais que apresentam alta complexidade, devido ao fato de serem naturais e terem seu comportamento influenciado por diversos fatores (composição granulométrica e mineralógica, umidade, índice de vazios, estado de tensões, trajetória de tensões, entre outros), as teorias clássicas de mecânica do contínuo, baseadas em conceitos de elasticidade, plasticidade e viscosidade, sofreram algumas modificações a fim de serem adaptadas a solos e rochas. Porém, as equações originadas deste processo foram se tornando cada vez mais complexas, fazendo com que a sua solução precisasse de métodos aproximados ou de novas simplificações. Assim, diversos pesquisadores ainda procuram uma forma de modelar satisfatoriamente o comportamento dos geomateriais, que incorpore as principais características do material e dos carregamentos que o solicitam (Dyminski, 2000).

1.1 – DEFINIÇÃO DO PROBLEMA

Uma das formas “não tradicionais” de se modelar o comportamento mecânico dos solos é a utilização de redes neurais (RN's). Assim, pode-se procurar simular a relação tensão-deformação dos solos, nas mais diversas condições, sem a necessidade de se explicitar a equação matemática que o faria (Dyminski, 2000). As RN's podem ser definidas como

sistemas paralelos compostos por unidades de processamento simples (nodos) que calculam determinadas funções matemáticas (geralmente não lineares). Tais unidades são dispostas em uma ou várias camadas, e interligadas por um grande número de conexões geralmente unidirecionais (Braga et al., 2000).

Em RN's o problema usual passa primeiro por uma fase de aprendizagem, em que um conjunto de exemplos é apresentado à rede, a qual extrai automaticamente as características necessárias para representar a informação fornecida. A capacidade de aprender por meio de exemplos e de generalizar a informação aprendida é, sem dúvida, o atrativo principal das RN's (Braga et al., 2000). Outras propriedades importantes das RN's em aplicações de engenharia são as seguintes (Ghaboussi, 1992):

- São capazes de produzir respostas corretas, ou muito aproximadamente corretas, quando apresentadas com estímulos parcialmente incorretos ou incompletos.
- Têm grande capacidade de generalização, ou seja, de produzir respostas corretas quando apresentadas com informação nova.
- As RN's são capazes de extrair informação a partir de dados com ruídos.
- Como modelos computacionais, as RN's são muito robustas. Nenhuma unidade de processamento é essencial na operação da RN. No caso de remoção de uma unidade de processamento, a RN só vai experimentar uma diminuição no seu desempenho.

Em seções posteriores serão abordados, com maior profundidade, alguns aspectos teóricos relevantes das RN's.

Um detalhe importante dos modelos computacionais baseados em RN's é a sua facilidade para desenvolver modelos que possam mapear relações tipo tensão-deformação ou a inversa, ou seja, deformação-tensão. De fato, há uma flexibilidade muito grande na definição das variáveis que vão ser usadas nas unidades de entrada e de saída. De alguma maneira, nesta flexibilidade está o ponto no qual o pesquisador toma as decisões sobre como vai representar a informação contida no comportamento do material. Esta é uma das diferenças mais importantes entre os modelos constitutivos baseados em RN's e os modelos convencionais de tipo paramétrico. O único requerimento básico é que a informação representada nas unidades de entrada e de saída, deve ser definida de tal forma que o mapeamento entre elas seja único. Por exemplo, para materiais cujo comportamento é fortemente dependente da trajetória de tensões, não são suficientes os estados de tensão e deformação para definir de uma forma única o estado do material. Nestes casos, as unidades

de entrada devem conter informação que represente de alguma forma o estado imediatamente anterior de tensões e deformações (Ghaboussi, 1992).

No caso dos solos, as relações tensão-deformação são em geral do tipo não linear, podendo-se representar de uma forma simbólica como na Figura 1.1.

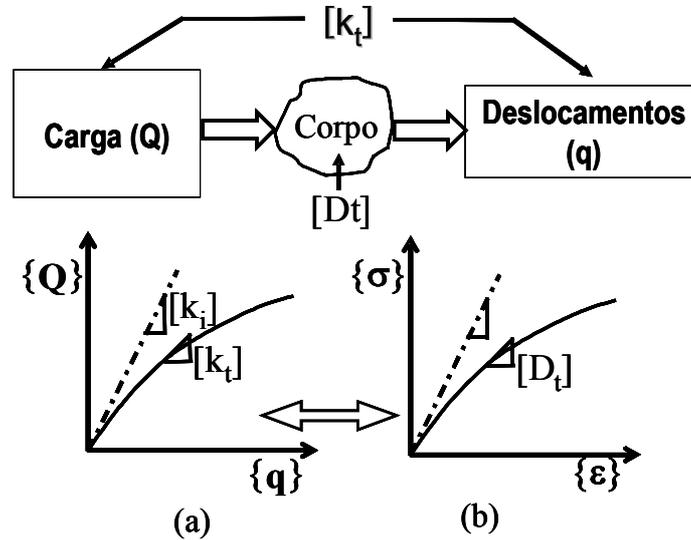


Figura 1.1. Representação geral do comportamento não linear: (a) carga-deslocamento; (b) tensão-deformação (Modificado de Desai & Siriwardane, 1984).

Este comportamento não linear manifesta-se por meio de mudanças na geometria (não linearidade geométrica) e nas propriedades do material (não linearidade física). A Figura 1.1 é uma representação simbólica da relação entre uma excitação (carga Q) e uma resposta correspondente (deslocamento, q). De uma maneira incremental, uma aproximação para esta relação pode se expressar segundo a Eq.1.1.

$$[k_t]\{dq\}=\{dQ\} \tag{1.1}$$

onde $[k_t]$ é a matriz de rigidez, $\{dq\}$ é o vetor de deslocamentos, e $\{dQ\}$ é o vetor de carregamentos. Se o material for linear,

$$[k_t]=[k_i] \tag{1.2}$$

onde $[k_i]$ é a matriz de rigidez inicial cujos coeficientes se mantêm constantes para qualquer magnitude de carregamento.

Em procedimentos numéricos como o método dos elementos finitos (MEF), a matriz de rigidez $[k_t]$ é usualmente expressa como na Eq.1.3.

$$[k_t]=\iiint_V [B]^T [D_t] [B] dV \tag{1.3}$$

onde $[B]$ é a matriz de transformação deformação–deslocamento, e V é o volume da região ou elemento. A matriz $[D_t]$ na Eq.1.3 é chamada matriz constitutiva ou matriz tensão–deformação, e representa o comportamento do material. Como o seu nome o indica, esta matriz representa a ligação entre acréscimos de tensões e deformações segundo a Eq.1.4.

$$\{d\sigma\} = [D_t]\{d\varepsilon\} \quad (1.4)$$

onde $\{d\sigma\}$ e $\{d\varepsilon\}$ são os vetores que representam os incrementos de tensão e deformação (Desai e Siriwardane, 1984).

Um dos fatores causadores da não linearidade é o estado de tensões (Figura 1.1). Na medida que o nível de carregamentos aumenta ou diminui, o material experimenta mudanças contínuas nas suas características físicas. Para um instante qualquer durante a ocorrência de uma mudança no carregamento, o material comporta-se diferente do que para um instante imediatamente anterior. Em outras palavras, os parâmetros mecânicos que definem a rigidez do material, experimenta mudanças permanentes na medida que os carregamentos vão mudando (ver Eqs.1.5 e 1.6).

$$[k_t]=[k_t(D_t)] \quad (1.5)$$

$$[D_t]=[D_t(\sigma)] \quad (1.6)$$

Além do estado de tensões, existem outros parâmetros que influem no comportamento não linear dos geomateriais. Os mais relevantes são: estado inicial de tensão, condição física do material (densidade, umidade, índice de vazios), tipo de carregamento (estático, dinâmico, repetitivo), velocidade de carregamento, trajetória de tensão e existência de descontinuidades (juntas, fraturas, trincas) (Desai e Siriwardane, 1984).

O ponto chave na modelagem constitutiva dos materiais é a definição da matriz constitutiva $[D_t]$ de tal forma que possa se usar em técnicas numéricas como MEF. No caso mais geral, com seis componentes de deformação e seis componentes de tensão, a matriz constitutiva teria 36 coeficientes, ou 21 considerando a simetria desta matriz.

Caso o material seja elástico linear, estes coeficientes são constantes e os valores dependem do módulo de elasticidade, E , e do coeficiente de Poisson, ν . Por outro lado, caso o material seja modelado segundo a teoria da plasticidade, a matriz constitutiva teria uma componente elástica, e uma componente plástica de coeficientes variáveis cujos valores dependem de três funções denominadas: função de plastificação, F ; lei de endurecimento, H ; e

função de potencial plástico ou lei de fluxo, G. A definição destas funções caracteriza um modelo constitutivo.

Numa abordagem com RN's, o modelo neural seria usado como operador constitutivo dentro de um código de elementos finitos no lugar da relação constitutiva $[D_t]$ ou sua inversa. Tal rede seria treinada para mapear um vetor de saída (acréscimos de tensões, por exemplo) em um estágio de carregamento determinado, a partir de um vetor de entradas (acréscimos de deformações e outras informações).

O conjunto de dados para o treinamento desta rede não terá de conter todas as possíveis trajetórias de tensão (o que seria impossível), mas deverá incluir um número suficiente de trajetórias para que a rede possa aprender o comportamento constitutivo não linear do material. Este conjunto de dados que inclui todo o conhecimento relevante é chamado de “conjunto abrangente”. Uma RN treinada com um “conjunto abrangente” de dados sobre o comportamento do material, deveria ser capaz de usar sua capacidade de generalização para produzir respostas aceitáveis quando for consultada com trajetórias de tensão não incluídas no conjunto de dados de treinamento (Ghaboussi, 1992).

1.2 – OBJETIVO

Nesta tese pretende-se criar, treinar e simular uma rede neural artificial que possa reproduzir, da maneira mais geral possível, o comportamento constitutivo das areias limpas submetidas a carregamentos do tipo monotônico - drenado. Tal modelo neural deverá posteriormente ser acoplado a um código de elementos finitos com o propósito de possibilitar sua aplicação na solução de problemas de contorno. A rede será treinada com uma base de dados composta por um conjunto amplo de ensaios de laboratório, correspondentes a diversas trajetórias de tensão e para vários tipos de areia.

Como operador constitutivo, a rede neural deverá ser capaz de prever corretamente os acréscimos de tensão correspondentes a um determinado conjunto de acréscimos de deformação, permitindo a simulação integral de trajetórias tridimensionais de tensão para uma ampla gama de condições físicas iniciais.

1.3 – METODOLOGIA

Como é usual neste tipo de pesquisas, a metodologia baseia-se numa seqüência de passos ou etapas, que visam levar o conhecimento adquirido desde os níveis mais gerais até os mais específicos. Em relação aos recursos necessários para o desenvolvimento desta pesquisa, estes se podem classificar em dois tipos. O primeiro corresponde à base de dados de ensaios para diferentes trajetórias e para diferentes tipos de areia. O segundo tipo de recurso está relacionado com a logística necessária para o processamento da informação. Estes recursos são um computador e um programa computacional. Em princípio será utilizado o programa Matlab, o qual possui um módulo especializado em redes neurais. As etapas propostas para o desenvolvimento desta tese são descritas a seguir:

1.3.1 – ESCOLHA DO CONJUNTO DE DADOS

A base de dados para o treinamento desta rede neural artificial (RNA) não terá de conter todas as possíveis trajetórias de tensão, mas deverá incluir um número suficiente delas para que a RNA possa aprender o comportamento constitutivo do solo. Assim a RNA seria capaz de usar sua capacidade de generalização para produzir respostas aceitáveis quando for consultada com trajetórias de tensão não incluídas no conjunto de dados de treinamento (Ghaboussi, 1992). Em consequência, o conjunto de dados deve ser do tipo “abrangente”, ou seja, deve conter toda a informação relevante para que a RN possa aprender o comportamento constitutivo do solo. Este conjunto de dados deverá incluir diversas trajetórias de tensão, estados iniciais de tensão, e condições físicas dos materiais. Quanto ao tipo de carregamento (estático, dinâmico, repetitivo) serão considerados apenas os resultados de ensaios com condições de carregamento estático e eventualmente com ciclos de descarregamento e re-carregamento.

Posteriormente, será realizado um processo de normalização dos dados para ajustá-los ao intervalo de -1 a 1. Esta normalização, tal como será explicado mais adiante está de acordo com a função de ativação escolhida. Isso acaba por tornar mais eficiente a etapa de treinamento da rede. Depois de se normalizar os dados, introduzi-los à rede e serem obtidas as saídas, as mesmas estarão também em sua forma normalizada. Para poderem ser analisadas e interpretadas, as saídas deverão ser transformadas novamente para seu domínio original (Dyminski, 2000).

A base de dados já coletada está composta por 12 areias diferentes (ver Tabela 1.1), cujos ensaios triaxiais foram digitalizados a partir dos gráficos originalmente apresentados nas referências mencionadas na própria tabela, correspondendo a um total de 171 testes cujas características gerais são apresentadas na Tabela A.1.

Tabela 1.1. Relação das areias consideradas na base de dados.

Material	e_{max}	e_{min}	$e_{max}-e_{min}$	Referência
Toyoura	1.1	0.61	0.49	Felin, 2002
	0.988	0.616	0.372	Cubrinovski & Ishihara, 2002
	0.973	0.612	0.361	Hoque & Tatsuoka, 1998
Calcareous Sand	1.66	1.01	0.65	Colliat-Dangus et al., 1998
Hostun	0.998	0.655	0.343	Colliat-Dangus et al., 1998
Hostun S32	0.782	0.548	0.234	Viggiani et al., 2000
Hostun S28	0.986	0.641	0.345	
Hostun	1.09	0.61	0.48	Felin, 2002
Ottawa	0.88	0.49	0.39	
Ottawa 50/200	0.805	0.55	0.255	Cubrinovski & Ishihara, 2002
Ottawa F-95	0.865	0.58	0.285	
Ottawa	0.78	0.48	0.3	
Ottawa C-109	0.83	0.5	0.33	
Ottawa	0.978	0.521	0.457	Penumadu & Zhao, 1999
Ticino	1.05	0.6	0.45	Felin, 2002
	0.934	0.582	0.352	Bellotti et al., 1996
	0.96	0.59	0.37	Hoque & Tatsuoka, 1998
SLB	0.9	0.49	0.41	Felin, 2002
	0.79	0.49	0.3	Hoque & Tatsuoka, 1998
Cambria	0.8	0.51	0.29	Oshiai & Lade, 1983
	0.767	0.538	0.229	Cubrinovski & Ishihara, 2002
Nevada 50/80	0.858	0.581	0.277	
Nevada 80/200	0.94	0.617	0.323	

Material	ϵ_{\max}	ϵ_{\min}	$\epsilon_{\max}-\epsilon_{\min}$	Referência
Monterrey	0.86	0.565	0.295	Lade & Duncan, 1973
Chattahoochee	0.978	0.521	0.457	Penumadu & Zhao, 1999
Antioch	1.14	0.75	0.39	Lee, 1970
San Fernando Dam	1.49	0.61	0.88	Lee et al., 1975
Kalsruhe Sand	0.843	0.529	0.3140	Kolymbas & Wu, 1990
	1	0.53	0.47	Felin, 2002
Shirasu	1.659	0.878	0.781	Haruyama, 1987

Tal como será apresentado no Capítulo 4, para os modelos neurais mais avançados, a base de dados de treinamento foi re-classificada e reduzida. Na Tabela 1.2 é apresentada a relação de testes para cada tipo de areia, enquanto que na Tabela 1.3. é apresentada a relação dos diferentes testes considerados classificados pelo tipo de trajetória de tensão.

Tabela 1.2. Relação das areias consideradas na base de dados re-classificada

AREIA	NÚMERO DE TESTES
Antioch	3
Calcareous	6
Cambria	18
Chattahoochee	2
Hostun	23
Karsruhe	28
S.Fernando Dam	2
Monterrey	12
Ottawa	4
SLB	1
Ticino	1
Toyoura	22
TOTAL	122

Tabela 1.3. Relação dos testes considerados na base de dados re-classificada e as trajetórias correspondentes.

TRAJETÓRIA	NÚMERO DE TESTES
CTC	69
CTE	6
Plane Strain	3
HC	5
RTC	2
RTE	4
TC ($\theta=0$)	16
TE ($\theta=60$)	3
$\theta=9$	1
$\theta=12$	1
$\theta=15$	1
$\theta=30$	2
$\theta=36$	1
$\theta=45$	3

1.3.2 – ESCOLHA DA ARQUITETURA DA REDE

Nesta fase são definidos o número e tipo de entradas a serem usadas, assim como o tipo de saídas necessárias. Segundo as informações obtidas da bibliografia analisada, as redes do tipo PMC (Perceptron de Múltiplas Camadas) com retropropagação do erro parecem ser as melhores ferramentas para a modelagem de problemas como o proposto neste trabalho. Como um aproximador universal de funções, um perceptron de múltiplas camadas pode ser visto como um veículo prático para realizar um mapeamento não-linear de entrada-saída de natureza geral. No entanto, a definição da arquitetura ótima para um problema específico de aproximação não é uma questão trivial, obrigando o usuário a testar sistematicamente diferentes arquiteturas até conseguir uma que apresente as melhores características em termos de tempo de aprendizagem, facilidade de implementação e capacidade de generalização.

1.3.3 – TREINAMENTO E TESTE DA REDE

Inicialmente, deve-se dividir o conjunto de dados em um conjunto de treinamento e um conjunto de teste. O conjunto de treinamento será composto de exemplos que serão utilizados na fase de treinamento da rede neural. Já o conjunto de teste será composto de exemplos que serão utilizados na tarefa de testar a capacidade de generalização da rede neural. Cabe salientar que os exemplos de teste não deverão ter sido utilizados no treinamento da rede, ou seja, o conjunto de treinamento e o conjunto de teste são formados por exemplos distintos entre si (Dyminski, 2000).

A primeira etapa consiste na apresentação dos exemplos do conjunto de treinamento ao sistema. A rede processará os parâmetros de entrada relativos a estes exemplos, fornecendo então as respostas (saídas da rede) a este estímulo. Estas saídas da RN deverão ser comparadas com os valores reais dos parâmetros de saída correspondentes aos exemplos do conjunto de treinamento, e desta comparação será obtido um valor de erro da fase de treinamento (Dyminski, 2000).

Procura-se, então minimizar o erro de treinamento por meio de um algoritmo matemático, e ao mesmo tempo controlar a capacidade de generalização da rede mediante a apresentação periódica de exemplos do conjunto de teste, obtendo-se assim o chamado erro de teste, através da comparação das saídas da rede com as respostas desejadas correspondentes aos exemplos de teste. Monitorando-se os dois valores de erro (de treinamento e de teste), deve-se procurar chegar a um ponto ótimo, equivalente à minimização de ambos os erros, em que a rede estará bem treinada e com sua capacidade de generalização mantida (Dyminski, 2000).

Assim, tem-se como objetivo da fase de treinamento ajustar da melhor maneira possível os valores dos pesos sinápticos da rede, procurando fazer com que as saídas fornecidas pela RN estejam bastante próximas dos valores reais de saída correspondentes a cada um dos exemplos apresentados, sem que se perca sua capacidade de generalização.

1.3.4 - SIMULAÇÃO FINAL E IMPLEMENTAÇÃO NO PROGRAMA DE ELEMENTOS FINITOS

O processo descrito nos passos anteriores deverá se repetir para diferentes arquiteturas neurais e diferentes formas de apresentação dos dados. Não é possível definir *a priori* qual arquitetura ou estratégia de treinamento vão apresentar os melhores resultados. Durante o

processo vão se descartando as redes que não oferecem resultados satisfatórios para chegar no final com um modelo neural robusto que possa ser implementado dentro de um código de elementos finitos como operador constitutivo. Neste caso, as unidades de processamento na camada de entrada do modelo podem, por exemplo, representar estados de tensão-deformação e acréscimos de deformação; enquanto as unidades de processamento na camada de saída representariam os correspondentes acréscimos de tensão (Ghaboussi, 1992). A unicidade do mapeamento entrada-saída dependeria da incorporação, no conjunto de entrada, de fatores que descrevessem adequadamente o tipo e estado inicial do solo, e a capacidade de generalização do modelo seria verificada mediante a realização de simulações nas quais a rede seja obrigada a gerar trajetórias completas de tensão compreendidas dentro do espaço de trajetórias usadas para o treinamento da mesma.

A implementação do modelo neural dentro de um programa de elementos finitos requer um grande esforço de programação e posteriormente de um processo intenso de verificação e validação, no qual múltiplas simulações deverão ser feitas antes de poder aprovar os resultados. Tal processo será realizado usando como plataforma o programa ALLFINE, solucionando alguns exemplos simples de problemas de contorno que são apresentados em capítulos posteriores. Pesquisas futuras deverão focar na análise da eficiência destes sistemas híbridos, e no melhoramento do seu desempenho de modo a os fazer competitivos em termos de velocidade de processamento principalmente.

1.4 – ESCOPO DA TESE

Seguindo o procedimento comum para este tipo de trabalho, o texto da tese foi estruturado em capítulos da seguinte maneira:

- Capítulo 1. Introdução. Apresenta-se ao leitor a definição do problema, mostram-se os objetivos do trabalho e finalmente a metodologia a ser usada.
- Capítulo 2. Revisão bibliográfica. Neste capítulo é inicialmente apresentado um breve histórico da evolução da técnica das redes neurais. Posteriormente se apresenta um resumo das principais aplicações que têm sido reportadas tanto na engenharia civil em geral, como em geotecnia.
- Capítulo 3. São apresentados os fundamentos teóricos das redes neurais. O capítulo compreende uma descrição básica do neurônio como elemento fundamental de processamento de informação, descrevendo de maneira detalhada o procedimento de aprendizado supervisionado. Posteriormente, dá-se ênfase nas redes neurais do tipo

PMC (Perceptron de Múltiplas Camadas) que são as ferramentas usadas nesta tese. No final do capítulo é apresentado um tipo especial de rede neural chamado rede competitiva que foi usada para pré-processar os dados de alguns dos modelos criados.

- Capítulo 4. Neste capítulo se apresentam, de uma forma detalhada, os diferentes modelos neurais criados e treinados para modelar o comportamento constitutivo das areias, a partir da base de dados de ensaios triaxiais com diversas trajetórias de tensão.
- Capítulo 5. Este capítulo descreve o processo de implementação da rede neural constitutiva dentro de um código de elementos finitos. O modelo neural, já incorporado dentro do programa de elementos finitos, age como operador constitutivo em substituição à relação constitutiva elastoplástica tradicional, fazendo o mapeamento entre acréscimos de deformação e acréscimos de tensão ao nível de um ponto (por exemplo o ponto de integração numa análise numérica pelo método dos elementos finitos). Nos resultados das simulações realizadas estão incluídas tanto simulações dos próprios testes da base de dados, como um problema simples de contorno.
- Capítulo 6. Este capítulo resume as conclusões gerais de cada uma das etapas da pesquisa, ressaltando tanto as vantagens como as dificuldades mais marcantes encontradas durante a elaboração do trabalho. Deixam-se algumas sugestões gerais para o estabelecimento de futuras pesquisas nesta área da engenharia geotécnica.

REVISÃO BIBLIOGRÁFICA

2.1 – BREVE HISTÓRICO SOBRE AS REDES NEURAIS

O trabalho sobre redes neurais artificiais tem sido motivado desde o começo pelo reconhecimento de que o cérebro humano processa informações de uma forma diferente do computador digital convencional (Haykin, 2001). O cérebro é o órgão no qual se alojam as habilidades cognitivas dos seres humanos. Foi o cientista espanhol Ramón y Cajal (prêmio Nobel em 1911) quem no final do século XIX apresentou a noção do neurônio como elemento estrutural constitutivo do cérebro (Coronel-Ávila & Parra-Grajales, 2003). O cérebro é um elemento de processamento de informação extremamente complexo, cujo modo de funcionamento é predominantemente paralelo e cujo comportamento não pode ser descrito por meio de modelos simples. Talvez a característica mais importante do cérebro é a sua capacidade de adaptação a estímulos externos por meio de dois procedimentos: criação de novas conexões sinápticas entre neurônios ou modificação das conexões já existentes. Tais procedimentos são simulados de uma forma muito simplificada nas redes neurais artificiais. Como o cérebro, as RNA's são processadores maciçamente paralelos com capacidade de armazenar conhecimento experimental para fazê-lo disponível para uso posterior, mediante um processo de aprendizagem.

Na sua forma mais geral, uma RNA é uma “máquina de inferência” projetada para modelar a maneira como o cérebro realiza uma tarefa particular ou função de interesse; a rede é normalmente implementada utilizando-se componentes eletrônicos ou é simulada por programação em um computador digital. Para alcançarem bom desempenho, as RNA's empregam uma interligação maciça de células computacionais chamadas neurônios, nodos, nós ou unidades de processamento (Haykin, 2001).

A Figura 2.1 apresenta um esquema de uma rede neural simples. Observa-se uma camada de entrada composta por três neurônios, os quais estão completamente interligados por meio de conexões com uma segunda camada de neurônios que representam as saídas da rede. O processo de fluxo de informação através da rede (de esquerda para direita no caso do esquema em questão) se dá de tal forma que cada uma das conexões toma pesos diferentes na medida que diversos vetores de treinamento são apresentados.

O procedimento para realizar o processo de aprendizagem é chamado de algoritmo de aprendizagem, cuja função é modificar os pesos das conexões entre os neurônios (pesos sinápticos) da rede de uma forma ordenada para alcançar um objetivo de projeto desejado (Haykin, 2001). O aprendizado da rede fica então gravado no conjunto final de pesos que as suas conexões adquiriram. Nos capítulos subseqüentes serão abordados em detalhe os conceitos de treinamento e aprendizado.

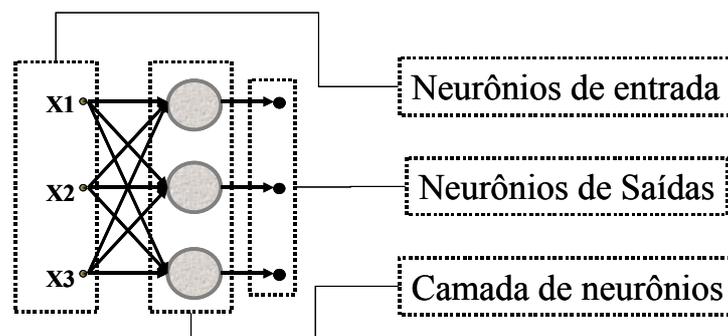


Figura 2.1. Esquema de uma rede neural simples.

A era moderna das redes neurais começou com o trabalho de McCulloch e Pitts em 1943 (“*A logical calculus of the ideas immanent in nervous activity*”) no qual os autores descrevem um cálculo lógico das redes neurais que unificava os estudos de neurofisiologia e da lógica matemática, assumindo que o seu modelo de neurônio seguia uma lei de “tudo ou nada” (booleano). Com um número suficiente dessas unidades simples e conexões sinápticas ajustadas apropriadamente e operando sincronicamente, os pesquisadores mostraram que uma rede assim constituída teria a capacidade de calcular qualquer função (Haykin, 2001).

Em 1949, Hebb publica o seu livro “*The Organization of Behavior*” no qual foi apresentada pela primeira vez a formulação explícita de uma regra de aprendizagem para a modificação sináptica. O livro de Hebb teve grande influência na comunidade de psicólogos, mas lamentavelmente não teve praticamente nenhum impacto na comunidade de engenharia (Haykin, 2001).

Cerca de 15 anos após a publicação do artigo de McCulloch e Pitts, uma nova abordagem para o problema de reconhecimento de padrões foi introduzida por Rosenblath (1958) em seu trabalho sobre o perceptron (“*The perceptron: a probabilistic model for information storage and organization in the brain*”), um método inovador de aprendizagem supervisionada. Em 1960, Widrow e Hoff introduziram o algoritmo do mínimo quadrado médio (LMS – *Least Mean Square*) e o usaram para formular o *Adaline (Adaptive Linear Element)*.

Durante o período clássico do perceptron nos anos 1960, parecia que as redes neurais poderiam realizar qualquer coisa. Mas, então, veio em 1969 o livro de Minsky e Papert (“*Perceptrons*”) que utilizaram a matemática para demonstrar que existem limites fundamentais para aquilo que os perceptrons de camada múltipla podem calcular. Em uma breve seção sobre perceptrons de múltiplas camadas eles afirmavam que não havia razão para supor que qualquer uma das limitações do perceptron de camada única poderia ser superada na versão de múltiplas camadas (Haykin, 2001).

Uma limitação severa do perceptron, apontada também por Minsky e Papert em 1969, é que ele pode ser treinado para aprender unicamente classes que são linearmente separáveis. Num espaço multidimensional, estas classes aparecem sempre separadas por hiperplanos. Num espaço bidimensional o hiperplano se reduz a uma linha tal como representado na Figura 2.2. Contudo, os perceptrons são ainda muito utilizados para resolução de problemas devido a sua arquitetura simples e a sua convergência incondicional quando são consideradas classes linearmente separáveis. O perceptron é um excelente discriminador linear (Kasabov, 1998).

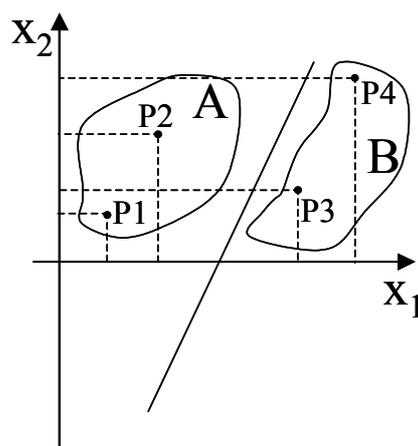


Figura 2.2. Classes linearmente separáveis A e B que podem ser aprendidas por um perceptron (Kasabov, 1998).

A conjectura feita por Minsky e Papert parece agora injustificada, pois se têm atualmente várias formas avançadas de RN's que são mais poderosas, do ponto de vista computacional, que o perceptron de Rosenblatt, como por exemplo o perceptron de múltiplas camadas treinado com o algoritmo de retropropagação que será discutido nas seções subseqüentes (Haykin, 2001). Porém, o livro de Minsky e Papert levou ao esmorecimento do interesse continuado em redes neurais nos anos 70.

Até aqui chegou o que pode ser chamado de entusiasmo inicial das redes neurais. Existia um problema fundamental a ser resolvido denominado “Problema de atribuição de crédito” o qual consistia em atribuir crédito aos neurônios ocultos da rede, cuja solução teve que aguardar até o aparecimento de melhoras tecnológicas nos computadores, o que aconteceu no começo dos anos 1980.

Nos anos 80 foram feitas importantes contribuições em várias frentes à teoria e ao projeto de redes neurais, e com isso houve um ressurgimento do interesse. Em 1982, Hopfield utilizou a idéia de uma função de energia para formular um novo modo de se entender a computação executada por redes recorrentes com conexões sinápticas simétricas. Essa classe particular de redes neurais com realimentação atraiu muita atenção nos anos 80 e no decorrer do tempo tornou-se conhecida como *redes de Hopfield*. Em 1986 foi relatado por Rumelhart, Hinton e Williams (“*Learning representations of back-propagation errors*”) o desenvolvimento do algoritmo de retropropagação (*back-propagation*) que emergiu como o algoritmo de aprendizagem mais popular para o treinamento de perceptrons de múltiplas camadas.

Após o desenvolvimento do algoritmo de retropropagação em meados dos anos 80, revelou-se que o mesmo havia sido descrito anteriormente por Werbos na sua tese de doutorado na universidade de Harvard, em 1974; a tese de Werbos foi a primeira descrição documentada da computação eficiente do gradiente em modo reverso que foi aplicada a modelos gerais de redes, sendo as redes neurais um caso especial (Haykin, 2001)

2.2 – AS REDES NEURAIIS EM ENGENHARIA CIVIL

As aproximações têm sido usadas na engenharia desde o início dessa especialidade. Recentemente, o uso das redes neurais artificiais como aproximadores universais tem se popularizado em todas as áreas da engenharia e das ciências no geral. As aplicações em engenharia civil começaram a aparecer a partir dos últimos anos da década de 80. Atualmente

o campo de aplicação é muito amplo abrangendo tópicos como otimização de processos, simulação de processos construtivos, previsão de ameaça sísmica, classificação de sinais provenientes de testes não destrutivos, estimativa de custos de construção, engenharia de transporte, análise estrutural, tratamento de águas, recuperação de aquíferos, previsão de vazões, avaliação do potencial de liquefação de solos e muitas outras relacionadas com o campo da engenharia geotécnica que serão apresentadas mais adiante.

São muitos os fatores que têm contribuído para estimular o interesse nas redes neurais, o mais notável é a sua capacidade de aprendizado e generalização a partir de exemplos, para produzir soluções confiáveis a problemas complexos ainda que a informação de entrada esteja contaminada com dados imprecisos ou incompletos. As redes neurais são capazes de adaptar soluções a problemas dependentes do tempo e de processar informações rapidamente e transferi-las entre sistemas computacionais. Outros fatores que têm contribuído para aumentar o interesse nas RN's têm a ver com o rápido desenvolvimento de computadores cada vez mais poderosos, com a capacidade necessária para manejar a grande carga computacional imposta pelos algoritmos de treinamento das RN's (Flood & Kartam, 1994).

As RN's não devem ser consideradas como uma ferramenta substitutiva das ferramentas computacionais tradicionais, mas sim como uma ferramenta complementar. De fato, muitos pesquisadores estão considerando cada vez com mais frequência a utilização de sistemas híbridos que integram RN's com sistemas especialistas ou códigos de elementos finitos, de tal forma que as vantagens de cada um dos métodos são potencializadas nestes sistemas híbridos. Por outro lado, as RN's tampouco devem ser consideradas a solução para todas as falhas dos sistemas computacionais tradicionais; de fato, elas apresentam muitas deficiências como por exemplo: a produção de soluções imprecisas, a deficiência de teoria para orientar sobre a escolha da arquitetura e tamanho ótimo para um dado problema, e o lento progresso do aprendizado durante o treinamento para uma grande quantidade de aplicações (Flood & Kartam, 1994).

Não existe uma maneira única para abordar um problema usando redes neurais. Na hora de projetar um modelo neural deve-se considerar o conjunto de características do problema específico para definir adequadamente a arquitetura do modelo, a forma de apresentação dos dados e de interpretação das respostas, o método de treinamento e a forma de operação dos neurônios. A seguir serão tratados de forma breve diferentes tipos de problemas que podem ser abordados com redes neurais em engenharia civil.

No seu artigo de revisão, Adeli (2001) apresenta de maneira detalhada os trabalhos sobre redes neurais em engenharia civil publicados entre 1989 e 2000. A maior parte dos trabalhos estão compreendidos na área da engenharia estrutural: previsão de cargas em vigas, definição de seções de vigas de concreto, seleção dos elementos estruturais de edificações, problemas de mecânica de fraturas, avaliação de danos em elementos estruturais, previsão do comportamento de conexões estruturais, previsão da resistência de elementos estruturais, determinação da relação carga-deslocamento em elementos estruturais, otimização dos processos de execução de projetos estruturais, identificação de sistemas dinâmicos estruturais, criação eficiente e automática de malhas de elementos finitos para análise estrutural, previsão das propriedades mecânicas e térmicas de materiais compostos, modelagem constitutiva de concreto e materiais compostos em geral.

Nesta última aplicação se destaca o trabalho de Theocaris e Panagiotopoulos (1995) que apresentaram uma metodologia para abordar o problema clássico de elasto-plasticidade com endurecimento por meio de uma seqüência de problemas de elasticidade anisotrópica. O método desenvolvido apóia-se numa identificação de parâmetros realizada por uma rede neural.

Zhao & Ren (2002) apresentaram dois modelos neurais treinados para prever a resistência do concreto sob condições de carregamento triaxial, o primeiro deles estruturado com funções de base radial, enquanto o segundo tratava-se de um perceptron multicamadas, treinado com o algoritmo de retropropagação. Os resultados dos modelos neurais foram comparados com os obtidos a partir de modelos de regressão. Embora os modelos neurais tenham apresentado melhor desempenho que os modelos paramétricos, o longo tempo de treinamento necessário e a dificuldade para definir uma estrutura neural ótima, foram as duas maiores desvantagens encontradas. No entanto, deve ser destacado que a complexidade do mapeamento entrada-saída do problema analisado, pode ser facilmente abordado pelo modelo neural mediante simples modificações de sua estrutura, enquanto que nos modelos paramétricos é indispensável escolher formulações explícitas baseadas em processos de tentativa e erro.

No Brasil, um dos primeiros trabalhos encontrados é o de Cassa et al. (1997) que apresentou resultados de pesquisas desenvolvidas sobre a previsão do comportamento do concreto, fazendo referência a trabalhos desenvolvidos desde 1994 na área de modelagem de materiais complexos em geral. Em segundo lugar se encontram as aplicações relacionadas com engenharia de construção: programação de obra, estimativa de custos de construção,

valoração de orçamentos de manutenção de obras e minimização de custo total de obras. A seguir se encontram as aplicações relacionadas com engenharia de recursos hídricos e engenharia ambiental (previsão de vazões, previsão da solubilidade de contaminantes, estudo da interação entre sistemas hídricos, previsão de enchentes), engenharia de tráfego e estradas (estimação dos volumes de tráfego, avaliação do desempenho de motoristas sob diferentes condições de tráfego, localização de acidentes, otimização da sinalização de interseções, avaliação do desempenho de pavimentos, previsão de trincas em pavimentos), e aplicações em engenharia geotécnica.

2.3 – APLICAÇÕES EM ENGENHARIA GEOTÉCNICA

A maioria das aplicações de redes neurais está compreendida nas áreas de reconhecimento de padrões, classificação, previsão e mapeamento de funções. Estas amplas áreas de aplicação abrangem um grande número de possíveis problemas de engenharia geotécnica. No entanto, em engenharia geotécnica os problemas não são usualmente abordados em termos de reconhecimento de padrões, classificação ou mapeamento. Normalmente, a maior parte dos problemas geotécnicos são considerados únicos, e de fato em muitos casos, as suas características e complexidades fazem com que assim seja. Também é muito comum em geotecnia incorporar a cada problema o que é chamado de “experiência” e “juízo”, sendo que às vezes os dois se encontram relacionados. A experiência e o juízo fornecem a cada projeto informações relacionadas com as características comuns a outros projetos similares. De fato, o estudo e análise de projetos anteriores são realizados de maneira habitual em geotecnia, produzindo resultados satisfatórios. Todo este processo não é mais que uma tentativa de entender o espaço solução de um problema particular a partir da observação de um conjunto discreto de pontos nesse espaço. As simulações numéricas fazem parte do mesmo processo de exploração da superfície-solução; cada simulação fornece informação relacionada com um ponto discreto no espaço solução. Os estudos paramétricos, nos quais as simulações numéricas são repetidas para um conjunto escolhido de valores dos parâmetros, fornecem informação sobre a vizinhança de um ponto no espaço solução (Ghaboussi, 1992). Nos parágrafos seguintes serão apresentadas algumas das aplicações mais comuns das redes neurais em geotecnia.

2.3.1 – CLASSIFICAÇÃO E CARACTERIZAÇÃO DE SOLOS E ROCHAS

Neste caso o problema consiste em determinar uma série de parâmetros para um material dado e aplicar uma série de regras para classificar ou caracterizar tal material. Algumas destas regras podem ser formuladas explicitamente, como no caso da determinação de resistências, porém outras que são o resultado da experiência acumulada do engenheiro são mais difíceis de serem expressas mediante formulações explícitas. No entanto, se aceita que existe sempre uma relação definida entre as características observáveis e mensuráveis, de tal forma que existe um espaço solução para cada problema. Neste espaço solução, as características do material são as coordenadas que localizam cada um dos pontos que representam uma solução particular do problema geral.

Neste tipo de problema está o trabalho de Najjar et al. (1996) que usaram redes neurais do tipo Perceptron de Múltiplas Camadas, PMC, para determinar a umidade ótima e a densidade seca máxima a partir das características granulométricas e de plasticidade de 39 solos finos. Os autores concluíram que os modelos neurais não só apresentaram melhor desempenho que modelos paramétricos convencionais, mas também mostraram maior versatilidade para se adaptar às complexidades inerentes à relação entre as diversas variáveis de entrada usadas, e as duas variáveis de saída escolhidas. Outras aplicações são: determinação do teor de umidade de areias (Mohamed & Hawas, 2004), previsão da permeabilidade de maciços rochosos (Ali & Chawathé, 2000), determinação da resistência de rochas a partir das características petrográficas (Singh et al., 2001), previsão de resultados de testes de adensamento em solos finos a partir das suas propriedades índice (Turk et al., 2001), e previsão de parâmetros tais como densidade relativa (D_r), coeficiente de empuxo em repouso (K_0) e relação de pre-adensamento (OCR) a partir de resultados de ensaios de campo de penetração com cone (CPT – *Cone Penetration Test*) para depósitos de areia (Juang et al., 2002).

2.3.2 – ESTUDO DO POTENCIAL DE LIQUEFAÇÃO DE AREIAS

Esta aplicação também forma parte dos problemas de classificação. A maneira mais usual de apresentar este tipo de problemas é formular a questão em termos de susceptibilidade de liquefação de um dado depósito de areia; usando uma escala qualitativa na qual os extremos podem ser, por exemplo, “*muito susceptível de apresentar liquefação*” e

“definitivamente não susceptível de apresentar liquefação”, passando por uma série de estágios intermediários entre os dois extremos. Neste caso, o habitual é definir uma escala numérica de valores para indexar cada um dos estágios da escala qualitativa.

Os dados de entrada para determinar o potencial de liquefação estão classificados em dois grupos: as características do local e as características do sismo. O local é usualmente caracterizado por meio das propriedades dos materiais, o perfil estratigráfico e as características topográficas. O sismo é caracterizado pela intensidade, a aceleração pico do terreno, a duração e o conteúdo de frequências (Ghaboussi, 1992).

Abordagens deste tipo podem ser consultadas nos trabalhos de Goh (1994) e Juang et al. (2001). Uma outra linha de trabalho relacionada com o comportamento dinâmico de terrenos compreende os problemas de análise de resposta sísmica. Nestes casos, o mapeamento é realizado entre parâmetros que definem as características do sismo e as propriedades do local afetado, e a resposta do depósito de solo em termos, por exemplo, do espectro de frequências ou de períodos. As aplicações mais modernas estão sendo encaminhadas à utilização de sistemas híbridos nos quais as RN's são usadas em conjunto com técnicas da teoria de controle. Uma aplicação deste tipo pode ser consultada em Garcia & Romo (2004).

2.3.3 – APLICAÇÕES EM PAVIMENTAÇÃO.

A maior parte das aplicações de RNA's em pavimentação estão concentradas nas áreas de construção e manutenção, no entanto, nos últimos anos tem sido visível o aumento do interesse na utilização destas ferramentas para análise estrutural de pavimentos assim como previsão do comportamento. Não obstante o grande avanço na velocidade e desempenho dos computadores, a modelagem numérica do comportamento estrutural dos pavimentos é ainda uma tarefa que consome tempo importante de processamento. Dependendo do tipo de aplicação, uma rede neural pode chegar a ser duas ou três vezes mais rápida que a maioria dos programas de elementos finitos (TRB/NRC, 1999). No entanto, se aceita que as RNA's não podem substituir completamente a versatilidade do método dos elementos finitos, e que estão ainda longe de poder prever para uma estrutura de pavimento qualquer, as tensões, deformações e as deflexões em qualquer ponto e para qualquer condição de carregamento. Em contraste, é tarefa simples treinar uma RNA para calcular, por exemplo, a deformação máxima num ponto determinado de uma estrutura de pavimento flexível devida a um carregamento dado. Uma compilação da TRB/NRC (1999) apresenta um resumo completo

das principais aplicações de RNA's em engenharia de pavimentos na década de 90. Os trabalhos referenciados versam principalmente sobre previsão de bacias de deflexão, previsão de módulos de deformação, estimação de espessura de camadas da estrutura de pavimento, previsão da durabilidade de agregados, previsão da rugosidade e detecção automática e classificação de trincas.

Mais recentemente, o Grupo de Geotecnia da Universidade de Brasília, tem incorporado esta ferramenta computacional em diversos projetos de pesquisa na área de pavimentação. Souza (2002) usou RNA's para prever o valor do Índice de Irregularidade Internacional (IRI – *International Roughness Index*) a partir de uma base de dados de desempenho de longo prazo de pavimentos para 42 estados dos Estados Unidos de América.

Macedo (2003) usou RNA's do tipo PMC treinadas com o algoritmo de retropropagação para fazer previsões de bacias de deflexão fornecidas por diferentes métodos de ensaios não destrutivos, sob diferentes condições de carregamento.

Dantas Neto (2004) estudou a influência das variáveis do processo de fabricação dos asfaltos-borracha nas propriedades físicas e no comportamento mecânico de misturas asfálticas confeccionadas com esses materiais. Neste caso o autor usou RNA's para a previsão das propriedades físicas do asfalto-borracha.

2.3.4 – MODELAGEM CONSTITUTIVA

Uma das primeiras publicações que descreve a modelagem do comportamento tensão-deformação de materiais usando redes neurais corresponde ao estudo das características mecânicas do concreto (Ghaboussi et al., 1991). Neste trabalho os autores concluem que um modelo constitutivo baseado em redes neurais poderia ser desenvolvido e acoplado num programa de elementos finitos como uma forma alternativa às relações atualmente usadas.

Ellis et al. (1995) conseguiram resultados satisfatórios trabalhando com arquiteturas alternativas de redes neurais (redes seqüenciais ou recorrentes), onde a saída da rede para um estágio atual de tensão-deformação serve como entrada para o próximo estágio. Estes autores utilizaram dados de ensaios triaxiais convencionais, envolvendo ciclos de carregamento e descarregamento, não drenados, realizados com diferentes areias.

Mais recentemente, Ghaboussi e Sidarta (1998) e Sidarta e Ghaboussi (1998) introduziram o conceito de rede neural auto-adaptável, no qual a RN vai aumentando gradualmente a quantidade de neurônios nas camadas ocultas, na medida que o volume de

informação vai aumentando. Estes autores modelaram o comportamento de areias em ensaios triaxiais e conseguiram pela primeira vez na literatura geotécnica consultada, acoplar uma RN a um programa de elementos finitos para reproduzir a deformada de uma amostra cilíndrica num ensaio triaxial convencional. Ao mesmo tempo, Zhu et al. (1998) apresentaram resultados de simulação de redes tipo perceptron treinadas para simular o comportamento tensão cisalhante-deformação de um solo residual fino e de uma areia limpa. O modelo reproduz satisfatoriamente o comportamento dilatante das amostras mais densas do solo residual, assim como alguns ciclos de carga-descarga da areia limpa.

Penumadu e Zhao (1999) modelaram o comportamento tensão-deformação de areia e brita, usando RN's treinadas e testadas para uma base de dados de 250 ensaios triaxiais convencionais numa tentativa de fornecer algoritmos gerais para serem usados na prática cotidiana da geotecnia. Os modelos apresentados podem representar satisfatoriamente os efeitos da mineralogia, forma e tamanho dos grãos, índice de vazios e tensão confinante.

Najjar et al. (1999) usaram RN's para reproduzir os resultados de ensaios triaxiais convencionais monotônicos sobre areias, e ensaios triaxiais cíclicos sobre solos finos. O modelo usado mostrou-se muito eficiente para caracterizar o comportamento tensão-deformação de ambos solos. Simultaneamente, Pernot e Lamarque (1999) usaram redes neurais do tipo perceptron de múltiplas camadas para simular o comportamento tensão-deformação de concreto sob carregamento axial simples, e areia sob condições de cisalhamento simples e finalmente areia sob carregamento triaxial convencional. Neste caso, além de variáveis de entrada que forneciam informação sobre o tipo de material, os autores introduziram tensões de referência que informavam ao modelo neural sobre a resistência do material para cada uma das condições de carregamento impostas.

Basheer (2000) destaca as vantagens e capacidades das RN's como substitutos dos modelos clássicos baseados nas teorias da elasticidade e plasticidade, cujas principais limitações são a complexidade das formulações, a idealização do comportamento e, às vezes, o número excessivo de parâmetros empíricos. O autor apresenta diferentes metodologias que podem ser usadas como referência para a modelagem constitutiva de geomateriais com RN's, e são analisadas em relação à sua capacidade para representar o comportamento real do material e a simplicidade da modelagem. As diferentes metodologias são testadas mediante a simulação de ciclos histeréticos tensão-deformação obtidos analiticamente. O melhor modelo é validado mediante a aplicação de dados reais obtidos experimentalmente para uma argila.

García et al. (2000) apresentam uma discussão sobre os diferentes fatores relacionados com a modelagem do comportamento dos materiais granulares usando RN's. Um conjunto de ensaios triaxiais não drenados foi realizado sob condições controladas para criar uma base de dados para o treinamento e teste dos diferentes modelos neurais apresentados. O trabalho apresenta uma detalhada comparação entre as diferentes funções de ativação usadas e os algoritmos de treinamento, visando esclarecer alguns pontos importantes relacionados com a precisão dos modelos e principalmente o tempo de processamento.

O interesse principal das aplicações de redes neurais mostradas pelos diferentes autores, nas quais o modelo é treinado diretamente a partir de informação experimental é obvio. Neste caso, um operador constitutivo convencional (do tipo paramétrico) poderia ser diretamente substituído por uma rede neural devidamente treinada. Nas diferentes referências já citadas, a criação da arquitetura dos modelos neurais está claramente determinada pelas grandezas físicas medidas nos experimentos ou simulações: tensões-deformações ou forças-deslocamentos. Uma aproximação fundamentalmente diferente foi apresentada por Lefik e Schrefler (2003), na qual o modelo constitutivo neural é treinado para fazer o mapeamento incremental entre tensões e deformações em contraste com as aproximações tradicionalmente apresentadas do tipo ϵ - σ . No seu artigo, os autores usaram informação experimental obtida a partir de ensaios biaxiais sobre amostras de cabos supercondutores, a qual revelou um comportamento não-linear altamente complexo que foi adequadamente capturado pelo modelo neural e posteriormente implementado dentro de um programa de elementos finitos.

2.3.5 – MODELAGEM “IN SITU” DE GEOMATERIAIS

Os modelos constitutivos de geomateriais são habitualmente desenvolvidos a partir de resultados de ensaios de laboratório. No entanto, o comportamento “in situ” dos geomateriais é usualmente diferente do comportamento das amostras de laboratório. A questão da determinação “in situ” das propriedades dos materiais está tornando-se uma parte cada vez mais importante do problema geral de caracterização de terrenos.

Seja qual for o método de amostragem usado, a relação entre as propriedades medidas no laboratório, as propriedades “in situ” e o comportamento constitutivo dos materiais, é altamente complexa, e como tal, ideal para ser abordada por meio de redes neurais que, como já se mencionou anteriormente, são boas para capturar relações complexas (Ghaboussi, 1992).

Outras aplicações apresentadas incluem previsão de deslocamentos verticais de fundações superficiais (Shahin et al., 2000), previsão de erosão de solos (Licznar & Nearing, 2003), previsão da capacidade de carga de estacas (Goh, 1995; Chan et al., 1995; Lee & Lee, 1996), avaliação da estabilidade geotécnica de terrenos (Yuanyou et al., 1997), previsão de deslocamentos verticais devidos à construção de túneis (Kim et al., 2001).

No Brasil, as primeiras publicações sobre aplicações geotécnicas das RN's datam de 1995. Mais recentemente, Dyminski (2000) apresentou três aplicações práticas diferentes de RN's em geotecnia. A primeira está relacionada com a simulação de provas de carga dinâmica em estacas, a segunda simula o comportamento tensão-deformação de dois tipos de solos diferentes (areia de Ipanema e solo residual do Rio de Janeiro). A terceira refere-se à simulação das características do subsolo do sítio da Usina Nuclear Angra 2.

FUNDAMENTOS TEÓRICOS

A técnica das redes neurais pode ser considerada como uma aproximação diferente para introduzir de alguma maneira o raciocínio humano em sistemas cuja fonte de conhecimento é a experiência. Este paradigma tem sido útil na representação de informação a partir de dados incompletos ou imprecisos, apresentando como uma grande vantagem a possibilidade de processamento em paralelo e aprendizado a partir de informação preexistente. Neste capítulo são apresentadas as noções teóricas básicas da técnica das redes neurais, dando-se ênfase num tipo particular e muito utilizado de rede neural, chamado Perceptron de Múltiplas Camadas, que constitui a ferramenta computacional básica empregada nesta tese para a modelagem constitutiva de areias.

3.1 – O NEURÔNIO ARTIFICIAL

A Figura 3.1 apresenta um esquema do primeiro neurônio desenvolvido em 1943 por Warren McCulloch e Walter Pitts, o qual é, ainda hoje, muito utilizado em arquiteturas neurais.

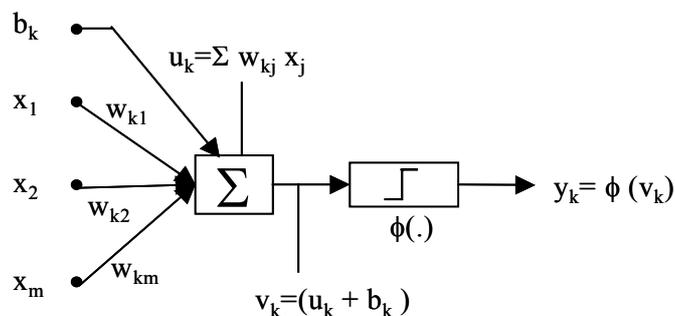


Figura 3.1. Modelo de neurônio de McCulloch-Pitts (Modificado de Ghaboussi, 1992)

O neurônio calcula uma variável dinâmica chamada ativação, v_k , que é uma função de uma combinação linear, x_m , que chegam através das conexões. Na maioria dos modelos, estas conexões estão associadas a pesos, w_{km} , os quais guardam o conhecimento representado no modelo, e servem para ponderar a entrada recebida por cada neurônio da rede (Braga et al., 2000). O sinal de entrada, x_m , numa conexão particular é multiplicado pelo peso sináptico da conexão, w_{km} . O sinal de saída de um neurônio (atividade do neurônio), é calculado como $y_k = \phi(v_k)$, onde $\phi(\cdot)$ é a chamada função de ativação. Pode-se também observar um valor de limiar ou threshold (positivo) ou bias (negativo), b_k , que possui um efeito de aumentar ou diminuir o valor da combinação linear das entradas na função de ativação.

De acordo com o valor de cada um dos parâmetros anteriormente apresentados, surgem diferentes tipos de neurônios. Os valores de entrada e saída de um neurônio podem ser binários, $\{0,1\}$; bivalentes, $\{-1,1\}$; contínuos, $[0,1]$; ou números discretos dentro de um intervalo definido (Kasabov, 1998).

As funções de ativação mais comumente usadas são apresentadas na Figura 3.2. Elas são (Kasabov, 1998):

- A função degrau. Se o valor de entrada v supera certo limiar o neurônio se ativa, caso contrário permanece inativo.
- A função limiar lógico. O valor de ativação aumenta linearmente na medida que aumenta o valor do sinal de entrada v , porém, a partir de certo limiar a saída fica saturada atingindo e permanecendo no seu valor máximo. Existem diferentes variantes dependendo do valor do limiar ou saturação (ver Figura 3.2).
- A função sigmóide. É uma função de transformação não linear com forma de S que apresenta as seguintes características:
 - a. Limitada, ou seja, seus valores estão restringidos entre duas fronteiras, por exemplo, $[0,1]$, $[-1,1]$.
 - b. Monotonamente crescente, ou seja, o valor de $\phi(v)$ nunca decresce quando v cresce.
 - c. Contínua e suave, ou seja, diferenciável ao longo de todo o seu domínio. Diferentes tipos de funções logísticas têm sido usados na prática. A mais comum é a logística, que em sua forma mais geral pode ser escrita como apresentado na Eq.(3.1).

$$\phi(v) = \frac{1}{1 + \exp(-av)} \quad (3.1)$$

na qual a é uma constante chamada parâmetro de inclinação. A razão pela qual esta função tem sido amplamente usada é porque muitos dos algoritmos empregados no processo de treinamento das redes neurais usam a derivada da função de ativação, a qual tem uma expressão bastante simples e que para o caso da sigmóide é:

$$\frac{\partial \phi}{\partial v} = \frac{a}{1 - a} \quad (3.2)$$

Uma outra função deste tipo é a tangente hiperbólica cuja expressão aparece na Eq.(3.3).

$$\tanh(v) = \frac{\exp(v) - \exp(-v)}{\exp(v) + \exp(-v)} \quad (3.3)$$

- A função gaussiana. É a função clássica de Gauss, em forma de sino invertido (ver Figura 3.2).

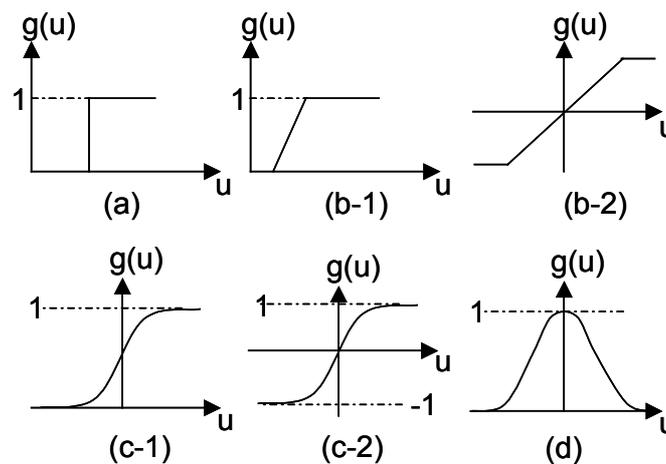


Figura 3.2. Funções de ativação mais usadas: (a) Degrau, (b-1 e b-2) Limiar lógico, (c) sigmóide: Logística (c-1), tangente hiperbólica (c-2) e Gaussiana (d). (modificada de Kasabov, 1998).

Em termos matemáticos um neurônio pode ser descrito pelo seguinte par de equações (Kasabov, 1998):

$$u_k = \sum_{j=1}^m w_{kj} x_j \quad (3.4)$$

$$y_k = \phi(u_k + b_k) \quad (3.5)$$

onde x_1, x_2, \dots, x_m são os sinais de entrada; $w_{k1}, w_{k2}, \dots, w_{km}$ são os pesos sinápticos do neurônio k ; u_k é a saída do combinador linear dos sinais de entrada; b_k é o bias; $\phi(\cdot)$ é a função de ativação; e y_k é o sinal de saída do neurônio. O uso do bias b_k tem o efeito de aplicar uma transformação afim à saída u_k do combinador linear do modelo da Figura 3.1, como mostrado a seguir:

$$v_k = u_k + b_k \quad (3.6)$$

Em particular, dependendo se o bias é positivo ou negativo, a relação entre o campo local induzido ou potencial de ativação v_k do neurônio k e a saída do combinador linear u_k é modificada na forma ilustrada na Figura 3.3. Note-se que como resultado desta *transformação afim*, o gráfico de v_k em função de u_k não passa mais pela origem.

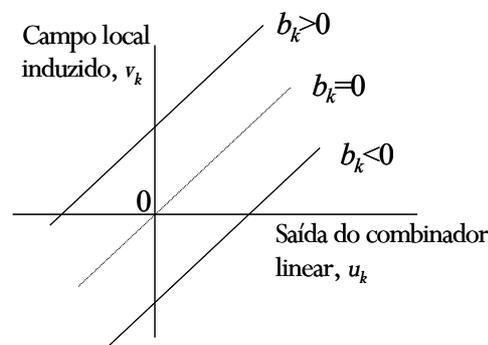


Figura 3.3. Transformação afim produzida pela presença de um bias; note que $v_k = b_k$ em $u_k = 0$ (modificado de Kasabov, 1998)

3.2 – AS REDES NEURAIS

Ainda que um neurônio seja capaz de realizar algumas tarefas simples de processamento de informação, o verdadeiro potencial da computação neural reside na possibilidade de conectar neurônios entre si para formar redes. A Figura 3.4 apresenta uma rede contendo quatro neurônios de entrada (também chamados nós), dois intermediários e um nó de saída. Os pesos associados às conexões são o resultado de um processo de treinamento usando um algoritmo de treinamento (aprendizado) (Kasabov, 1998). Os neurônios de entrada transmitem seus valores, neste exemplo entradas binárias, através das conexões à segunda camada (camada oculta) de neurônios. Para cada conexão existe um peso associado que multiplica o valor transmitido. Os valores já afetados pelos correspondentes pesos que convergem em um neurônio da camada oculta, são somados e o valor gerado é usado como argumento de uma

função simples (função de ativação) cujo sinal de saída é chamado nível de atividade do neurônio. No exemplo da Figura 3.4, usou-se uma função de ativação do tipo degrau com um valor de limiar igual a zero, e um valor de bias nulo.

Os diferentes níveis de atividade dos neurônios da camada oculta são transmitidos através das suas conexões de saída aos neurônios da camada de saída. Da mesma forma como foi apresentado para a camada oculta, na camada de saída os valores transmitidos são multiplicados pelos pesos das conexões, somados e o resultado usado para produzir o nível de ativação dos neurônios da camada de saída. O nível de ativação gerado no neurônio de saída é a solução da rede ao problema apresentado nas entradas. Todos os neurônios dentro de uma camada neste tipo de rede, operam de maneira simultânea no sentido que para qualquer instante no tempo eles estarão no mesmo estágio de processamento (Flood & Kartam, 1994).

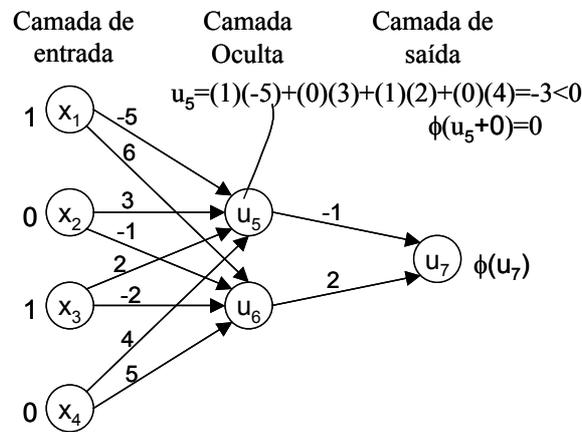


Figura 3.4. Uma rede neural simples. Os pesos das conexões são produto de um processo de treinamento. O valor de ativação do nó 5 (u_5) é apresentado (modificado de Kasabov, 1998).

Tipicamente se terá uma grande quantidade de neurônios em cada uma das camadas da rede, incluindo a camada de saída, e freqüentemente haverá mais de uma camada oculta. O seguinte sistema de equações fornece uma descrição geral do modo de operação deste tipo de rede, independente do número de neurônios em cada camada:

$$y_k = \phi \left(\sum_{j=1}^m (w_{k,j} \cdot y_j) + b_k \right) \quad (3.7)$$

$$y_s = \phi \left(\sum_{l=1}^n (w_{s,l} \cdot y_l) + b_s \right) \quad (3.8)$$

onde y_l é o nível de atividade gerado no l -ésimo neurônio da camada oculta; y_s é o nível de atividade gerado no s -ésimo neurônio da camada de saída; $w_{k,j}$ e $w_{s,l}$ são os pesos das

conexões das camadas oculta e de saída, respectivamente; b_k e b_s são os biases na camada oculta e na camada de saída, respectivamente; e $\phi(\cdot)$ é a função de ativação (Flood & Kartam, 1994).

A resposta da rede é determinada pelo tipo de função de ativação usada, a topologia das conexões, e os valores dos pesos das conexões. Usualmente, a função de ativação e a topologia das conexões são selecionadas em primeiro lugar, deixando para um momento posterior a determinação dos pesos necessários para a rede fazer a tarefa programada (Flood & Kartam, 1994).

Existe uma ampla variedade de RN's. Algumas possuem a capacidade de armazenar informação em memórias associativas. Outras são especialmente adaptadas para solucionar problemas de otimização. Muitas são usadas em problemas de reconhecimento e classificação de padrões, formação de categorias e associação de padrões. Algumas outras têm a capacidade de se adaptar, enquanto que outras são usadas exclusivamente para tarefas muito específicas que não precisam mudanças nos pesos das conexões.

As principais variáveis a levar em conta no desenvolvimento de uma RN são: arquitetura da rede (número de camadas e de nodos, tipo de conexão entre os nodos e topologia da rede), algoritmos de ativação, de propagação e de aprendizagem (Ghabousi, 1992). Alguns exemplos de arquiteturas neuronais são apresentados na Figura 3.5. Quanto ao número de camadas, pode-se ter (Braga et al, 2000):

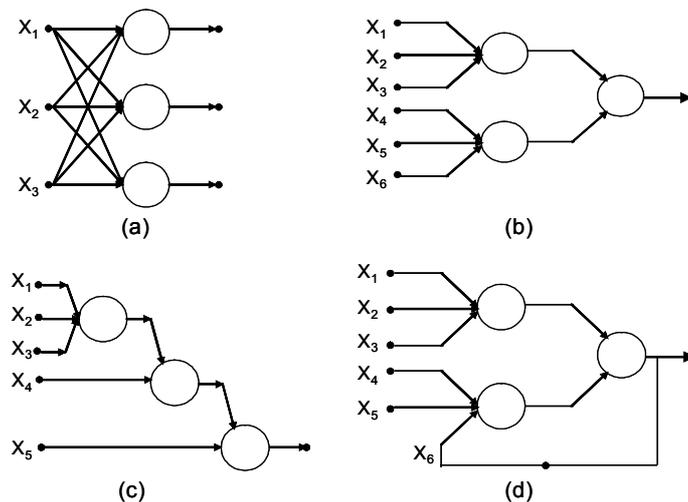


Figura 3.5. Algumas arquiteturas neurais (modificado de Braga et al., 2000).

- Redes de camada única. Só existe um nó entre qualquer entrada e qualquer saída da rede (Figura 3.5a).

- Redes de múltiplas camadas. Existe mais de um neurônio entre alguma entrada e alguma saída da rede (Figura 3.5b,c,d).

Os nodos podem ter conexões do tipo:

- *Feedforward* ou acíclica. A saída de um neurônio na i -ésima camada da rede não pode ser usada como entrada de nodos em camadas de índice menor ou igual a i . (Figura 3.5a,b,c).
- *Feedback* ou cíclica. A saída de algum neurônio na i -ésima camada da rede é usada como entrada de nodos em camadas de índice menor ou igual a i . (Figura 3.5d).

O número de neurônios e de camadas está associado à complexidade do modelo pretendido, refletida no número de pesos (parâmetros) que se vai estimar. Deste modo, algumas recomendações gerais devem ser seguidas. O número de neurônios não pode ser muito pequeno se o problema a ser resolvido apresentar grande complexidade. Este mesmo número também não deve ser grande demais, a ponto de se prejudicar a capacidade de generalização da rede, gerando o problema conhecido como *overfitting* (ajuste excessivo) (Dyminski, 2000).

3.3 – O APRENDIZADO SUPERVISIONADO

Uma das propriedades mais interessantes das RN's é a sua capacidade de aprender a partir de exemplos de treinamento por meio da utilização de um algoritmo de treinamento. Os algoritmos de treinamento estão classificados em três grupos principais:

- **Supervisionado.** Os exemplos de treinamento estão compostos por vetores de entrada x e os correspondentes vetores de saída y . A rede pode então aprender a aproximar a função $y=f(x)$ representada pelo conjunto de exemplos de treinamento (x,y) .
- **Não-supervisionado.** Neste caso só os vetores de entrada x são fornecidos à rede que deve aprender as características internas próprias do conjunto de treinamento apresentado.
- **Aprendizado com reforço.** É uma combinação dos paradigmas anteriores na qual só os vetores de entrada são apresentados e a saída que a rede gera é

analisada. Caso essa saída seja “boa” a rede é recompensada mediante o reforço dos pesos das conexões; caso a saída seja “ruim” a rede é penalizada mediante a diminuição dos valores dos pesos das conexões. Este aprendizado com reforço é equivalente a *aprender com um crítico*, e contrário a *aprender com um professor* (Kasabov, 1998). Por se tratar do método mais simples e popular de aprendizado, e o que melhor se acomoda às características do problema que pretende se resolver, este algoritmo será tratado com maior detalhe nas seções seguintes.

Tipicamente, o processo de aprendizagem com reforço começa com uma rede composta por um número arbitrário de neurônios ocultos, uma topologia fixa, e um conjunto aleatório de valores pequenos para os pesos das conexões. Um conjunto de padrões ou vetores de treinamento, cada um dos quais contem um exemplo do problema a ser resolvido (*inputs*) e a sua correspondente solução (*outputs*) são apresentados á rede. Cada exemplo do problema gera, por parte da rede, uma resposta que deve ser comparada com a solução ou resposta alvo, e o valor do erro é calculado para o conjunto de treinamento apresentado. Os pesos das conexões são então ajustados de acordo com um algoritmo de minimização do erro que deve ser definido previamente, de tal forma que o próximo exemplo apresentado à rede gere um erro menor, aproximando-se gradativamente da solução alvo. O processo é repetido numerosas vezes até que a rede seja capaz de reproduzir, com uma determinada precisão, as soluções correspondentes a cada um dos exemplos do problema. A seguir, deve ser verificada a capacidade de generalização da rede, ou seja a sua capacidade de aprendizagem, mediante a apresentação de exemplos que não foram usados para o treinamento da mesma (Flood & Kartam, 1994).

Uma outra forma de ver o aprendizado supervisionado é como uma aproximação do mapeamento entre um domínio e um espaço solução de um problema: $X \rightarrow D$, quando conhecido o conjunto de treinamento (x,d) tal que $x \in X$, $d \in D$, $x=(x_1, x_2, \dots, x_n)$, $d=(d_1, d_2, \dots, d_n)$. Inicialmente é apresentado o primeiro vetor de entrada x . A rede calcula o vetor de saída y . Compara-se o vetor de saída desejado, d , com o vetor produzido pela rede, y , e o erro é calculado. Corrigem-se os pesos das conexões de tal forma que a próxima vez que o vetor x seja apresentado, a resposta da rede, y , esteja mais próxima dele. Repete-se a seqüência de passos até a rede atingir o ponto de convergência (Kasabov, 1998).

A avaliação do erro de aproximação pode ser feita de diferentes formas. As mais usadas são o chamado erro instantâneo (Eq. 3.9) e o erro médio quadrático (Eq. 3.10):

$$e_i = (y_i - d_i) \quad (3.9)$$

$$e_i = \frac{(y_i - d_i)^2}{2} \quad (3.10)$$

A maneira pela qual o sinal de erro é usado para controlar os ajustes dos pesos sinápticos do neurônio depende de um processo de otimização realizado sobre uma função de custo ou índice de desempenho, $\mathcal{E}(w)$, previamente definida em termos do sinal de erro e continuamente diferenciável de um vetor de peso w . Ela é uma medida de como escolher o vetor de peso w de um algoritmo de aprendizado de modo que ele se comporte de maneira ótima. Deseja-se encontrar a solução ótima w^* que satisfaz a condição:

$$\mathcal{E}(w^*) \leq \mathcal{E}(w) \quad (3.11)$$

A condição necessária para a otimização aparece na Eq.(3.12). onde ∇ é o operador gradiente.

$$\nabla \mathcal{E}(w^*) = 0 \quad (3.12)$$

Uma classe de algoritmo de otimização está baseada na idéia da descida iterativa local (Haykin, 2001). Iniciando com uma suposição inicial representada por $w(0)$, gera-se uma seqüência de vetores de peso $w(1), w(2), \dots$, de modo que a função de custo $\mathcal{E}(w)$ seja reduzida a cada iteração do algoritmo, como mostrado por:

$$\mathcal{E}(w(n+1)) < \mathcal{E}(w(n)) \quad (3.13)$$

onde o argumento n representa o instante de tempo discreto do processo iterativo envolvido no ajuste dos pesos sinápticos de um neurônio dado, $w(n)$ é o valor antigo do vetor de peso e $w(n+1)$ é o seu valor atualizado.

A seguir serão descritos três métodos de otimização que se baseiam da idéia da descida iterativa.

3.3.1 – MÉTODO DO GRADIENTE DESCENDENTE

Uma alternativa muito usada é fazer com que os ajustes sucessivos aplicados ao vetor de peso w sejam na direção da descida mais íngreme, isto é, em uma direção oposta ao *vetor gradiente* $\nabla \mathcal{E}(w)$. Por conveniência de apresentação escreve-se (Haykin, 2001):

$$\delta = \nabla \mathcal{E}(w) = \left[\frac{\partial \mathcal{E}}{\partial w_1}, \frac{\partial \mathcal{E}}{\partial w_2}, \dots, \frac{\partial \mathcal{E}}{\partial w_m} \right]^T \quad (3.14)$$

Correspondentemente o algoritmo da descida mais íngreme é descrito formalmente por:

$$w(n+1) = w(n) - \eta \delta(n) \quad (3.15)$$

onde η é uma constante positiva chamada taxa de aprendizagem, que pode assumir valores contínuos no intervalo aberto $(0,1)$, e $\delta(n)$ é o vetor gradiente calculado no ponto $w(n)$ passando da iteração n para $n+1$, o algoritmo aplica a correção:

$$\Delta w(n) = w(n+1) - w(n) \quad (3.16)$$

$$\Delta w(n) = -\eta \delta(n) \quad (3.17)$$

O método da descida mais íngreme converge lentamente para a solução ótima w^* . Além disso, o parâmetro de taxa de aprendizagem η tem uma influência no seu comportamento quanto à convergência (Haykin, 2001):

- Quando η é pequeno a resposta transitória do algoritmo é sobreamortecida, sendo que a trajetória traçada por $w(n)$ segue um caminho suave no plano \mathbf{W} como mostrado na Figura 3.6 (a).
- Quando η é grande a resposta transitória do algoritmo é subamortecida, sendo que a trajetória traçada por $w(n)$ no plano \mathbf{W} segue um caminho em zigzague, como ilustrado na Figura 3.6 (b).
- Quando η excede um valor crítico o algoritmo se torna instável e não há convergência.

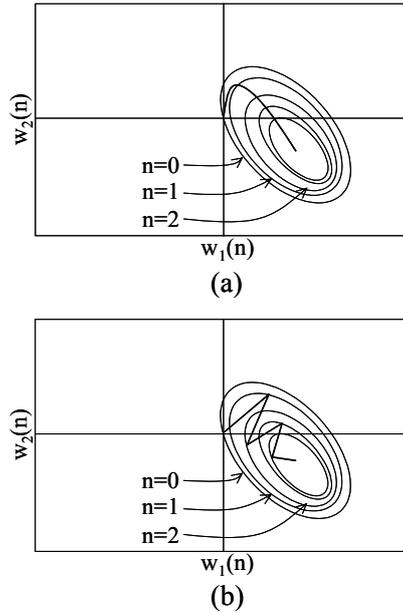


Figura 3.6. Trajetória do método da descida mais íngreme em um espaço bidimensional para dois valores diferentes de parâmetro de taxa de aprendizagem: (a) η pequeno, (b) η grande. As coordenadas w_1 e w_2 são elementos do vetor de peso W .

3.3.2- MÉTODO DE NEWTON

Segundo apresentado por Haykin (2001) a idéia básica do método de Newton é minimizar, em cada iteração do algoritmo, a aproximação quadrática da função de custo $\mathcal{E}(w)$ em torno do ponto $w(n)$. Usando a aproximação de Taylor de segunda ordem da função de custo em torno do ponto pode-se escrever:

$$\begin{aligned} \Delta\mathcal{E}(w(n)) &= \mathcal{E}(w(n+1)) - \mathcal{E}(w(n)) \\ &\cong \partial^T(n)\Delta w(n) + \frac{1}{2}\Delta w^T(n)H(n)\Delta w(n) \end{aligned} \quad (3.18)$$

em que o vetor $\delta(n)$ é o gradiente da função de custo $\mathcal{E}(w)$ calculada no ponto $w(n)$. A matriz $H(n)$ é a matriz hessiana de $\mathcal{E}(w)$, também calculada no ponto $w(n)$. A hessiana é definida por:

$$H = \nabla^2 \mathcal{E}(\mathbf{w}) = \begin{bmatrix} \frac{\partial^2 \mathcal{E}}{\partial w_1^2} & \frac{\partial^2 \mathcal{E}}{\partial w_1 \partial w_2} & \dots & \frac{\partial^2 \mathcal{E}}{\partial w_1 \partial w_m} \\ \frac{\partial^2 \mathcal{E}}{\partial w_2 \partial w_1} & \frac{\partial^2 \mathcal{E}}{\partial w_2^2} & \dots & \frac{\partial^2 \mathcal{E}}{\partial w_2 \partial w_m} \\ \vdots & \vdots & \dots & \vdots \\ \frac{\partial^2 \mathcal{E}}{\partial w_m \partial w_1} & \frac{\partial^2 \mathcal{E}}{\partial w_m \partial w_2} & \dots & \frac{\partial^2 \mathcal{E}}{\partial w_m^2} \end{bmatrix} \quad (3.19)$$

Diferenciando a Eq. (3.18) em relação a Δw , a variação $\Delta \mathcal{E}(\mathbf{w})$ é minimizada quando:

$$\partial(n) + H(n)\Delta w(n) = 0 \quad (3.20)$$

Resolver esta equação para $\Delta w(n)$ resulta:

$$\Delta w(n) = -H^{-1}(n)\partial(n) \quad (3.21)$$

Isto é:

$$w(n+1) = w(n) + \Delta w(n) = w(n) - H^{-1}(n)\partial(n) \quad (3.22)$$

onde $H^{-1}(n)$ é a inversa da hessiana de $\mathcal{E}(\mathbf{w})$.

O método de Newton converge rapidamente de modo assintótico e não apresenta o comportamento zigzagueante que frequentemente se apresenta no método da descida mais íngreme. Para que o método de Newton funcione, a hessiana deve ser uma matriz positiva e definida para todo n .

3.3.3 – MÉTODO DE GAUSS-NEWTON

Este método é aplicável a uma função de custo expressa como a soma dos erros quadrados (Haykin, 2001):

$$\mathcal{E}(\mathbf{w}) = \frac{1}{2} \sum_{i=1}^n e^2(i) \quad (3.23)$$

¹ Duas características de uma matriz positiva e definida são: (a) ela é não singular, isto é, existe a inversa, (b) seus autovalores, ou raízes da equação característica, $\det(R)$, são todos positivos.

Na equação (3.23) o fator de escala $\frac{1}{2}$ é incluído para simplificar a análise subsequente. Todos os termos de erro nesta fórmula são calculados com base no vetor de peso w , que é fixo dentro de todo o intervalo de observação $1 \leq i \leq n$.

O sinal de erro $e(i)$ é uma função do vetor de peso ajustável w . Dado um ponto de operação $w(n)$, se lineariza a dependência de $e(i)$ em relação a w escrevendo:

$$e'(n, w) = e(n) + J(n)(w - w(n)) \quad (3.24)$$

onde $e(n)$ é o vetor de erro:

$$e(n) = [e(1), e(2), \dots, e(n)]^T \quad (3.25)$$

e $J(n)$ é a matriz jacobiana de $e(n)$:

$$J(n) = \begin{bmatrix} \frac{\partial e(1)}{\partial w_1} & \frac{\partial e(1)}{\partial w_2} & \dots & \frac{\partial e(1)}{\partial w_m} \\ \frac{\partial e(2)}{\partial w_1} & \frac{\partial e(2)}{\partial w_2} & \dots & \frac{\partial e(2)}{\partial w_m} \\ \vdots & \vdots & & \vdots \\ \frac{\partial e(n)}{\partial w_1} & \frac{\partial e(n)}{\partial w_2} & \dots & \frac{\partial e(n)}{\partial w_m} \end{bmatrix}_{w=w(n)} \quad (3.26)$$

A jacobiana é a transposta da matriz de gradiente $\nabla e(n)$, onde:

$$\nabla e(n) = [\nabla e(1), \nabla e(2), \dots, \nabla e(n)] \quad (3.27)$$

O vetor de pesos atualizado, $w(n+1)$, é calculado a partir da minimização da norma euclidiana (comprimento do vetor) do sinal de erro linearizado na Eq.(3.24).

$$w(n+1) = \min_w \left\{ \frac{1}{2} \|e'(n, w)\|^2 \right\} \quad (3.28)$$

Usando a Eq.(3.24) chega-se a:

$$\frac{1}{2} \|e'(n, w)\|^2 = \frac{1}{2} \|e(n)\|^2 + e^T(n)J(n)(w - w(n)) + \frac{1}{2} (w - w(n))^T J^T(n)J(n)(w - w(n)) \quad (3.29)$$

Diferenciando esta expressão em relação a w e igualando a zero se obtém:

$$J^T(n)e(n) + J^T(n)J(n)(w - w(n)) = 0 \quad (3.30)$$

Resolvendo esta equação para w , pode-se escrever a partir da Eq. (3.28) que descreve a forma pura do método de Gauss-Newton:

$$w(n+1) = w(n) - (J^T(n)J(n))^{-1} J^T(n)e(n) \quad (3.31)$$

Diferentemente do método de Newton, o método de Gauss-Newton requer apenas da matriz jacobiana do vetor de erro $e(n)$. Entretanto, para que a iteração de Gauss-Newton seja computável, a matriz produto $J^T(n)J(n)$ deve ser não singular, para isto, as n linhas de $J(n)$ devem ser linearmente independentes. Infelizmente, não há garantia de que esta condição seja sempre satisfeita, razão pela qual é costume adicionar a matriz diagonal αI à matriz $J^T(n)J(n)$. O parâmetro α é uma constante positiva pequena para assegurar que $J^T(n)J(n) + \alpha I$ seja definida positivamente para todo n . Baseado nisto, o método de Gauss-Newton é implementado na forma ligeiramente modificada:

$$w(n+1) = w(n) - (J^T(n)J(n) + \alpha I)^{-1} J^T(n)e(n) \quad (3.32)$$

3.4 – PERCEPTRONS DE MÚLTIPLAS CAMADAS

3.4.1 - INTRODUÇÃO

Até aqui tem se tratado exclusivamente sobre o perceptron simples, mostrando alguns algoritmos de otimização, que são usualmente usados para corrigir o valor dos pesos das conexões durante o processo de treinamento. Um perceptron de múltiplas camadas (PMC) consiste de um conjunto de unidades de processamento (nós) que constituem a camada de entrada, uma ou mais camadas ocultas de nós computacionais e uma camada de saída (Figura 3.7). O sinal de entrada se propaga para frente através da rede, camada por camada. Um conjunto de saídas é produzido como a resposta real da rede.

Um perceptron de múltiplas camadas tem três características distintivas:

- O modelo de cada neurônio da rede inclui uma função de ativação não-linear, suave e diferenciável em qualquer ponto.
- A rede contém uma ou mais camadas de neurônios ocultos que capacitam a rede a aprender tarefas complexas extraindo, progressivamente, as características mais significativas dos padrões de entrada.
- A rede exhibe um alto grau de conectividade.

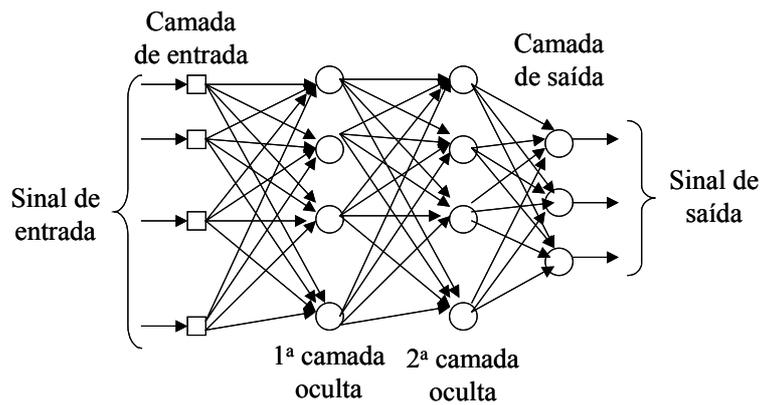


Figura 3.7. Esquema de um perceptron de múltiplas camadas com duas camadas ocultas (Modificado de Haykin, 2002).

O algoritmo básico de treinamento para o perceptron pode ser resumido nos passos que são apresentados na **Erro! Auto-referência de indicador não válida..** Pode-se concluir que o perceptron somente aprende quando erra classificando um vetor de entrada tomado do conjunto de treinamento. O aprendizado significa uma mudança nos valores dos pesos sinápticos, caso o perceptron consiga prever adequadamente o vetor apresentado, os pesos se manterão invariáveis.

P1	Adotam-se valores aleatórios iniciais para os pesos sinápticos $w(n)$.
P2	Calcula-se a saída do combinador linear, u , devida aos sinais de entrada x segundo a Eq. 3.4.
P3	Soma-se o valor do bias e aplica-se uma função de transferência $\phi(\cdot)$ para obter um sinal de saída y (Eq. 3.5) da primeira camada oculta.
P4	As saídas da camada anterior são usadas como entradas na camada oculta seguinte e assim sucessivamente até alcançar a camada de saída.
P5	Calcula-se o erro mediante a comparação entre a saída do PMC e a resposta desejada.
P6	Usa-se um algoritmo de otimização para calcular os novos pesos $w(n+1)$ das conexões sinápticas.
P7	Repetem-se os passos P2 a P7 até a convergência do PMC, ou seja, até que o valor do erro seja suficientemente pequeno

Figura 3.8. Resumo do algoritmo de treinamento para um perceptron. O perceptron “aprende” quando erra na classificação de um vetor de treinamento (Modificada de Kasabov, 1998).

3.4.2 – O ALGORITMO DE RETROPROPAGAÇÃO.

Os perceptrons de múltiplas camadas têm sido aplicados com sucesso para resolver diversos problemas difíceis, mediante o seu treinamento de forma supervisionada com um algoritmo muito popular conhecido como *algoritmo de retropropagação de erro (error back-propagation)*, baseado na regra de aprendizagem por correção de erro (Haykin, 2001). Para aliviar a carga matemática envolvida na derivação do algoritmo, apresenta-se primeiro um resumo das notações utilizadas (Haykin, 2002):

- Os índices i, j, k se referem a diferentes neurônios na rede; com os sinais de propagação através da rede da esquerda para direita, o neurônio j se encontra numa camada à direita do neurônio i .
- Na iteração (passo de tempo) n , o n -ésimo padrão de treinamento (exemplo) é apresentado à rede.

- O símbolo $\mathcal{E}(n)$ se refere à soma instantânea dos erros quadráticos ou energia do erro na iteração n . A média de $\mathcal{E}(n)$ sobre todos os valores de n produz a energia média de erro, \mathcal{E}_{med} .
- O símbolo $e_j(n)$ se refere ao sinal de erro na saída do neurônio j , para a iteração n .
- O símbolo $d_j(n)$ se refere à resposta desejada para o neurônio j e é usada para calcular $e_j(n)$.
- O símbolo $y_j(n)$ se refere ao sinal funcional que aparece na saída do neurônio j , na iteração n .
- O símbolo $w_{ji}(n)$ representa o peso sináptico conectando a saída do neurônio i à entrada do neurônio j , na iteração n . A correção aplicada a este peso na iteração n é representada por $\Delta w_{ji}(n)$.
- O campo local induzido no neurônio j (soma ponderada de todas as entradas sinápticas acrescida do bias) na iteração n é representado por $v_j(n)$; constitui o argumento aplicado na função de ativação associada com o neurônio j .
- A função de ativação, que descreve a relação funcional de entrada-saída da não linearidade associada ao neurônio j , é representada por $\phi_j(\cdot)$.
- O bias aplicado no neurônio j é representado por b_j . O seu efeito é representado por uma sinapse de peso $w_{j0}=b_j$ conectada a uma entrada fixa igual a $+1$.
- O i -ésimo elemento do vetor de entrada é representado por $x_i(n)$.
- O k -ésimo elemento do vetor global de saída é representado por $o_k(n)$.
- O parâmetro da taxa de aprendizagem é representado por η .
- O símbolo m_l representa o tamanho da camada l (número de nós) do PMC; $l=0,1,2,\dots,L$, onde L é a “profundidade” da rede. Assim m_0 representa o tamanho da camada de entrada, m_1 o tamanho da primeira camada oculta e m_L representa o tamanho da camada de saída.

Basicamente, a aprendizagem por retropropagação de erro consiste de dois passos através das diferentes camadas da rede: um passo para frente, a propagação, e um passo para trás, a retropropagação. No passo para frente, um vetor de entrada é aplicado aos nós de

entrada da rede e seu efeito é propagado através dela, camada por camada. Finalmente, um conjunto de saídas é produzido como a resposta real da rede. Durante o passo de propagação, os pesos sinápticos da rede são todos fixos. Durante o passo para trás, por outro lado, os pesos sinápticos são ajustados de acordo com um algoritmo de minimização do erro. Especificamente, a resposta real da rede é subtraída de uma resposta desejada para produzir um sinal de erro. Este sinal de erro é então propagado para trás através da rede, contra a direção das conexões sinápticas. Os pesos sinápticos são ajustados para fazer com que a resposta real da rede se mova para mais perto da resposta desejada, em um sentido estatístico (Haykin, 2001).

O sinal de erro na saída do neurônio j (ver Figura 3.9), na iteração n (i.e., a apresentação do n -ésimo exemplo de treinamento), é definido pela Eq. (3.33) que é uma generalização da Eq. (3.9).

$$e_j(n) = d_j(n) - y_j(n) \quad (3.33)$$

onde $d_j(n)$ representa o valor desejado de saída do neurônio j e $y_j(n)$ representa sua estimativa na iteração n .

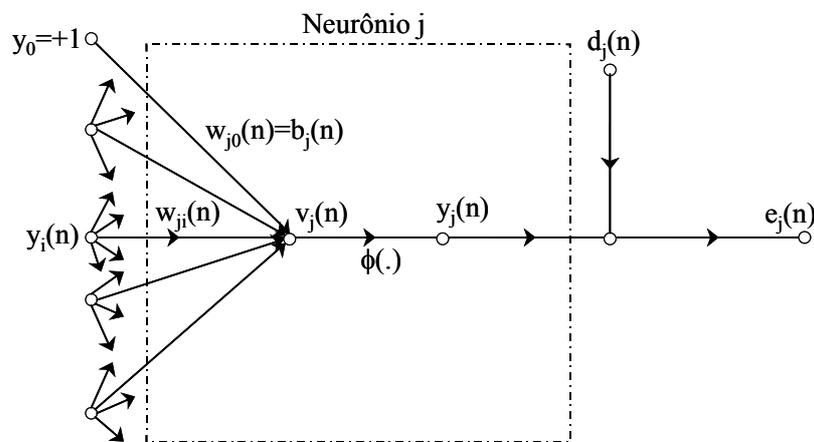


Figura 3.9. Fluxo de sinais ressaltando os detalhes do neurônio de saída j (Haykin, 2001).

Definindo-se o valor instantâneo do erro quadrático para o neurônio j como $e_j^2(n)/2$, então o valor instantâneo do erro total $\mathcal{E}(n)$ pode ser obtido somando-se os termos $e_j^2(n)/2$ de todos os neurônios da camada de saída que são os únicos neurônios visíveis para os quais os sinais de erro podem ser calculados diretamente. Pode-se escrever então:

$$\mathcal{E}(n) = \frac{1}{2} \sum_{j \in C} e_j^2(n) \quad (3.34)$$

onde o conjunto C inclui todos os neurônios da camada de saída da rede. Considere que N representa o número total de exemplos do conjunto de treinamento, então o erro médio quadrático pode ser obtido somando-se os valores de $\mathcal{E}(n)$ e normalizando em relação ao tamanho do conjunto tal como se apresenta a seguir:

$$\mathcal{E}_{\text{med}} = \frac{1}{N} \sum_{n=1}^N \mathcal{E}(n) \quad (3.35)$$

Para um dado conjunto de treinamento, \mathcal{E}_{med} é uma função de todos os parâmetros livres da rede (pesos e bias) e representa a função de custo como uma medida do desempenho de aprendizagem. O objetivo do processo de aprendizagem é ajustar os parâmetros livres da rede para minimizar \mathcal{E}_{med} (Haykin, 2001).

Na Figura 3.9, o campo local induzido $v_j(n)$ na entrada da função de ativação associada ao neurônio j é:

$$v_j(n) = \sum_{i=0}^m w_{ji}(n) y_i(n) \quad (3.36)$$

onde m é o número de entradas (excluindo o bias) aplicadas ao neurônio j. O peso sináptico w_{j0} (correspondendo à entrada fixa $y_0=+1$) é igual ao bias b_j aplicado ao neurônio j. Assim o sinal funcional $y_j(n)$ que aparece na saída do neurônio j na iteração n é:

$$y_j(n) = \phi_j(v_j(n)) \quad (3.37)$$

O algoritmo de retropropagação aplica uma correção $\Delta w_{ji}(n)$ ao peso sináptico $w_{ji}(n)$, que é proporcional ao gradiente instantâneo $\partial \mathcal{E}(n) / \partial w_{ji}(n)$. De acordo com a regra da cadeia pode-se expressar este gradiente como:

$$\frac{\partial \mathcal{E}(n)}{\partial w_{ji}(n)} = \frac{\partial \mathcal{E}(n)}{\partial e_j(n)} \frac{\partial e_j(n)}{\partial y_j(n)} \frac{\partial y_j(n)}{\partial v_j(n)} \frac{\partial v_j(n)}{\partial w_{ji}(n)} \quad (3.38)$$

Diferenciando ambos os lados da Eq. (3.34) em relação a $e_j(n)$ se obtém:

$$\frac{\partial \mathcal{E}(n)}{\partial e_j(n)} = e_j(n) \quad (3.39)$$

Diferenciando ambos os lados da Eq. (3.33) em relação a $y_j(n)$ se obtém:

$$\frac{\partial e_j(n)}{\partial y_j(n)} = -1 \quad (3.40)$$

Diferenciando a Eq. (3.37) em relação a $v_j(n)$ se obtém a Eq. (3.41) na qual o uso do apóstrofe significa a diferenciação em relação ao argumento.

$$\frac{\partial y_j(n)}{\partial v_j(n)} = \phi'_j(v_j(n)) \quad (3.41)$$

Finalmente, diferenciar a Eq. (3.36) em relação a $w_{ji}(n)$ produz:

$$\frac{\partial v_j(n)}{\partial w_{ji}(n)} = y_i(n) \quad (3.42)$$

Tomando desde a Eq. (3.39) até a Eq. (3.42) e aplicando-as na Eq. (3.38) se obtém:

$$\frac{\partial \mathcal{E}(n)}{\partial w_{ji}(n)} = -e_j(n) \phi'_j(v_j(n)) y_i(n) \quad (3.43)$$

A correção $\Delta w_{ij}(n)$ aplicada a $w_{ij}(n)$ é definida pela regra delta que é uma generalização da Eq. (3.17) e que é apresentada a seguir:

$$\Delta w_{ij}(n) = -\eta \frac{\partial \mathcal{E}(n)}{\partial w_{ji}(n)} \quad (3.44)$$

O uso do sinal negativo na Eq. (3.44) indica a descida do gradiente no espaço de pesos. O uso da Eq. (3.43) na Eq. (3.44) produz:

$$\Delta w_{ji}(n) = \eta \cdot e_j(n) \cdot \phi'_j(v_j(n)) \cdot y_i(n) = \eta \cdot \delta_j(n) \cdot y_i(n) \quad (3.45)$$

onde $\delta_j(n)$ é o gradiente local definido por:

$$\delta_j(n) = -\frac{\partial \mathcal{E}(n)}{\partial v_j(n)} = e_j(n) \cdot \phi'_j(v_j(n)) \quad (3.46)$$

Das duas equações anteriores se nota que um fator-chave envolvido no cálculo do ajuste de peso é o sinal de erro na saída do neurônio. Neste contexto são identificáveis dois casos distintos (Haykin, 2001):

- Se o neurônio j é um nó de saída $\delta_j(n)$ é dado pela Eq. (3.46).
- Se o neurônio j é um nó oculto, não existe uma resposta desejada especificada para aquele neurônio. Conseqüentemente o sinal de erro para aquele neurônio deve ser calculado recursivamente, em termos dos sinais de erro de todos os

neurônios aos quais o neurônio oculto se encontra diretamente ligado. O gradiente local do neurônio oculto j pode ser definido como:

$$\delta_j(\mathbf{n}) = -\frac{\partial \mathcal{E}(\mathbf{n})}{\partial v_j(\mathbf{n})} = \frac{\partial \mathcal{E}(\mathbf{n})}{\partial y_j(\mathbf{n})} \frac{\partial y(\mathbf{n})}{\partial v_j(\mathbf{n})} = \frac{\partial \mathcal{E}(\mathbf{n})}{\partial y_j(\mathbf{n})} \phi'_j(v_j(\mathbf{n})) \quad (3.47)$$

Para calcular a derivada parcial $\frac{\partial \mathcal{E}(\mathbf{n})}{\partial y_j(\mathbf{n})}$ procede-se como segue:

$$\mathcal{E}(\mathbf{n}) = \frac{1}{2} \sum_{k \in C} e_k^2(\mathbf{n}) \quad (3.48)$$

A equação anterior é a mesma Eq.(3.34) na qual o índice j foi substituído pelo índice k correspondente a um neurônio de saída da rede. Diferenciando a Eq. (3.48) em relação ao sinal funcional $y_j(\mathbf{n})$ tem-se:

$$\frac{\partial \mathcal{E}(\mathbf{n})}{\partial y_j(\mathbf{n})} = \sum_k e_k(\mathbf{n}) \frac{\partial e_k(\mathbf{n})}{\partial y_j(\mathbf{n})} = \sum_k e_k(\mathbf{n}) \frac{\partial e_k(\mathbf{n})}{\partial v_k(\mathbf{n})} \frac{\partial v_k(\mathbf{n})}{\partial y_j(\mathbf{n})} \quad (3.49)$$

Uma vez que o neurônio k está na saída da rede o valor do sinal de erro pode ser calculado como segue:

$$e_k(\mathbf{n}) = d_k(\mathbf{n}) - y_k(\mathbf{n}) = d_k(\mathbf{n}) - \phi_k(v_k(\mathbf{n})) \quad (3.50)$$

Então:

$$\frac{\partial e_k(\mathbf{n})}{\partial v_k(\mathbf{n})} = -\phi'_k(v_k(\mathbf{n})) \quad (3.51)$$

Para o neurônio k , o campo local induzido é:

$$v_k(\mathbf{n}) = \sum_{j=0}^m w_{kj}(\mathbf{n}) y_j(\mathbf{n}) \quad (3.52)$$

Diferenciando a Eq.(3.52) em relação a $y_j(\mathbf{n})$ produz:

$$\frac{\partial v_k(\mathbf{n})}{\partial y_j(\mathbf{n})} = w_{kj}(\mathbf{n}) \quad (3.53)$$

Utilizando as Eqs.(3.51) e (3.53) na Eq.(3.49) se obtém a derivada parcial desejada:

$$\frac{\partial \mathcal{E}(\mathbf{n})}{\partial y_j(\mathbf{n})} = -\sum_k e_k(\mathbf{n}) \phi'_k(v_k(\mathbf{n})) w_{kj}(\mathbf{n}) = -\sum_k \delta_k(\mathbf{n}) w_{kj}(\mathbf{n}) \quad (3.54)$$

Utilizando esta última equação na Eq. (3.47) se obtém:

$$\delta_j(n) = \phi'_j(v_j(n)) \sum_k \delta_k(n) w_{kj}(n) \quad (3.55)$$

O fator $\phi'_j(v_j(n))$ envolvido no cálculo do gradiente local $\delta_j(n)$ na equação anterior depende unicamente da função de ativação associada ao neurônio oculto j . O somatório sobre k , depende de dois conjuntos de termos. O primeiro conjunto de termos, correspondente aos $\delta_k(n)$, requer conhecimento dos sinais de erro, $e_k(n)$, para todos os neurônios que se encontram na camada imediatamente à direita do neurônio oculto j , e que estão diretamente conectados a ele. O segundo conjunto de termos, os $w_{kj}(n)$, consiste dos pesos sinápticos associados com estas conexões. Em conclusão, para aplicar o algoritmo de retropropagação do erro em um neurônio oculto de um PMC o cálculo do gradiente local, $\delta_j(n)$, é igual ao produto da derivada associada $\phi'_j(v_j(n))$ pela soma ponderada dos $\delta_k(n)$, calculados para os neurônios na próxima camada, k , que estão conectados ao neurônio j .

3.4.3 – TREINAMENTO

Freqüentemente é dito que o projeto de uma rede neural utilizando o algoritmo de retropropagação é mais uma arte do que uma ciência, significando que muitos dos numerosos fatores envolvidos no projeto são o resultado da experiência particular de cada um. Há alguma verdade nesta afirmação. Entretanto existem métodos que melhoram significativamente o desempenho do algoritmo de retropropagação (Haykin, 2001).

3.4.3.1 – TREINAMENTO SEQUÊNCIAL E POR LOTE

Dependendo da forma como o conjunto de treinamento é apresentado a uma RNA durante a aprendizagem por retropropagação, podem ser identificadas duas formas básicas de treinamento: por lote e seqüencialmente. No treinamento por lote, o ajuste dos pesos é realizado exclusivamente depois de ter apresentado à rede o conjunto de treinamento na sua totalidade o que constitui uma “época” de treinamento. Já no caso seqüencial, também chamado modo estocástico, os pesos da rede são atualizados após a apresentação de cada exemplo de treinamento.

Do ponto de vista operacional, o modo seqüencial é preferível em relação ao modo por lote, porque requer menos armazenamento local para cada conexão sináptica. Além disso, dado que os exemplos são apresentados à rede de forma aleatória, o ajuste dos pesos de padrão em padrão torna a busca no espaço de pesos de natureza estocástica, fazendo com que seja menos provável que o algoritmo fique preso em um mínimo local. Por outra parte, o uso do treinamento por lote fornece uma estimativa precisa do vetor gradiente garantido assim a convergência do algoritmo de uma maneira simples (Haykin, 2001).

3.4.3.2 – ABRANGÊNCIA DO CONJUNTO DE TREINAMENTO

Um outro fator que pode influenciar de maneira significativa a capacidade de aprendizado e generalização da rede é o número de exemplos de treinamento. Aumentando o número de exemplos de treinamento consegue-se fornecer maior quantidade de informação acerca da forma da superfície solução, e assim incrementa-se o nível potencial de precisão que pode ser alcançado pela rede. No entanto, um conjunto muito grande de vetores de treinamento pode, em alguns dos casos, saturar o algoritmo de treinamento utilizado ou pode gerar um estancamento prematuro da rede em mínimos locais da superfície solução. Em conseqüência, não há garantia alguma de que aumentando o número de exemplos de treinamento o desempenho da rede melhore na mesma proporção.

Deve ser levado em conta, então que existe um limite prático para o tamanho do conjunto de treinamento que pode ser eficientemente modelado por uma rede com um número fixo de neurônios; a partir deste limite o ganho obtido pelo aumento do tamanho do conjunto de treinamento rapidamente se atenua (Flood & Kartam, 1994).

Existe uma série de fatores práticos que podem limitar o número de exemplos que podem ser utilizados no processo de treinamento. Para muitos problemas, um conjunto fixo de exemplos de treinamento pode ser tudo o que se tenha, sendo impossível ou muito trabalhoso aumentar o tamanho desse conjunto de treinamento. Adicionalmente, o tempo requerido para o treinamento do modelo aumenta consideravelmente com o aumento do número de exemplos. Conseqüentemente, pode resultar que mesmo sendo possível aumentar o tamanho do conjunto de treinamento, isto possa gerar uma diminuição tão grande na evolução do processo de aprendizado que o torne inaceitável (Flood & Kartam, 1994).

Geralmente, nos casos em que é possível ter algum controle sobre o processo de coleta do conjunto de dados de treinamento, o que se faz é realizar uma série de testes para

determinar o tamanho ótimo de tal conjunto. O primeiro tipo de experimento consiste em usar uma amostra relativamente pequena de dados de treinamento (provavelmente entre 10 e 100) bem distribuídos no espaço domínio do problema. O desempenho do modelo é monitorado permanentemente durante o treinamento, observando o nível de precisão de convergência do modelo e os tempos necessários para atingir tal nível. O experimento é repetido incrementando o tamanho do conjunto de treinamento. Um segundo experimento deve ser realizado depois para um número intermediário de exemplos. Os resultados destes experimentos fornecerão ao usuário a informação necessária para determinar a sensibilidade do desempenho da rede em comparação ao tamanho do conjunto de treinamento (Flood & Kartam, 1994).

A distribuição dos exemplos de treinamento dentro do espaço-domínio do problema pode ser muito significativa no desempenho do aprendizado e da capacidade de generalização do modelo. Devido ao fato de as RN's serem muito limitadas como extrapoladores, os exemplos de treinamento devem idealmente atingir os extremos do espaço-domínio do problema em todas as dimensões do mesmo. De igual forma, é importante que os exemplos de treinamento estejam homogeneamente distribuídos dentro desta região. Não sendo assim, o treinamento terá uma tendência inevitável a se concentrar nas regiões onde os exemplos estão mais densamente localizados, descartando ou minimizando a importância daqueles localizados em regiões menos densamente amostradas (Flood & Kartam, 1994).

Sendo que a distribuição dos exemplos de treinamento no espaço-domínio do problema depende da maneira como eles são amostrados ou gerados, não é sempre possível controlar a sua distribuição ao longo do espaço-domínio do problema. Existem várias maneiras por meio das quais os exemplos de treinamento podem ser coletados, as mais frequentes são:

- Coleta de informação histórica a partir de dados publicados.
- Observação direta da situação a ser modelada. Dados experimentais.
- Consulta com um especialista humano.
- Análise de um modelo alternativo da situação a ser considerada.
- Análise de um modelo inverso da situação a ser considerada

Obviamente os dois primeiros métodos de coleta de informação não oferecem a possibilidade de controlar totalmente a distribuição dos exemplos de treinamento no espaço-domínio do problema. O controle de tal distribuição é maior nos outros três métodos, no entanto, tampouco é absoluto (Flood & Kartam, 1994).

3.4.3.3 – ARQUITETURA DA REDE

Nesta fase devem ser definidos o número e tipo de entradas a serem usadas, assim como o tipo de saídas necessárias. Segundo as informações obtidas da bibliografia analisada, as redes do tipo PMC com retropropagação do erro aparentemente fornecem a melhor ferramenta para a modelagem de problemas como o proposto neste trabalho. Como um aproximador universal de funções, um perceptron de múltiplas camadas pode ser visto como um veículo prático para realizar um mapeamento não-linear de entrada-saída de natureza geral. No entanto, a definição da arquitetura ótima para um problema específico de aproximação, não é uma questão trivial, obrigando o usuário a testar sistematicamente diferentes arquiteturas até conseguir uma que apresente as melhores características em termos de tempo de aprendizagem, facilidade de implementação e capacidade de generalização.

Em relação ao número de camadas ocultas, necessário para solucionar um problema determinado, pode-se dizer que, tal como se verá mais adiante, uma camada oculta de neurônios (operando com funções de ativação do tipo sigmóide) é suficiente para modelar qualquer superfície solução de interesse prático, no entanto, podem existir superfícies extremamente difíceis de serem modeladas com uma só camada oculta de neurônios com funções de ativação sigmóide. Neste caso, duas camadas garantem a flexibilidade necessária para modelar superfícies solução de formas complexas.

A utilização de uma RN com duas camadas, é justificada pelo fato de os neurônios tenderem a interagir entre si globalmente em redes de uma única camada, o que torna difícil melhorar a aproximação em um ponto sem piorá-la em outro. Já com duas camadas intermediárias, o processo se torna mais gerenciável, pois as características locais são extraídas da primeira camada, enquanto que as características globais são extraídas da segunda, na qual cada neurônio combina as saídas dos neurônios da primeira camada operando numa região particular do espaço (Haykin, 2001).

Existem duas exceções para as quais definitivamente não faz sentido adotar duas camadas ocultas numa rede do tipo perceptron:

A primeira consiste em modelos com um só neurônio de entrada.

A segunda consiste em problemas nos quais a informação de entrada assume valores binários exclusivamente. Em ambos os casos, a solução pode ser encontrada usando uma só camada oculta de neurônios (Flood & Kartam, 1994).

Em relação ao número de neurônios a serem usados, pode-se dizer que não existe uma regra geral que permita determinar de maneira precisa o tamanho das camadas ocultas, e esse

problema torna-se mais complexo na medida que aumenta o número dessas camadas ocultas. Embora intuitivamente possa se concluir que o aumento do número de neurônios incrementa a capacidade do modelo para encontrar uma solução para o problema apresentado, também é certo que um número exagerado de neurônios pode conduzir a uma superfície solução que, enquanto se ajusta bem aos pontos de treinamento, desvia-se consideravelmente dos pontos de teste ou validação localizados em posições intermediárias. Adicionalmente, um número exagerado de neurônios ocultos faz com que o processo de treinamento, e posteriormente de utilização da rede, seja excessivamente lento. Em contraste, se o problema for de alta complexidade é provável que um modelo com poucos neurônios não seja capaz de modelar adequadamente todas as feições e detalhes da superfície solução do mesmo.

Numa tentativa por resolver este dilema o que tradicionalmente tem sido feito é considerar e testar uma ampla gama de configurações diferentes, descartando as que apresentam os piores desempenhos e deixando só a melhor. Para consegui-lo pode se começar com uma rede com um número grande de neurônios, da qual vão se eliminando gradativamente aqueles que não contribuem significativamente com a solução até atingir um número ótimo. Uma outra aproximação consiste em começar com um número pequeno de neurônios e aumentar progressivamente o tamanho da rede até atingir o tamanho ideal que fornece a melhor resposta de generalização.

Em relação às funções de ativação, Haykin (2001) menciona que um perceptron de múltiplas camadas treinado com o algoritmo de retropropagação pode, em geral, aprender mais rápido (em termos do número de iterações de treinamento necessárias) quando a função de ativação sigmóide incorporada no modelo for antissimétrica (*tangente hiperbólica*) do que quando ela for não-simétrica (*logística*). A minimização do tempo de aprendizagem depende de se evitar o uso de entradas com média diferente de zero. Agora, considerando-se o vetor de sinal x aplicado a um neurônio na primeira camada de um PMC, é fácil remover a média de cada elemento de x antes de aplicá-lo à rede. Mas o que dizer dos sinais aplicados nas restantes camadas ocultas e na camada de saída? A resposta a esta questão se encontra no tipo de função de ativação usada na rede. Se a função for não-simétrica, como o caso da *logística*, a saída de cada neurônio está restrita ao intervalo $[0,1]$. Uma escolha assim introduz uma fonte de *bias* sistemático para aqueles neurônios localizados além da primeira camada oculta da rede. Para superar este problema, precisamos usar uma função de ativação anti-simétrica tal como a *tangente hiperbólica*. Com esta escolha, permite-se que a saída de cada neurônio

assuma valores tanto positivos como negativos no intervalo $[-1,1]$, e neste caso é provável que a sua média seja zero.

3.4.3.4 – NORMALIZAÇÃO DAS ENTRADAS

O progresso do treinamento pode ser pouco equilibrado se os exemplos de treinamento se encontrarem definidos em regiões relativamente estreitas em um sentido e alongadas em outro, tal como a mostrada na Figura 3.10a. Nestas situações, é claro que a forma da superfície-solução pode mudar rapidamente nas direções para as quais o domínio de treinamento é estreito, ou muito lentamente nas direções onde o domínio se alarga. O problema pode ser contornado mediante a normalização do domínio de treinamento ao longo de cada dimensão tal como se visualiza na Figura 3.10b (Flood & Kartam, 1994).

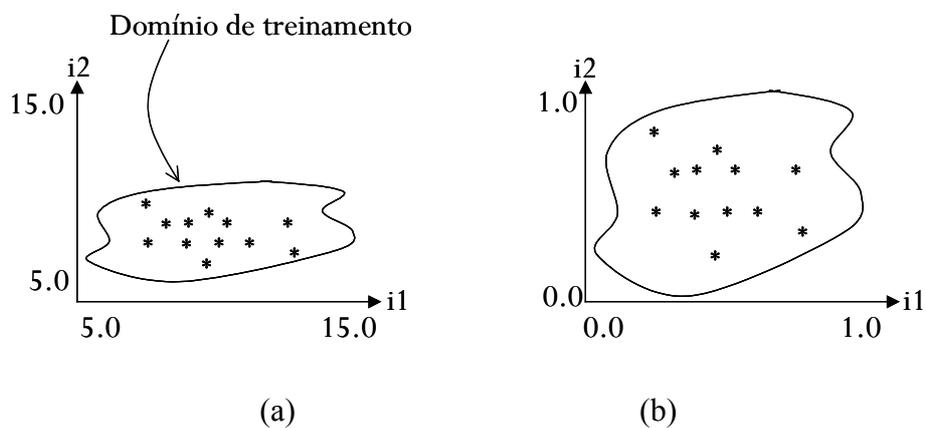


Figura 3.10. Efeito da normalização do conjunto de treinamento. (a) domínio de treinamento antes da normalização; (b) domínio de treinamento após a normalização. (modificado de Flood & Kartam, 1994).

Tal normalização pode ser aplicada também nos valores das saídas do problema (output) de tal forma que a amplitude da superfície-solução para cada saída seja a mesma. Isso pode ser conveniente em situações nas quais existem diferenças importantes nas amplitudes das superfícies-solução correspondentes a cada saída (output). Em caso de não se normalizar os valores de saída, pode acontecer que o treinamento se concentre, pelo menos nos estágios iniciais do processo, exageradamente no aprendizado daquelas saídas que apresentam as maiores amplitudes (Flood & Kartam, 1994). No entanto, deve ser levado em conta que, segundo a recomendação dada no item anterior sobre a escolha da função de ativação, a normalização mais eficiente para funções do tipo *tangente hiperbólica*, será a que resulte no

intervalo $[-1,1]$. Assim, o valor médio de cada variável de entrada, calculado sobre todo o conjunto de treinamento, será próximo de zero ou pelo menos pequeno comparado com o desvio padrão.

Existe ainda uma outra recomendação relacionada com o pré-processamento dos dados de entrada. Quando o conhecimento das características do problema é escasso pode acontecer que um número exagerado de variáveis de entrada seja usado nas primeiras fases de projeto, o que aumenta a probabilidade de que algumas de essas variáveis estejam correlacionadas de maneira estreita, dificultando o processo de aprendizagem ou, no mínimo, fazendo-o muito lento. Para evitar esta situação deve-se realizar uma análise de componentes principais para diminuir o número de variáveis eliminando a correlatividade entre elas. Uma vez feito isto, procede-se à normalização as variáveis.

3.4.4 – GENERALIZAÇÃO

O processo de treinamento supervisionado pode ser visualizado graficamente representando, em um espaço adequado, os pontos de entrada (*inputs*) e de saída (*outputs*) correspondentes a um dado exemplo. Suponha-se um caso no qual o conjunto de treinamento seja composto por sete vetores de treinamento cada um deles constituído por duas entradas e uma saída. Neste caso (ver Figura 3.11), cada vetor de treinamento pode ser visto como um ponto no espaço que indique o seu valor de saída (o_1) versus os dois valores de entrada (i_1 e i_2).

Coletivamente, os pontos representados desta forma definem uma superfície de solução para o problema. O objetivo é treinar a rede para que possa fornecer uma aproximação aceitável a esta superfície alvo. A intenção, no entanto, não é simplesmente treinar a rede para reproduzir as soluções dos exemplos do conjunto de treinamento, mas sim, treiná-la de tal forma que ela possa identificar uma solução generalizada aplicável ainda a todo o universo de exemplos que o problema possa apresentar (Flood & Kartam, 1994).

Na aprendizagem por retropropagação, alimenta-se a rede com tantos exemplos de treinamento quanto possível esperando que a mesma seja capaz de generalizar. Diz-se que uma rede generaliza bem quando o mapeamento entrada-saída calculado pela rede for correto para dados de teste não utilizados para o treinamento da rede. Aqui se assume que os dados de teste são retirados da mesma população usada para gerar os dados de treinamento (Haykin, 2001).

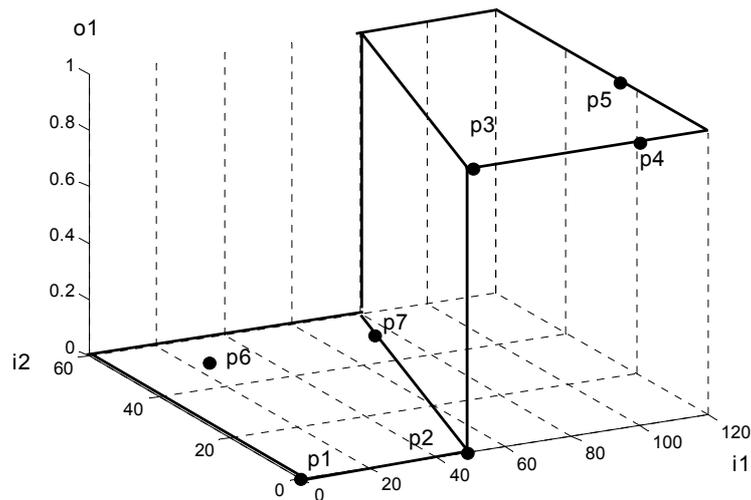


Figura 3.11. Localização espacial de pontos para um caso com sete vetores de treinamento (modificado de Flood & Kartam, 1994).

Na Figura 3.12 são apresentadas três soluções alternativas (representadas pelas linhas contínuas) para um problema do qual só se conhece um conjunto de vetores de treinamento, alguns dos quais foram separados para testar o desempenho da rede. Os conjuntos de treinamento (o) e teste (x) são os mesmos para todos os gráficos. A primeira solução se ajusta exatamente aos pontos de treinamento, no entanto falha na hora de fornecer uma solução generalizada válida para os pontos de teste. Uma solução mais aceitável pode ser a representada no segundo gráfico, cuja aproximação é bem mais admissível para os pontos de teste, mantendo-se bem ajustada aos pontos de treinamento. Por outro lado é possível que os valores que compõem o conjunto de treinamento contenham erros (provavelmente devido a imprecisões nas medições ou devido a mudanças dos valores no tempo), e neste caso a segunda solução poderia ser considerada uma interpretação excessivamente literal dos pontos de treinamento. Sendo este o caso, a curva que melhor identifica uma tendência dos pontos de treinamento é mais adequada, ou seja a curva representada no terceiro gráfico (Flood & Kartam, 1994).

O grau de generalização que a rede fornece depende do número e configuração dos neurônios na camada ou camadas ocultas, do número e dispersão dos vetores ou exemplos de treinamento, e igualmente do tipo de rede utilizada e do algoritmo de treinamento adotado. Por exemplo, se o número de neurônios é pequeno, então a rede não será capaz de modelar todas as características inerentes à forma da superfície solução, dando como resultado uma resposta parecida com a apresentada na Figura 3.12c. Em contraste, se o número de neurônios

é excessivo, a superfície solução dada pela rede pode conter formas falsas (como depressões ou pontas) em lugares intermediários aos pontos de treinamento (ver Figura 3.12a). De maneira similar, um conjunto muito limitado de exemplos de treinamento pode ser insuficiente para que a rede possa localizar e representar detalhes importantes da superfície solução diminuindo a sua capacidade de fornecer respostas adequadas nas regiões localizadas entre os pontos de treinamento (Flood & Kartam, 1994).

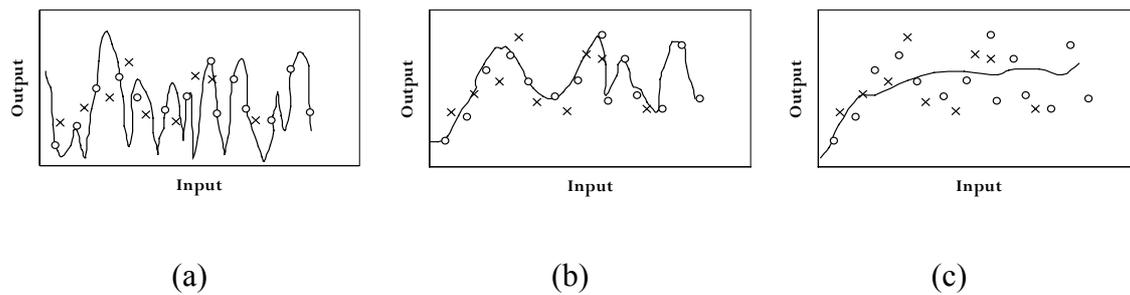


Figura 3.12. Diferentes alternativas de generalização para um conjunto de treinamento: (a) interpolação deficiente; (b) interpretação literal; (c) identificação de tendência. (modificado de Flood & Kartam, 1994).

A eficiência do treinamento é, então, avaliada de acordo com a minimização do erro de teste mais do que com a minimização do erro de treinamento. Em alguns casos, o treinamento excessivo conduz à memorização dos exemplos de treinamento por parte da rede, ocorrendo o chamado *overfitting* (ajuste excessivo com interpolação deficiente). Neste caso, a rede acaba encontrando uma característica (devido ao ruído, por exemplo) que está presente nos dados de treinamento, mas não na função subjacente que deve ser modelada (Haykin, 2001). Uma das maneiras de evitar este fenômeno é usar a técnica da *parada antecipada* do processo de treinamento, na qual o treinamento é interrompido quando o erro de teste alcança o seu valor mínimo não importando se o erro de treinamento continua a diminuir. Caso a rede esteja aprendendo uma solução com um grau aceitável de generalização para o problema dado, a curva de evolução do erro dos exemplos de teste deve acompanhar a tendência da curva de evolução do erro dos exemplos de treinamento (ver Figura 3.13). No outro extremo, se a rede apenas está aprendendo a tendência dada pelos exemplos de treinamento, a curva de evolução do erro dos exemplos de teste não acompanha a tendência da curva de evolução do erro de treinamento.

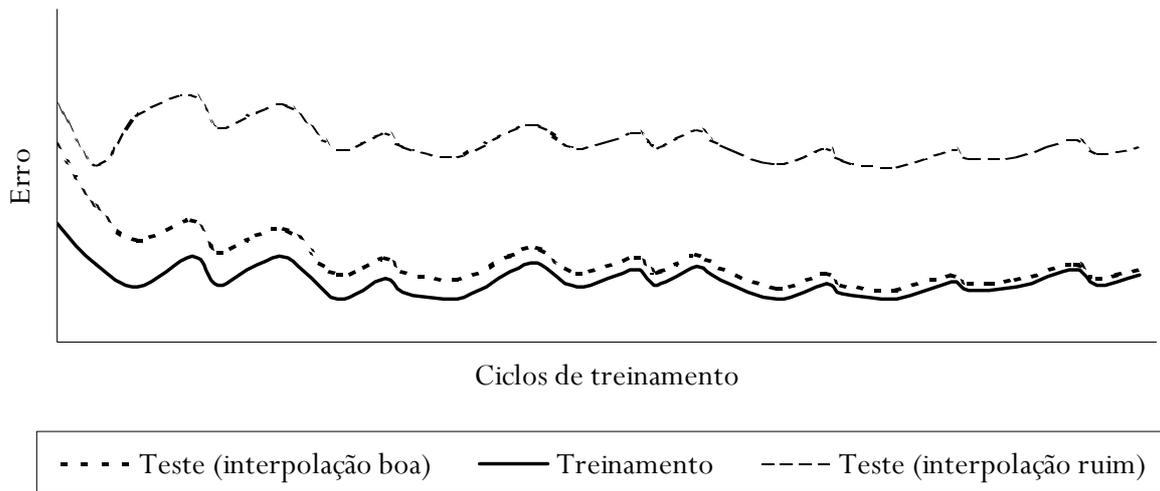


Figura 3.13. Monitoramento do desempenho da rede durante o processo de treinamento (modificado de Flood & Kartam, 1994).

3.4.5 – APROXIMAÇÃO DE FUNÇÕES

Um PMC treinado com o algoritmo de retropropagação pode ser visto como um aproximador universal não-linear de entrada-saída. Para estimar a capacidade do PMC do ponto de vista do mapeamento entrada-saída, surge a questão fundamental sobre qual é o número mínimo de camadas ocultas que fornece uma aproximação aceitável de qualquer mapeamento contínuo?

A resposta a esta questão está incorporada no Teorema de Aproximação Universal que pode ser formulado como (Haykin, 2001):

Suponha que $\phi(\cdot)$ seja uma função contínua não-constante, limitada e monotonamente crescente. Suponha que I_{m_0} represente o hiper-cubo unitário $[0, 1]^{m_0}$ de dimensão m_0 . O espaço das funções contínuas em I_{m_0} é representado por $C(I_{m_0})$. Então, dada qualquer função $f \in C(I_{m_0})$ e $\varepsilon > 0$, existe um conjunto de constantes reais α_i , b_i e w_{ij} tais que podemos definir:

$$F(x_1, x_2, \dots, x_{m_0}) = \sum_{i=1}^{m_1} \alpha_i \phi \left(\sum_{j=1}^{m_0} w_{ij} x_j + b_i \right) \quad (3.56)$$

como uma aproximação da função $f(\cdot)$; isto é,

$$\left| F(x_1, x_2, \dots, x_{m_0}) - f(x_1, x_2, \dots, x_{m_0}) \right| < \varepsilon \quad (3.57)$$

para todo x_1, x_2, \dots, x_{m_0} que se encontra no espaço de entrada.

O Teorema de Aproximação Universal é diretamente aplicável aos PMC's. Notamos primeiro que a função logística utilizada como a não-linearidade em um modelo neural do tipo PMC é de fato uma função não-constante, limitada e monotonamente crescente; ela portanto, satisfaz as condições impostas pela função $\phi(\cdot)$. A Eq.(3.56) representa a saída de um PMC descrito como segue (Haykin, 2001):

A rede tem m_0 nós de entrada e uma camada oculta com m_1 neurônios; as entradas estão representadas por x_1, \dots, x_{m_0} .

O neurônio oculto i tem pesos sinápticos w_{i1}, \dots, w_{im_0} , e bias b_i .

A saída da rede é uma combinação linear das saídas dos neurônios ocultos, com $\alpha_1, \dots, \alpha_{m_1}$ definindo os pesos sinápticos da camada de saída.

O teorema afirma que uma única camada oculta é suficiente para um PMC calcular uma aproximação para um dado conjunto de treinamento representado pelo conjunto de entradas x_1, \dots, x_{m_0} e a saída desejada (alvo) $f(x_1, \dots, x_{m_0})$. Entretanto, o teorema não diz que a única camada é ótima no sentido de tempo de treinamento, facilidade de implementação, ou (mais importante) generalização (Haykin, 2001).

3.5 – AS REDES COMPETITIVAS

Até aqui a análise tem se concentrado no treinamento supervisionado de redes neurais. Uma maneira alternativa, porém não menos importante de treinamento é o chamado treinamento não supervisionado. Neste caso, os vetores de treinamento carecem de soluções (outputs) deixando à rede o trabalho de determiná-las de maneira automática ao longo do processo de treinamento (Flood & Kartam, 1994).

Um caso particular de redes neurais que usam o treinamento não supervisionado é constituído pelas chamadas redes auto-organizáveis ou simplesmente mapas auto-organizáveis (SOM – *Self Organizing Maps*). Redes que utilizam o paradigma do aprendizado supervisionado, em que um supervisor externo fornece informações sobre respostas desejadas para os padrões de entrada, limitam o seu campo de aplicação àqueles onde estes dados são conhecidos. Existem várias aplicações em que é necessário que a rede seja treinada sem um supervisor externo ou punição/recompensa. Neste caso, a única informação fornecida está no conjunto de padrões de entrada, permitindo à rede definir seus parâmetros por si própria, sem auxílio externo (Braga et al., 2000).

Este tipo de redes é especialmente indicado para resolver problemas de classificação de padrões, nos quais as classes não são conhecidas *a priori*. Existem basicamente dois algoritmos de treinamento para as redes do tipo SOM: o aprendizado competitivo e o aprendizado hebbiano.

No aprendizado competitivo, os nós competem entre si pelo direito de atualizar seus pesos, já no caso do aprendizado hebbiano o peso de uma conexão sináptica é ajustado se houver sincronismo entre os níveis de atividade das entradas e saídas. Neste caso, se dois neurônios, em lados distintos da sinapse, são ativados sincronicamente, ter-se-á o fortalecimento desta sinapse. Em contraste, se os neurônios forem ativados assincronicamente, a sinapse será enfraquecida ou mesmo eliminada. A seguir serão tratadas com certo grau de detalhe, as redes do tipo SOM competitivas, por terem sido usadas no processo de pré-processamento dos dados de entrada de alguns dos modelos neurais que serão descritos nos capítulos subseqüentes.

As redes SOM competitivas foram desenvolvidas por Tevuo Kohonen na década de 80, baseado na estrutura do mapa topológico presente no córtex cerebral. Sabe-se que o cérebro dos seres mais sofisticados possui áreas que são responsáveis por funções específicas. Existem áreas, por exemplo, dedicadas à fala, à visão, ao controle motor, à sensibilidade ao toque etc. Cada uma destas áreas contém subáreas. Cada subárea mapeia internamente respostas do órgão sensorial representado por ela (Barga et al., 2000).

Observa-se que esses neurônios estão espacialmente ordenados dentro destas áreas, e assim neurônios topologicamente próximos tendem a responder a padrões ou estímulos semelhantes. Esta ordenação topológica é resultado do uso de interação lateral entre as células do córtex cerebral. Esta interação lateral é geralmente modelada por uma função conhecida como “chapéu mexicano”, segundo a qual cada neurônio influencia o estado de ativação dos seus vizinhos de três formas possíveis (Figura 3.14):

Excitatória, se os vizinhos estão próximos dele, por exemplo, dentro de uma área de raio R1.

Inibitória, se os vizinhos estão fora da área anterior, mas dentro de uma segunda área, por exemplo, de raio R2.

Levemente excitatória, se os vizinhos estão fora das duas áreas anteriores, mas dentro de uma terceira área de raio maior, R3.

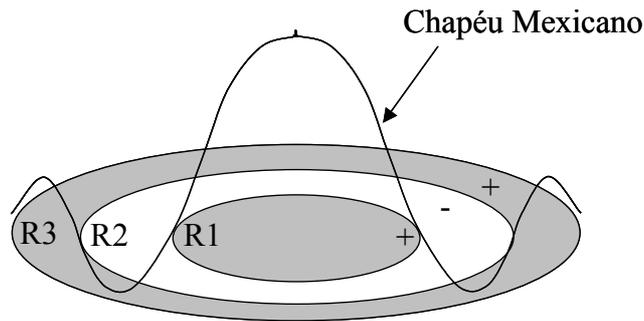


Figura 3.14. Tipos de estímulos produzidos de acordo com a função chapéu mexicano (modificada de Braga et al., 2000)

A rede SOM utiliza um algoritmo de aprendizado competitivo em que os nós da camada de saída competem entre si para se tornarem ativos, ou seja, para ver quem gera o maior valor de saída. Assim, para cada padrão de entrada, apenas um nó de saída se torna ativo. Esta competição é conhecida como “*winner-takes-all*”, o vencedor leva tudo.

Quanto à arquitetura, os neurônios numa rede SOM se organizam em uma grade multidimensional. Cada neurônio recebe todas as entradas e funciona como um discriminador de características. Quanto mais semelhante a entrada for do vetor de pesos de um nodo, maior o valor da sua saída. Durante a fase de aprendizado os nós se organizam topologicamente fazendo com que os padrões detectados por um dado nodo estejam relacionados com as coordenadas da posição do nodo dentro do reticulado. Assim, as localizações espaciais dos nodos correspondem a características dos padrões de entrada, daí o nome de mapas auto-organizáveis. O estado de ativação de um neurônio é determinado pela distância entre seu peso e o vetor de entrada. A função de ativação da rede SOM é baseada na medida de distância euclidiana:

$$d_j = \sqrt{\sum_i (x_i - w_{ij})^2} \quad (3.58)$$

O treinamento de uma rede SOM é competitivo e não supervisionado. Cada vez que um novo padrão é apresentado à rede, os nodos competem entre si para determinar qual gera a maior saída. Definido o nodo vencedor segue o processo de atualização dos pesos. Neste processo apenas o nodo vencedor e seus vizinhos dentro de uma certa área de vizinhança atualizam seus pesos segundo as equações que se apresentam a seguir (Kasabov, 1998):

$$W_j(t+1) = w_j(t) + \alpha \cdot (x - w_j(t)), \text{ se } j \in N_t. \quad (3.59)$$

$$W_j(t+1) = w_j(t), \text{ se } j \notin N_t. \quad (3.60)$$

onde $w_j(t)$ é o vetor de pesos correspondente ao neurônio j , α é a taxa de aprendizado e N_t é a região de vizinhança do nodo vencedor, tudo no instante de tempo t . Na Figura 3.15 apresenta-se um resumo do algoritmo de treinamento para uma rede SOM. Após cada exemplo de treinamento ter sido apresentado, o neurônio vencedor é encontrado e os pesos das conexões na sua região de vizinhança N_t são incrementados, enquanto que os pesos das conexões fora dessa região permanecem iguais. É recomendável que o número de ciclos de treinamento seja pelo menos 500 vezes o número de neurônios de saída (Kasabov, 1998).

K0	Adotam-se números aleatórios pequenos para os vetores de pesos iniciais, $w_j(t=0)$, para cada neurônio j da saída da rede.
K1	Apresenta-se um vetor de entrada x no momento correspondente ao seguinte passo de tempo, t .
K2	Calcula-se a distância d_j (no espaço n -dimensional) entre x e os vetores de peso $w_j(t)$ para cada neurônio j . Num espaço Euclidiano essa distância é calculada como $d_j = \sqrt{\sum (x_i - w_{ij})^2}$.
K3	O neurônio k que é o mais próximo de x é declarado vencedor. Torna-se então o centro de uma região de vizinhança N_t .
K4	Mudam-se todos os vetores de peso na área de vizinhança segundo: $W_j(t+1) = w_j(t) + \alpha \cdot (x - w_j(t)), \text{ se } j \in N_t.$ $W_j(t+1) = w_j(t), \text{ se } j \notin N_t.$
K5	Todos os passos de K1 até K4 são repetidos para todos os exemplos de treinamento. N_t e α decrescem com o tempo. O processo de treinamento é repetido com os mesmos exemplos de treinamento até a convergência.

Figura 3.15. Resumo do procedimento de treinamento do algoritmo de Kohonen (Kasabov, 1998).

CAPÍTULO 4

APRESENTAÇÃO DOS MODELOS CONSTITUTIVOS NEURAIS PARA AREIAS

As coisas que se pode perceber, ver, ouvir ou construir podem ser explicadas por meio do uso de certos princípios e leis da natureza: conservação da massa, energia, momentum linear e angular, as leis do fluxo eletromagnético e da termodinâmica. Estes são os princípios fundamentais nos quais se baseia a mecânica. Um dos primeiros e notáveis sucessos da mecânica foi sua aplicação no estudo do movimento dos planetas do sistema solar, o que constituía um problema misterioso de tempos antigos (Desai & Siriwardane, 1984).

A Mecânica dos meios contínuos se baseia nos princípios fundamentais da natureza, os quais são independentes da constituição interna do material. No entanto, a resposta de um sistema ou meio, submetido à ação de forças externas, não pode ser determinada unicamente a partir das equações obtidas daqueles princípios fundamentais. A constituição interna do material representa um campo fundamental na ciência (Desai & Siriwardane, 1984).

Neste capítulo são apresentadas algumas idéias básicas sobre a modelagem constitutiva de geomateriais, tanto por meio da utilização de modelos tradicionais do tipo paramétrico, baseados na teoria mecânica da elasto-plasticidade, como por meio de RNA's, o que constitui o objetivo desta tese. São apresentados em detalhe os diferentes modelos de RNA's que foram desenvolvidos para a modelagem constitutiva de areias, começando desde o modelo mais simples até o mais complexo. Os resultados das simulações obtidas são apresentados para cada um dos modelos descritos junto com explicações detalhadas sobre as diferentes fases do projeto: pré-processamento de dados, criação do modelo, treinamento e validação.

4.1 – INTRODUÇÃO

Uma lei constitutiva ou modelo constitutivo representa um modelo matemático que descreve as idéias sobre o comportamento de um material. Em outras palavras, tenta simular um comportamento físico tal como foi percebido mentalmente pelo autor do modelo. A principal vantagem de estabelecer um modelo matemático é a sua aplicação para resolver problemas quantitativamente complexos. Não obstante, o poder de um modelo constitutivo depende fundamentalmente do grau de conhecimento do fenômeno físico e a capacidade para o simular (Desai & Siriwardane, 1984).

As equações constitutivas dos diversos materiais englobam as características e aspectos relevantes de cada material e as situações de carregamento a que estão expostos. Para tal, estas equações possuem parâmetros que devem ser determinados, geralmente por meio de experimentos realizados com o material de interesse. Em se tratando de solos e rochas, este procedimento é seguido realizando-se ensaios de laboratório e de campo, idealizados para representar o comportamento destes materiais (Dyminski, 2000).

A seção seguinte descreve o que nesta tese tem se chamado de modelos paramétricos ou modelos constitutivos convencionais, os quais estão baseados nas teorias da elasticidade e plasticidade. A seção subsequente apresenta uma abordagem fundamentalmente diferente baseada na utilização de RNA's.

4.2 – MODELOS PARAMÉTRICOS

A maioria dos modelos constitutivos usados em geomecânica está baseada nos conceitos da teoria da elasticidade e da plasticidade. Ao se considerar o solo como um material elástico, supõe-se que o estado de tensão é função somente do estado atual de deformação, não sofrendo deformações permanentes. Entre os modelos que utilizam este conceito, podem ser citados os de Cauchy e Green, de primeira ordem, segunda ordem ou ordem superior. Estes modelos são úteis para estabelecer um marco geral para entender o comportamento do material, porém, eles não englobam o comportamento complexo dos materiais geológicos incluindo não-linearidade, endurecimento, amolecimento e dependência do nível de deformação.

A hipoelasticidade baseia-se em uma lei incremental, na qual o comportamento do material pode ser representado por incrementos de tensão, função da trajetória de tensão e dos incrementos de deformação. Com esta consideração, pode-se simular o amolecimento ou

endurecimento de materiais. Assim, os modelos hipoeelásticos já apresentam uma melhor capacidade de representar o comportamento de materiais geotécnicos do que modelos fundamentados na teoria da elasticidade (Dyminski,2000).

Modelos mais avançados foram desenvolvidos com base na teoria da plasticidade a qual visa entender o comportamento de materiais que são capazes de reter parte das suas deformações durante o processo de descarregamento. Observações experimentais mostram que muitos materiais, como os metais por exemplo, apresentam comportamento elástico até um certo nível de tensão (ponto A na Figura 4.1), recuperando seu estado original quando o carregamento é retirado. Se o material for solicitado além do ponto A, por exemplo até o ponto B, e depois descarregado, haverá uma deformação permanente ou irrecuperável no corpo. Diz-se então que o material experimentou deformações plásticas (Desai & Siriwardane, 1984).

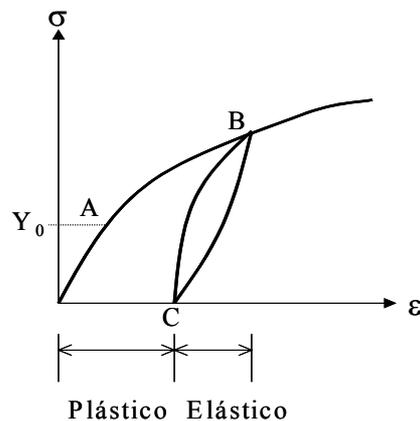


Figura 4.1. Curva típica tensão-deformação para um metal sob condições de carregamento uniaxial simples (modificada de Desai & Siriwardane, 1984)

O primeiro modelo matemático conhecido envolvendo um critério de plastificação aplicável a solos foi formulado por Coulomb em 1773. Desde então e até meados do século XX, vários autores formularam critérios de plastificação que serviram como base para a unificação da teoria da plasticidade levando posteriormente à formulação de modelos mais avançados como o Cam-Clay e o Cam-Clay modificado que apareceram no final da década de 60. Entretanto, este modelo foi desenvolvido para condições específicas de sollicitação, não apresentando desempenho satisfatório fora do plano de tensões axis-simétrico. Nakai & Matsuoka (1986) apresentaram um modelo constitutivo que simula o comportamento de argilas para condições de carregamento gerais, ou seja, tridimensionais verdadeiras. Posteriormente, Nakai (1989) apresentou o modelo equivalente para solos arenosos. Os

modelos foram chamados de T_{ij} –Clay e T_{ij} –Sand, devido ao fato de os autores terem proposto um novo tensor de tensões chamado tensor t_{ij} (Pedroso, 2002).

Na situação unidimensional, a verificação do início da plastificação é imediata, bastando comparar a tensão atuante com a tensão de plastificação ou escoamento, Y_0 (Figura 4.1). Essa comparação pode ser feita por uma expressão como:

$$F(\sigma_x) = \sigma_x - Y_0 \quad (4.1)$$

onde σ_x é a tensão atuante e Y_0 é a tensão de plastificação. Se $F=0$, então $\sigma_x=Y_0$, indicando que o material está plastificado. Se $F<0$ o material está no regime elástico. Não há possibilidade de F ser maior que zero.

Da mesma forma, em casos tridimensionais, utiliza-se a função de plastificação para indicar o início da plastificação. Essa é uma função cujas variáveis independentes são os seis componentes do tensor simétrico de tensões. Portanto, a função de plastificação delimita uma região fechada num espaço hexadimensional por meio de uma hiper-superfície de plastificação. Essa região é referida como domínio elástico. Para um material isotrópico, pode-se esboçar a superfície no espaço tridimensional de tensões principais (Figura 4.2).

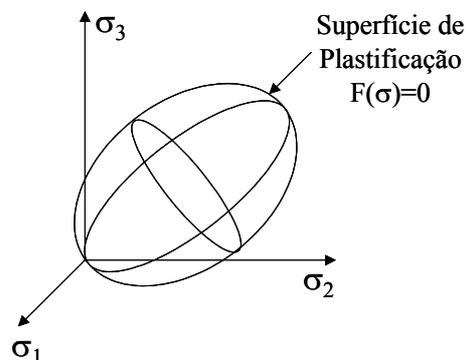


Figura 4.2. Superfície de plastificação no espaço de tensões principais.

A tensão que indica a plastificação Y_0 , nos materiais submetidos a estado unidimensional de tensão e deformação, poderá aumentar (endurecimento; *hardening*) ou diminuir (amolecimento; *softening*) com a ocorrência de deformações plásticas. A forma como isto acontece é regulada pela *lei de endurecimento*.

A direção do incremento de deformação plástica, no caso unidimensional, é a mesma do único componente de tensão e sempre no sentido positivo. Em situações gerais, a direção na qual ocorre o fluxo plástico é definida pela *lei de fluxo* que é uma equação que relaciona os incrementos de deformações plásticas com os gradientes de uma superfície de *potencial*

plástico, $Q(\sigma)$. Esta função pode ser determinada experimentalmente ou pode ser a mesma função de plastificação. Neste último caso diz-se que o modelo constitutivo utiliza uma *lei de fluxo associada*.

Os modelos constitutivos são calibrados a partir de informação experimental, geralmente obtida de ensaios triaxiais convencionais tanto de compressão como de extensão. No entanto, a utilização de ensaios mais avançados é necessária para explorar outras regiões do espaço de tensões. Adicionalmente, os ensaios de laboratório utilizam amostras homogêneas de solo, livres de imperfeições e descontinuidades, o que representa uma limitação inerente à capacidade de simular a resposta do solo no campo.

Estas equações constitutivas definem o comportamento do material usando um conjunto fixo de equações, cuja vantagem principal é a capacidade de generalização. A partir de alguns ensaios de laboratório, e usando um conjunto de hipóteses sobre o comportamento geral tensão-deformação do solo, e as leis de conservação da teoria da mecânica, o modelo consegue extrapolar o comportamento tensão-deformação para regiões do espaço de tensões e deformações não cobertas pelos ensaios de laboratório. Não obstante, uma desvantagem a destacar deste tipo de modelos é a sua limitada ou quase nula capacidade de aprendizagem. Um dado modelo constitutivo possui uma capacidade muito limitada de aprendizado a partir do ajuste das propriedades do modelo. A forma fundamental da relação tensão-deformação não muda. Visando melhorar o desempenho de generalização de uma lei constitutiva um novo modelo tem que ser desenvolvido com um novo conjunto de equações e uma nova superfície de plastificação (Marulanda, 2005).

4.3 – MODELAGEM USANDO REDES NEURAIAS ARTIFICIAIS

Uma das formas “não tradicionais” de se modelar o comportamento mecânico dos solos, é a utilização de RNA's, cujo fundamento matemático já foi descrito no capítulo anterior. Assim, pode-se procurar simular as relações tensão-deformação dos solos, nas mais diversas condições sem a necessidade de se explicitar o conjunto de equações matemáticas que o fariam (Dyminski, 2000). Nesta aproximação *não-simbólica*, o comportamento constitutivo do solo seria capturado diretamente pela RNA e *armazenado* para posteriores generalizações. O interesse de tal aplicação das RNA's, na qual o modelo é construído diretamente a partir de informação experimental é obvio; nesse caso a RNA devidamente treinada poderá substituir

uma relação paramétrica tradicional. A fonte de informação da RNA não é uma fórmula paramétrica mas sim um conjunto de dados experimentais (Lefik & Schrefler, 2003).

No caso de solos arenosos, se aceita que o seu comportamento mecânico depende fundamentalmente de dois fatores: (a) a sua natureza física, e (b) o seu estado físico. O primeiro fator está relacionado com a composição da areia como material granular, e tem sido frequentemente descrito por meio do uso de propriedades como a distribuição granulométrica, o conteúdo de finos, a forma dos grãos e a mineralogia entre outros. Por outro lado, o estado físico da areia está relacionado com as condições físicas “in situ” nas quais o solo existe, e tem sido tradicionalmente descrito com a ajuda de variáveis como a densidade relativa (D_r), o estado de tensão efetiva e o arranjo entre os grãos (Cubrinovski & Ishihara, 2002).

Conseqüentemente, numa aproximação baseada em aprendizado a partir de experiência, onde uma RNA age como um operador constitutivo, o conjunto de variáveis de entrada para o treinamento, deverá conter não só informação sobre o estado de tensão-deformação do solo, mas também informação sobre a natureza física da areia e outras variáveis (Martinez-Carvajal & Farias, 2004). O comportamento constitutivo do material aprendido pela RNA depende da informação disponível usada para o treinamento, e portanto, a resposta do modelo neural será confiável só dentro do espaço-domínio definido pelo conjunto de treinamento.

Um modelo constitutivo neural não possui as mesmas capacidades de generalização (em termos de extrapolação) de um modelo paramétrico convencional que usa um conjunto de hipóteses para definir a relação tensão-deformação dentro de um espaço-domínio correspondente. A RNA requer uma quantidade de informação suficiente para treinamento dentro de todo o espaço-domínio do problema tratado, para poder garantir uma boa resposta dentro dos limites práticos possíveis que o problema possa assumir. Alguns ensaios de laboratório não são suficientes para desenvolver um modelo constitutivo neural. Não obstante, uma vantagem importante dos modelos neurais é a sua flexibilidade de aprendizado. O modelo neural não está limitado por uma relação tensão-deformação pré-definida, nem por uma superfície de plastificação pré-definida. A sua flexibilidade lhe permite, por exemplo, se atualizar facilmente para incorporar as características de algum novo comportamento observado, mediante a realização de uma nova fase de treinamento incluindo novos dados experimentais. Conseqüentemente, os modelos neurais tornam-se a melhor opção para modelagem constitutiva de materiais, nos casos em que a complexidade do comportamento dificulta a sua representação matemática, quando o entendimento físico do problema é escasso, quando as simplificações necessárias para a parametrização do problema são tão

grandes que o modelo paramétrico torna-se não-representativo ou quando se requer uma atualização permanente dos dados do problema.

Nesta seção serão apresentados de maneira seqüencial os resultados dos modelos neurais criados para resolver o problema apresentado. Para a implementação de cada um deles foi adotada a mesma metodologia apresentada nas seções anteriores, sendo que o primeiro e o segundo foram construídos a partir de uma base de dados *piloto* de pequeno porte, enquanto que os modelos subseqüentes incluem no seu treinamento a base de dados completa apresentada na Tabela A.1. Cabe ressaltar que para alguns dos modelos foi necessário realizar um processo de depuração da base de dados o que reduziu o tamanho final do conjunto de treinamento, porém melhorou o desempenho dos modelos. Tal processo será oportunamente apresentado nas seções seguintes.

Tal como foi mencionado no Capítulo 1, a base de dados já coletada está composta por 12 areias diferentes (ver Tabela 1.1), cujos ensaios triaxiais foram digitalizados a partir dos gráficos originalmente apresentados nas referências mencionadas na própria tabela; correspondendo a um total de 171 testes cujas características gerais são apresentadas na Tabela A.1. Os dados correspondentes à areia de Toyoura foram fornecidos pelo NIT (Nagoya Institute of Technology) mediante convênio de cooperação com a Universidade de Brasília

Na Figura 4.3 se apresentam os passos realizados para a recuperação dos pares de dados tensão-deformação necessários para a composição da base de dados. Da imagem original são recuperados, mediante digitalização manual com o auxílio do programa *grabit*, os pontos ao longo das curvas originais (passo 1). Esses pontos são usados para alimentar modelos simples de interpolação polinomial (passo 2) com cujas equações é possível aumentar a densidade de pontos ao longo da curva completa mantendo certo controle sobre o tamanho dos acréscimos de deformação impostos (passo 3).

Uma vez prontos os pares de valores tensão-deformação é possível recuperar para cada ponto das curvas os valores das tensões principais e as correspondentes deformações principais, que constituem o corpo principal de dados do conjunto de treinamento dos diferentes modelos.

4.4 – MODELO NEURAL 1

Nesta seção apresenta-se o procedimento realizado para simular uma RNA do tipo PMC, treinada com um banco de dados experimentais de ensaios triaxiais verdadeiros para

diferentes trajetórias de tensão, com o propósito de reproduzir a relação tensão-deformação de três areias típicas (Toyoura, Cambria e Monterrey). A identificação do tipo de areia é feita a partir do índice de vazios em cada ponto experimental (e_i), do fator de forma das partículas (n_s) e da diferença entre os índices de vazios máximo e mínimo ($e_{\max} - e_{\min}$).

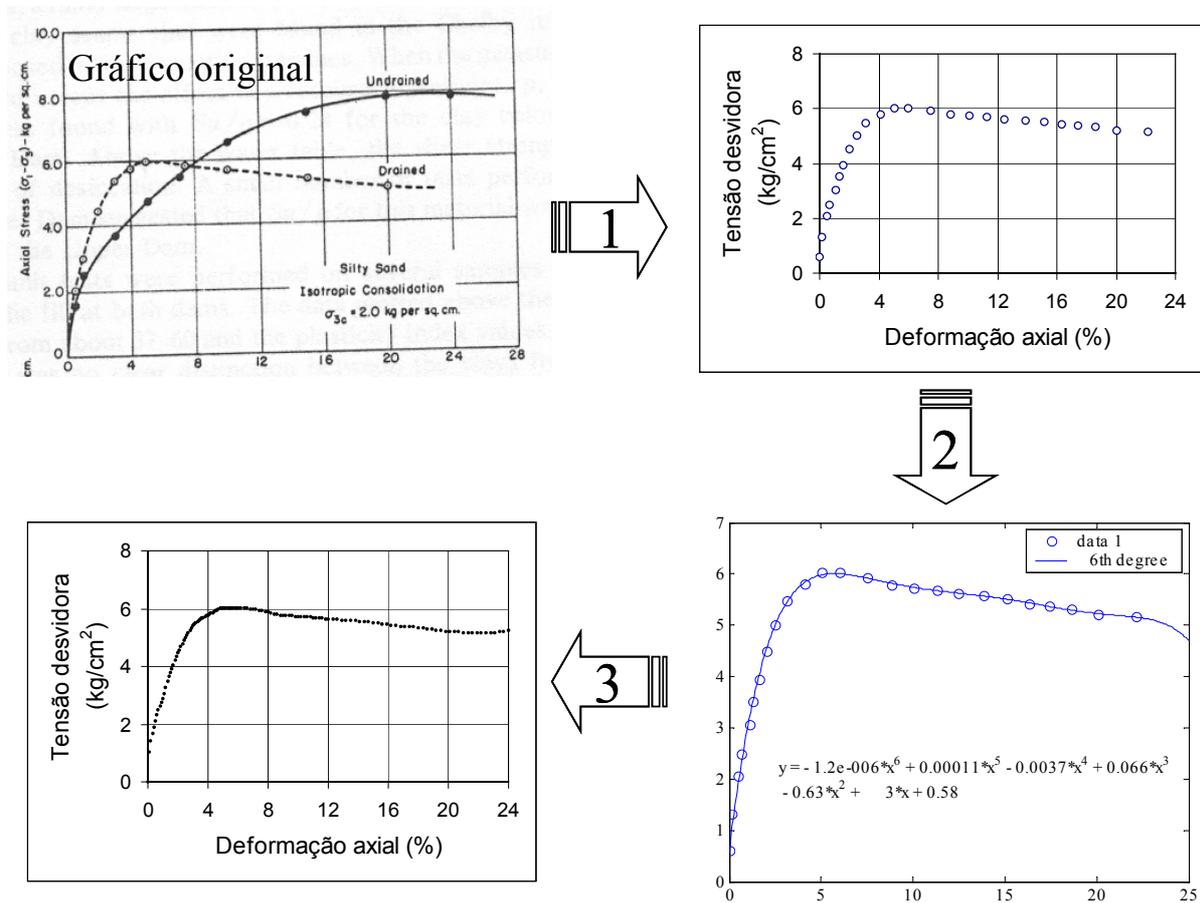


Figura 4.3. Processo de recuperação dos pares de dados tensão-deformação a partir dos gráficos originais dos testes usados na base de dados.

4.4.1 – A BASE DE DADOS

Os principais fatores a serem levados em conta no processo de modelagem constitutiva das areias são: mineralogia, forma das partículas, características granulométricas, índice de vazios e tensão confinante inicial. Desta forma, foram escolhidos para formar uma base de dados “piloto” 32 testes realizados sobre três areias que têm sido amplamente estudadas: Cambria, Toyoura e Monterrey. O efeito da mineralogia não foi considerado, pois as três areias escolhidas são do tipo quartzoso, porém foi incluído um fator de forma dos grãos (n_s),

que, adotando a definição de Hardin (1985) vale 25 para grãos angulosos, 20 para sub-angulosos, 17 para sub-arredondados e 15 para esféricos.

Com relação às características granulométricas, os fatores que podem ser considerados para descrever a condição física das areias são: o diâmetro médio (D_{50}), o coeficiente de uniformidade (C_u), o coeficiente de curvatura (C_c) e a densidade relativa (D_r). Porém, pesquisas recentes sobre caracterização mecânica de areias mostram que a diferença entre os índices de vazios máximo e mínimo ($e_{\max} - e_{\min}$) abrange os efeitos combinados do D_{50} , da granulometria, do teor de finos e da forma das partículas; podendo ainda fornecer informação valiosa e única sobre comportamento de deformação volumétrica desses materiais (Cubrinovski & Ishihara, 2002).

A Figura 4.4 apresenta as trajetórias de tensão no espaço de tensões principais dos ensaios utilizados que correspondem às primeiras 32 colunas da Tabela A1 (ver Apêndice A), formando uma base de dados de 1333 pontos experimentais.

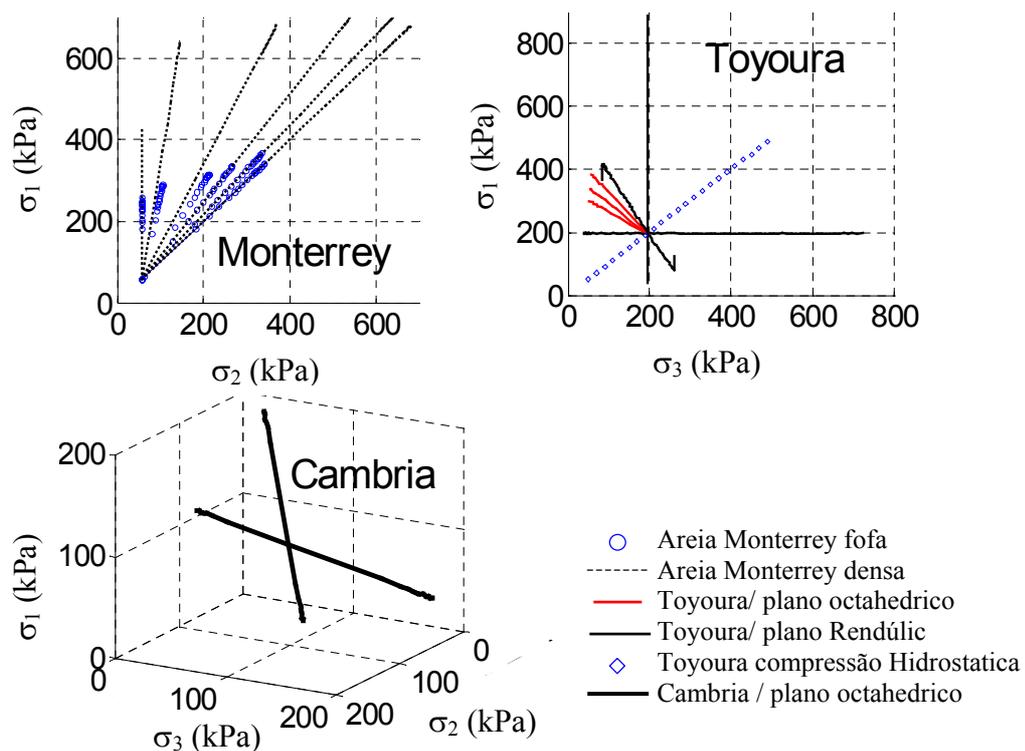


Figura 4.4. Trajetórias de tensão dos ensaios de tensão usados no modelo neural 1

As informações correspondente às areias de Monterrey e de Cambria foram obtidas diretamente a partir da digitalização dos gráficos apresentados pelos autores das diversas referências revisadas. Já os dados da areia de Toyoura, fornecidos pelo NIT (Nagoya Institute of Technology), não precisaram de modificação alguma sendo incorporados diretamente na

base de dados. É importante ressaltar que durante o processo de digitalização dos dados é praticamente impossível conseguir um espaçamento homogêneo entre pontos sucessivos, e conseqüentemente não se tem controle sobre a distribuição final dos valores recuperados, sendo que em alguns casos a densidade de pontos na parte inicial da curva experimental foi grande comparada com a parte final da mesma, enquanto que em outros casos aconteceu o inverso. Por outro lado, a distribuição da quantidade de pontos por ensaio ficou marcadamente heterogênea com 58% dos testes possuindo 20 ou menos pontos, enquanto que o restante, em proporções eqüitativas, está composto por testes com quantidades variáveis entre 40 e 190 pontos tal como se pode apreciar na Figura 4.5 e na Figura 4.6.

Tal como se verá mais adiante na análise do segundo modelo, para aperfeiçoar o desempenho da rede foi necessário modificar as distribuições iniciais dos dados de maneira que a distribuição final do número de pontos por teste ficasse mais homogênea.

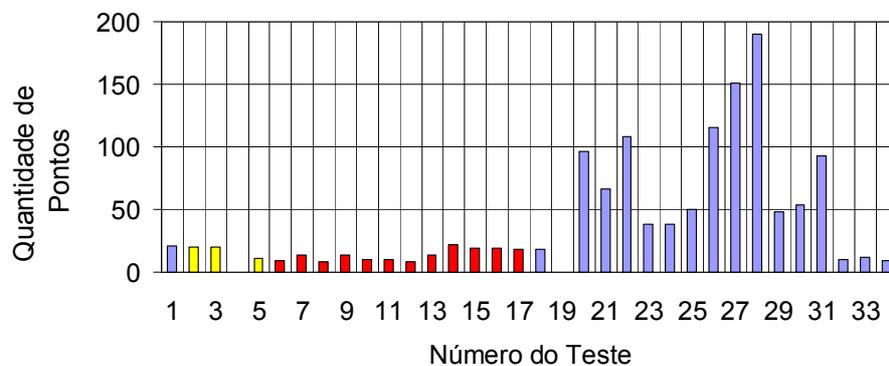


Figura 4.5. Distribuição da quantidade de pontos por teste. Em amarelo os testes da areia de Cambria, em vermelho os testes de Monterrey, em azul os testes de Toyoura.

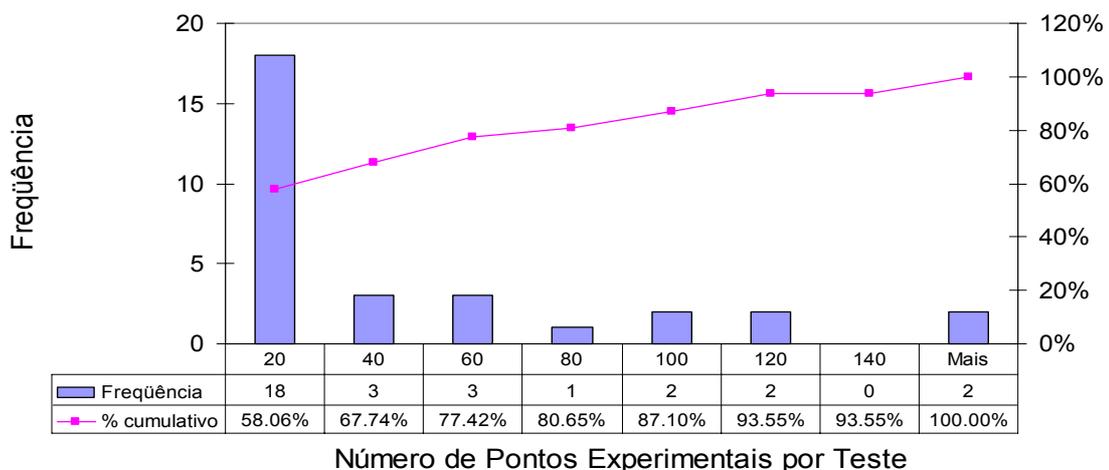


Figura 4.6. Histograma de frequência da quantidade de pontos experimentais por teste.

4.4.2 - ARQUITETURA E TREINAMENTO

Para modelar o comportamento drenado tensão–deformação das areias, criou-se uma rede do tipo PMC com duas camadas intermediárias de 30 neurônios cada uma, uma camada de entrada com 16 neurônios e uma camada de saída com 3 neurônios. A rede foi treinada usando o algoritmo do gradiente descendente simples implementado no programa Matlab, cujo desempenho, em termos do erro mínimo, mostrou-se melhor do que outros algoritmos mais rápidos como o Levenberg-Marquardt. Adicionalmente, usou-se o recurso da parada antecipada, ou validação cruzada, para evitar o efeito de sobre-treinamento. O conjunto de treinamento corresponde a 50% dos pontos experimentais disponíveis, enquanto que os 50% restantes foram divididos em dois conjuntos iguais: um para validação da rede durante o treinamento e o outro para avaliar a capacidade de generalização.

A utilização de uma RNA com duas camadas é justificada pelo fato de os neurônios tenderem a interagir entre si globalmente em redes de uma única camada, o que torna difícil melhorar a aproximação em um ponto sem piorá-la em outro. Já com duas camadas intermediárias, o processo se torna mais gerenciável, pois as características locais são extraídas da primeira camada, enquanto que as características globais são extraídas da segunda, na qual cada neurônio combina as saídas dos neurônios da primeira camada operando numa região particular do espaço (Haykin, 2001).

Na determinação das entradas a serem usadas, procura-se fazer com que o mapeamento entradas-saídas seja único. Caso contrário, a RNA não fornecerá respostas confiáveis, pois para um certo conjunto de valores de entrada poderiam existir várias possíveis respostas ou valores de saída.

Na modelagem constitutiva de solos, uma RNA poderia substituir a matriz constitutiva numa análise usando o Método dos Elementos Finitos, fazendo o mapeamento entre acréscimos de deformação e acréscimos de tensão (deformação controlada). Neste caso (Figura 4.7), para um dado estágio de carregamento atual (i), definido por $(\{\sigma_i\}, \{\varepsilon_i\})$, dado o acréscimo de deformações $\{\Delta\varepsilon_i\}$, encontra-se o acréscimo de tensões $\{\Delta\sigma_i\}$, que gera o estado de tensão-deformação posterior ($i+1$).

Considerando que o comportamento do solo depende dos estados anteriores de carregamento, é importante incluir no conjunto de treinamento, dados relativos ao estágio de carregamento anterior ($i-1$), ou seja $(\{\sigma_{i-1}\}, \{\varepsilon_{i-1}\})$. Em consequência, as entradas da RNA foram: tensão de confinamento inicial (σ_{co}), $(e_{max}-e_{min})$, índice de vazios (e), fator de forma

(n_s), três tensões e três deformações principais correspondentes ao estado de carregamento $i-1$ ($\sigma_{1-i-1}, \sigma_{2-i-1}, \sigma_{3-i-1}, \varepsilon_{1-i-1}, \varepsilon_{2-i-1}, \varepsilon_{3-i-1}$), três deformações principais do estado de carregamento i ($\varepsilon_{1i}, \varepsilon_{2i}, \varepsilon_{3i}$) e, por último, os três acréscimos de deformação principal ($\Delta\varepsilon_{1i}, \Delta\varepsilon_{2i}, \Delta\varepsilon_{3i}$) que permitem definir o estado de deformação do estágio $i+1$. As saídas são os três acréscimos de tensão principal ($\Delta\sigma_{1i}, \Delta\sigma_{2i}, \Delta\sigma_{3i}$) que permitem definir o estado de tensão do estágio posterior. Em resumo, o padrão de treinamento está formado por 16 entradas e 3 saídas, tal como apresentado na Eq.(4.2).

$$\{\Delta\sigma_i\} = f(n_s, \sigma_0, e_{\max} - e_{\min}, e_i, \{\varepsilon_{i-1}\}, \{\sigma_{i-1}\}, \{\varepsilon_i\}, \{\Delta\varepsilon_i\}) \quad (4.2)$$

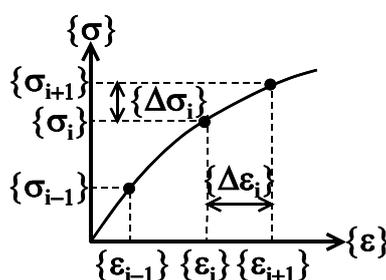


Figura 4.7. Representação dos estados de tensão-deformação necessários para o treinamento da RNA.

Observa-se que não foi considerado o vetor $\{\sigma_i\}$ dentro do conjunto de treinamento, porquanto queria-se saber se a RNA era capaz de fornecer respostas adequadas com a menor quantidade possível de informação sobre os estados de tensão do solo. Adicionalmente, cabe esclarecer que o índice de vazios (e_i) foi fornecido para cada ponto experimental do conjunto de treinamento, com o propósito de prover permanentemente informação sobre o estado de deformação volumétrica do solo e facilitar o mapeamento entrada-saída da RNA. Posteriormente, foi entendido que a incorporação do (e_i), embora facilitando o trabalho da RNA, não representa uma alternativa razoável diante do objetivo de usar o modelo neural como operador constitutivo dentro de um código de elementos finitos, já que o usuário precisaria ter, “a priori” o conhecimento da evolução da deformação volumétrica do material ponto a ponto, coisa que é inviável. Foi por isso que se decidiu, para os modelos subsequentes, substituir o (e_i) pelo (e_0) que representa exclusivamente o valor do índice de vazios inicial do solo.

A utilização de uma rede com duas camadas intermediárias, já foi mencionada e justificada, porém nada foi dito ainda sobre o tamanho das mesmas que no caso foi de 30 neurônios. Em relação a esse assunto só pode ser dito que o tamanho foi escolhido de maneira

aleatória procurando que a rede fosse de tamanho médio considerando a complexidade do problema analisado assim como o tamanho da amostra. Ressalta-se que em se tratando do primeiro teste o objetivo, mais do que buscar uma arquitetura ótima, era vislumbrar se a linha de trabalho indicada apresentava-se promissora e conseqüentemente determinar suas possibilidades futuras para desenvolver a tese ora proposta. Para isso, mantendo a arquitetura constante, a rede foi treinada com quatro algoritmos diferentes que se encontram implementados no módulo de redes neurais do programa Matlab com os seguintes nomes: “*trainlm*”, “*traingdx*”, “*traingdm*” e “*traingd*”. Cada um deles corresponde a uma variação do algoritmo básico de retropropagação no qual os pesos da rede são ajustados segundo a direção negativa do gradiente da função de desempenho, que para o caso considerado foi o erro quadrado médio (MSE) calculado entre as saídas da rede e os valores reais. A Eq. (4.3), que é uma cópia da Eq. (3.15) representa uma iteração deste algoritmo básico (“*traingd*”), na qual $w(n+1)$ é o vetor de pesos atuais, $\delta(n)$ é o gradiente atual e η é a taxa de aprendizado.

$$w(n+1) = w(n) - \eta \delta(n) \quad (4.3)$$

O parâmetro η determina o tamanho das mudanças que serão produzidas nos pesos da rede na direção negativa do gradiente. Se for muito grande o algoritmo torna-se altamente instável. Caso contrário, o algoritmo tardará muito tempo para convergir. Uma característica importante desse algoritmo é que o seu desempenho na busca de um mínimo na superfície de erro depende exclusivamente do valor do gradiente local, sendo fácil, portanto, que a rede fique presa num mínimo local. Este ponto pode ser evitado incluindo um parâmetro que leve em conta a tendência recente da superfície de erro, de tal maneira que a rede possa ignorar pequenas irregularidades na superfície sendo capaz então de escapar de mínimos locais de pouca profundidade. Esse parâmetro, chamado *momentum* é incorporado no algoritmo “*traingdm*”, fazendo com que a mudança nos novos pesos seja igual à soma entre uma fração das mudanças da iteração anterior e as mudanças sugeridas pela regra de retropropagação (Demuth & Beale, 2000).

Observe-se que para os dois algoritmos apresentados anteriormente, a taxa de aprendizado é mantida constante ao longo de todo o processo de treinamento, o que leva o usuário obrigatoriamente a ter que testar sistematicamente, vários valores deste parâmetro até conseguir um que seja ótimo. Uma alternativa mais racional consiste em permitir que a taxa de aprendizado mude durante o processo de treinamento, assumindo o maior valor possível para acelerar a convergência da rede, mas controlando ao mesmo tempo sua estabilidade. Isso

é implementado no algoritmo “traingdx” no qual a primeira iteração do algoritmo é calculada segundo a regra básica de retropropagação, obtendo assim as respostas e o erro da rede. Se o novo erro excede ao anterior em uma quantidade maior do que um valor predefinido, tipicamente 1,04, então os novos pesos são descartados e a taxa de aprendizado é multiplicada por um fator de diminuição, tipicamente 0,7. Caso contrário, ou seja, se o novo erro for menor que o anterior, os pesos são mantidos e a taxa de aprendizado é multiplicada por um fator de aumento, tipicamente 1,05.

Até agora, os três algoritmos apresentados estão baseados no algoritmo básico de retropropagação apresentado na Eq. (4.3), e suas variações foram desenvolvidas a partir da análise do desempenho desse algoritmo básico. Nos algoritmos baseados na técnica básica de retropropagação, a única fonte de informação sobre a superfície de erro é o vetor gradiente. A inclusão do termo de *momentum* é uma maneira simples de introduzir informação de segunda ordem sobre a superfície de erro, o que agiliza o processo de convergência, mas também aumenta o número de parâmetros que o usuário deve ajustar. Para produzir uma melhora significativa no desempenho da rede deve-se usar informação de ordem mais elevada, a qual pode ser incorporada mediante aproximações do tipo quadrático da superfície de erro tal como foi exposto no Capítulo 3 nos itens correspondentes ao método de otimização de Newton e de Gauss-Newton.

Assim, uma outra categoria de algoritmos foi desenvolvida a partir do uso destas técnicas de otimização numérica, como o gradiente conjugado e o algoritmo de *Levenberg-Marquardt*. Para o caso deste primeiro modelo analisado, só o *Levenberg-Marquardt* (“*trainlm*”) foi testado. Neste caso, é preciso estimar a matriz Hessiana (H) a partir da Jacobiana (J) segundo a expressão seguinte:

$$H(n) \cong J^T(n)J(n) \quad (4.4)$$

O método calcula as correções dos pesos sinápticos da rede segundo a formulação apresentada na Eq. (4.5) que é uma cópia da Eq. (3.32).

$$w(n+1) = w(n) - (J^T(n)J(n) + \alpha I)^{-1} J^T(n)e(n) \quad (4.5)$$

Quando α é zero, o algoritmo é equivalente ao método básico de Newton, no qual o gradiente corresponde a $J(n)^T$, e a hessiana é aproximada segundo a Eq. (4.4).

Durante o processo de treinamento os algoritmos “*trainlm*”, “*traingdx*” e “*traingdm*” apresentaram convergências muito rápidas atingindo valores de erro quadrático médio (MSE)

mínimo da ordem de 0,01 em apenas 100 iterações ou menos. No entanto, o algoritmo “traingd”, embora muito mais lento, alcançou valores de erro uma ordem de magnitude menor.

Em relação à divisão do conjunto de dados, a mesma foi feita de maneira sistemática seguindo uma seqüência 1:4:1333 (começando do primeiro ponto se tomam valores de quatro em quatro até o ponto final) para o conjunto de teste, e 2:4:1333 para o conjunto de validação. O restante foi usado para o treinamento da rede.

4.4.3 – RESULTADOS DA SIMULAÇÃO COM O MODELO NEURAL 1

Após terem sido treinados vários modelos, o melhor em termos de erro de teste (que foi chamado de *Magola 3*) foi testado para verificar sua capacidade de generalização. Para maior clareza na visualização dos resultados, cada tipo de areia foi simulada individualmente, e os resultados comparados com as trajetórias experimentais. O erro quadrático mínimo de treinamento foi de $4,7 \times 10^{-3}$ em 50000 iterações de treinamento. O erro de teste foi ligeiramente maior: $5,1 \times 10^{-3}$.

A comparação inicial entre as respostas da RNA e os dados experimentais, apresentada na Figura 4.8, foi feita para cada acréscimo de tensão principal $\Delta\sigma_1$, $\Delta\sigma_2$ e $\Delta\sigma_3$, separando os resultados obtidos com a totalidade dos dados, daqueles correspondentes aos dados de teste. Observando a semelhança entre os coeficientes de correlação (R), conclui-se que o critério de parada foi adequadamente aplicado e que a RNA não ficou sobre-treinada.

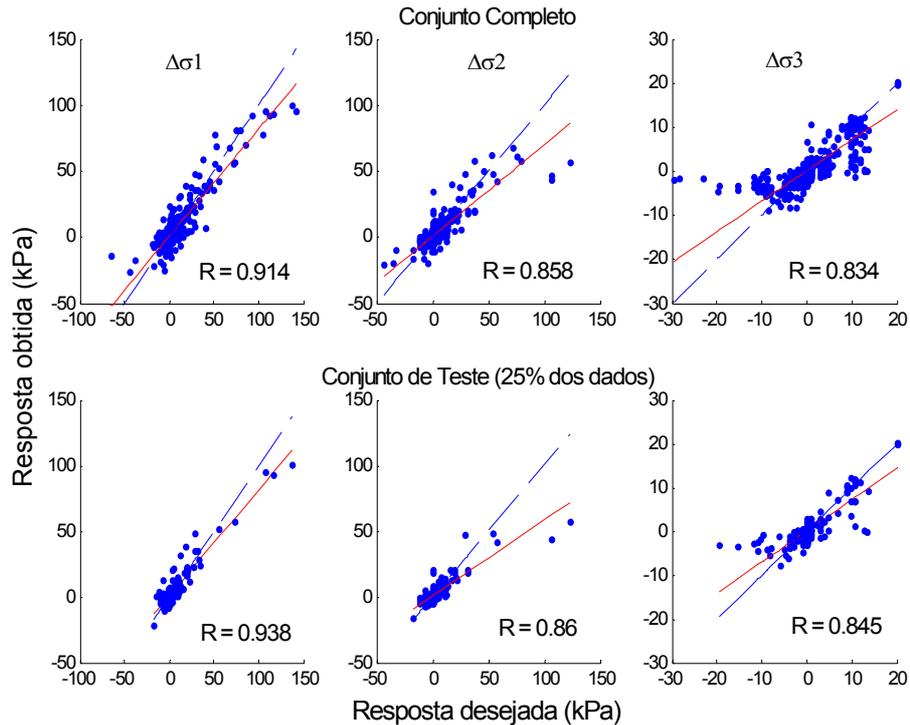


Figura 4.8. Correlação entre as respostas da rede *Magola 3* e os valores experimentais (linha contínua). Em tracejado a linha $y=x$.

A Figura 4.9 apresenta os resultados da simulação feita para a areia de Cambria, para quatro trajetórias no plano octaédrico. Observa-se como a RNA consegue identificar o sentido da trajetória, porém a reprodução da curva experimental ainda é imprecisa. Uma possível explicação pode ser o fato de existir uma diferença muito grande entre a magnitude dos primeiros acréscimos de tensão e os últimos, ou seja, a dispersão dos valores de $\Delta\sigma_1$, $\Delta\sigma_2$ e $\Delta\sigma_3$ é considerável. Na Figura 4.8 se observa que os maiores erros da RNA ocorrem nos pontos mais afastados das nuvens centrais, onde os acréscimos de tensão são maiores. Desta forma, partindo do eixo hidrostático, a RNA é obrigada a simular trajetórias nas quais os primeiros acréscimos de tensão são grandes, cometendo erros significativos que alteram a forma da trajetória.

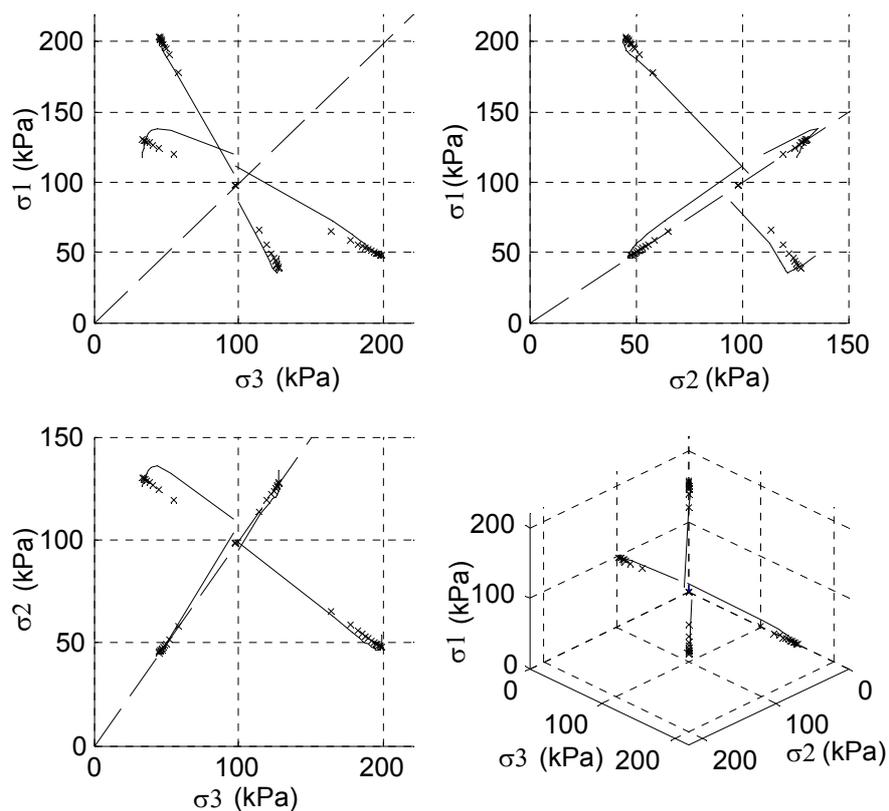


Figura 4.9. Resultados da simulação da rede *Magola 3* para a areia de Cambria. Os pontos são os dados experimentais, as linhas são a resposta da RNA.

Na Figura 4.10 são apresentados os resultados para a areia de Monterrey tanto em estado denso ($e_0=0,57$), como fofo ($e_0=0,78$). Observa-se que, em geral, todas as aproximações são bastante boas, principalmente nos pontos localizados nos extremos das trajetórias, em contraste com o início das mesmas, onde os acréscimos de tensão são muito grandes, ocorrendo maiores erros.

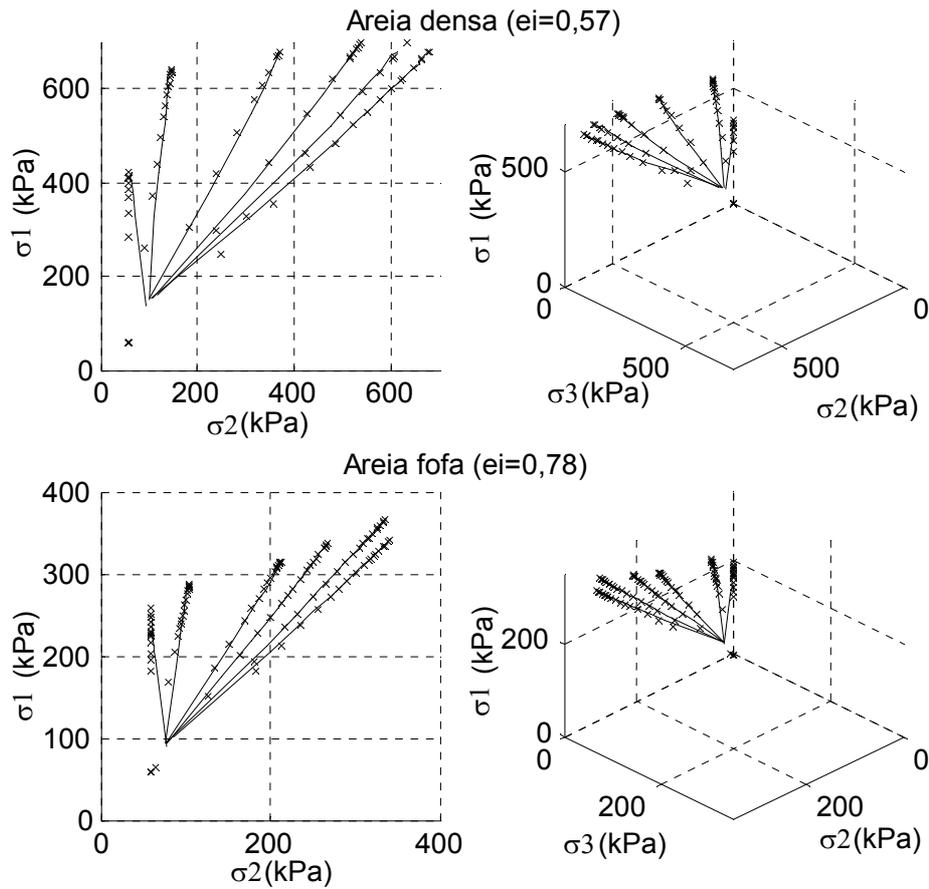


Figura 4.10. Resultados da simulação da rede *Magola 3* para a areia de Monterrey. Os pontos são os dados experimentais, as linhas, a resposta da RNA.

Os resultados das simulações da areia de Toyoura são apresentados na Figura 4.11. Observa-se em todos os gráficos, uma excelente resposta da RNA, que consegue acompanhar de perto as trajetórias experimentais.

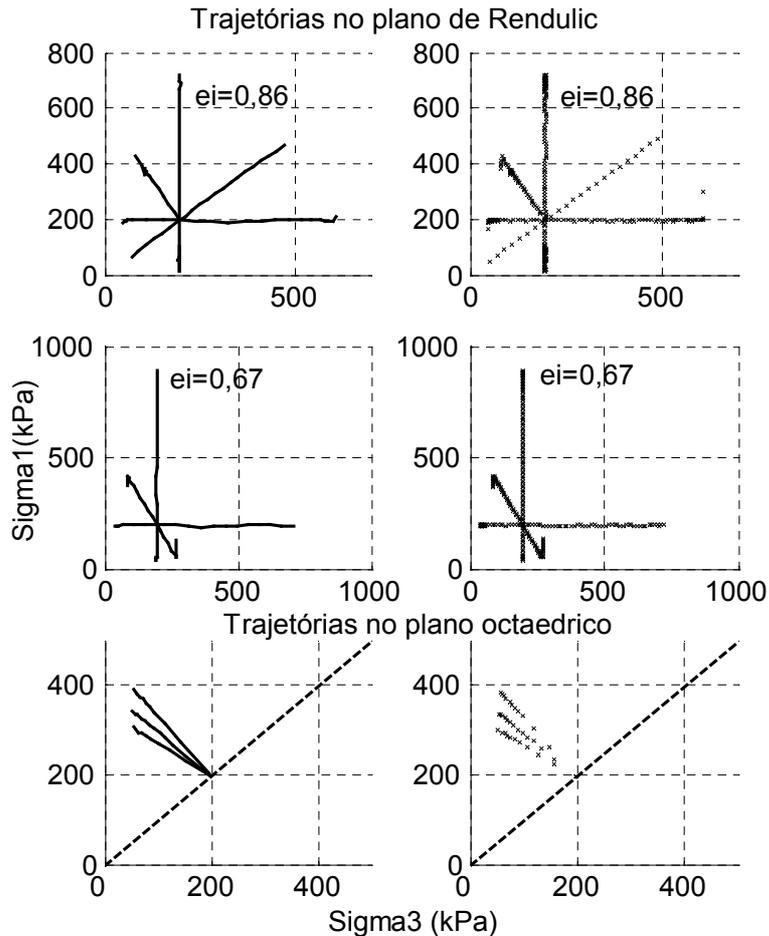


Figura 4.11. Resultados da simulação da rede *Magola 3* para a areia de Toyoura. Em tracejado o eixo hidrostático. À direita os dados experimentais, à esquerda a resposta da rede.

4.4.4 – DISCUSSÃO

A utilização de redes PMC para modelagem constitutiva de solos apresenta como maior dificuldade a necessidade de usar os valores absolutos das tensões nas entradas da rede, porque ao usar os acréscimos das tensões corre-se o risco de elas ficarem especializadas só nas regiões de entrada correspondentes à faixa de valores dos acréscimos usados, não sendo capazes, posteriormente, de fornecer uma generalização adequada. Desta forma, deve-se adotar entradas adicionais que viabilizem a unicidade entre o mapeamento de entradas e saídas. Neste modelo, as entradas adicionais foram o índice de vazios (e_i), a diferença entre o índice de vazios máximo e mínimo ($e_{\max}-e_{\min}$), o fator de forma das partículas (n_s) e a tensão de confinamento inicial (σ_0). No entanto, deve-se acrescentar que não foi o uso

exclusivamente do e_{\max} - e_{\min} o que fez possível um adequado desempenho de validação do modelo. A inclusão do índice de vazios em cada ponto experimental dos exemplos de treinamento permite complementar a informação incluída no e_{\max} - e_{\min} .

4.5 – MODELO NEURAL 2

Geralmente, as redes simuladas estaticamente, como foi o caso de *Magola 3*, apresentam respostas de validação muito precisas. Entretanto, as mesmas não podem ser usadas dessa maneira dentro de um ambiente computacional como operadores constitutivos. Uma maneira mais racional é o método recorrente, também chamado incremental ou autônomo, o qual foi usado no modelo que se descreve nesta seção.

Em relação ao processo de simulação da rede, cabe ressaltar que existem basicamente duas formas de simular um PMC dependendo da maneira como os exemplos de treinamento ou validação são apresentados: *estática* e *recorrente*. A simulação estática, também chamada “*ponto-a-ponto*” consiste em usar informação experimental sobre os estados de tensão-deformação do solo em todos e cada um dos pontos de simulação. A rede então produz uma resposta que posteriormente não é usada para calcular o estado tensão-deformação seguinte. No caso da simulação *recorrente*, também chamada *incremental* ou *autônoma*, a rede acumula a resposta tensão-deformação atual estimada para um certo ponto, e a utiliza para fabricar o vetor de entrada seguinte e assim estimar estados subseqüentes de tensão-deformação (ver Figura 4.12).

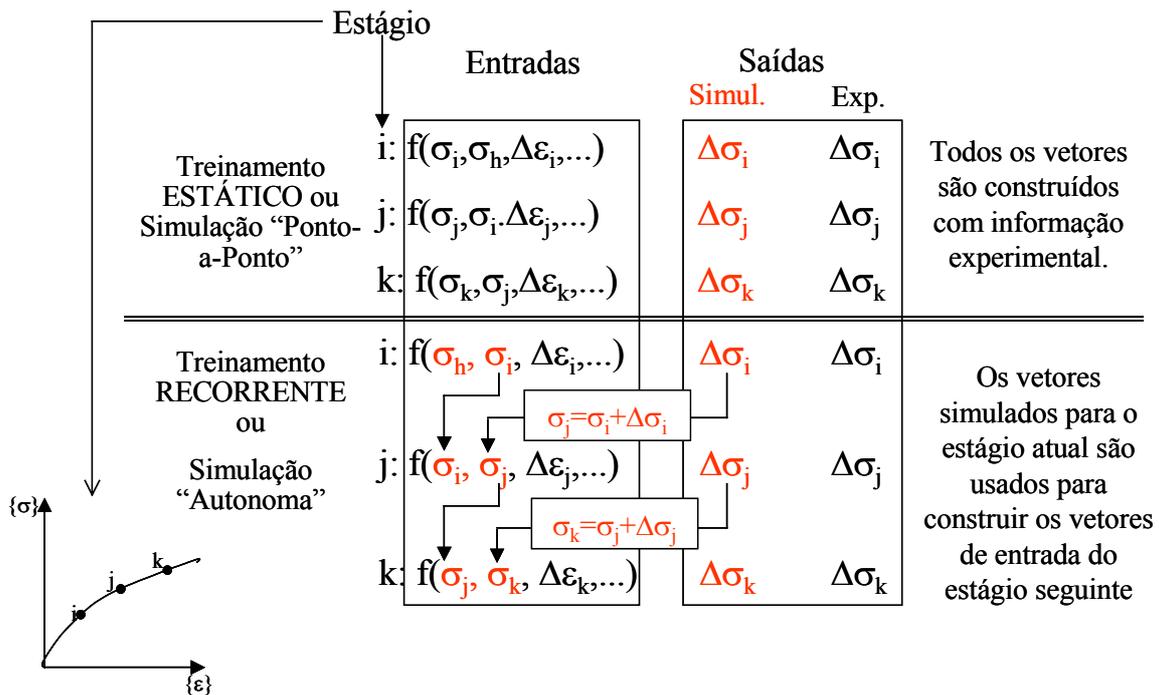


Figura 4.12. Representação esquemática dos processos de treinamento *estático* e *recorrente*, e de simulação “*Ponto-a-Ponto*” e “*Autônoma*”.

4.5.1 – A BASE DE DADOS

Em contraste com o modelo *Magola 3*, no qual a base de dados era formada por 1333 pontos, neste novo modelo, o número de pontos aumentou para 1465 devido ao fato de que alguns testes da areia de Cambria foram re-digitalizados com o intuito de diminuir o valor dos acréscimos de tensão de modo a melhorar o desempenho da rede. Isto se justifica pelo fato de que as simulações do modelo anterior, realizadas para a areia de Cambria, exibiram desempenhos pobres, devidos provavelmente à existência de uma grande diferença entre a magnitude dos acréscimos de tensão da parte inicial das trajetórias e a magnitude da parte final das mesmas, o que pode ser verificado pela observação direta dos pontos experimentais apresentados na Figura 4.9. Seguindo o mesmo critério do modelo anterior, a base de dados foi dividida em três subconjuntos da seguinte forma: 50% dos exemplos para treinamento e o restante em partes iguais para os conjuntos de teste durante treinamento e para validação final.

Adicionalmente, houve uma modificação na estrutura da base de dados modificando a forma de apresentação do primeiro ponto experimental de cada teste. Isto se explica na Figura

4.13 a qual corresponde a um conjunto de treinamento típico de um teste com 11 pontos experimentais.

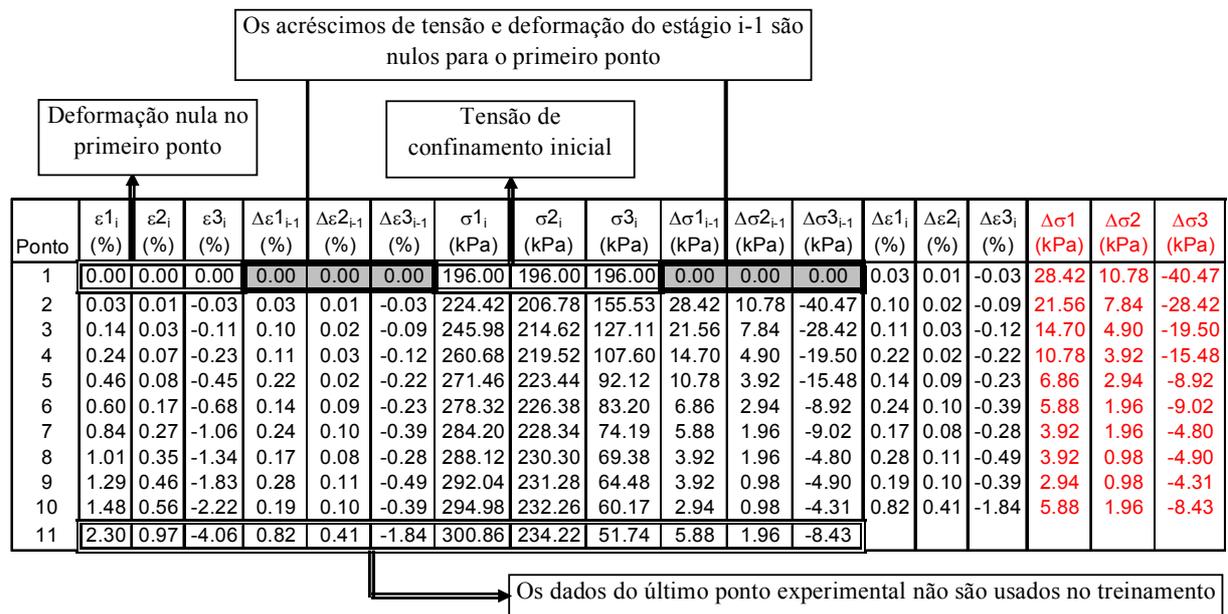


Figura 4.13. Estrutura geral de um conjunto de treinamento que inclui o primeiro ponto experimental. Em preto os dados de entrada e em vermelho os dados de saída.

As quatro entradas iniciais (n_s , σ_0 , e_{\max} - e_{\min} , e_0) não foram apresentadas por razões de espaço. Uma vez que o primeiro ponto carece de estado de tensão-deformação anterior, as primeiras quatro posições são inicializadas com valores nulos. No caso do modelo *Magola 3*, o conjunto de vetores de treinamento começava no segundo ponto, sendo que o primeiro era usado exclusivamente para definir o estado tensão-deformação das componentes com sub-índice i-1 correspondentes ao ponto 2. O fato de usar essa estrutura modificada, na qual o primeiro ponto participa ativamente do treinamento, permite uma adequada simulação autônoma já que, partindo da apresentação apenas do primeiro ponto experimental, a rede será capaz de fornecer valores para os acréscimos de tensão subsequentes reproduzindo assim a trajetória completa. Neste caso, durante o processo de treinamento o conjunto de vetores de entrada é continuamente criado pelo processo e continuamente modificado na medida que o erro vai diminuindo (Martinez & Farias, 2004).

4.5.2 – ARQUITETURA E TREINAMENTO

Foram criados e treinados diferentes modelos neurais variando sistematicamente as funções de transferência e algoritmos de aprendizado, assim como os parâmetros que controlam o desempenho das mesmas. A Tabela 4.1 apresenta um resumo das diferentes combinações consideradas, as quais geraram um total de 204 modelos neurais diferentes, todos eles com duas camadas intermediárias, 19 nós de entrada e 3 nós de saída. Conseqüentemente, o código numérico usado na primeira coluna da tabela refere-se unicamente ao número de neurônios das camadas ocultas. Por outro lado, da mesma maneira que para o modelo *Magola 3*, neste caso os nomes das funções e dos seus parâmetros são mantidos segundo a nomenclatura do Matlab®, na qual “*tansig*” faz referência à função tangente hiperbólica, e “*purelin*” faz referência à função limiar lógico.

Observa-se que a função de transferência sigmóide foi descartada porque ela sempre produz respostas positivas para o intervalo completo de entrada. Essa característica certamente afetaria o desempenho das redes pois muitas das trajetórias de tensão consideradas estão submetidas a incrementos negativos de tensão. Tal observação indica então que o intervalo ótimo de normalização dos dados é $[-1,1]$, razão pela qual foram usadas funções de transferência de domínio mais amplo como as que aparecem na Tabela 4.1. Na mesma tabela pode-se apreciar que foram cinco os algoritmos de treinamento ou aprendizado que foram testados, todos eles implementados no módulo de redes neurais do programa Matlab®: *Gradiente Descendente Simples* (traingd), *Taxa de Aprendizado Variável com Momentum* (traingdx), *Retropropagação Ressiliente* (trainrp), *Gradiente Conjugado Fletcher-Reeves* (traingcf) e *Gradiente Conjugado Modificado* (trainscg).

Tabela 4.1. Resumo das combinações usadas para otimizar a arquitetura do modelo neural.

Arquitetura	Função de Transferência	Algoritmo de Aprendizado	Parâmetros de Treinamento
30-10 / 30-20 / 30-30 60-20 / 60-40 / 60-60 90-30 / 90-45	tansig-tansig-purelin	Gradient Descent (traingd)	Taxa de aprendizado (LR)= 0.05, 0.2, 0.5
	tansig-tansig-tansig		
30-10 / 30-20 / 30-30 60-20 / 60-40 / 60-60 90-30 / 90-45 / 90-60 / 90-75 / 90-90	tansig-tansig-tansig	Variable Learning Rate (traingdx)	Taxa de aprendizado (LR)= 0.05, 0.2, 0.5 Momentum (M)=0.9, 0.6, 0.3 LR Incremento= 1.05 LR Decremento =0.7
30-20 40-20 / 40-30 / 40-40 50-30 / 50-40 / 50-50 60-20 / 60-40 / 60-60	tansig-tansig-tansig	Resilient BackPropagation (trainrp)	Incremento nos pesos =1.2 Decremento nos pesos =0.5
		Conjugate Gradient (traincgf)	(Fletcher-Reeves)
		Conjugate Gradient (trainscg)	(Scaled Conjugate Gradient)

O algoritmo do *Gradiente Descendente Simples* foi explicado em seções anteriores. No caso do algoritmo de *Taxa de Aprendizado Variável*, o desempenho do algoritmo do *Gradiente Simples* é melhorado permitindo que a taxa de aprendizado mude durante o processo de treinamento tal como se explica a seguir: Primeiro, as saídas iniciais da rede e o erro correspondente são calculados. Para cada iteração novos pesos são calculados usando a taxa de aprendizado atual, e novas saídas assim como um novo erro são também calculados. Se o erro novo excede o antigo numa quantidade maior do que um fator predefinido (tipicamente 1,04) então os novos pesos são descartados e a taxa de aprendizado multiplicada por um fator de diminuição predefinido (tipicamente 0,7). Caso contrário, os novos pesos são mantidos. Se o novo erro é menor do que o antigo, a taxa de aprendizado é multiplicada por um fator de aumento predefinido (tipicamente 1,05). Adicionalmente, o fator de momentum permite que a rede responda não só as variações do gradiente local, mas também às tendências recentes da superfície de erro, possibilitando que a rede ignore pequenas irregularidades da superfície evitando assim que fique presa em mínimos locais (Demuth & Beale, 2000).

Por outra parte, no algoritmo de *Retropropagação Resiliente*, unicamente o sinal da derivada é usado para determinar a direção de atualização dos pesos; a magnitude da derivada não tem efeito. Isto é justificado pelo fato de as funções do tipo sigmóide apresentarem derivadas que tendem a zero na medida que os valores de entrada aumentam, o que ocasiona problemas quando se usa o algoritmo do Gradiente Descendente, uma vez que gradientes

pequenos geram mudanças pequenas nos pesos não importando que esses pesos estejam ainda longe dos seus valores ótimos. A atualização dos valores dos pesos é feita então por meio de um fator de aumento que é aplicado sempre que a derivada da função de desempenho apresente o mesmo sinal em duas iterações sucessivas. Em contraste, os pesos são afetados por um fator de diminuição sempre que a derivada da função de desempenho mude de sinal em relação à iteração anterior.

Nos algoritmos de *Gradiente Conjugado* a busca da direção na qual serão ajustados os pesos baseia-se no critério das direções conjugadas. Na primeira iteração o algoritmo inicia a rede usando o algoritmo do gradiente descendente simples. Na iteração seguinte, o algoritmo determina a direção de atualização dos pesos conjugando a direção da iteração anterior com a nova direção de descida mais íngreme como apresentado na Eq. (4.5), na qual $\delta(n)$ é o gradiente atual, $P(n)$ e $P(n-1)$ são, respectivamente, as direções de busca atual e anterior, e $\beta(n)$ é a constante que define o tipo de combinação que vai ser realizada e que diferencia as diversas variações do algoritmo.

$$P(n) = -\delta(n) + \beta(n)P(n-1) \quad (4.5)$$

Em relação ao padrão de treinamento, a diferença principal entre este modelo e o anterior é a inclusão do vetor $\{\sigma_i\}$, o qual aumentou o número de entradas de 16 para 19 tal como se apresenta na Eq. (4.6). Perante a necessidade de suprimir a utilização do índice de vazios em cada ponto experimental de treinamento (e_i) e substituí-lo pelo índice de vazios inicial, e_0 , achou-se importante aumentar a quantidade de informação relacionada com o estado de tensão da areia e por conseguinte decidiu-se incluir o $\{\sigma_i\}$. Adicionalmente, o estado anterior de tensão-deformação foi apresentado de forma incremental, representado na Eq. (4.6) pelos vetores $\{\Delta\varepsilon_{i-1}\}$ e $\{\Delta\sigma_{i-1}\}$, de forma a tornar as entradas compatíveis com o processo de simulação autônoma.

$$\{\Delta\sigma_i\} = f(n_s, \sigma_0, e_{\max} - e_{\min}, e_0, \{\varepsilon_i\}, \{\Delta\varepsilon_{i-1}\}, \{\sigma_i\}, \{\Delta\sigma_{i-1}\}, \{\Delta\varepsilon_i\}) \quad (4.6)$$

Durante os processos de treinamento realizados para determinar as melhores funções de transferência e algoritmos de treinamento, todas as redes testadas foram apresentadas no modo “por lote” e validadas segundo a técnica do “ponto-a-ponto”. Ao finalizar o processo de treinamento e verificação foi visto que o melhor desempenho era obtido para redes com 60 ou 90 neurônios nas duas camadas ocultas, com função de transferência do tipo “*tansig*” e com o algoritmo de treinamento do tipo “*trainseg*” (Gradiente Conjugado Modificado). Já nas etapas

seguintes do processo também foram feitas simulações autônomas ou recorrentes. O fato de usar essa estrutura modificada, na qual o primeiro ponto participa ativamente do treinamento, permite uma adequada simulação autônoma pois partindo da apresentação só do primeiro ponto experimental, a rede fornece valores para os acréscimos de tensão que conduzem ao segundo ponto e, conseqüentemente, reproduz cada um dos pontos seguintes até obter a trajetória completa. Neste caso, o conjunto de vetores de entrada é continuamente criado pelo próprio processo.

4.5.3 – RESULTADOS DA SIMULAÇÃO

Um novo modelo neural (*Magola 6*) com duas camadas ocultas do tipo 90-90 com função de transferência do tipo “*tansig*” foi treinado recorrentemente durante 10000 iterações com o algoritmo “*trainscg*”. Finalizado o treinamento, o modelo foi simulado no modo autônomo para todos os testes. Inicialmente os resultados são apresentados mediante a comparação das respostas da rede com as respostas desejadas correspondentes aos valores experimentais. Tal comparação é apresentada na Figura 4.14 tanto para os acréscimos de tensão principal (coluna esquerda) como para os valores absolutos das tensões (coluna direita).

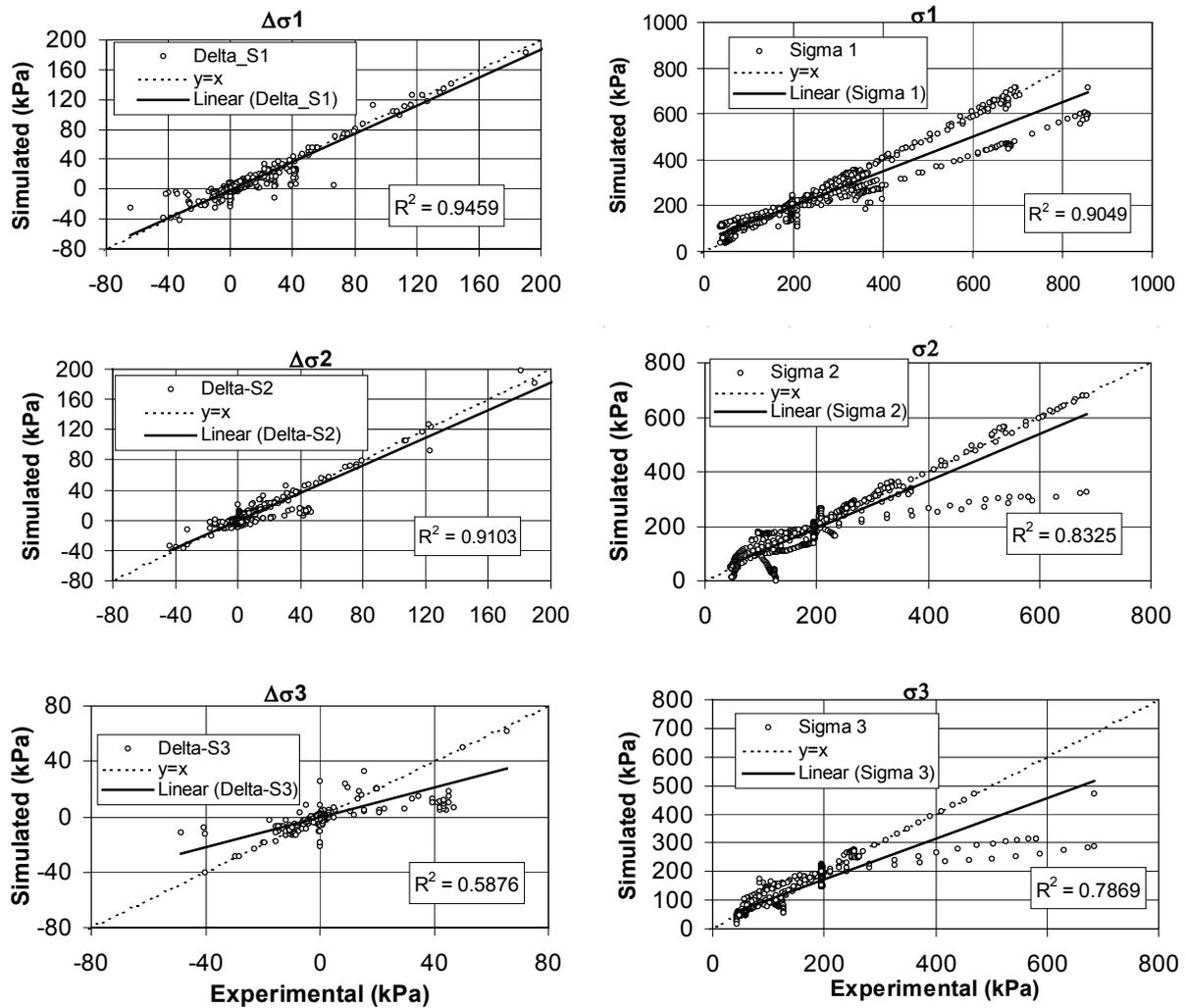


Figura 4.14. Comparação dos resultados para a rede *Magola 6*.

Em cada um dos gráficos apresentados foi acrescentada a linha $y=x$ que corresponde à simulação perfeita. Quanto mais próximo desta linha ficarem os pontos, melhor é a aproximação calculada pelo modelo. Para o caso das componentes 1 e 2, tal aproximação pode ser considerada boa, já para o caso da componentes 3 é destacável a presença de alguns ensaios com dispersão alta. No entanto, de uma maneira global, a aproximação da rede pode ser considerada aceitável.

Uma outra forma de visualizar os resultados é mediante a apresentação das curvas tensão-deformação de cada um dos testes simulados, tal como apresentado na Figura 4.15. Nesta figura foi usada a relação τ_{oct}/p para representar as tensões, pois o efeito combinado das três componentes de tensão está incluído nesta relação, tal como pode ser visto nas Eqs. (4.7) e (4.8). Nos gráficos apresentados na Figura 4.15 as ordenadas correspondem à relação τ_{oct}/p e as abscissas às deformações. Cada gráfico está composto de três curvas, uma para cada

componente de deformação principal. Em todos os gráficos os pontos representam os dados experimentais e as linhas contínuas os resultados das simulações da rede. Para o teste 1, correspondente a um ensaio de compressão hidrostática na areia de Toyoura, as ordenadas correspondem à tensão média, p , e as abscissas à deformação volumétrica. Tanto o tipo de trajetória como dados relacionados com o tipo de areia de cada teste podem ser acompanhados da Tabela A 1 no Apêndice 1.

$$p = \frac{1}{3}(\sigma_1 + \sigma_2 + \sigma_3) \quad (4.7)$$

$$\tau_{\text{oct}} = \frac{1}{3}\sqrt{(\sigma_1 - \sigma_2)^2 + (\sigma_2 - \sigma_3)^2 + (\sigma_1 - \sigma_3)^2} \quad (4.8)$$

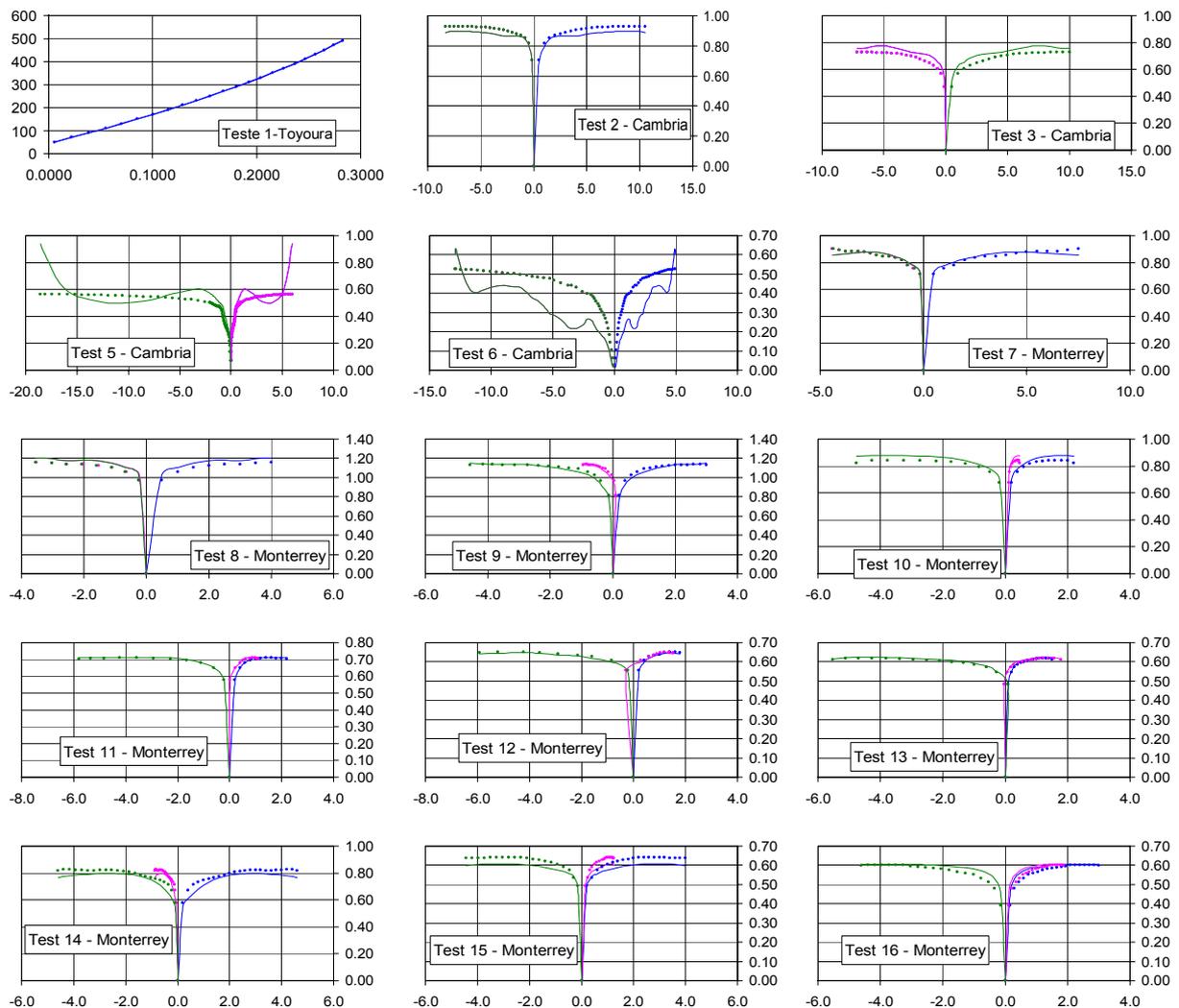


Figura 4.15. Resultados da simulação incremental da rede *Magola 6*.

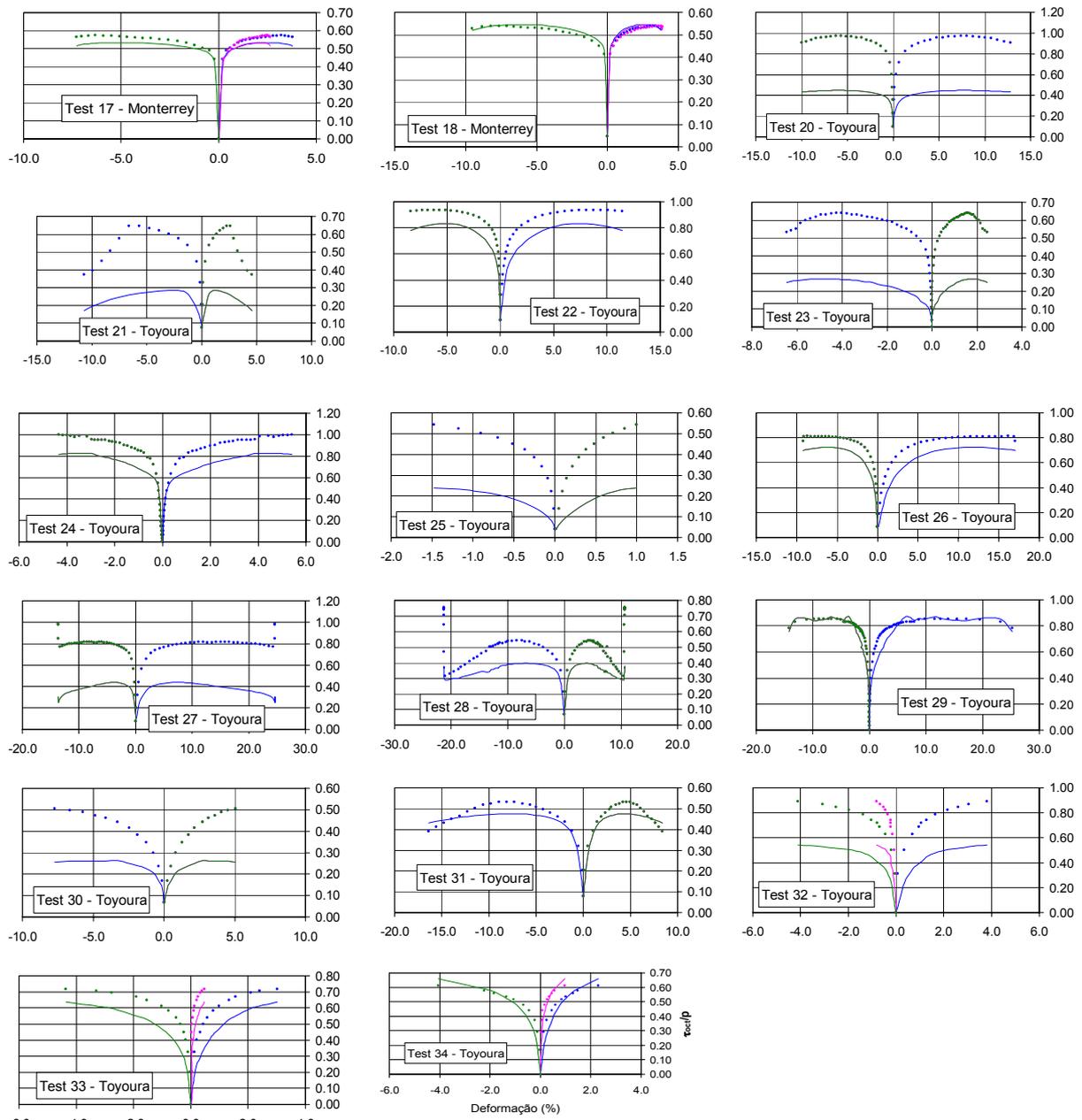


Figura 4.15. (Continuação) Resultados da simulação incremental da rede *Magola*.

4.5.4 – DISCUSSÃO

Observa-se como os resultados de simulação da rede *Magola 6* para os testes da areia de Monterrey (teste 8 a 18) foram muito bem aproximados às trajetórias experimentais; em contraste os resultados das duas areias restantes apresentaram erros de simulação consideráveis. Uma observação detalhada das curvas tensão-deformação dos testes com piores resultados de simulação permite-nos reconhecer dois tipos de erro: erro de forma da curva

(testes 5 e 6) e erro de posicionamento da curva (testes 20, 21, 23, 24, 25, 27, 30, 32 e 33). O primeiro refere-se a trajetórias com comportamento oscilante geradas provavelmente por algum tipo de sobre-treinamento em testes com grande quantidade de exemplos de treinamento como são o teste 5 e o teste 6 que juntos armazenam 23,9% dos exemplos de treinamento disponíveis. O segundo, relacionado com trajetórias mal posicionadas ou transladadas em relação à real, porém com forma adequada, parece resultar de uma má distribuição dos dados dos conjuntos de generalização. Nestes testes, foi detectado que dentro do conjunto de validação existem variáveis cujos intervalos de valores são diferentes dos seus equivalentes no conjunto de treinamento (Martinez-Carvajal & Farias, 2004). Isto pode ser apreciado nos histogramas de frequência das deformações e dos acréscimos de deformação do teste 21 (Figura 4.16).

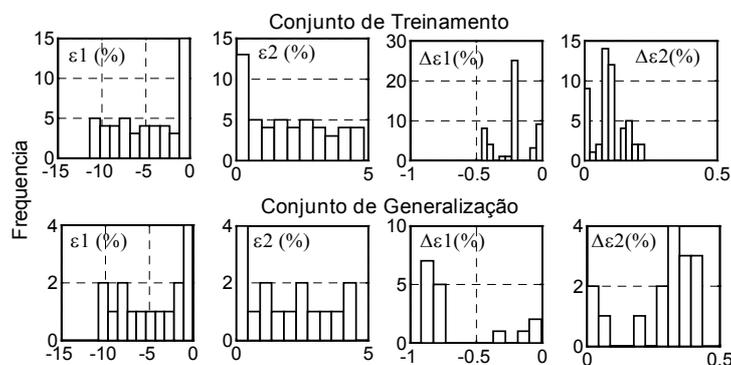


Figura 4.16. Histogramas de frequência das deformações e dos acréscimos de deformação do teste 21 para os conjuntos de treinamento e generalização.

4.6 – MODELO NEURAL 3

Visando corrigir os erros descritos na apresentação do modelo anterior e mantendo ao mesmo tempo a necessidade de melhorar os resultados de simulação autônoma da rede, foram introduzidas algumas mudanças na apresentação dos dados assim como no padrão de treinamento. Também decidiu-se trabalhar com a totalidade dos pontos de treinamento disponíveis por considerar que as potencialidades da base de dados piloto haviam sido completamente esgotadas do ponto de vista dos objetivos desta tese.

4.6.1 – A BASE DE DADOS

Para desenvolver um modelo mais robusto foram aumentados o número de exemplos de treinamento e a variedade de materiais considerados. Em consequência, a totalidade dos dados disponíveis (171 testes) foi considerada no treinamento deste modelo. Para cada um dos testes digitalizados, com exceção dos pertencentes às areias de Toyoura, Cambria e Monterrey que já haviam sido processados nas etapas anteriores deste trabalho, foi obtida, com ajuda do módulo de ajuste polinomial do Matlab, a melhor equação de ajuste com a qual foram recalculados os valores dos acréscimos de tensão correspondentes a acréscimos de deformação impostos dentro do intervalo [-3.3%, 6%]. Neste processo procurou-se, sempre que foi possível, adotar valores para os acréscimos de deformação, menores que 0.5% principalmente na região das pequenas deformações de cada teste. Já na parte final de algumas das curvas experimentais foi necessário impor valores de acréscimos de deformação maiores, para evitar a ocorrência de acréscimos de deformação exageradamente pequenos. A distribuição final dos valores dos acréscimos de deformação é apresentada no histograma da Figura 4.17, no qual pode se apreciar como efetivamente, a maior concentração de valores está dentro da faixa menor que 0.5% em valor absoluto.

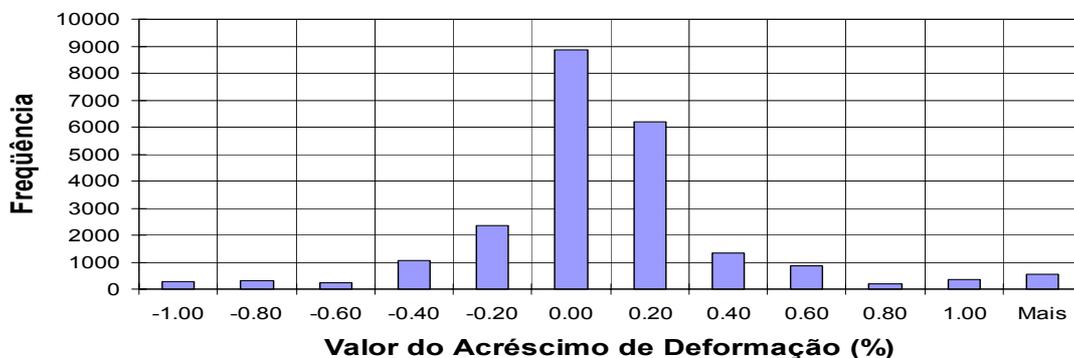


Figura 4.17. Histograma de frequência dos acréscimos de deformação para a base de dados completa (171 testes).

O conjunto de dados foi dividido para compor um conjunto de treinamento com 134 testes e 32212 exemplos, e um conjunto de validação com 37 testes e 9960 exemplos. Para cada uma das entradas do vetor de treinamento foram selecionados os testes que continham os valores máximo e mínimo das mesmas, e incorporados dentro do conjunto de treinamento. Isto foi feito com o propósito de impor que o conjunto de treinamento contivesse obrigatoriamente os valores extremos de cada uma das entradas, garantindo assim que o

processo de simulação posterior fosse realizado com a rede trabalhando como interpolador. Desta forma foram selecionados 54 testes enquanto que os 80 restantes foram escolhidos aleatoriamente até quando a proporção de pontos de treinamento foi de aproximadamente 75% do total de pontos disponíveis que era de 42172.

4.6.2 – ARQUITETURA E TREINAMENTO

O padrão de treinamento considerado neste modelo foi modificado com relação ao do modelo anterior e pode ser representado conforme aparece na Eq. (4.9). Observe-se que, dado um estágio de carregamento atual (i), o estado tensão-deformação do estágio anterior ($i-1$) é definido não por meio dos acréscimos correspondentes, mas sim usando os valores absolutos. Adicionalmente, cabe ressaltar que neste caso, além do primeiro ponto experimental das trajetórias de treinamento, também foi considerado o último ponto das mesmas, mediante a inclusão de uma fila de zeros nas colunas correspondentes aos acréscimos de deformação o que necessariamente conduz à inclusão de valores nulos para a última fila das colunas dos acréscimos de tensão (ver Figura 4.18).

As tensões e deformações do estágio $i-1$ são nulos para o primeiro ponto

Ponto	Deformação nula no primeiro ponto			Tensão de confinamento inicial									Acréscimos de deformação nulos são associados a acréscimos de tensão nulos					
	$\varepsilon_{1i}(\%)$	$\varepsilon_{2i}(\%)$	$\varepsilon_{3i}(\%)$	$\varepsilon_{1_{i-1}}(\%)$	$\varepsilon_{2_{i-1}}(\%)$	$\varepsilon_{3_{i-1}}(\%)$	$\Delta\varepsilon_{1i}(\%)$	$\Delta\varepsilon_{2i}(\%)$	$\Delta\varepsilon_{3i}(\%)$	$\sigma_{1i}(\text{kPa})$	$\sigma_{2i}(\text{kPa})$	$\sigma_{3i}(\text{kPa})$	$\sigma_{1_{i-1}}(\text{kPa})$	$\sigma_{2_{i-1}}(\text{kPa})$	$\sigma_{3_{i-1}}(\text{kPa})$	$\Delta\sigma_1(\text{kPa})$	$\Delta\sigma_2(\text{kPa})$	$\Delta\sigma_3(\text{kPa})$
1	0.00	0.00	0.00	0.00	0.00	0.00	0.50	-0.79	-0.79	98.10	98.10	98.10	0.00	0.00	0.00	-31.57	15.79	15.79
2	0.50	-0.79	-0.79	0.00	0.00	0.00	0.50	-1.01	-1.01	66.53	113.89	113.89	98.10	98.10	98.10	-11.42	5.71	5.71
3	1.00	-1.80	-1.80	0.50	-0.79	-0.79	0.50	-1.19	-1.19	55.10	119.60	119.60	66.53	113.89	113.89	-5.51	2.75	2.75
4	1.50	-2.99	-2.99	1.00	-1.80	-1.80	0.50	-1.33	-1.33	49.60	122.35	122.35	55.10	119.60	119.60	-3.28	1.64	1.64
5	2.00	-4.32	-4.32	1.50	-2.99	-2.99	0.50	-1.44	-1.44	46.32	123.99	123.99	49.60	122.35	122.35	-2.30	1.15	1.15
6	2.50	-5.76	-5.76	2.00	-4.32	-4.32	0.50	-1.51	-1.51	44.02	125.14	125.14	46.32	123.99	123.99	-1.79	0.89	0.89
7	3.00	-7.27	-7.27	2.50	-5.76	-5.76	0.50	-1.54	-1.54	42.23	126.04	126.04	44.02	125.14	125.14	-1.43	0.71	0.71
8	3.50	-8.81	-8.81	3.00	-7.27	-7.27	0.50	-1.53	-1.53	40.80	126.75	126.75	42.23	126.04	126.04	-1.12	0.56	0.56
9	4.00	-10.34	-10.34	3.50	-8.81	-8.81	0.50	-1.49	-1.49	39.69	127.31	127.31	40.80	126.75	126.75	-0.85	0.42	0.42
10	4.50	-11.83	-11.83	4.00	-10.34	-10.34	0.50	-1.40	-1.40	38.84	127.73	127.73	39.69	127.31	127.31	-0.63	0.32	0.32
11	5.00	-13.23	-13.23	4.50	-11.83	-11.83	0.00	0.00	0.00	38.21	128.05	128.05	38.84	127.73	127.73	0.00	0.00	0.00

Figura 4.18. Estrutura geral de um conjunto de treinamento que inclui tanto o primeiro como o último ponto experimental. Em preto os dados de entrada e em vermelho os dados de saída.

Com esta modificação conseguiu-se incluir informação importante que não tinha sido considerada, que é precisamente o fato de relacionar acréscimos de tensão nulos com acréscimos de deformação também nulos.

$$\{\Delta\sigma_i\} = f(n_s, \sigma_0, e_{\max} - e_{\min}, e_0, \{\varepsilon_i\}, \{\varepsilon_{i-1}\}, \{\Delta\varepsilon_i\}, \{\sigma_i\}, \{\sigma_{i-1}\}) \quad (4.9)$$

4.6.2.1 – ESTIMATIVA DO TEMPO DE TREINAMENTO

Levando em conta que a base de dados era consideravelmente volumosa, decidiu-se testar inicialmente uma arquitetura grande para estimar os tempos de treinamento prováveis de tal forma que fossem razoáveis em relação ao recurso computacional disponível (processador AMD XP1800). Para isso, foi escolhida uma rede com duas camadas intermediárias; a primeira com 120 neurônios e a segunda com 90 neurônios, totalmente interconectados entre si com todas as funções de transferência do tipo tangente hiperbólica, algoritmo de treinamento do tipo gradiente conjugado escalado (*trainscg*), 19 entradas e 3 saídas. A rede foi treinada de maneira recorrente durante uma iteração só, apresentando os primeiros 107 pontos experimentais, e o tempo de treinamento foi medido. Posteriormente o processo foi repetido com 1058 pontos experimentais e assim sucessivamente aumentando cada vez a quantidade de exemplos até alcançar os 32212 pontos de treinamento. No gráfico da Figura 4.19 pode-se observar a evolução do tempo de treinamento para os diferentes pacotes de informação apresentados à rede.

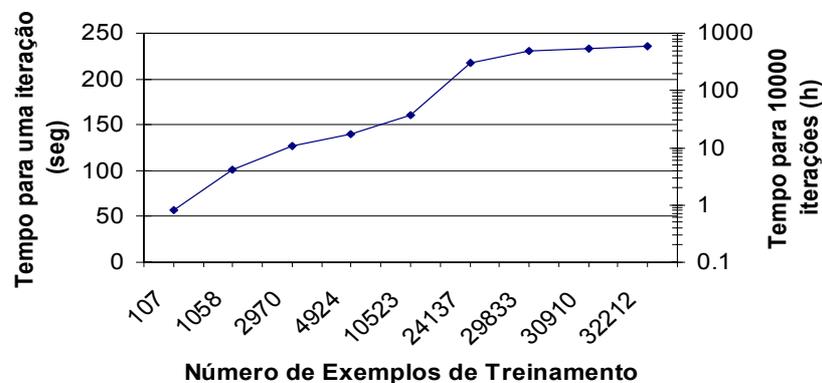


Figura 4.19. Evolução do tempo de treinamento com o número de exemplos para uma rede 19x120x90x3 treinada durante uma iteração com o algoritmo de gradiente conjugado modificado.

Segundo a experiência acumulada nas etapas anteriores, previa-se que os novos modelos neurais poderiam necessitar de aproximadamente 10000 épocas de treinamento contínuo para começar a apresentar evidências de aprendizado, razão pela qual foram estimados os tempos que a rede gastaria para treinar durante 10000 iterações a mesma seqüência de pacotes de

informação apresentada na Figura 4.19. Observou-se então que a utilização da totalidade dos pontos de treinamento consumiria tempos enormes de processamento, tornando-a inviável do ponto de vista prático. Aceitando como razoável um tempo de treinamento da ordem de 12 a 15 horas (uma noite), foi reduzido o tamanho do conjunto de dados para 6653 pontos dos quais 5242 são de treinamento e os 1411 restantes são para validação. A eliminação dos pontos foi feita por escolha aleatória para cada teste conservando entre 15% e 18% em cada caso e eliminando o restante.

4.6.2.2 – ESCOLHA DA ARQUITETURA DA REDE

Foram testadas 6 arquiteturas diferentes com os seguintes números de neurônios das duas camadas ocultas: 150x150, 120x90, 90x90, 90x60, 60x60 e 60x30. Com cada uma das arquiteturas foram criadas quatro redes diferentes com pesos iniciais escolhidos aleatoriamente. O melhor modelo de cada caso foi guardado, os restantes foram eliminados. A seqüência de treinamento usada consistiu nas etapas que se apresentam a seguir:

- a) Treina-se a rede usando o modo “por lote” e o critério de parada antecipada com um número máximo de 1000 iterações.
- b) Realiza-se uma simulação “autônoma” da rede usando o conjunto de validação, e outra simulação, também “autônoma” usando o próprio conjunto de teste.
- c) Calcula-se o erro de simulação de todos os testes, tanto de validação como de treinamento, mediante comparação direta dos valores obtidos com os valores reais. Estima-se também o erro médio de validação e o erro médio de treinamento.
- d) Separam-se os testes que apresentam erros de treinamento maiores do que o erro médio, e cria-se com eles uma base de dados temporal usada para treinar a rede no modo recorrente.
- e) Esta nova fase de treinamento se prolonga até quando o erro de validação recorrente dos testes que compõem a nova base de dados é igual ou menor que o erro médio de treinamento calculado no passo (c).
- f) Repete-se o processo a partir do passo (c) e os erros de treinamento e de validação vão sendo plotados para observar a evolução do aprendizado (ver Figura 4.20. Nota: a curva do erro de validação da rede 120x90 se perdeu acidentalmente).

Usando esta seqüência de passos, chamada de *treinamento seletivo*, o aprendizado da rede vai se dando de tal forma que ele sempre é mais intenso para aqueles testes que oferecem

maior dificuldade para serem aprendidos, enquanto que os testes que são mais facilmente aprendidos não são apresentados de maneira excessiva, minimizando com isso a possibilidade de aparecer o fenômeno de sobre-treinamento. Mesmo assim, esta técnica apresenta a grande desvantagem de não ser possível, em muitos casos, identificar o momento certo de parada do processo, pois o fato de a evolução dos erros se apresentar de maneira oscilatória, tal como se pode apreciar na Figura 4,20, faz com que sempre se tenha dúvida se um determinado mínimo na curva representa o momento ótimo de parada. Isto implica em armazenar os parâmetros da rede cada vez que aparece um ponto de mínimo na curva de validação, para posteriormente selecionar os correspondentes ao melhor modelo e descartar os restantes.

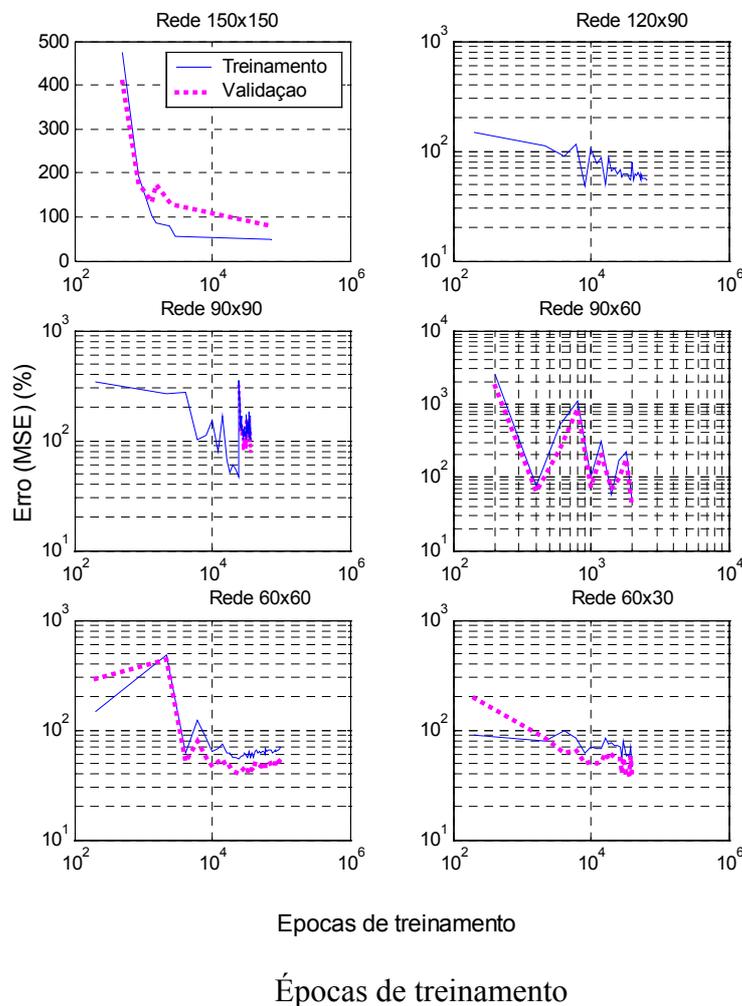


Figura 4.20. Evolução dos erros de treinamento e de validação de seis redes treinadas com a técnica de treinamento seletivo.

A partir da observação direta do comportamento dos diferentes modelos, resolveu-se escolher o modelo 90x60 que apresentou uma convergência relativamente rápida para um valor de erro de validação mínimo (38%), e testar com ele algumas outras estratégias diferentes de treinamento para otimizar seu desempenho. As estratégias referidas são, na realidade, variantes da técnica de treinamento seletivo já apresentada, nas quais a etapa (e) é realizada um número fixo de iterações definido pelo usuário, alternando com etapas curtas de treinamento “*por lote*” nas quais a totalidade dos pontos é apresentada à rede com o propósito de mantê-la informada sobre o problema global.

Foram testadas 7 estratégias diferentes as quais se descrevem a seguir, e seus resultados se apresentam na Figura 4.21.

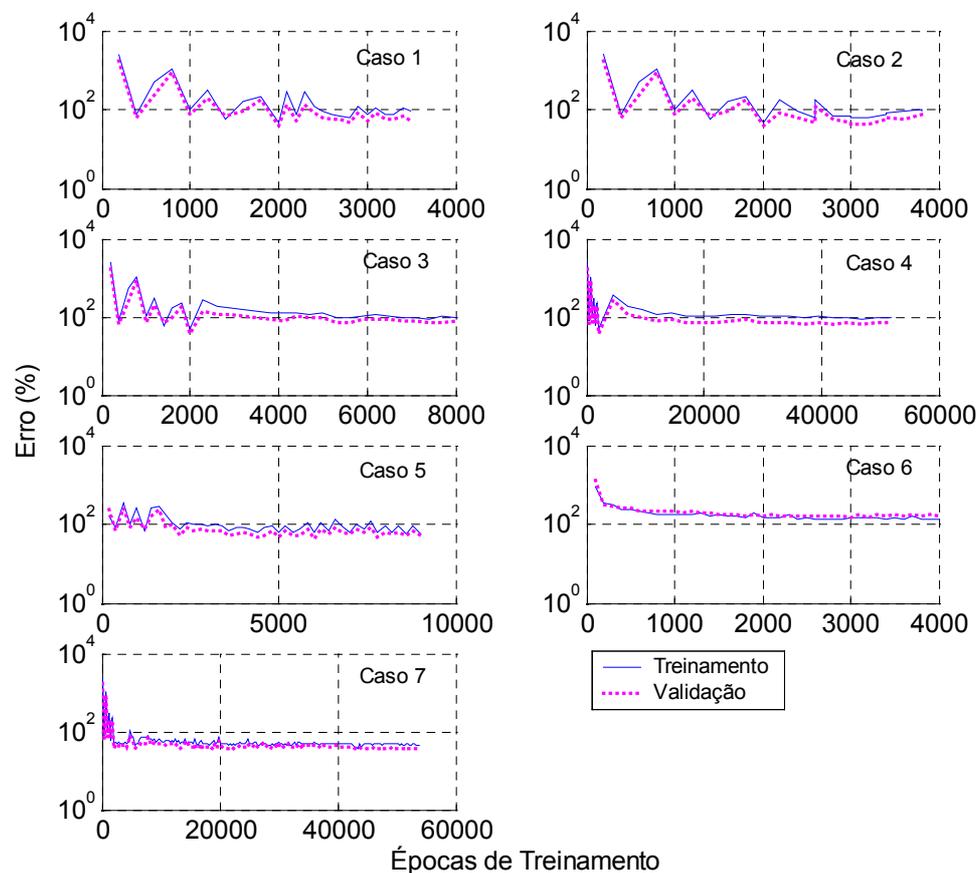


Figura 4.21. Gráficos de evolução do treinamento de uma rede 90x60 treinada seletivamente com sete estratégias diferentes.

De acordo com os resultados apresentados na Figura 4.21, observa-se que:

- Caso 1. Seqüências de 100 iterações de treinamento seletivo até o erro apresentar tendência assintótica horizontal. Para o caso analisado o número total de iterações foi de 3500, o erro mínimo de validação foi 38%. É importante esclarecer que o erro que aparece nos gráficos da figura em questão corresponde àquele que é medido durante o processo de treinamento e validação da rede dentro do programa Matlab, ou seja o erro calculado a partir dos valores dos vetores de treinamento e validação transformados e normalizados. Por outro lado, o erro de validação a que se refere a análise dos diferentes casos ora apresentados, corresponde àquele medido a partir da comparação real entre tensões geradas pela rede e as tensões reais de cada teste.
- Caso 2. Seqüências de 200 iterações de treinamento seletivo até o erro apresentar tendência assintótica horizontal. Para o caso analisado o número total de iterações foi de 3800, o erro mínimo de validação foi 38%.
- Caso 3. Seqüências de 100 iterações de treinamento “por lote” usando o conjunto total de dados, alternadas com seqüências de 100 iterações de treinamento seletivo até o erro apresentar tendência assintótica horizontal. Para o caso analisado o número total de iterações foi de 8000, o erro mínimo de validação foi 38%.
- Caso 4. Seqüências de 200 iterações de treinamento seletivo, alternadas com seqüências de 50 iterações de treinamento “por lote” usando o conjunto total de dados até o erro apresentar tendência assintótica horizontal. Para o caso analisado o número total de iterações foi de 52000, o erro mínimo de validação foi 38%.
- Caso 5. Reiniciando a rede levou-se a cabo a mesma seqüência descrita no Caso 2 para verificar a semelhança no comportamento. O número total de iterações foi 9000 enquanto que o erro mínimo de validação foi 44%.
- Caso 6. Reiniciando a rede levou-se a cabo a mesma seqüência descrita no Caso 1 para verificar a semelhança no comportamento. O número total de iterações foi 4000 enquanto que o erro mínimo de validação foi 154%.
- Caso 7. Seqüências de 50 iterações de treinamento seletivo até o erro apresentar tendência assintótica horizontal. Para o caso analisado o número total de iterações foi de 53750, o erro mínimo de validação foi 33%.

4.6.2.3 – INDEPENDÊNCIA DAS SAÍDAS

Devido ao fraco desempenho de simulação obtido com os modelos testados na seção imediatamente anterior, considerou-se adequado explorar a possibilidade de testar três modelos adicionais, cada um dos quais seria treinado para aprender uma das componentes do vetor de acréscimos de tensão, a partir do mesmo conjunto de treinamento.

Para isso foram consideradas quatro arquiteturas para as camadas ocultas, a saber: 30x15, 30x30, 40x20 e 40x30. Todas elas completamente conectadas, treinadas com o algoritmo de gradiente conjugado modificado (*trainscg*), inicialmente para a primeira componente do vetor de acréscimos de tensões ($\Delta\sigma_1$). Todas elas com funções de transferência do tipo tangente hiperbólico (*tansig*) em todas as camadas, exceto a rede 40x20 que possui função de transferência linear (*purelin*) na ligação entre a segunda camada intermediária e a de saída.

A rede 40x20 foi treinada com seqüências de 500 iterações usando o conjunto completo de dados no modo “por lote” alternadas com seqüências de 200 iterações de treinamento seletivo. As restantes foram treinadas com seqüências de 100 iterações usando o conjunto completo de dados no modo “por lote” alternadas com seqüências de 300 iterações de treinamento seletivo. Os resultados tanto para o erro de treinamento como de simulação são apresentados na Figura 4.22. Nela pode se observar que nenhum dos modelos testados ofereceu desempenhos que pudessem ser considerados satisfatórios, uma vez que a magnitude dos erros de validação, mais importantes que os próprios erros de treinamento, é ainda muito grande para que o modelo possa ser considerado adequado para funcionar como operador constitutivo.

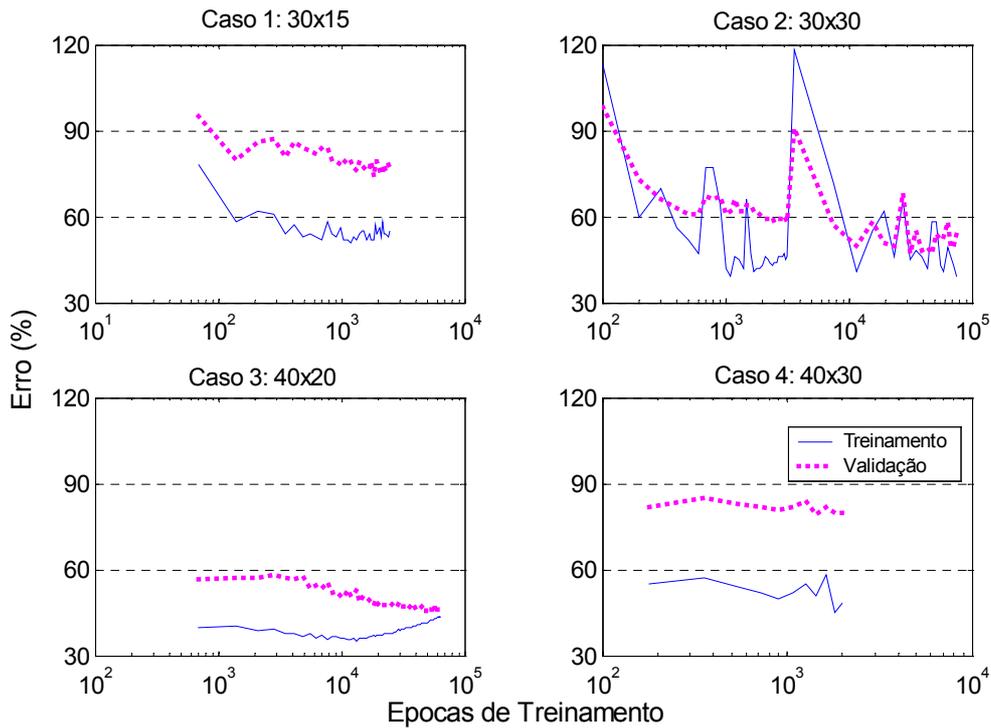


Figura 4.22. Erros de treinamento e validação de quatro redes treinadas seletivamente para a primeira componente do vetor de acréscimos de tensão ($\Delta\sigma_1$).

4.7 – MODELO NEURAL 4

Os resultados obtidos até o momento, provenientes de testes realizados com múltiplas estratégias de treinamento e uma grande variedade de arquiteturas para o modelo 3, não são satisfatórios, para nenhuma das situações analisadas. Sendo que modelos anteriores (modelos 1 e 2) foram capazes de resolver o problema apresentado, e levando em conta que a única diferença entre esses modelos e o modelo da seção anterior (modelo 3), é o fato de ter aumentado a quantidade de materiais envolvidos; é razoável pensar que a causa principal do fraco desempenho do modelo 3 fundamenta-se num problema de má configuração do espaço de treinamento em relação ao tipo de materiais usados ou às suas condições físicas.

Na seção seguinte é apresentado o processo por meio do qual foram classificados os materiais envolvidos no treinamento das RNA's subsequentes, a partir das suas propriedades físicas, usando redes neurais auto-organizáveis do tipo competitivo.

4.7.1 – PRÉ-CLASSIFICAÇÃO DOS TESTES DE TREINAMENTO

Observando a grande diversidade de materiais envolvidos e a variedade de condições de carregamento que está sendo considerada, o problema pode ser qualificado como de alta complexidade. Na pretensão que um modelo neural possa aprender de maneira generalizada o comportamento tensão-deformação das areias consideradas, sob qualquer condição de carregamento, estado inicial, independente da magnitude dos acréscimos de deformação impostos e, sobretudo, levando em conta efeitos não-lineares, é razoável pensar agora que pré-classificar os dados antes da sua apresentação ao modelo neural, seja uma alternativa que facilite o processo de aprendizado. Neste caso, seriam necessários tantos modelos neurais quanto as classes resultantes, razão pela qual uma classificação muito minuciosa acarretaria um número excessivo de classes, enquanto que uma classificação demasiado grosseira não traria ganho nenhum para o modelo a treinar.

Fica claro então que o ponto chave é a definição do critério de classificação que deve ser usado, uma vez que, devido à alta dimensionalidade do espaço que está sendo tratado, não é óbvia a definição das variáveis a considerar, as quais são basicamente de dois tipos: o primeiro tem a ver com aquelas variáveis que servem para definir o tipo de solo e a sua condição física inicial, enquanto o segundo tipo está relacionado com a história de carregamento ao qual o solo é submetido. Dentro do primeiro grupo estão sendo consideradas neste trabalho o fator de forma (n_s), a tensão de confinamento inicial (σ_{c0}), o intervalo do índice de vazios ($e_{\max}-e_{\min}$) e o índice de vazios inicial (e_0). Já no segundo grupo, tal como pôde ser visto nas seções anteriores, se encontram os vetores que definem os diferentes estados de tensão-deformação do solo, os quais estão sendo considerados em um esquema que pode ser chamado de dois pontos, pois para cada exemplo de treinamento apresentado à rede é incluído tanto o ponto do estágio atual de carregamento como o ponto do estágio imediatamente anterior.

Caso se adotasse o critério de classificação a partir do segundo grupo de variáveis, ou seja, a partir da análise das diferentes trajetórias consideradas, o modelo neural final aprenderia o comportamento tensão-deformação das areias para um grupo limitado de trajetórias de tensão. Em contraste, se o critério de pré-classificação estiver baseado nas variáveis do primeiro grupo, o modelo neural final aprenderia o comportamento de um conjunto limitado de areias, mas não seria limitado quanto às trajetórias consideradas. Sendo que esta última opção parece mais próxima ao objetivo da tese, decidiu-se pré-classificar a

base de dados a partir da informação fornecida pelo n_s , o σ_{c0} , o $e_{\max}-e_{\min}$ e o e_0 , com ajuda de uma rede do tipo auto-organizável ou competitiva.

4.7.2 – PRÉ-PROCESSAMENTO DA BASE DE DADOS

Foram criadas três redes competitivas do tipo Kohonen, e foram treinadas para classificarem todos os testes considerados na base de dados. A primeira, chamada *koho1* possui 4 neurônios competitivos na camada de saída; a segunda, *koho2*, tem 5 neurônios e finalmente a terceira, *koho3* tem 7 neurônios.

O número de neurônios da rede determina a quantidade de classes que vão ser consideradas. As classificações feitas por cada uma das redes foram comparadas para avaliar a consistência das mesmas. Tal comparação foi realizada mediante o cálculo da frequência de cada uma das classes expressa em porcentagem, tal como se pode apreciar no histograma da Figura 4.23. Nele pode-se observar que existe uma marcada preponderância da classe 1 sobre as demais, e que a porcentagem de testes dessa classe permanece aproximadamente constante para as três redes consideradas.

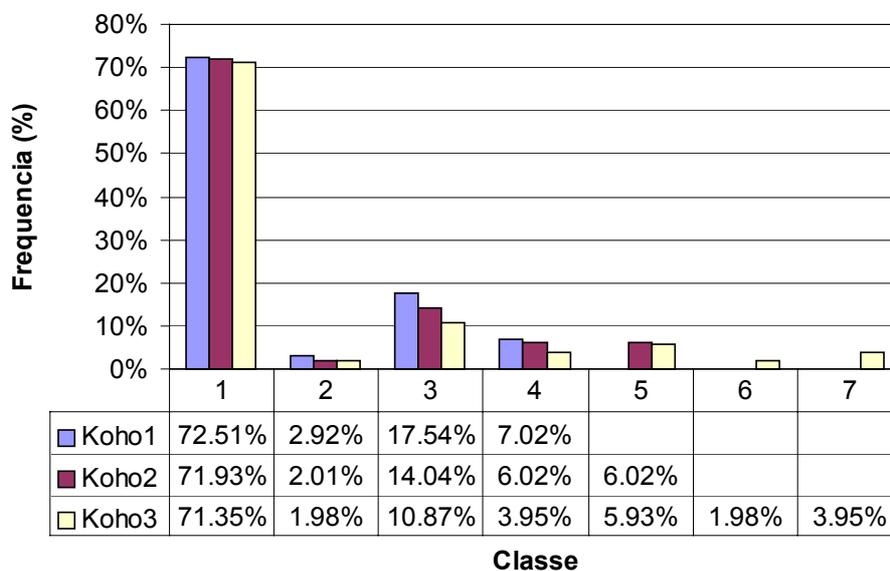


Figura 4.23. Comparação dos resultados de classificação de três redes auto-organizáveis do tipo Kohonen para o conjunto completo de testes.

Adicionalmente foram comparados os resultados individuais de classificação de cada um dos testes para as três redes, visando selecionar só aqueles que tinham sido agrupados dentro da classe 1 em todos os casos, para assim formar uma nova base de dados na qual ficaram

congregados um total de 122 testes. Desses foram separados aqueles que pertenciam ao conjunto original de validação para formar um novo conjunto de validação agora com 13 testes correspondentes a 416 pontos. Os 109 ensaios restantes, equivalentes a 4698 pontos, conformam o novo conjunto de treinamento. Ambos conjuntos encontram-se relacionados na Tabela B1, no apêndice B.

Embora seja humanamente impossível imaginar sequer a distribuição dos pontos de treinamento das redes Kohonen em um espaço de quatro dimensões, é interessante tentar pelo menos, supor a partir da observação das características dos testes rejeitados (Tabela B.1), os critérios de classificação implícitos nas redes mencionadas.

Em relação ao valor de $e_{\max}-e_{\min}$, parece claro que os testes da areia *Shirasu* foram separados em virtude do alto valor deste parâmetro. Em se tratando de uma areia vulcânica altamente estruturada, ou seja, com presença de pontes minerais ligando os grãos entre si, parece razoável separá-la das outras que definitivamente são de caráter puramente granular, sendo que nelas o comportamento mecânico depende exclusivamente dos fenômenos próprios da interação física entre os grãos.

Por outro lado, do mesmo modo é evidente que os testes realizados para valores altos do σ_{c0} também foram segregados, fato que poderia ter sido previsto com antecipação uma vez que efeitos de quebra de partículas, principalmente para areias fofas e angulosas, podem começar a aparecer a tensões de confinamento da ordem de 700 kPa, modificando substancialmente o comportamento do material. Este fenômeno, dependente principalmente de fatores como o índice de vazios inicial, a magnitude da tensão de confinamento e a dureza mineral dos grãos (Penumadu & Zhao, 1999), pode ser implicitamente considerado pelos modelos neurais aqui apresentados porquanto os três fatores de dependência mencionados foram efetivamente incorporados no vetor de treinamento; porém a insuficiente quantidade de ensaios deste tipo, provoca um condicionamento da base de dados que interfere com o processo de aprendizado da rede.

4.7.3 – ARQUITETURA E TREINAMENTO

De acordo com os resultados das etapas de treinamento dos modelos anteriores, foi observado que as redes com 90 neurônios na primeira camada intermediária oferecem melhores possibilidades para aprender o problema proposto. Por tal motivo, e usando a nova base de dados pré-classificada com a rede de Kohonen, foram criadas duas novas redes do

tipo PMC, a primeira com arquitetura 90x60 e a segunda com 90x90, completamente conectadas. As duas possuem funções de transferência do tipo tangente hiperbólico (*tansig*) na ligação entre as três primeiras camadas (uma de entrada e duas intermediárias), enquanto a ligação entre a segunda camada intermediária e a de saída é feita por uma função de transferência linear (*purelin*). Ambas foram treinadas usando a técnica de *treinamento seletivo* com o algoritmo de gradiente conjugado modificado (*trainscg*).

Observando periodicamente a evolução dos erros de treinamento e de validação foi possível construir os gráficos que se apresentam na Figura 4.24, nos quais é evidente a enorme instabilidade da rede 90x60, em contraste com a rápida convergência da rede 90x90.

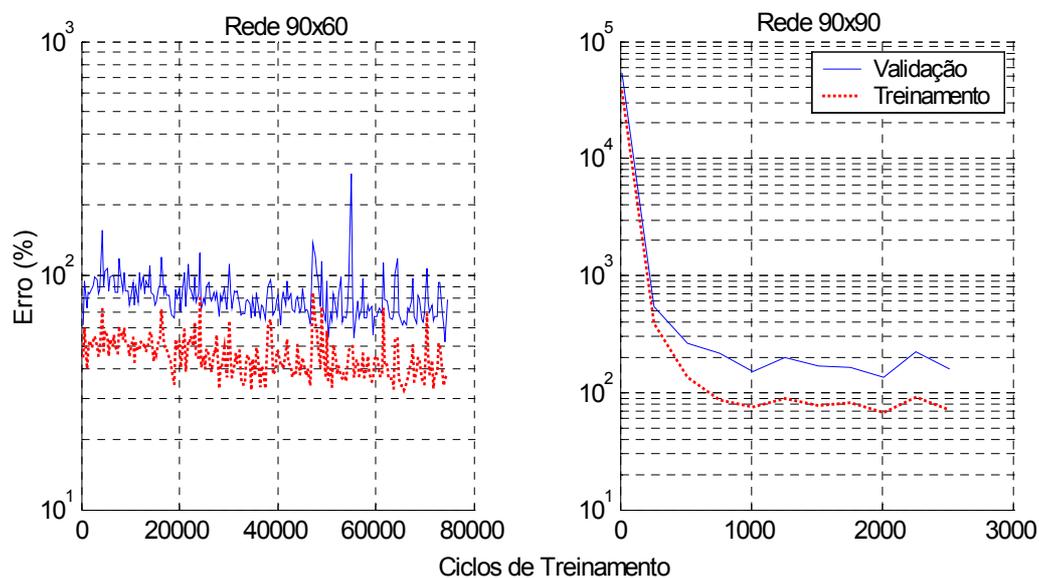


Figura 4.24. Evolução dos erros de treinamento e validação para duas redes do tipo PMC treinadas seletivamente com uma base de dados previamente classificada por uma rede de Kohonen.

Em relação à magnitude dos erros, é impossível negar que nenhuma das duas redes mencionadas ofereceu desempenhos satisfatórios. Mesmo assim, a rede 90x90 pode ser novamente testada levando em conta algumas modificações na normalização dos dados de entrada e principalmente na estratégia de apresentação dos exemplos de treinamento, tal como se explica a seguir.

Considerando que o $e_{\max}-e_{\min}$ em combinação com o fator de forma das partículas, n_s , são talvez os parâmetros físicos que individualmente fornecem maior quantidade de informação sobre o comportamento mecânico da areia, e observando que materiais diferentes estavam

sendo representados por valores com pequenas diferenças entre si, achou-se adequado modificar a forma como estava sendo feita a normalização destes fatores. Inicialmente, sobre os dados originais aplicou-se uma transformação que busca realçar as diferenças para os valores pequenos, mantendo-a invariável para os valores grandes. Uma transformação deste tipo pode ser vista no gráfico da Figura 4.25, tanto para o fator de forma (n_s) como para o $e_{\max}-e_{\min}$. As equações correspondentes se apresentam a seguir:

$$(e_{\max} - e_{\min})_{\text{mod}} = 7(e_{\max} - e_{\min})^3 - 15(e_{\max} - e_{\min})^2 + 11(e_{\max} - e_{\min}) - 2 \quad (4.10)$$

$$(n_s)_{\text{mod}} = -0,7(n_s)^2 + 37,2(n_s) - 398 \quad (4.11)$$

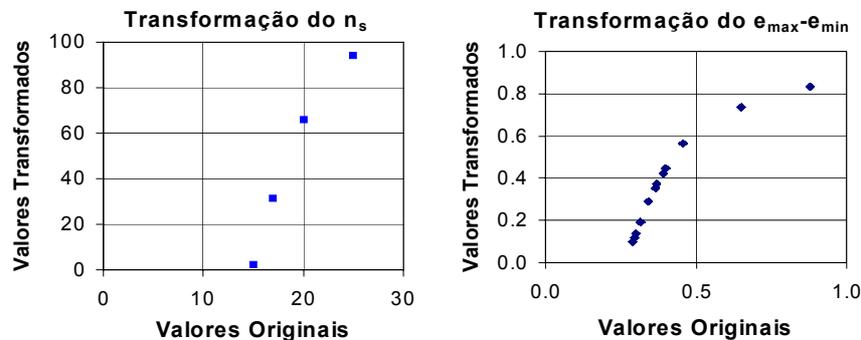


Figura 4.25. Transformação dos valores do $e_{\max}-e_{\min}$ e do n_s .

Finalmente, os dados transformados são normalizados para o intervalo $[-1 \ 1]$, e incorporados em substituição dos valores antigos, na base de dados cuja estrutura permaneceu inalterada. Os valores máximos e mínimos das variáveis de treinamento da rede são apresentados na Tabela C.1, no apêndice C, na qual as três últimas linhas, em vermelho, representam as saídas.

Uma nova rede do tipo PMC 90x90, chamada *Magola9*, foi criada e treinada com a base de dados modificada, composta como já foi mencionado, por 4698 vetores correspondentes a 109 testes. Para efeitos de validação foram reservados 416 vetores correspondentes a 13 testes.

Visando facilitar o processo de aprendizado, a base de dados de treinamento foi fragmentada em 6 porções consecutivas, cada uma das quais incorpora os dados da anterior além de dados novos. A rede neural foi treinada então com as diferentes sub-bases de dados de maneira sucessiva e usando o modo “por bloco”, alternando periodicamente procedimentos de treinamento e validação. Após a rede ter assimilado corretamente a informação constitutiva

fornecida por uma das porções apresentadas, seu conhecimento é atualizado incorporando então nova informação constitutiva contida na porção seguinte. Este método de subdivisão permite à rede potencializar melhor sua capacidade de aprendizado, coisa que não acontece quando é obrigada a assimilar a totalidade da base de dados diretamente (Pernor & Lamarque, 1999).

A Figura 4.26 mostra a evolução do processo de treinamento em termos do erro médio quadrático usando a técnica de subdivisão da base de dados. Deve-se ressaltar que o terceiro bloco de informação contendo 2406 vetores foi apresentado a primeira vez com a base de dados não transformada, a segunda vez após ter transformado o $e_{\max}-e_{\min}$ e o n_s , (1-2406b na legenda da figura). Neste ponto foi decidido aplicar no valor do e_0 a mesma transformação que foi aplicada ao $e_{\max}-e_{\min}$ e assim continuar com o processo de treinamento do terceiro bloco correspondente à legenda 1-2406c. Já as porções subseqüentes incluem a base de dados completamente transformada.

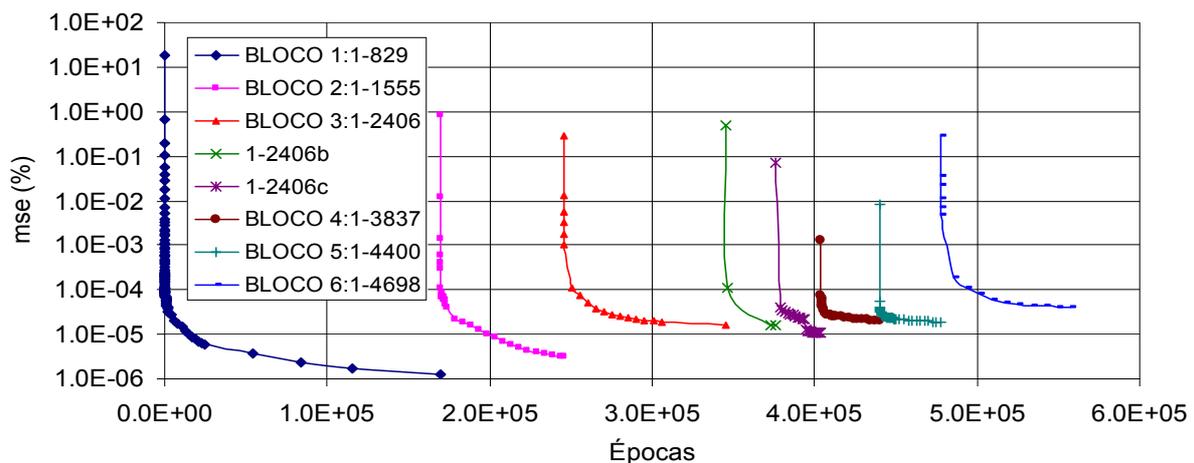


Figura 4.26. Evolução do erro de treinamento em função do número de iterações para a rede *Magola 9*.

Observa-se na mesma figura que o erro não diminui além do limite de 4×10^{-5} . É notável a presença de grandes picos de erro ao início de cada ciclo de treinamento, imputáveis ao esquema de subdivisão imposto: cada vez que a base de dados é atualizada, a rede deve aprender novos padrões de comportamento, os quais não são assimilados instantaneamente gerando aumento significativo do erro.

Considerando que a apresentação das diferentes bases de dados durante o processo de treinamento foi feita no modo de bloco, e que as simulações são feitas no modo recorrente, é

claro que durante o processo de simulação a rede gera uma nova base de dados diferente daquela que constitui o conjunto de treinamento original. Apresentado só o primeiro ponto de cada trajetória, a partir do segundo e até o final da mesma, cada resposta da rede diferirá do valor experimental original uma vez que o modelo neural sempre simula com erro.

4.7.4 – RESULTADOS DA SIMULAÇÃO

Uma primeira forma de visualizar os resultados de simulação do modelo é por meio de gráficos de valores reais versus valores simulados. Na Figura 4.27 apresenta-se a comparação para os acréscimos de tensão e para os correspondentes valores totais das tensões, obtidos com uma simulação estática da rede, na qual foram usados de maneira unificada os dados de treinamento e validação.

Por outro lado, na Figura 4.28 são apresentadas comparações similares, porém feitas a partir de uma simulação autônoma da rede só para os três componentes dos acréscimos de tensão.

Observa-se como nos três gráficos da linha superior da Figura 4.28 aparece uma dispersão grande gerada por um número pequenos de pontos e conseqüentemente baixos coeficientes de correlação. Analisando a base de dados original foi possível detectar a origem desses pontos dispersos. Todos eles, num total de 12, pertencem à parte final do teste 146 que corresponde a um ensaio de compressão hidrostática no qual, a tensão confinante final atinge o valor limite do intervalo de treinamento das tensões (ver Tabela C.1 no apêndice C).

Tendo-se em conta que nas regiões do espaço de treinamento que se encontram perto das fronteiras do mesmo, a concentração de dados é geralmente baixa e mal distribuída, é razoável imputar a origem da simulação deficiente ao fato de ter solicitado à rede com pontos localizados precisamente numa região de fronteira, como é a parte final do ensaio em questão. Foram então retirados os 12 pontos mencionados, obtendo os resultados que se apresentam nos três gráficos da linha inferior da Figura 4.28 nos quais é evidente a melhora no desempenho.

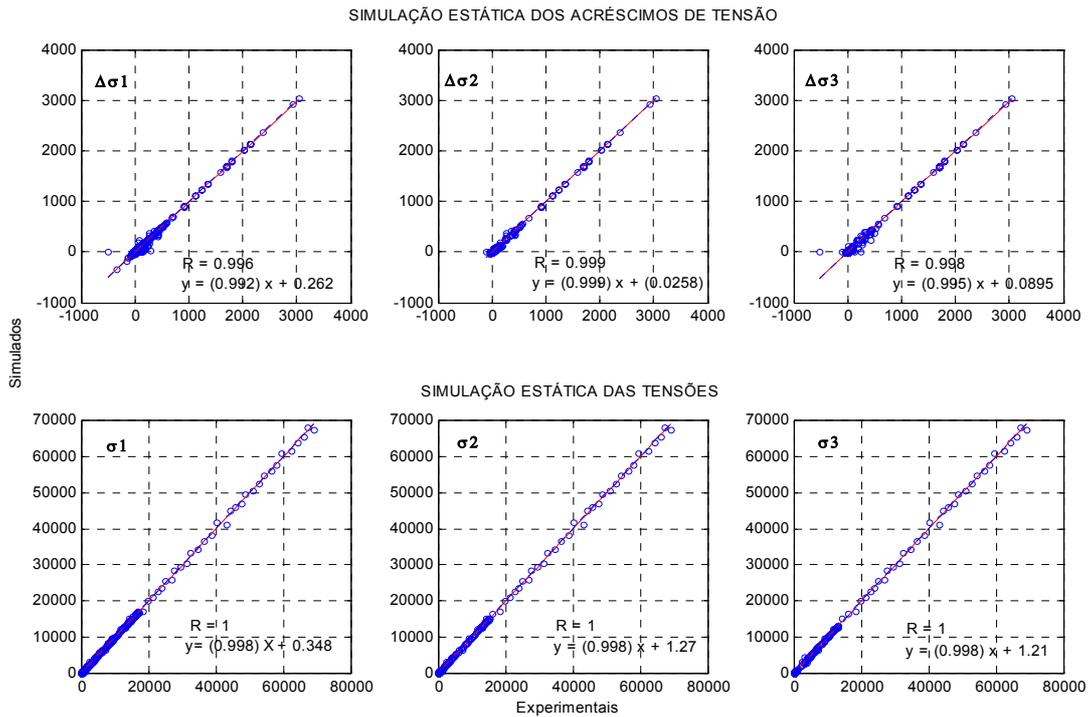


Figura 4.27. Resultados de simulação estática da rede *Magola9* treinada com a técnica de subdivisão da base de dados, para os acréscimos de tensão e para as tensões correspondentes.

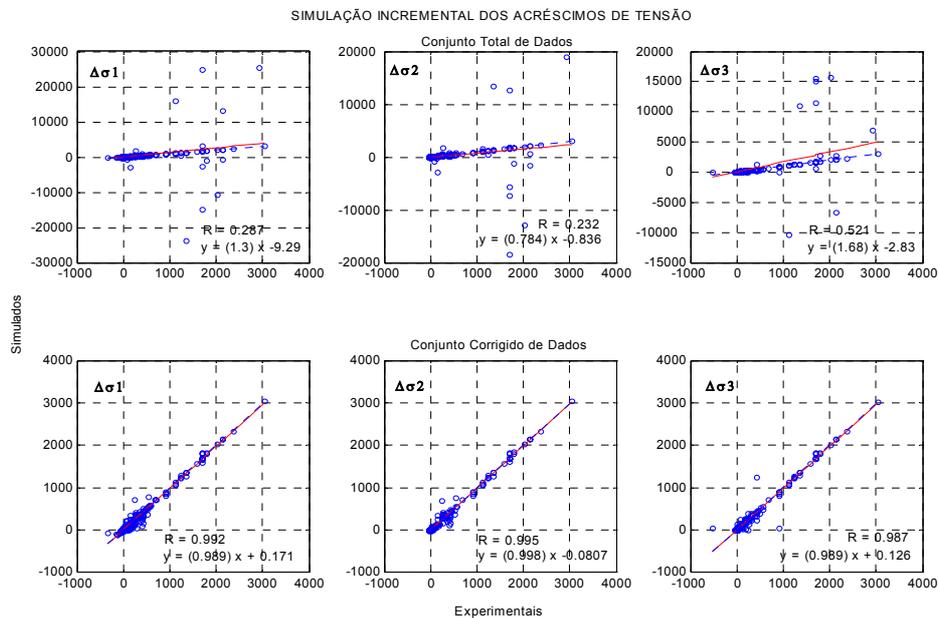


Figura 4.28. Resultados de simulação autônoma da rede *Magola9* treinada com a técnica de subdivisão da base de dados, para os três componentes dos acréscimos de tensão.

Uma análise similar pode ser feita a partir dos resultados de simulação das tensões. As diferenças entre os ajustes anteriores à eliminação dos 12 pontos finais do teste 146, e os ajustes posteriores à mesma, são marcantes. Isso pode ser observado na Figura 4.29.

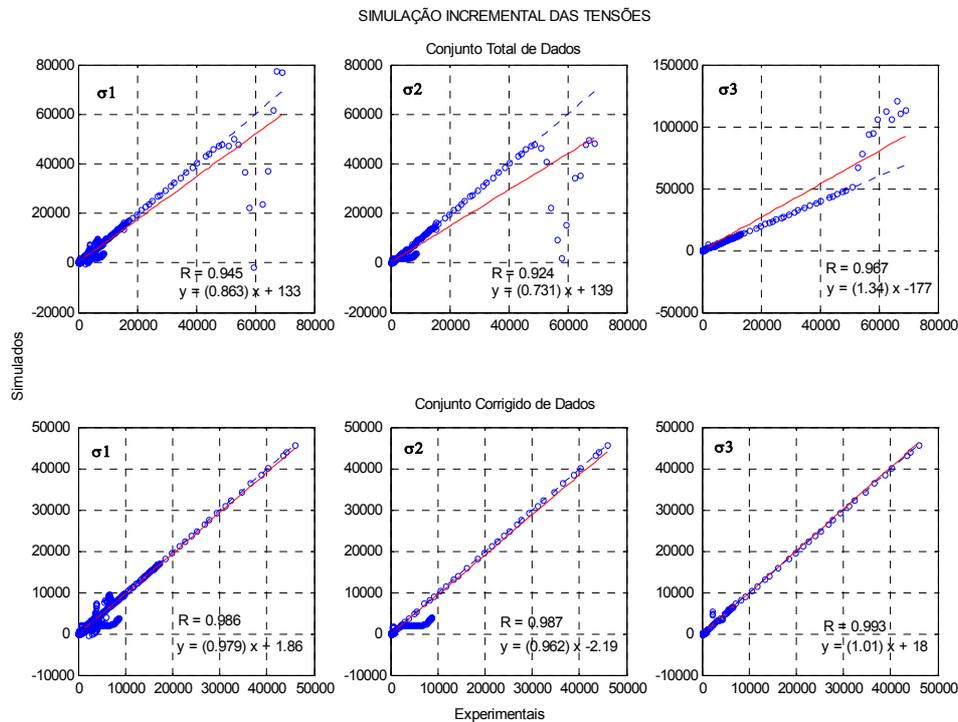


Figura 4.29. Resultados de simulação incremental ou autônoma da rede MLP 90x90 *Magola9* treinada com a técnica de subdivisão da base de dados, para os três componentes de tensão.

Uma outra forma de apresentar os resultados das simulações é por meio de gráficos tensão-deformação. Neste caso, tal como foi exposto anteriormente, as tensões são representadas pela razão τ_{oct}/p , nas ordenadas, enquanto que nas abscissas são representadas as deformações. Cada gráfico está composto de três curvas, uma para cada componente de deformação principal. Em todos os gráficos os pontos representam os dados experimentais e as linhas contínuas os resultados das simulações da rede. Para os testes de compressão hidrostática, as ordenadas correspondem à tensão média, p , e as abscissas à deformação volumétrica. No apêndice C, Tabela C.2, encontra-se um resumo das diferentes trajetórias para cada um dos tipos de areia. Observa-se, em relação ao tipo de teste, que as trajetórias do tipo CTC e TC são as mais abundantes. Por outro lado, em relação ao tipo de areia, é evidente a preponderância de Karlsruhe, Hostun, Toyoura, Cambria e Monterrey sobre as demais.

A combinação dos dois critérios anteriores, tipo de areia e tipo de trajetória, foi usada para definir a ordem de apresentação das simulações tensão-deformação de cada um dos 122 testes

considerados. As primeiras figuras correspondem às areias mais abundantes na base de dados enquanto as últimas são as menos representadas. Adicionalmente, para cada tipo de areia em particular, primeiro são apresentadas as simulações correspondentes às trajetórias mais abundantes, e posteriormente as simulações das trajetórias menos comuns.

A Figura 4.30 contém as simulações da areia de Karlsruhe, todas elas com trajetórias de tensão do tipo CTC. Por se tratar da areia e da trajetória mais abundante na base de dados, todas as simulações foram bem aproximadas dos dados experimentais.

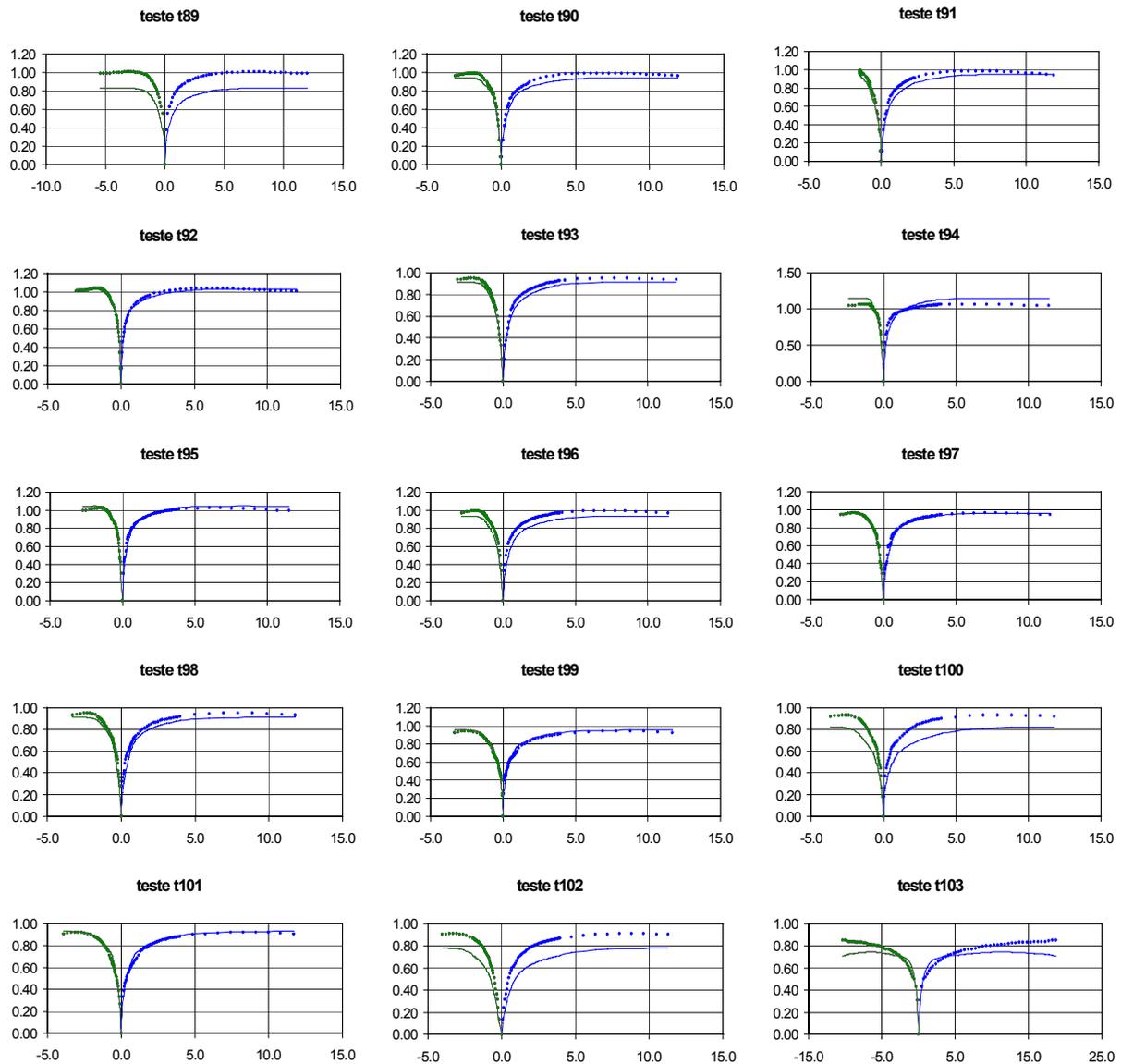


Figura 4.30. Simulação do modelo *Magola 9* para areia Karlsruhe. Trajetórias CTC.

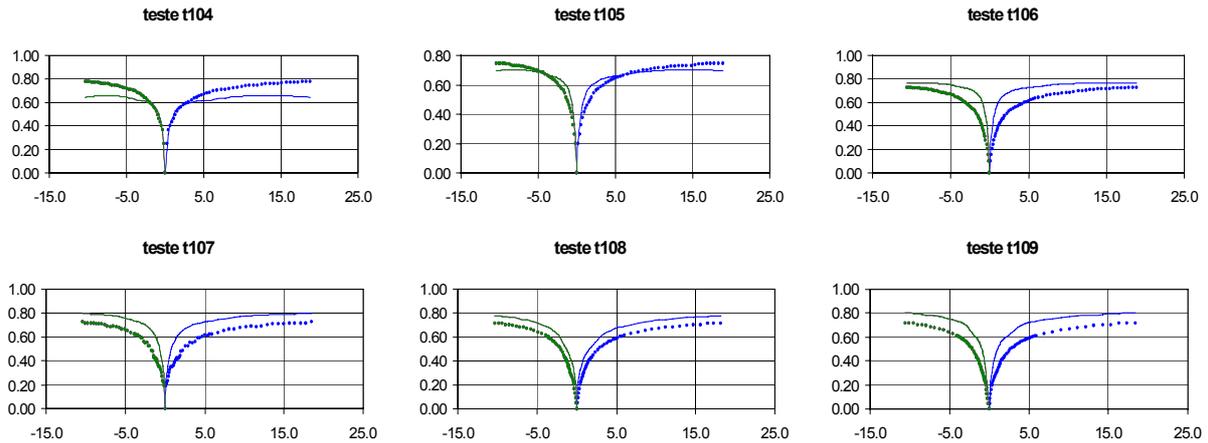


Figura 4.30. (Continuação) Simulação do modelo *Magola 9* para areia Karlsruhe. Trajetórias CTC.

A partir da Figura 4.31 até a Figura 4.33, são apresentados os da areia Hostun. Observaram-se simulações de pobre qualidade nos testes cíclicos (122, 123 e 124) e nos testes realizados a níveis de deformação muito pequenos (142, 143, 144 e 145). Os demais resultados para testes do tipo CTC são aceitáveis com exceção do teste 84 que apresentou um comportamento oscilatório não explicado. Igualmente aceitáveis são os resultados das trajetórias restantes.

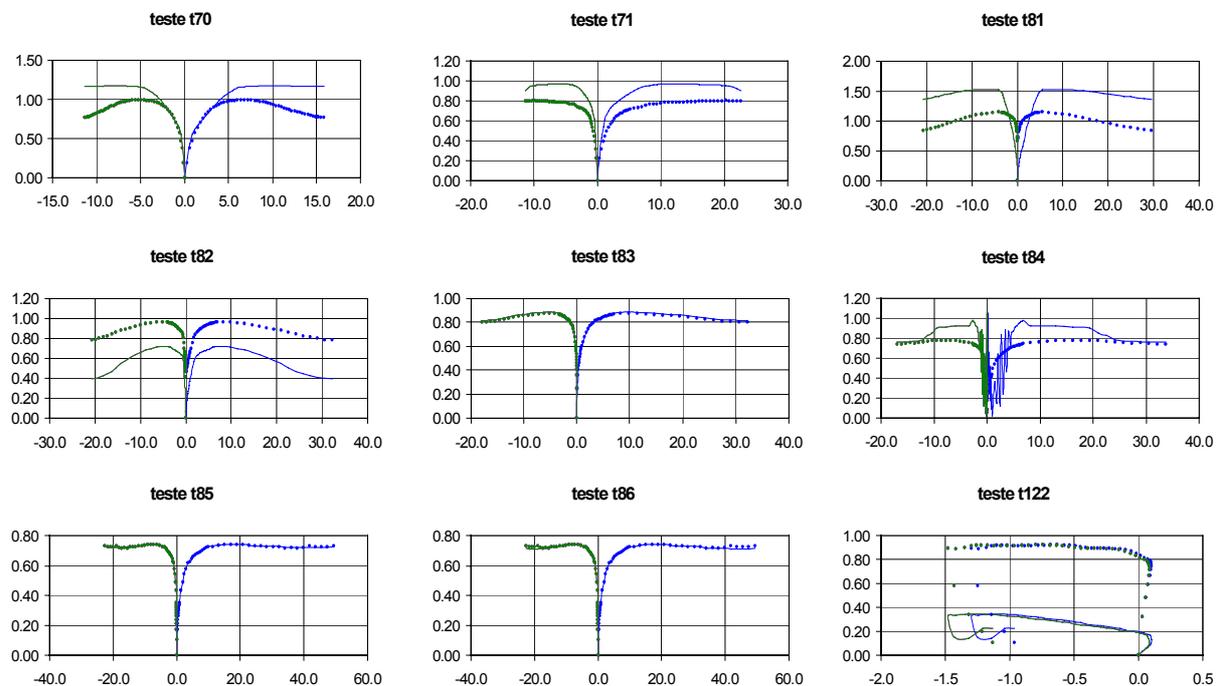


Figura 4.31. Simulação do modelo *Magola 9* para a areia Hostun. Trajetórias CTC.

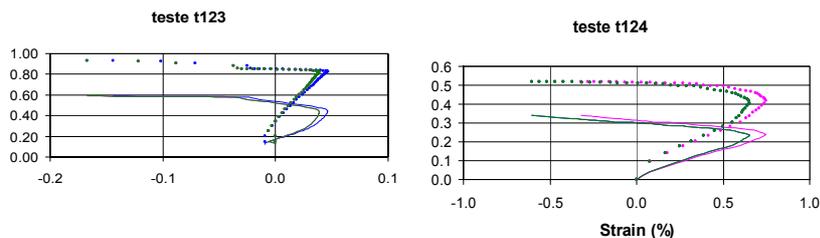


Figura 4.31. (Continuação) Simulação do modelo *Magola 9* para a areia Hostun. Trajetórias CTC.

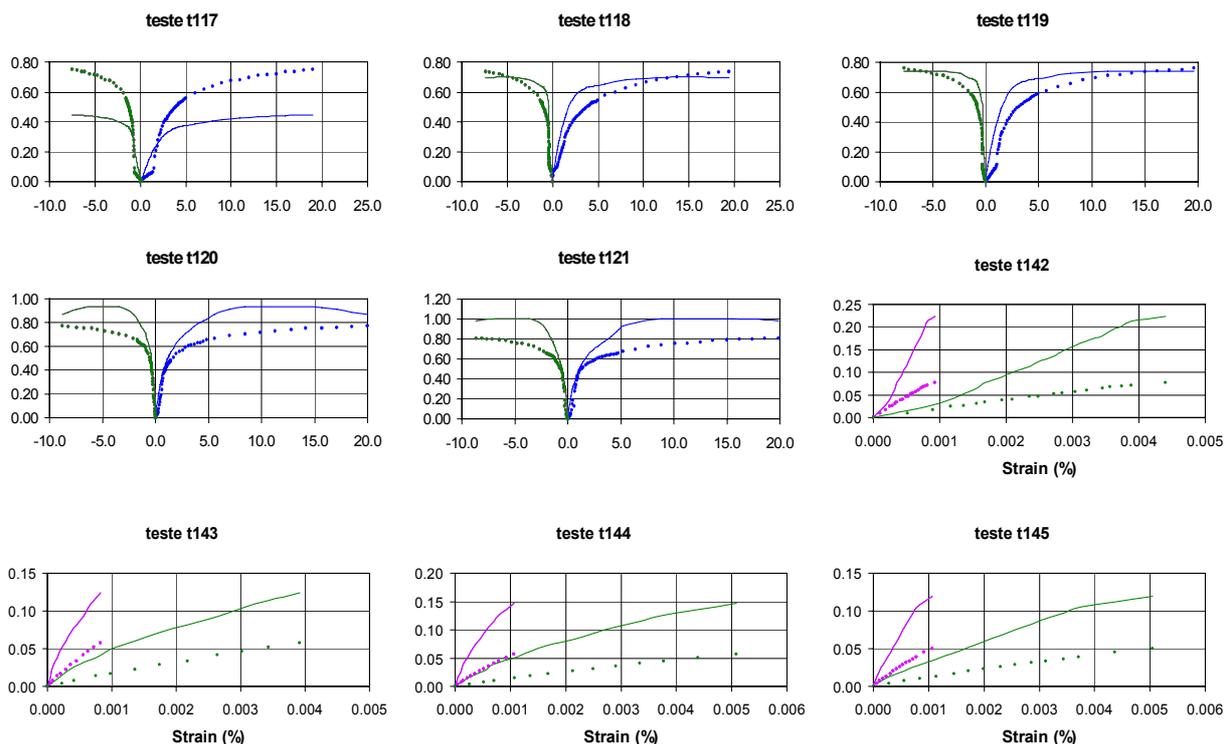


Figura 4.32. Simulação do modelo *Magola 9* para a areia Hostun. Trajetórias TC.

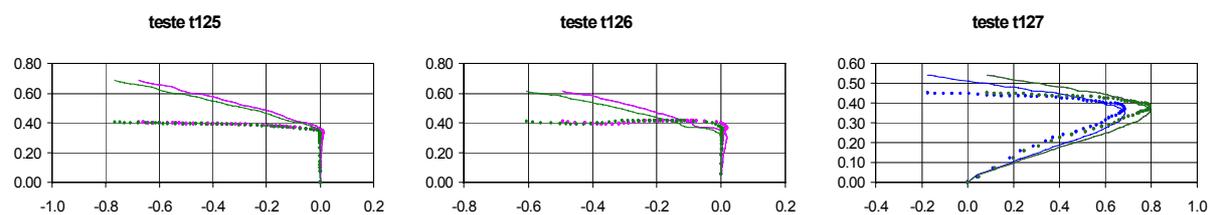


Figura 4.33. Resultados de simulação do *Modelo 9* para a areia Hostun. Trajetórias tipo RTE.

A partir da Figura 4.34 até a Figura 4.40, são apresentados os resultados da areia de Toyoura. De novo, os melhores resultados correspondem às trajetórias CTC, sendo evidente a deterioração da qualidade nas simulações à medida que se analisam os gráficos subsequentes. No entanto, é interessante ver como o modelo conseguiu simular perfeitamente a trajetória

HC, e por outro lado, aproximar muito bem a forma das trajetórias no plano octaédrico (Figura 4.40).

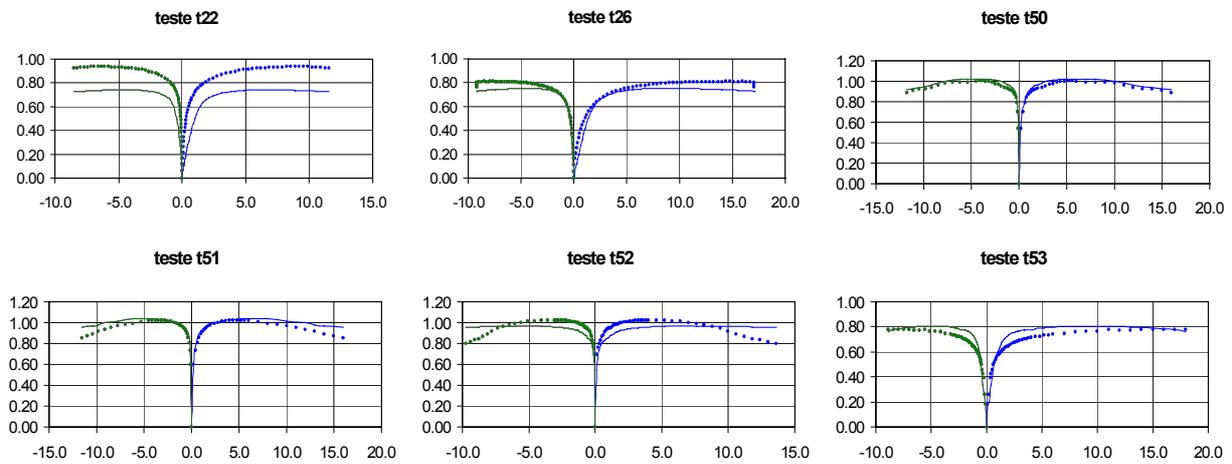


Figura 4.34. Resultados de simulação do modelo *Magola 9* para areia Toyoura. Trajetórias CTC.

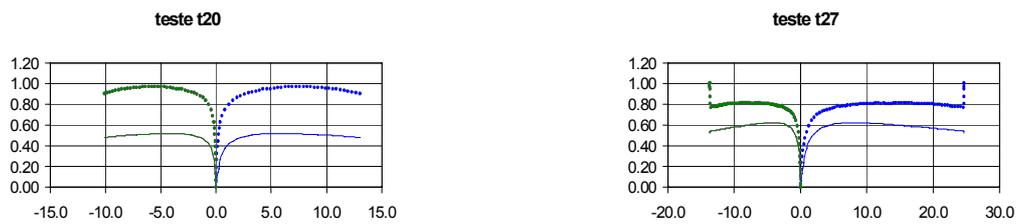


Figura 4.35. Resultados de simulação do modelo *Magola 9* para a areia Toyoura. Trajetórias TC.



Figura 4.36. Resultados de simulação do modelo *Magola 9* para areia Toyoura. Trajetórias CTE.



Figura 4.37. Resultados de simulação do modelo *Magola 9* para a areia Toyoura. Trajetórias TE.



Figura 4.38. Resultados de simulação do modelo *Magola 9*, areia Toyoura. Trajetórias RTC.



Figura 4.39. Resultados de simulação do modelo *Magola 9* para areia Toyoura. Trajetórias RTE.

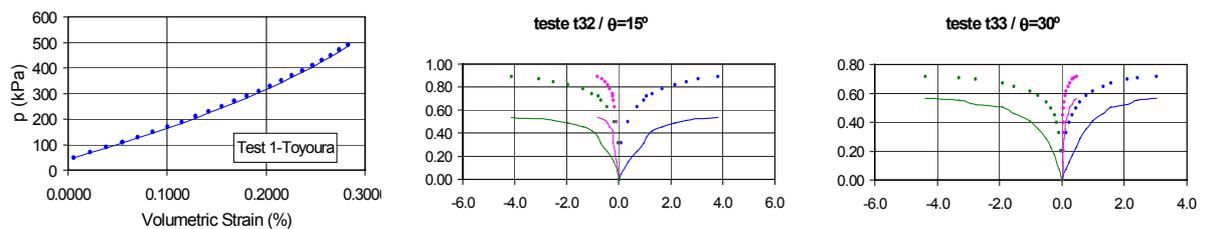


Figura 4.40. Resultados de simulação do modelo *Magola 9* para a areia Toyoura. O teste 1 corresponde a uma trajetória HC, os restantes correspondem a trajetórias no plano octaédrico.

A partir da Figura 4.41 até a Figura 4.46 são apresentados os resultados para a areia de Cambria. Observa-se como as trajetórias CTC foram bem simuladas tanto na forma da curva como na magnitude das tensões. Em contraste, nas trajetórias TC o modelo falhou na previsão da magnitude das tensões. Nos testes HC e de deformação plana o modelo apresentou um

desempenho satisfatório, porém alguns dos testes com trajetórias de extensão (CTE e TE) não foram bem simulados, o que era esperado já que, como já foi explicado, este tipo de trajetória não era abundante no conjunto de treinamento.



Figura 4.41. Resultados de simulação do modelo *Magola 9* para areia Cambria. Trajetórias CTC.

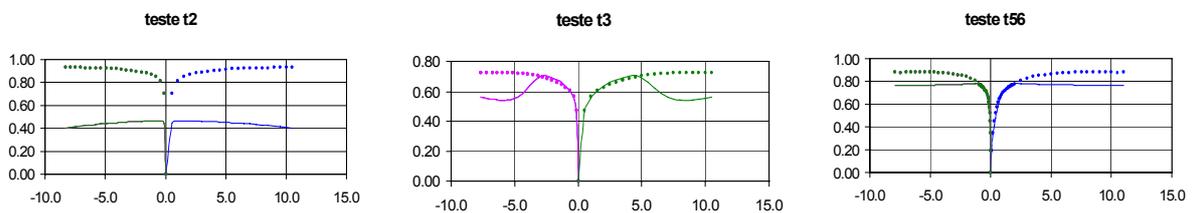


Figura 4.42. Resultados de simulação do modelo *Magola 9* para areia Cambria. Trajetórias TC.

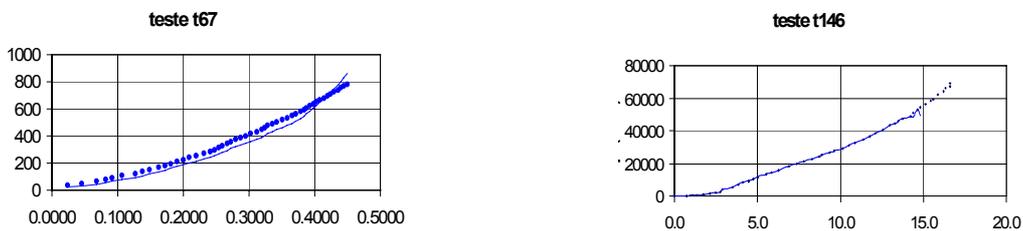


Figura 4.43. Resultados de simulação do modelo *Magola 9* para areia Cambria. Trajetórias HC.

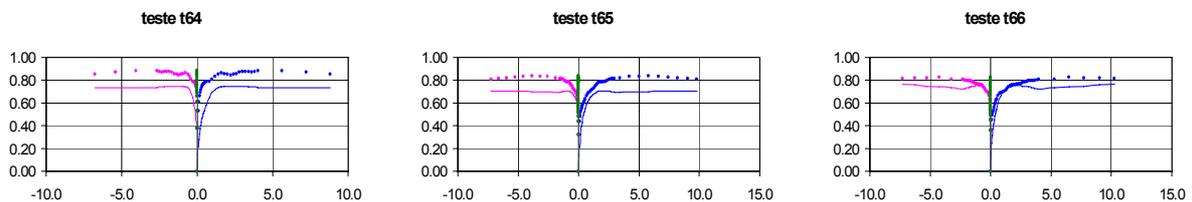


Figura 4.44. Resultados de simulação do modelo *Magola 9* para areia Cambria. Trajetórias D.P.

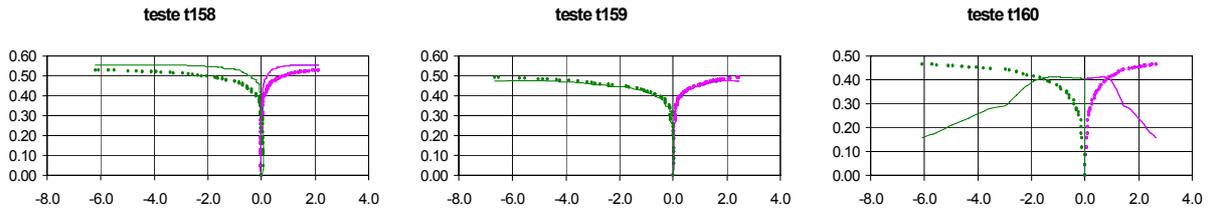


Figura 4.45. Resultados de simulação do modelo *Magola 9* para areia Cambria. Trajetórias CTE.



Figura 4.46. Resultados de simulação do modelo *Magola 9* para areia Cambria. Trajetórias TE.

A areia de Monterrey foi simulada para diversas trajetórias no plano octaédrico. Inicialmente (Figura 4.47) são apresentadas as simulações das trajetórias do tipo TC ($\theta=0^\circ$), e posteriormente (Figura 4.48) as do tipo TE ($\theta=60^\circ$). Em contraste, as trajetórias correspondentes a ângulos intermediários (Figura 4.49), consideradas mais difíceis, foram melhor aproximadas do que as trajetórias TC.



Figura 4.47. Resultados de simulação do modelo *Magola 9* para areia Monterrey. Trajetórias TC.



Figura 4.48. Resultados de simulação do modelo *Magola 9* para a areia Monterrey. Trajetórias tipo TE.

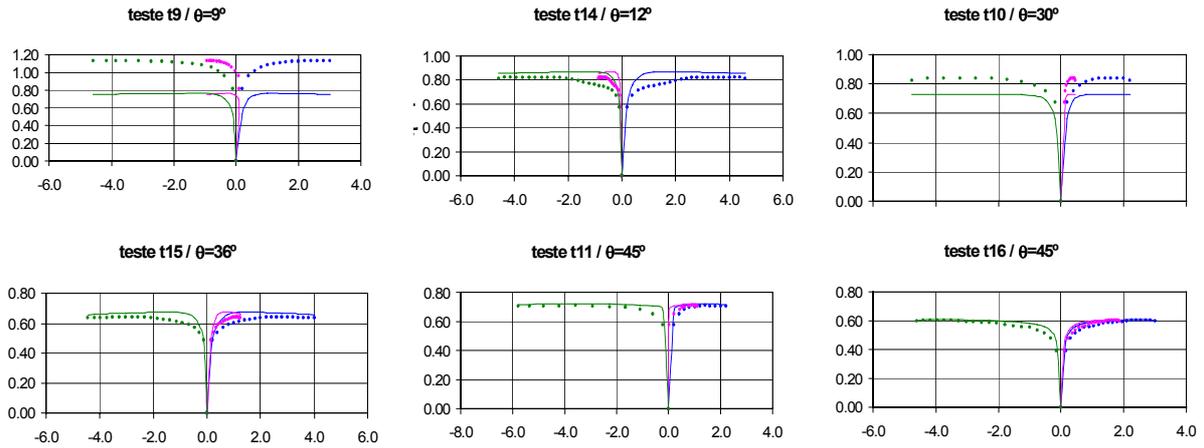


Figura 4.49. Resultados de simulação do modelo *Magola 9* para a areia Monterrey. Trajetórias no plano octaédrico.

A partir da Figura 4.50 até a Figura 4.54 são apresentadas as simulações dos testes CTC das areias Calcárea, Ottawa, Antioch, Chattahoochee e San Fernando, todas elas com níveis aceitáveis de precisão. Finalmente, na Figura 4.55 estão as simulações de duas trajetórias HC, correspondentes às areias SLB e Ticino, ambas muito próximas dos valores experimentais

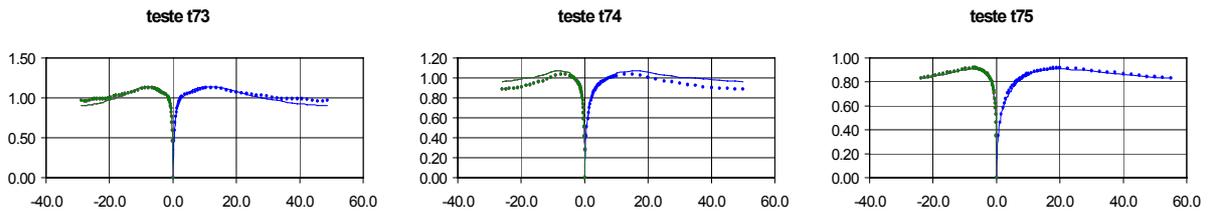


Figura 4.50. Resultados de simulação do modelo *Magola 9* para a areia Calcárea. Trajetórias tipo CTC.

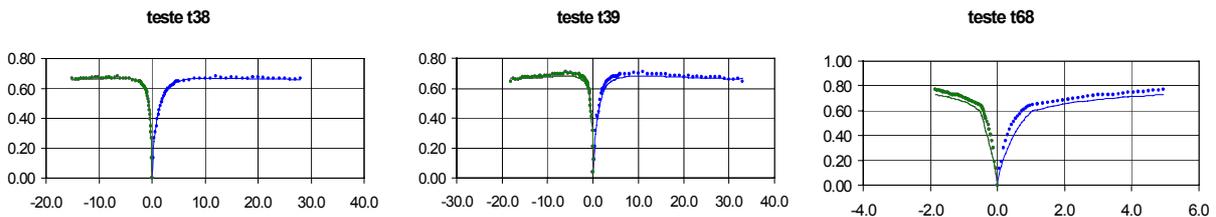


Figura 4.51. Resultados de simulação do modelo *Magola 9* para a areia Ottawa. Trajetórias tipo CTC.

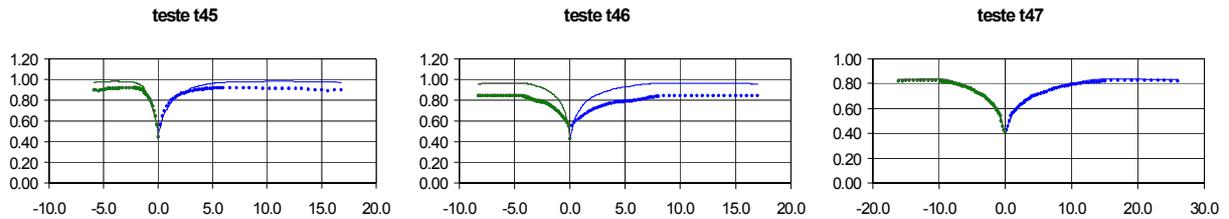


Figura 4.52. Resultados de simulação do modelo *Magola 9* para a areia Antioch. Trajetórias tipo CTC.



Figura 4.53. Resultados de simulação do modelo *Magola 9* para a areia Chattahoochee. Trajetórias CTC.



Figura 4.54. Resultados de simulação do modelo *Magola 9* para a areias San Fernando. Trajetórias CTC.



Figura 4.55. Resultados de simulação das areias SLB e Ticino. Trajetórias tipo HC.

4.7.5 – DISCUSSÃO

O histograma da Figura 4.56 mostra a distribuição das frequências dos erros de simulação das tensões para o modelo *Magola 9*. Observa-se que para o 63,3% dos testes simulados o erro do modelo neural foi menor que 10%, e para o 75,2% dos testes o erro foi menor do que 20%. Esses resultados mostram que trajetórias de tensão 3D de alta complexidade podem ser

bem simuladas por redes neurais artificiais do tipo perceptron de múltiplas camadas adequadamente treinadas com conjuntos abrangentes de dados experimentais, para uma ampla variedade de materiais sob condições de tensão iniciais diversas.

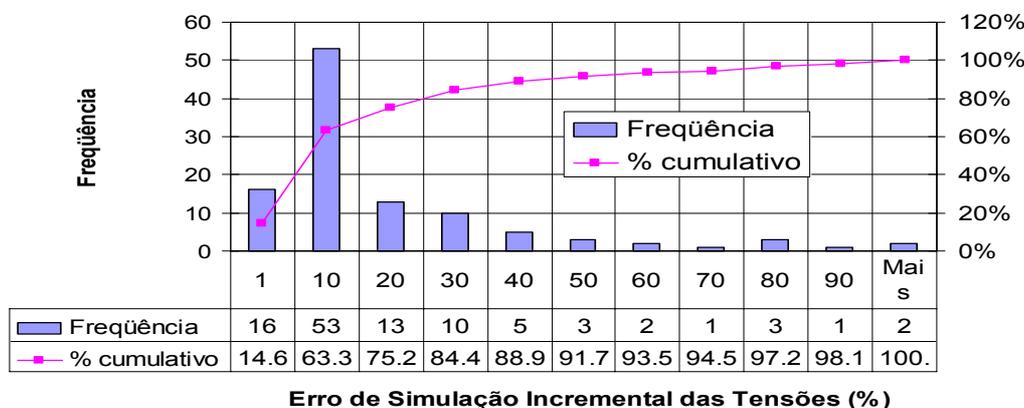


Figura 4.56. Histograma de frequência dos erros de simulação incremental das tensões para a rede *Magola9*.

Duas alternativas de treinamento foram investigadas: estática e recorrente; e também duas formas de simulação: ponto-a-ponto e autônoma. Embora as simulações ponto-a-ponto apresentem respostas muito precisas, elas não são úteis em aplicações numéricas como a técnica dos elementos finitos.

Foram quatro as variáveis usadas para identificar o tipo de areia e a sua condição inicial: fator de forma (n_s), tensão confinante inicial (σ_{c0}), e_{\max} - e_{\min} , e o índice de vazios inicial (e_0). Destas quatro variáveis talvez seja o parâmetro e_{\max} - e_{\min} o que apresenta maior variabilidade natural, devido à dificuldade experimental na determinação precisa do e_{\max} e do e_{\min} .

O valor do e_{\max} e e_{\min} para uma areia específica não pode ser considerado único ainda que sejam usados procedimentos experimentais padronizados na sua determinação. Isto porque os testes de laboratório sempre permitem um certo grau de variação nos detalhes do procedimento e em consequência de imprecisão no resultado final. Para ilustrar tal variação, Cubrinovski & Ishihara (2002) compilaram os valores destas duas variáveis para a areia de Toyoura a partir de 13 estudos independentes publicados na revista *Soils and Foundations* entre 1992 e 2002. Os valores reportados nesses estudos foram obtidos com o procedimento padrão da JGS (*Japanese Geotechnical Society*) e são apresentados na Figura 4.57. Observe-se que os dados estão bem agrupados e apresentam pouca dispersão em torno dos valores médios do e_{\max} e o e_{\min} . Os desvios padrões são 0,8% e 1,14%, respectivamente. As

dispersões máximas correspondentes são 1,6% e 3,6%. Em relação ao $e_{\max}-e_{\min}$, os mesmos autores reportaram um valor médio de 0,364 com desvio padrão de 1,3% e coeficiente de variação de 3,57%.

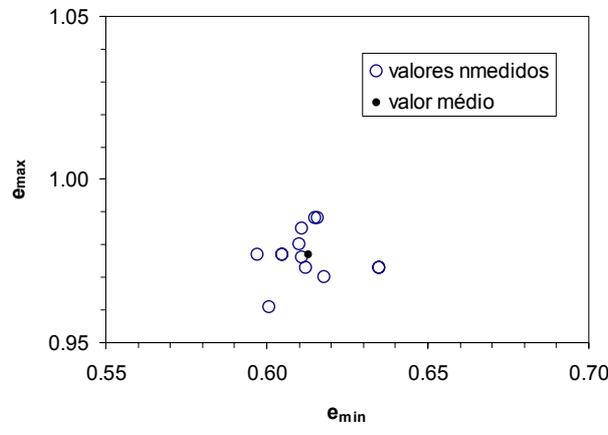


Figura 4.57. Valores do e_{\max} e do e_{\min} para a areia de Toyoura determinados a partir de 13 estudos independentes usando procedimentos padronizados da JGS – *Japanese Geotechnical Society* (modificada de Cubrinovski & Ishihara, 2002).

A Figura 4.58 apresenta o histograma de freqüência dos valores do $e_{\max}-e_{\min}$ de todos os exemplos de treinamento da rede *Magola9*. Nela pode-se observar que os dados apresentam grande concentração ao redor do valor médio que é 0,34.

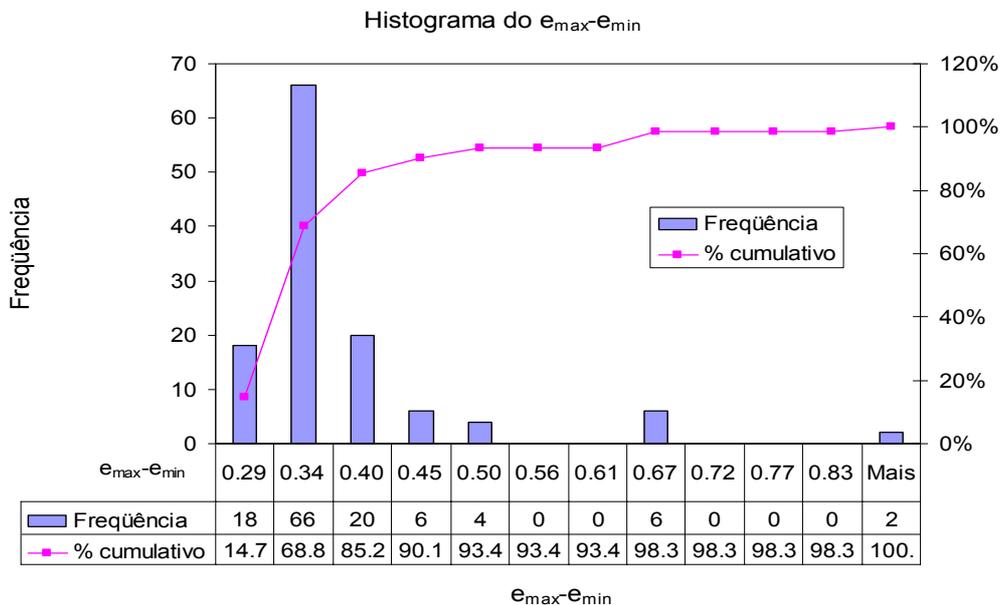


Figura 4.58. Histograma de freqüência do $e_{\max}-e_{\min}$ de todos os exemplos de treinamento da rede *Magola9*.

O fato de os valores de $e_{\max}-e_{\min}$ estarem muito concentrados, indica que os diferentes materiais considerados apresentam, obviamente, valores próximos deste parâmetro e, em conseqüência, a análise de sensibilidade deverá considerar variações cuja magnitude seja compatível com a variação natural do mesmo. Tomando como ponto de partida os valores dos coeficientes de variação do e_{\max} e do e_{\min} reportados por Cubrinovski & Ishihara (2000), e a alta concentração do parâmetro $e_{\max}-e_{\min}$ mostrada na Figura 4.58, assumiu-se que a avaliação da sensibilidade do modelo neural poderia ser realizada considerando variações máximas da ordem de 5% nos valores do e_{\max} e do e_{\min} . Foram considerados oito casos diferentes tal como se apresenta na Figura 4.59. Em cada caso é incorporada uma variação de 2% ou 5% no valor do e_{\max} ou do e_{\min} e para cada um deles foram simulados de maneira autônoma todos os testes da base de dados. Posteriormente foram calculados os erros de cada um dos testes mediante comparação direta entre os valores reais das tensões totais e os valores correspondentes produzidos pela rede neural. Adicionalmente, para cada uma das componentes de tensão principal foram realizados ajustes de correlação linear entre os valores reais e os resultados do modelo. O valor do coeficiente de correlação, R, indica a proximidade entre as respostas do modelo e os valores experimentais.

Na Figura 4.60 são apresentados em forma de barras os valores do coeficiente R para cada componente de tensão e para cada um dos casos analisados.

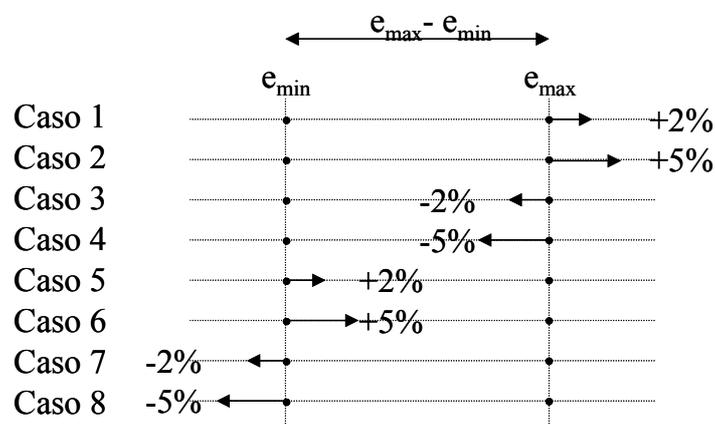


Figura 4.59. Diferentes casos para analisar a sensibilidade do modelo *Magola9* à variações do e_{\max} e do e_{\min} .

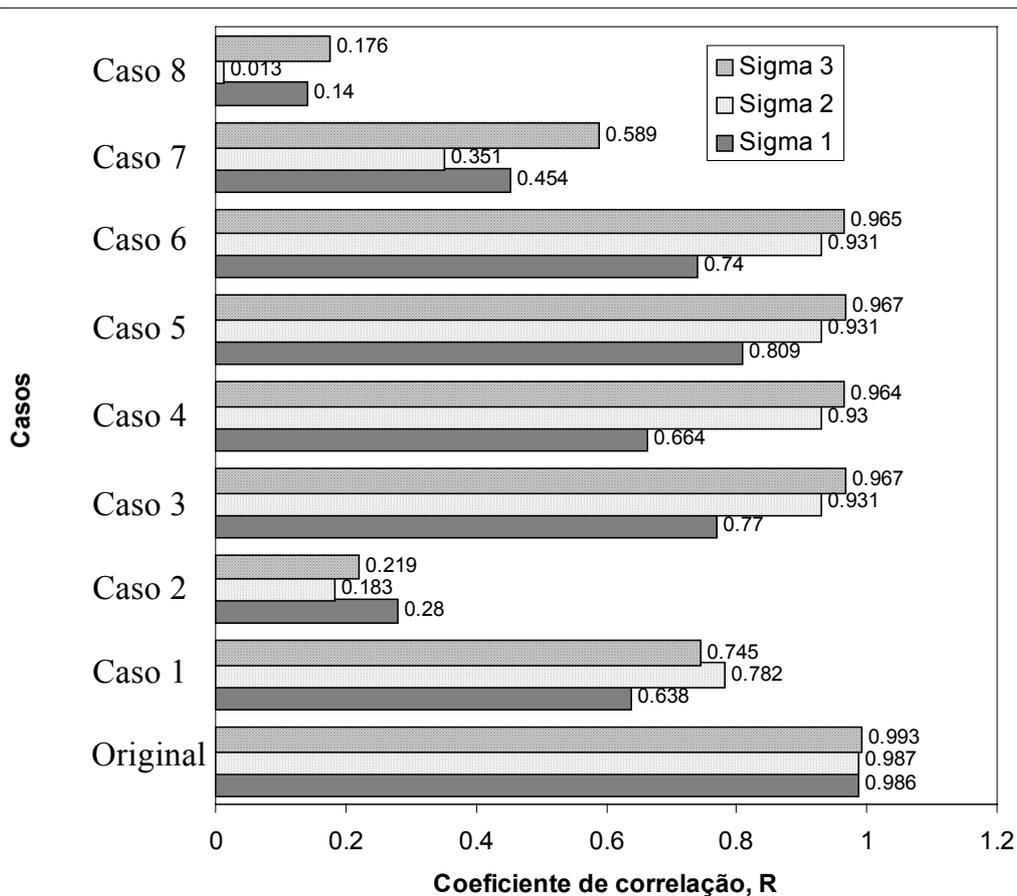


Figura 4.60. Gráfico de coeficientes de correlação para cada uma das componentes de tensão e para cada um dos casos considerados na análise de sensibilidade do modelo *Magola9*.

Na Figura 4.60 os valores do coeficiente R aparecem indicados à frente de cada barra. Cada caso está representado por três barras, uma para cada componente de tensão principal. Na parte inferior do gráfico são apresentadas, para efeito de comparação, as barras correspondentes ao modelo validado com os valores originais do ϵ_{\max} - ϵ_{\min} . As barras seguintes correspondem às validações dos diferentes casos considerados. A observação direta do comprimento das barras permite identificar que os casos 3, 4, 5 e 6 apresentam maior similitude com o caso original. Os casos restantes mostraram desempenhos fracos com coeficientes de correlação baixos.

Uma das principais características dos modelos neurais é a sua robustez. Isto significa que eles possuem a capacidade de lidar com dados que apresentam ruídos devidos a erros naturais de procedimento experimental ou devidos à variabilidade natural das variáveis incorporadas no treinamento. Essa robustez é válida também para o procedimento de validação, fazendo com que estes modelos forneçam respostas confiáveis quando as variáveis do exemplo de teste apresentado incorporam variações naturais. Não obstante, para o caso da

rede *Magola9* se identificou que a sensibilidade do modelo é variável e dependente da forma como as variações do parâmetro $e_{\max}-e_{\min}$ são dadas. Os casos nos quais a variação deste parâmetro induz uma redução dos valores de $e_{\max}-e_{\min}$ (casos 3, 4, 5 e 6), Figura 4.59, a sensibilidade é baixa, ou seja, a resposta do modelo validado com os valores modificados não varia muito em relação à resposta do modelo validado com os valores originais. A Figura 4.61 mostra que para os casos 3, 4, 5 e 6 há um aumento na proporção dos valores pequenos do $e_{\max}-e_{\min}$, situação que não induz variações grandes na resposta de simulação do modelo.

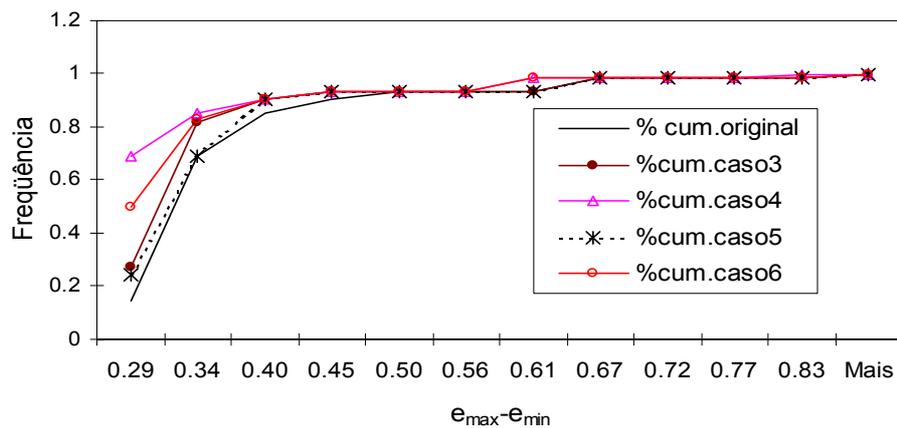


Figura 4.61. Curvas de frequência acumulada do $e_{\max}-e_{\min}$ para os casos 3, 4, 5 e 6 da rede *Magola9*.

Por outro lado, os casos nos quais a variação do $e_{\max}-e_{\min}$ induz um aumento dos valores do mesmo (casos 1, 2, 7 e 8), Figura 4.59, a sensibilidade é alta, ou seja, a resposta do modelo validado com os valores modificados varia muito em relação à resposta do modelo validado com os valores originais. A Figura 4.62 mostra que para os casos em questão há uma diminuição na proporção dos valores pequenos do $e_{\max}-e_{\min}$, situação que induz variações grandes na resposta de simulação do modelo.

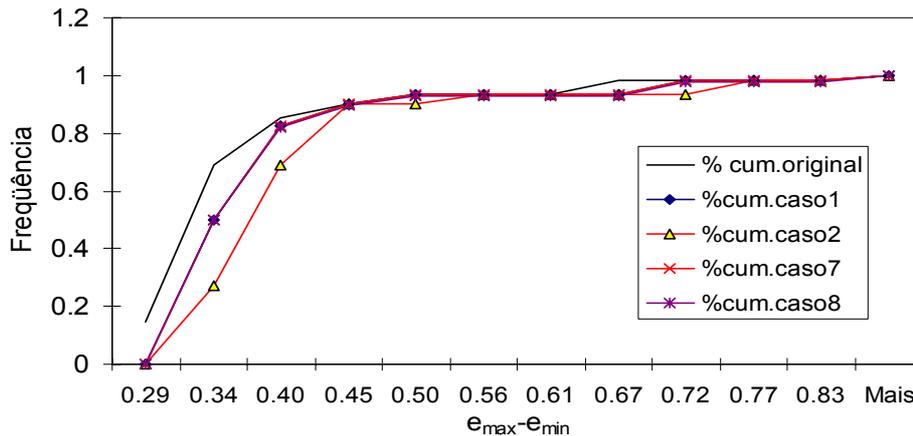


Figura 4.62. Curvas de frequência acumulada do $e_{\max} - e_{\min}$ para os casos 1, 2, 7 e 8 da rede *Magola9*.

O $e_{\max} - e_{\min}$ é um parâmetro do material indicativo das propriedades inerentes da areia tais como a distribuição granulométrica, conteúdo de finos, forma e empacotamento dos grãos, e conseqüentemente, ele fornece informação única sobre o comportamento mecânico do solo. Sendo assim, e considerando adicionalmente, que os valores de treinamento deste parâmetro estão estreitamente concentrados ao redor de um valor médio pequeno (ver histograma da Figura 4.58), pode-se explicar a alta sensibilidade do modelo nos casos para os quais a proporção de valores pequenos diminuiu em relação ao caso original.

4.8 – MODELO NEURAL 5

Visando entender os fatores que ocasionaram desempenho deficiente do modelo para algumas situações particulares, foram realizadas análises adicionais para cada uma das variáveis de entrada usadas no treinamento.

Em primeiro lugar foi observado que, como era de esperar, a rede simulou com maior precisão aqueles testes que estavam mais representados na base de dados como os correspondentes às areias Hostun e Karlsruhe, e as trajetórias do tipo CTC. Com o intuito de melhorar o desempenho da rede para aqueles testes menos representados e para as trajetórias mais complexas, a base de dados foi reorganizada de tal forma que os dados correspondentes a essas trajetórias, ocupassem as primeiras posições, enquanto que os dados correspondentes às trajetórias mais simples ou às areias mais representadas, ocupassem as últimas posições dentro da base de dados. Em conseqüência, quando o processo de treinamento for retomado

usando a técnica de subdivisão da base de dados apresentada na seção anterior, os primeiros lotes de informação conterão precisamente os dados mais complexos e por conseguinte serão apresentados ao longo de todo o processo de treinamento.

O critério de reorganização da base de dados segundo o tipo de trajetória é o seguinte:

- Primeiro são apresentados os testes com trajetórias no plano octaédrico na seguinte ordem: $\theta=9^\circ$, $\theta=12^\circ$, $\theta=15^\circ$, $\theta=30^\circ$, $\theta=36^\circ$, $\theta=45^\circ$, $\theta=54^\circ$.
- A seguir os testes com trajetórias de redução e de deformação plana: RTC, DP, RTE.
- Posteriormente os testes com trajetórias CTE, HC e TE.
- Finalmente os testes de compressão triaxial, TC, e convencionais, CTC.

4.8.1 - PRÉ-PROCESSAMENTO DA BASE DE DADOS

Algumas modificações foram incluídas nos parâmetros de identificação do tipo de solo, a saber:

- O fator de forma foi modificado segundo a Eq. (4.11) apresentada na Figura 4.25, ou seja, se manteve igual ao modelo anterior.
- O $e_{\max}-e_{\min}$ foi transformado para realçar os valores pequenos e manter os valores grandes na sua proporção original. A Figura 4.63 apresenta a transformação aplicada, a qual difere ligeiramente da transformação do modelo anterior no valor do intercepto para $e_{\max}-e_{\min}=0$. Neste caso a transformação foi deslocada para a esquerda uma unidade visando evitar a ocorrência de valores transformados muito próximos de zero ou inclusive nulos, quando ocorrerem pequenas variações do valor original perto do seu extremo inferior. A equação de transformação é:

$$(e_{\max} - e_{\min})_{\text{mod}} = 7(e_{\max} - e_{\min})^3 - 15(e_{\max} - e_{\min})^2 + 11(e_{\max} - e_{\min}) - 1 \quad (4.12)$$

- O índice de vazios inicial, e_0 , foi igualmente transformado usando a mesma equação do $e_{\max}-e_{\min}$. O propósito é o mesmo, ou seja, realçar o contraste entre os valores menores deste parâmetro, mantendo o contraste original para os valores maiores.

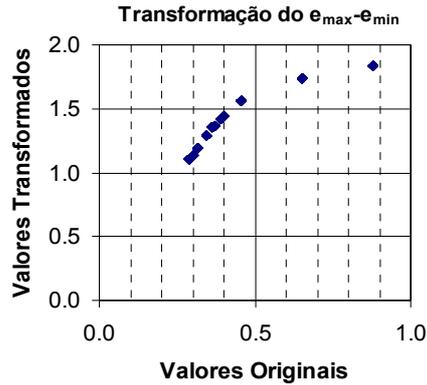


Figura 4.63. Transformação do $e_{\max}-e_{\min}$ para o modelo neural 5.

- A Figura 4.64 apresenta as curvas de frequência acumulada dos valores das tensões principais e da tensão de confinamento inicial usadas na base de dados. Observa-se uma alta concentração de valores no início de cada uma delas, indicando uma grande quantidade de valores pequenos em comparação com a quantidade de valores grandes. Numa tentativa por equilibrar um pouco esta distribuição foi aplicada uma transformação nas tensões, tal como se apresenta nas equações seguintes, obtidas empiricamente:

$$(\sigma_{c0})_{\text{mod}} = -2E^{-6}(\sigma_{c0})^2 + 0,0185(\sigma_{c0}) + 1,1132 \quad (4.13)$$

$$(\sigma)_{\text{mod}} = -7E^{-9}(\sigma)^2 + 0,0017(\sigma) + 0,8176 \quad (4.14)$$

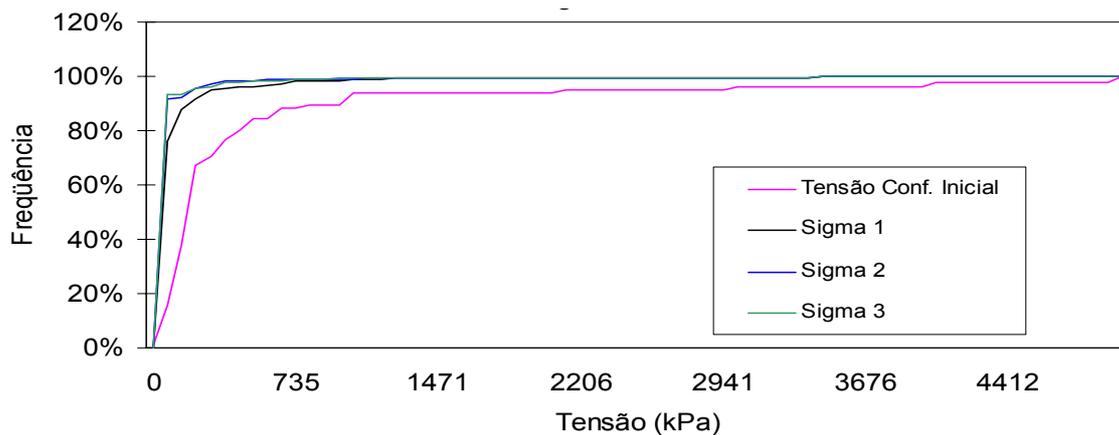


Figura 4.64. Curvas de frequência acumulada para os valores das tensões principais e a tensão de confinamento inicial da base de dados do modelo neural 5.

4.8.2 – ARQUITETURA E TREINAMENTO

Uma nova rede do tipo PMC 90x90, chamada *Magola10*, foi criada e treinada com a base de dados modificada, composta como já foi mencionado, por 4698 vetores correspondentes a 109 testes. Para efeitos de validação foram reservados 416 vetores correspondentes a 13 testes. A arquitetura desta rede é similar à usada no modelo anterior, a única diferença entre ambos os modelos consiste no pré-processamento da base de dados.

Visando facilitar o processo de aprendizado, a base de dados de treinamento foi fragmentada em 4 porções consecutivas, cada uma das quais incorpora os dados da anterior além de dados novos. A rede neural foi treinada então com as diferentes sub-bases de dados de maneira sucessiva e usando o modo “por bloco”, alternando periodicamente procedimentos de treinamento e validação. A Figura 4.65 apresenta a evolução do erro de treinamento da rede *Magola10*. Observa-se que neste caso os blocos foram apresentados sucessivamente até a rede atingir um erro aceitável.

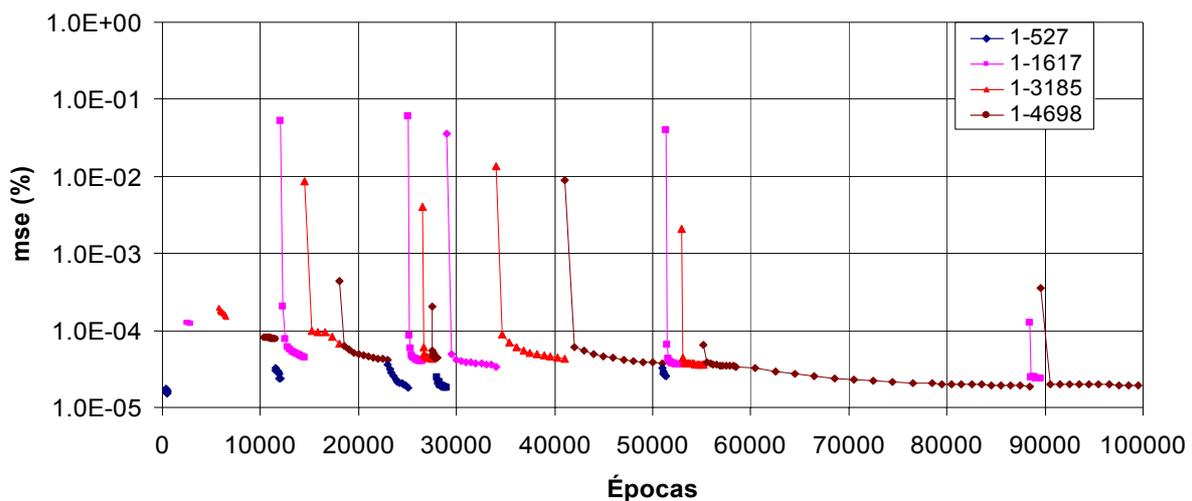


Figura 4.65. Evolução do erro de treinamento da rede *Magola10*.

4.8.3 – RESULTADOS DA SIMULAÇÃO

Uma vez terminado o processo de treinamento a rede foi simulada de maneira autônoma para a totalidade dos testes. Os valores das tensões fornecidas pela rede são comparados com os valores experimentais, e os correspondentes coeficientes de correlação são obtidos para determinar a precisão da aproximação do modelo. Estes resultados são apresentados nos gráficos da Figura 4.66.

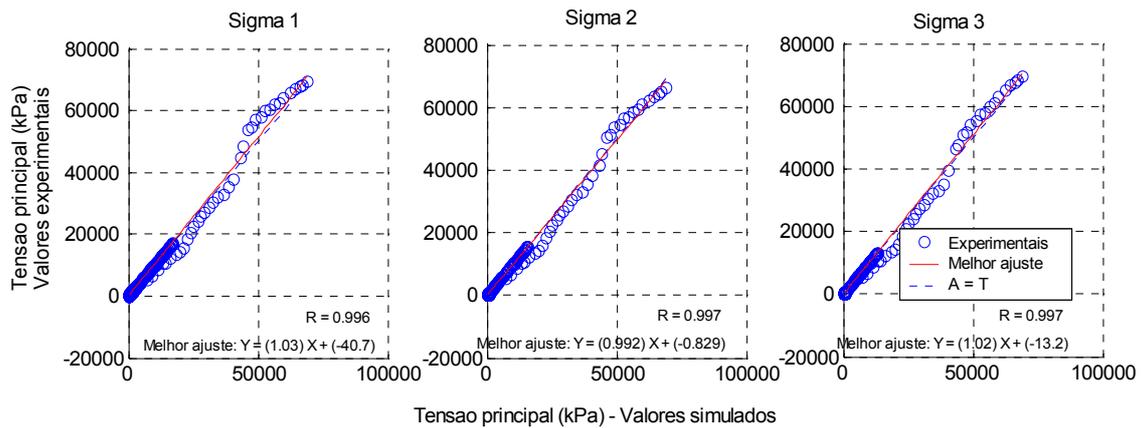


Figura 4.66. Resultados de simulação incremental das tensões para o modelo *Magola10*.

Observa-se que este modelo apresentou coeficientes de correlação ligeiramente maiores que os correspondentes para o modelo anterior. No entanto, o desempenho do modelo atual para valores altos de tensão foi melhor, não havendo-se observado dispersão considerável para os valores das tensões relativas aos extremos superiores dos intervalos correspondentes, tal como foi observado no modelo *Magola9* (Figura 4.29).

A seguir são apresentados os gráficos tensão-deformação para cada um dos testes, simulados de maneira autônoma com a rede *Magola10*. Em cada caso são apresentados adicionalmente os gráficos da tensão octaédrica (τ_{oct}) e de tensão desviadora (σ_d) versus deformação desviadora (ϵ_d). Inicialmente são apresentados os gráficos correspondentes à areia Karlsruhe para testes tipo CTC (Figura 4.67), observando-se boa aproximação entre os valores de simulação (linhas contínuas) e os valores experimentais (pontos).

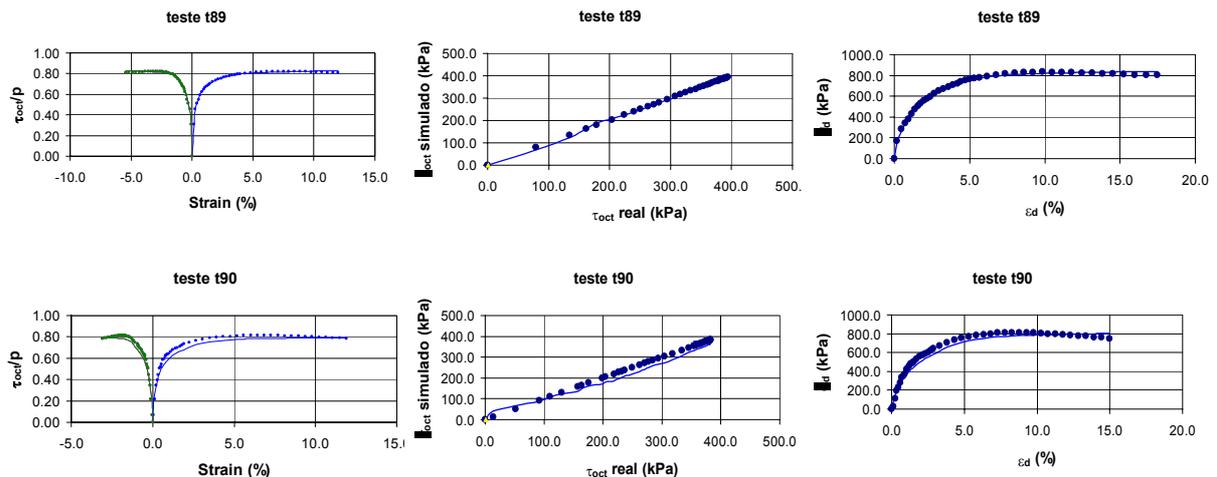


Figura 4.67. Gráficos tensão-deformação para a areia Karlsruhe. Testes CTC

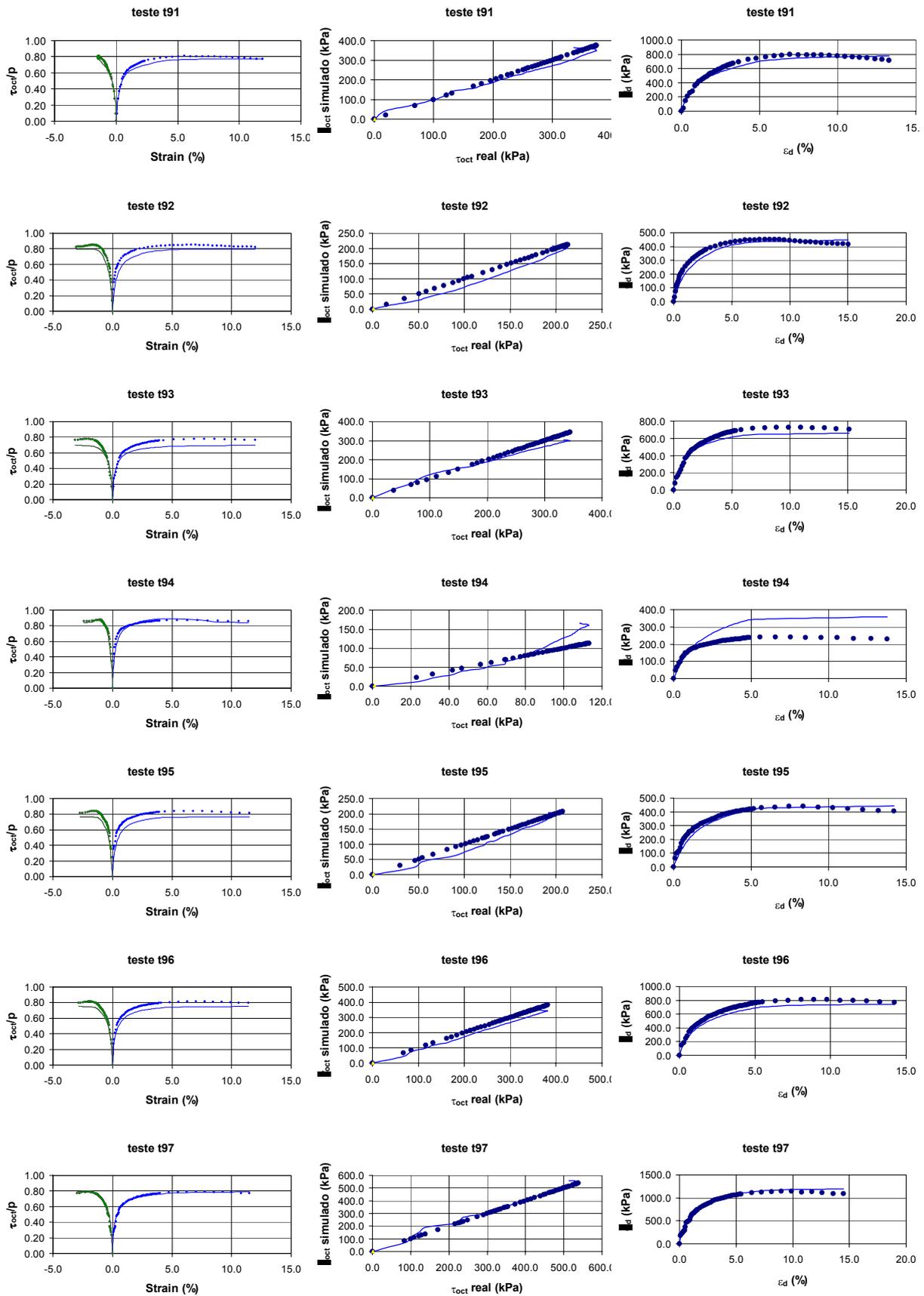


Figura 4.67.(Continuação) Gráficos tensão-deformação para a areia Karlsruhe. Testes CTC

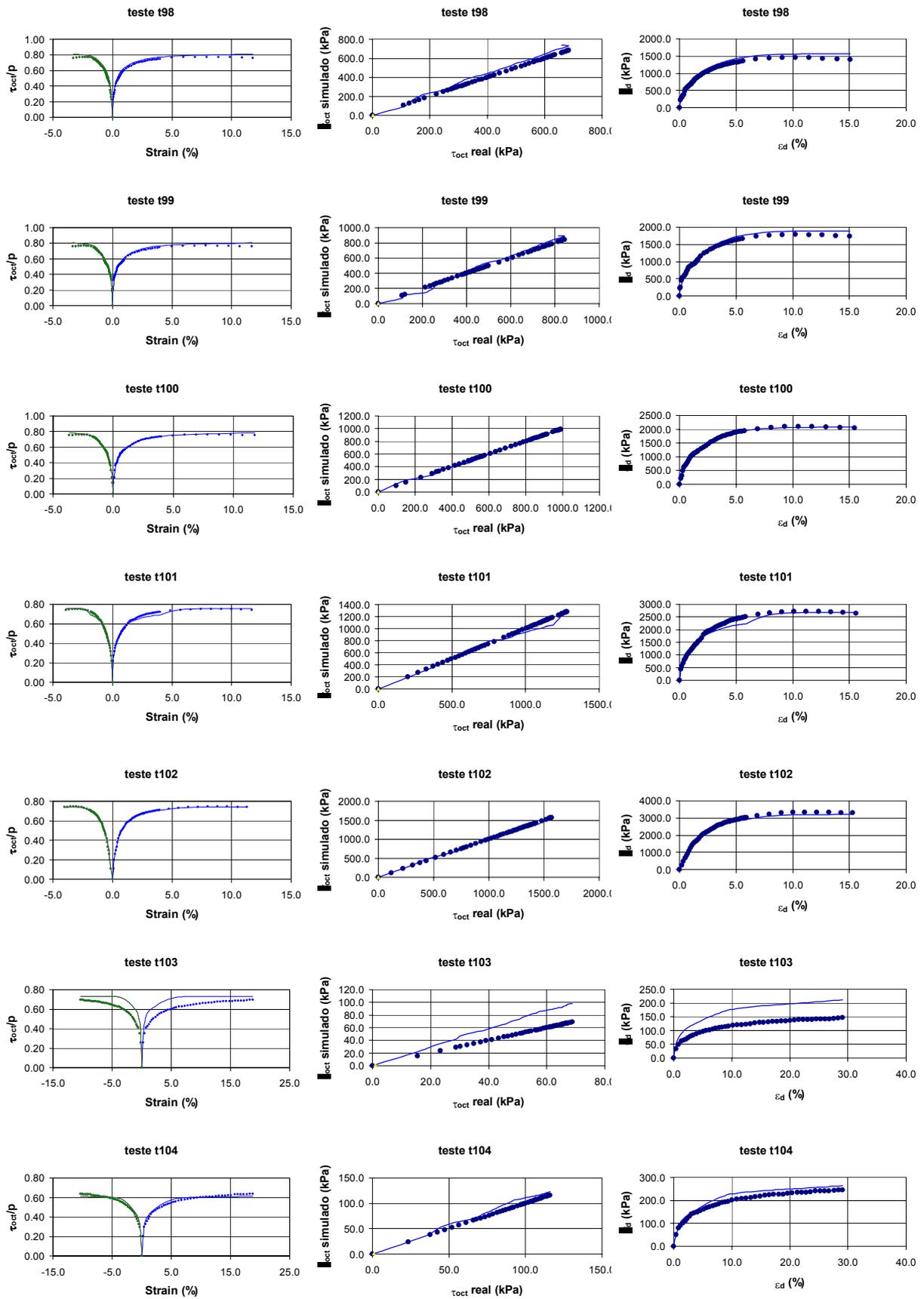


Figura 4.67.(Continuação) Gráficos tensão-deformação para a areia Karlsruhe. Testes CTC

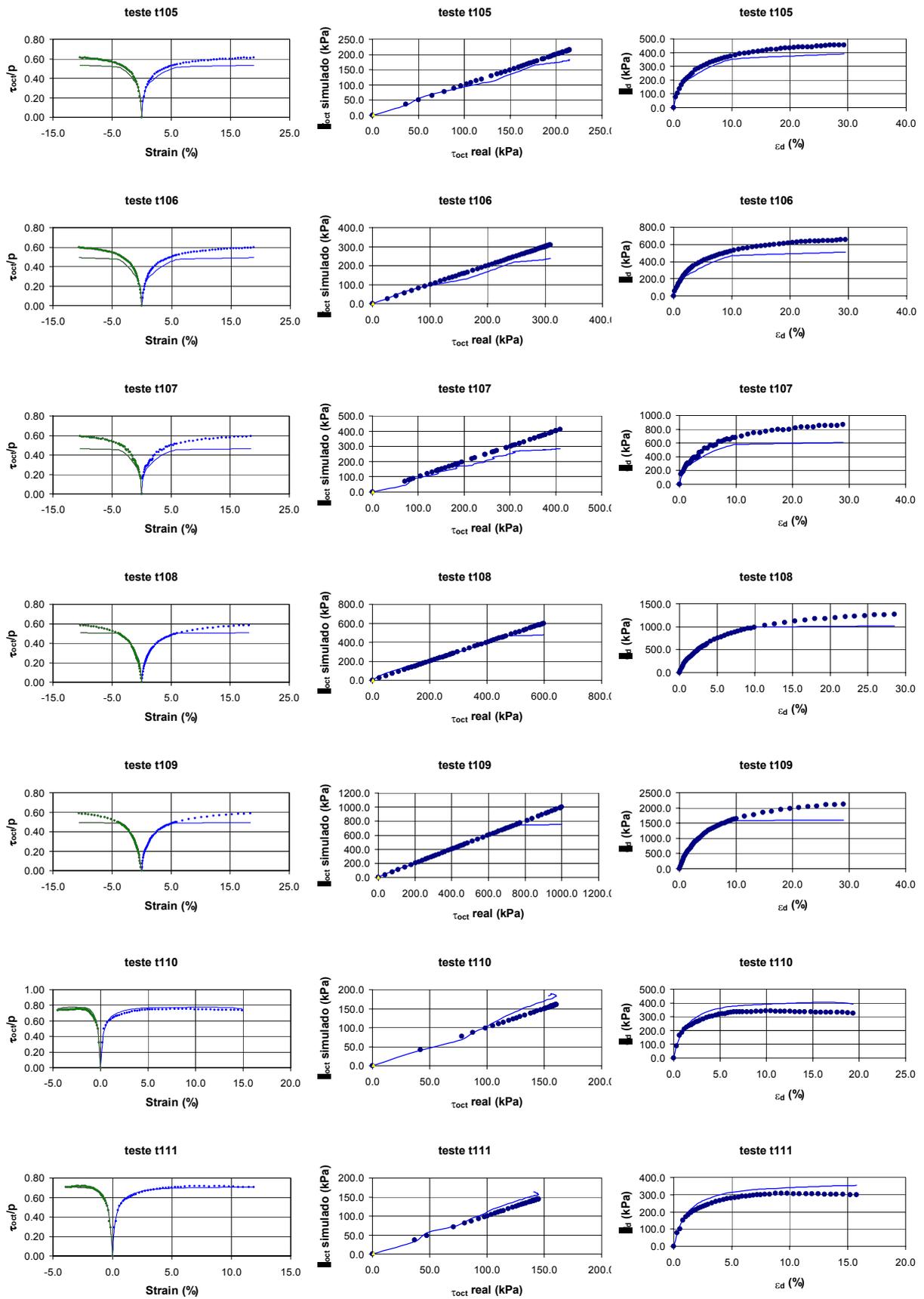


Figura 4.67.(Continuação) Gráficos tensão-deformação para a areia Karlsruhe. Testes CTC

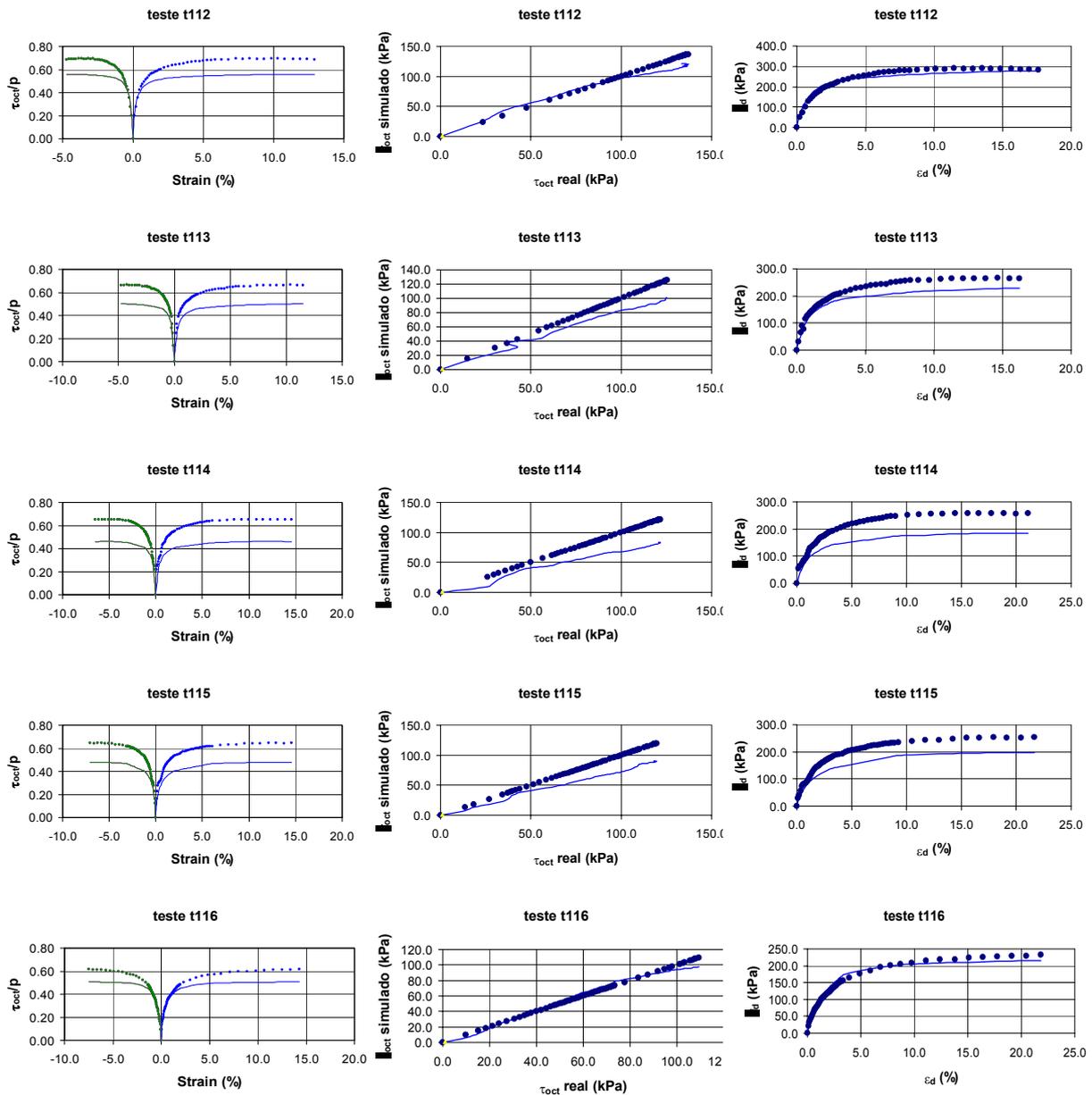


Figura 4.67.(Continuação) Gráficos tensão-deformação para a areia Karlsruhe. Testes CTC

A Figura 4.68 apresenta os resultados para os testes tipo CTC na areia de Hostun. Observa-se um desempenho fraco na simulação dos testes cíclicos (122, 123, 124) e uma definição imprecisa nas tensões de ruptura dos testes 71 e 142. O teste 84 foi eliminado por causa do péssimo desempenho não explicado da rede.

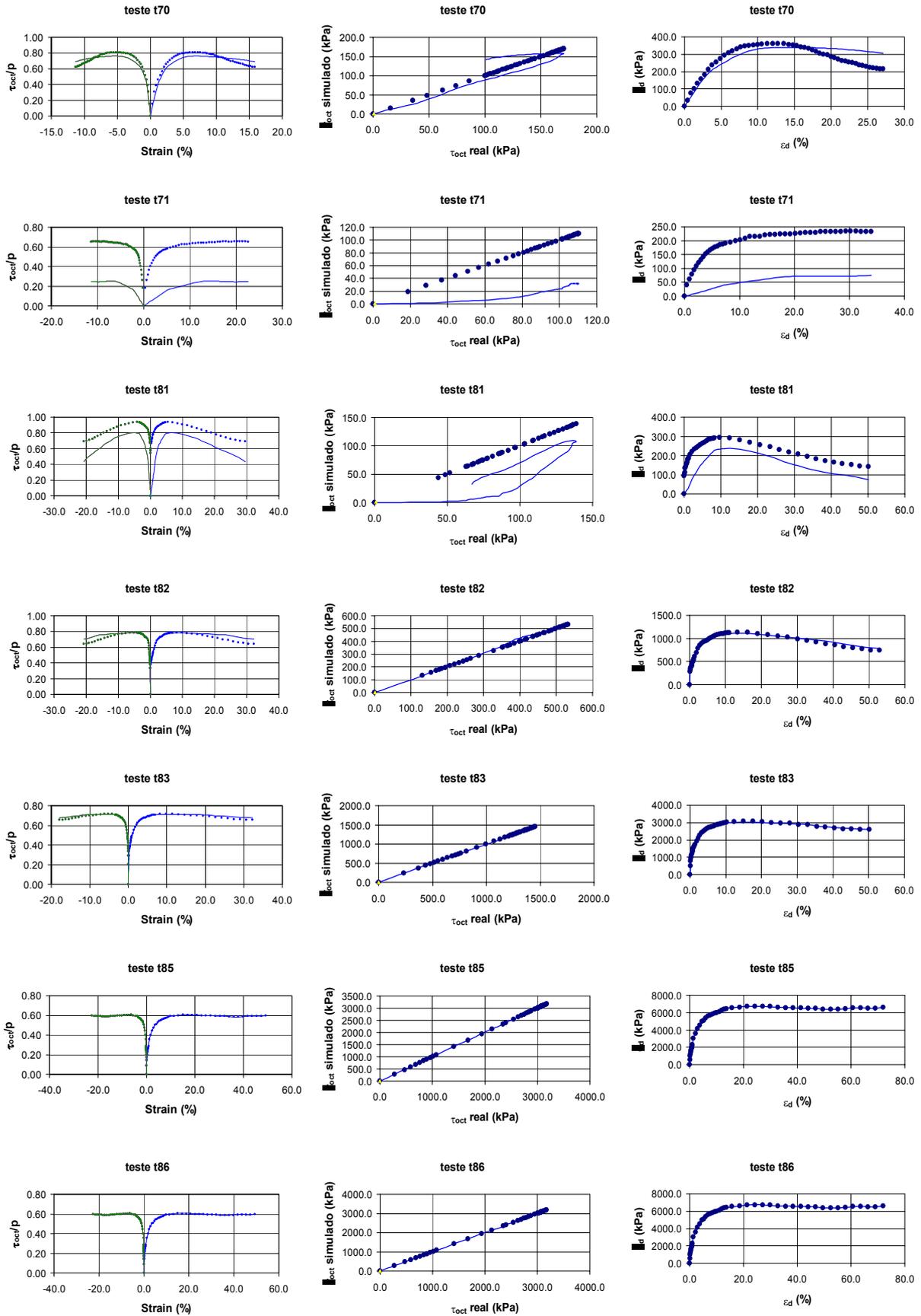


Figura 4.68. Gráficos tensão-deformação para a areia Hostun. Testes CTC

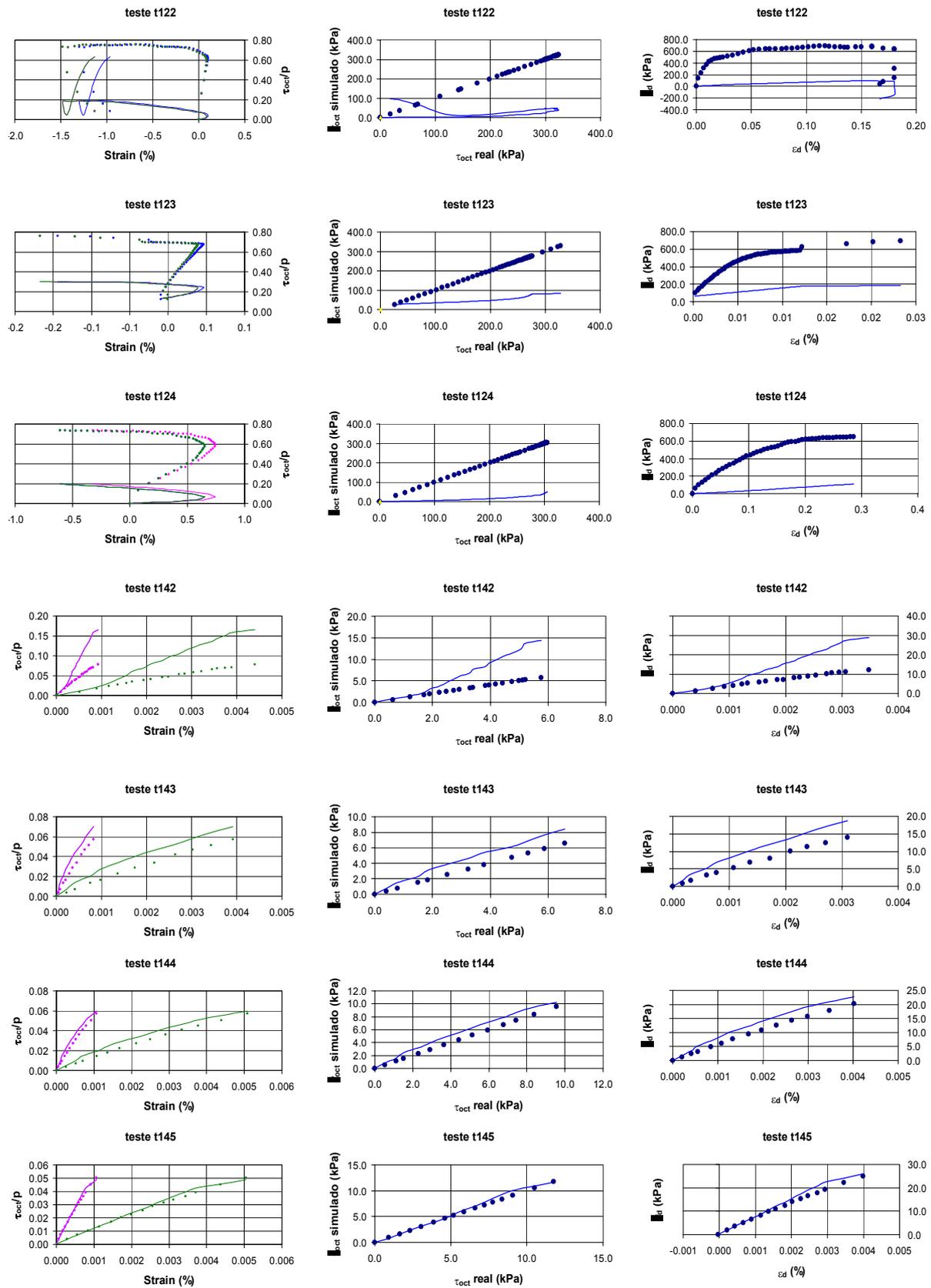


Figura 4.68. (Continuação) Gráficos tensão-deformação para a areia Hostun. Testes CTC

A Figura 4.69 apresenta os gráficos correspondentes à areia de Hostun para testes do tipo TC. Observa-se no geral uma boa aproximação na forma das curvas porém para alguns testes (117, 118 e 119) a previsão da tensão de ruptura foi muito conservadora.

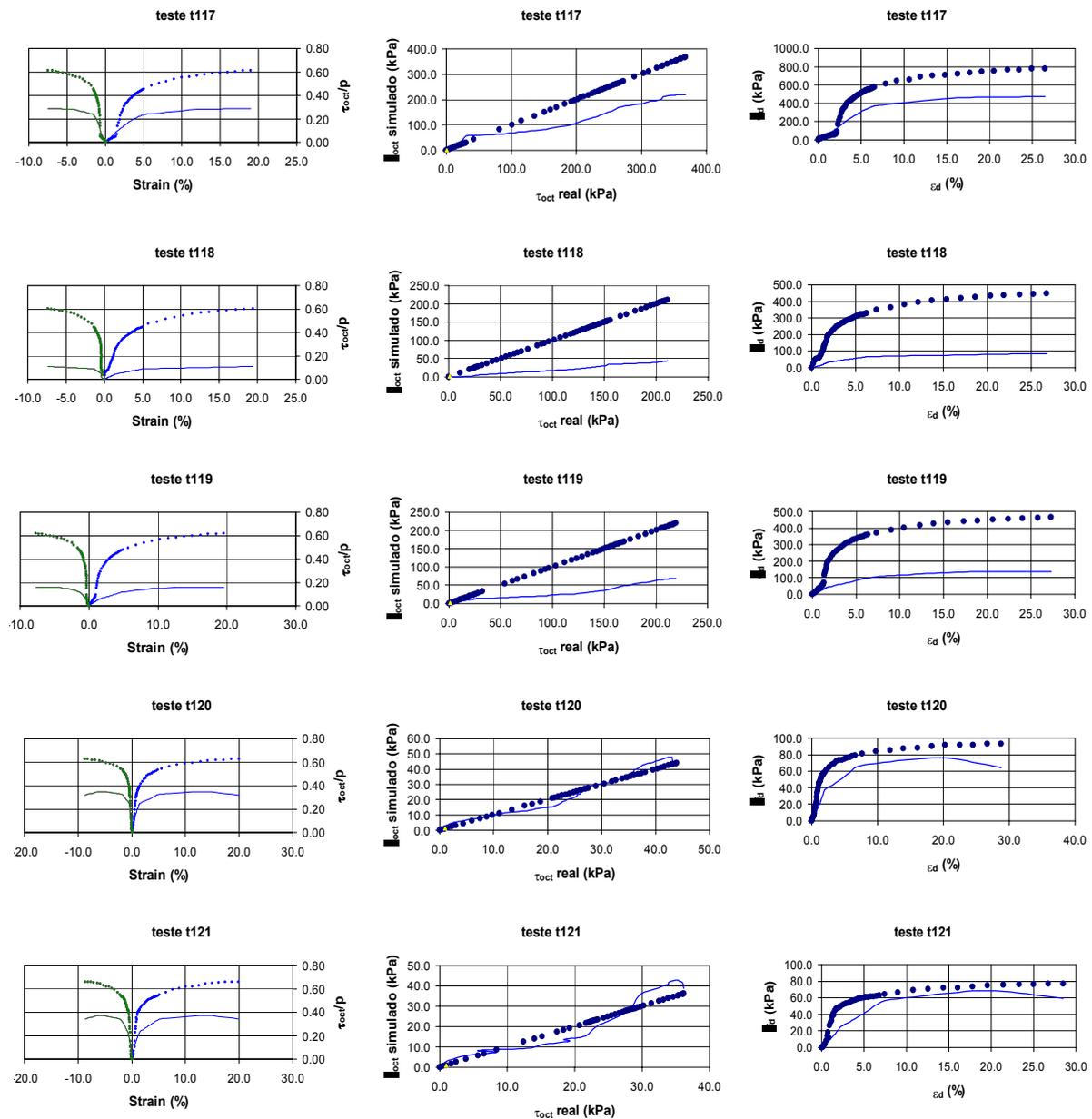


Figura 4.69. Gráficos tensão-deformação para a areia Hostun. Testes TC.

A Figura 4.70 apresenta os resultados para a areia de Hostun com testes de redução tipo RTE. Nenhum deles apresentou desempenho satisfatório quanto a rigidez, porém a rede identificou os valores de tensão de ruptura em cada um dos testes.

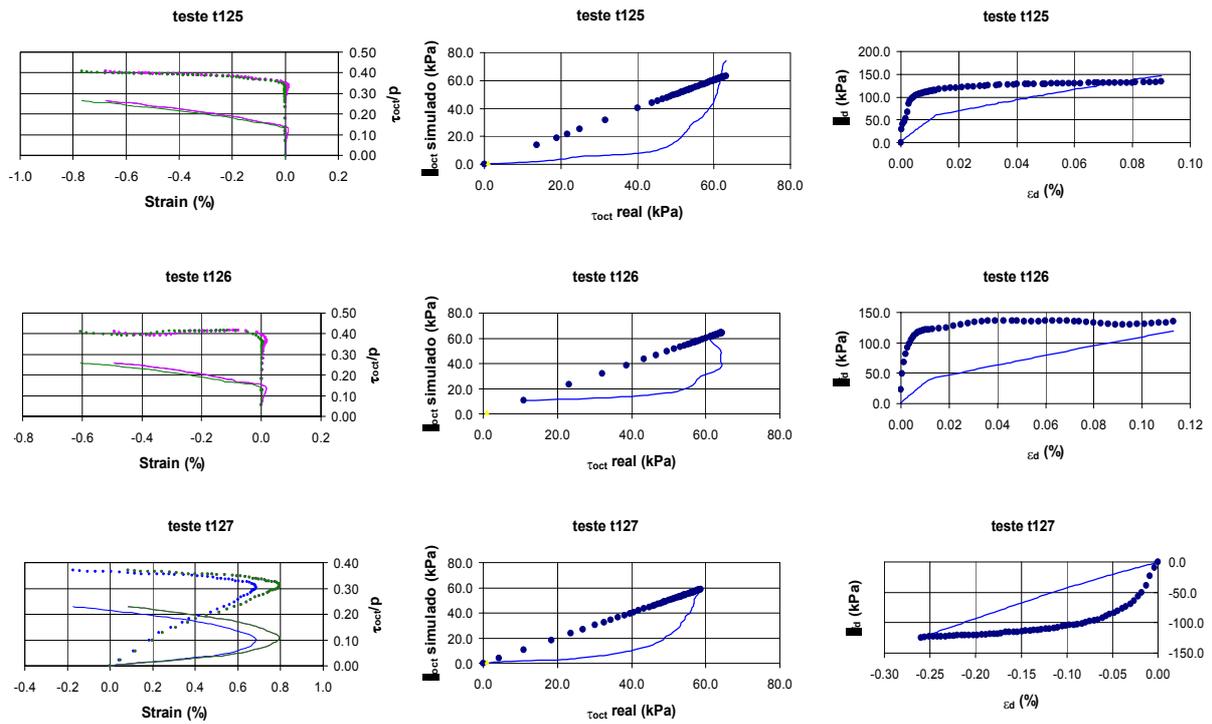


Figura 4.70. Gráficos tensão-deformação para a areia Hostun. Testes RTE.

A Figura 4.71 apresenta os resultados para a areia de Toyoura, testes CTC. Todas as simulações apresentaram boa aproximação aos dados experimentais, com exceção dos testes 22 e 26 nos quais a rede aproximou só a forma da trajetória porém falhou na previsão da tensão de ruptura.

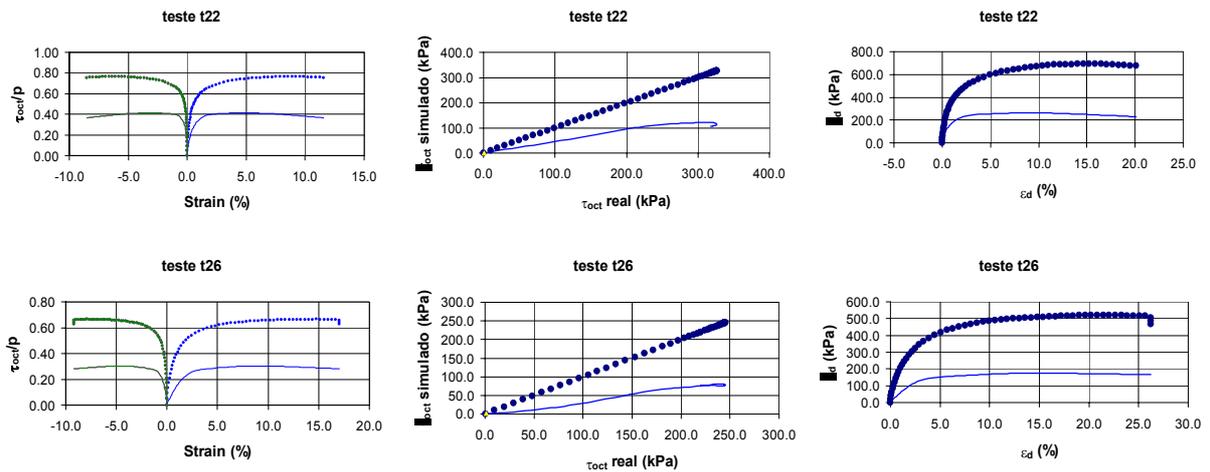


Figura 4.71. Gráficos tensão-deformação para a areia Toyoura. Testes CTC

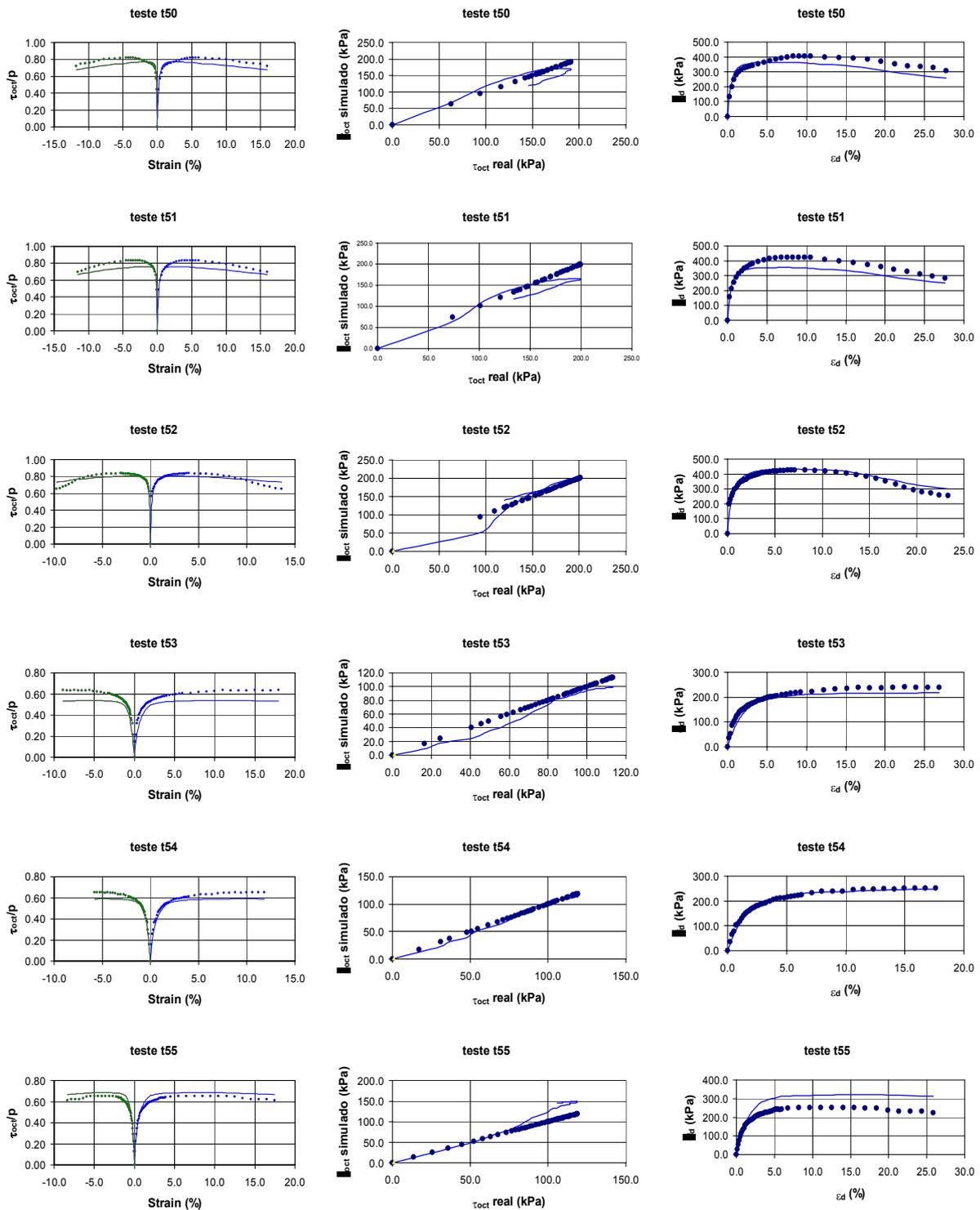


Figura 4.71. Gráficos tensão-deformação para a areia Toyoura. Testes CTC.

A Figura 4.72 apresenta os gráficos tensão-deformação para a areia Toyoura. Testes TC, CTE, TE, RTC, RTE e no plano octaédrico. Em todos eles a rede aproximou razoavelmente a forma das curvas porém falhou na determinação da tensão de ruptura.

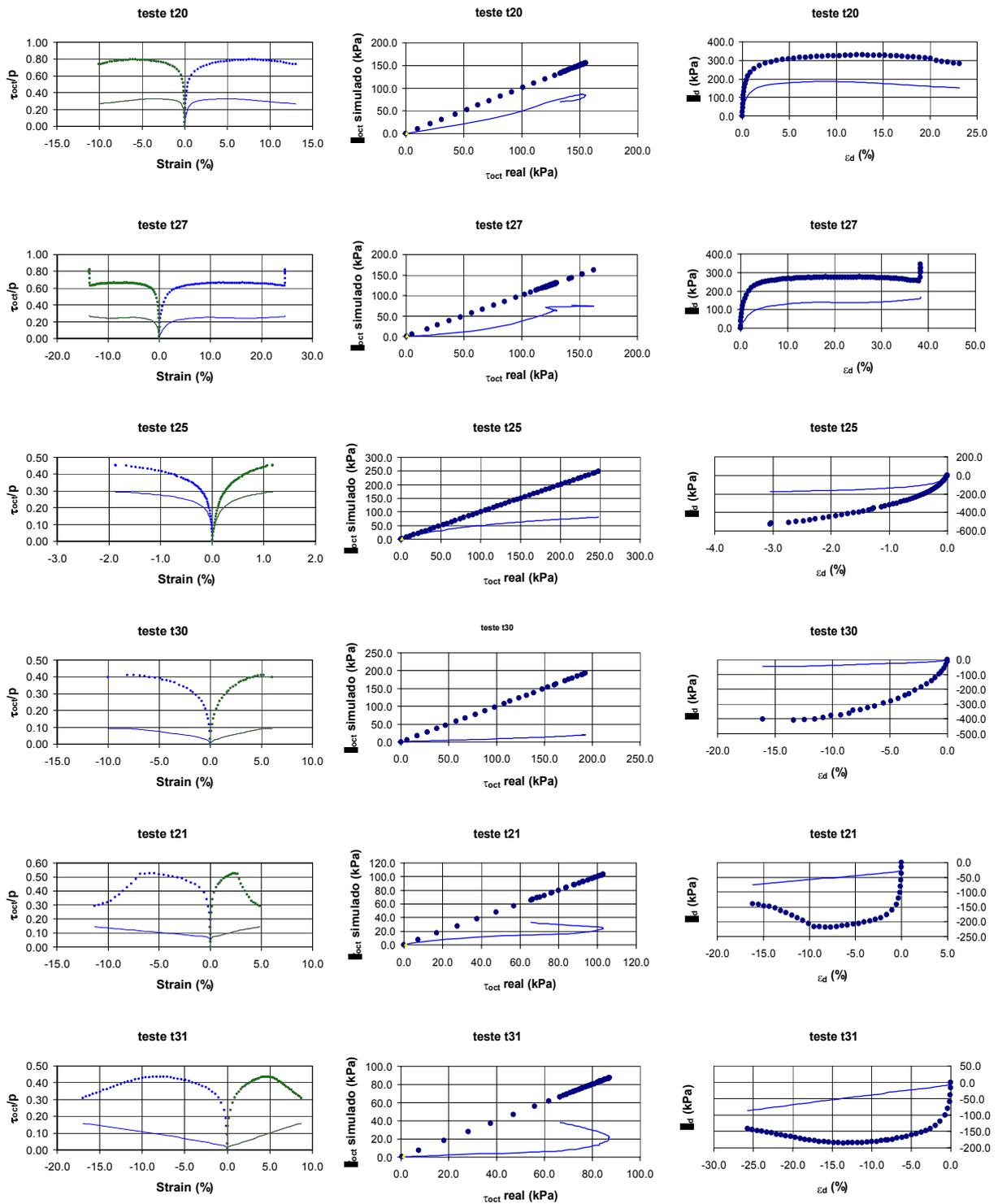


Figura 4.72. Gráficos tensão-deformação para a areia Toyoura. Testes TC, CTE, TE, RTC, RTE e no plano octaédrico.

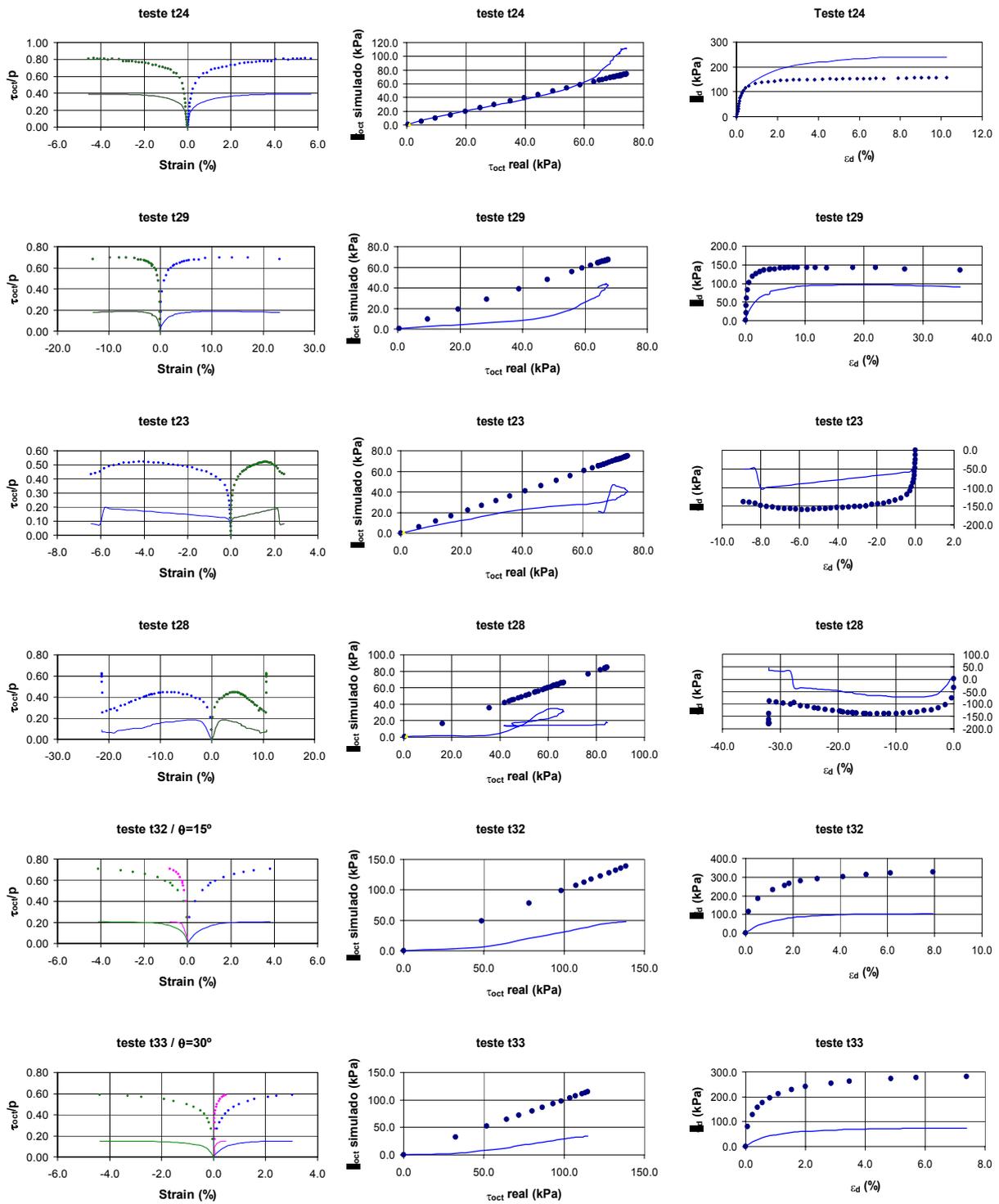


Figura 4.72. (Continuação) Gráficos tensão-deformação para a areia Toyoura. Testes TC, CTE, TE, RTC, RTE e no plano octaédrico.

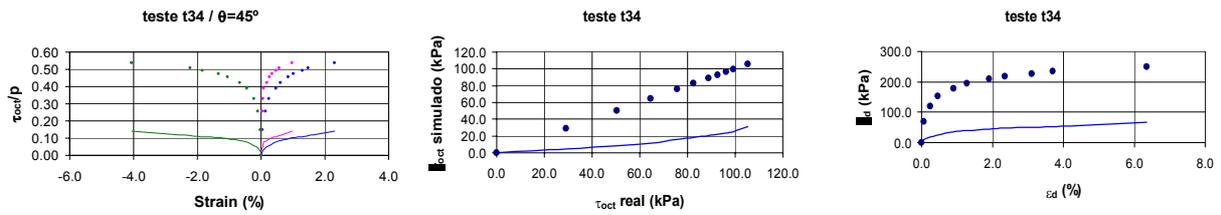


Figura 4.72. (Continuação) Gráficos tensão-deformação para a areia Toyoura. Testes TC, CTE, TE, RTC, RTE e no plano octaédrico.

A Figura 4.73 apresenta os resultados de simulação para os testes do tipo CTC e TC da areia de Cambria, todos eles apresentando boa aproximação da forma da trajetória e boa previsão da tensão de ruptura.

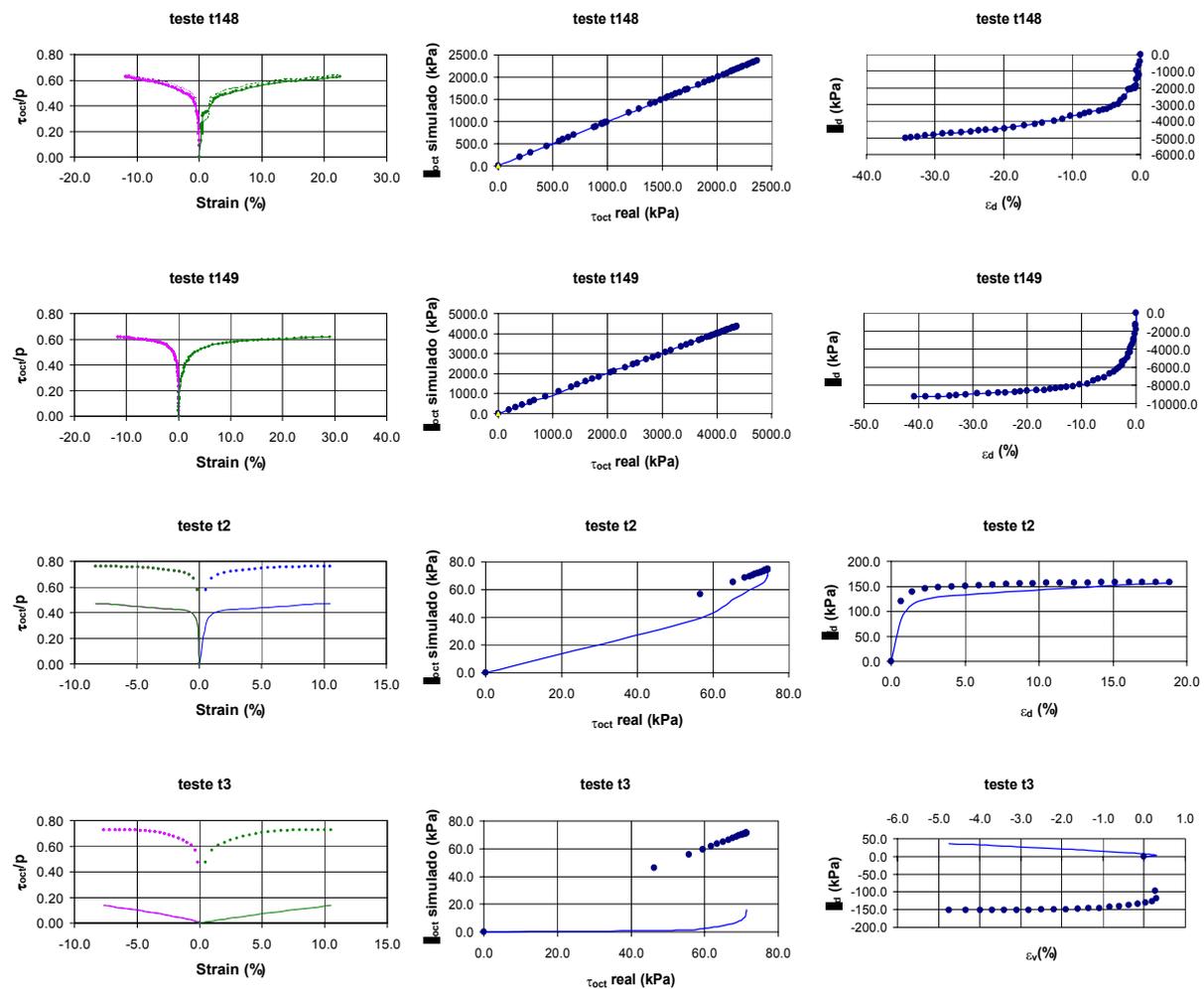


Figura 4.73. Gráficos tensão-deformação para a areia de Cambria. Testes CTC e TC.

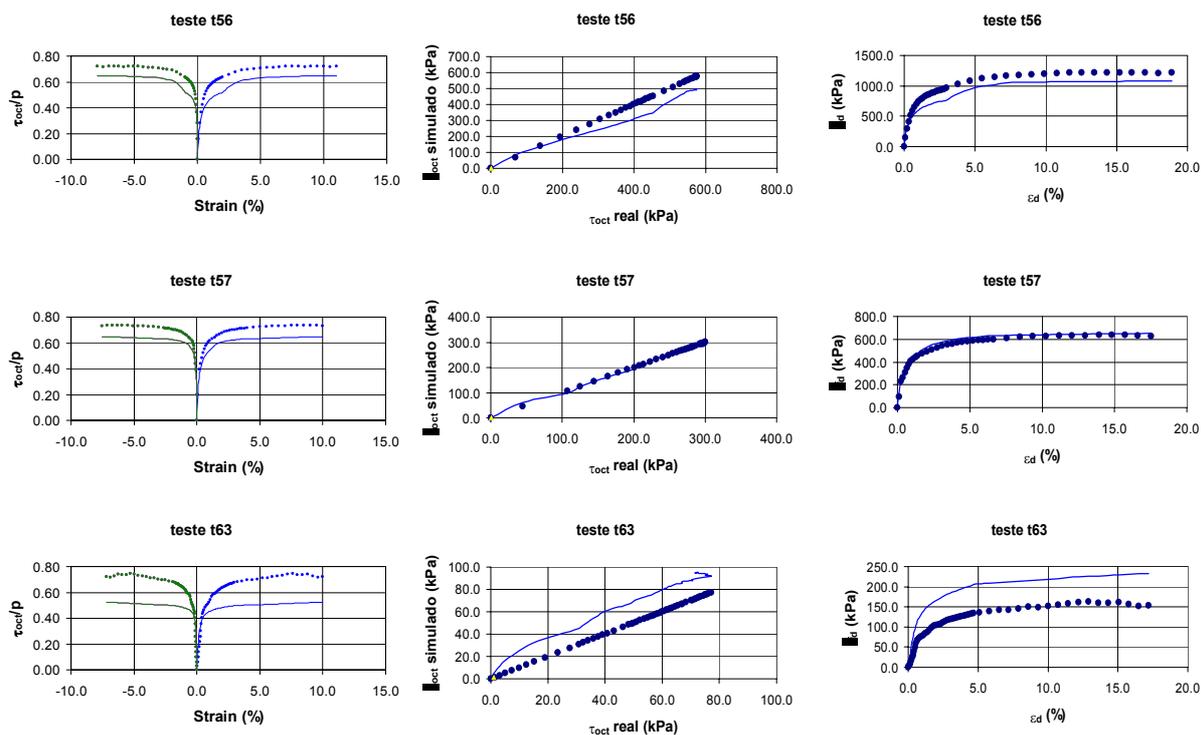


Figura 4.73. (Continuação) Gráficos tensão-deformação para a areia de Cambria. Testes CTC e TC.

A Figura 4.74 mostra os gráficos de simulação para os testes de deformação plana na areia de Cambria. Todos eles foram razoavelmente bem simulados considerando a escassa informação sobre este tipo de trajetória na base de dados de treinamento.

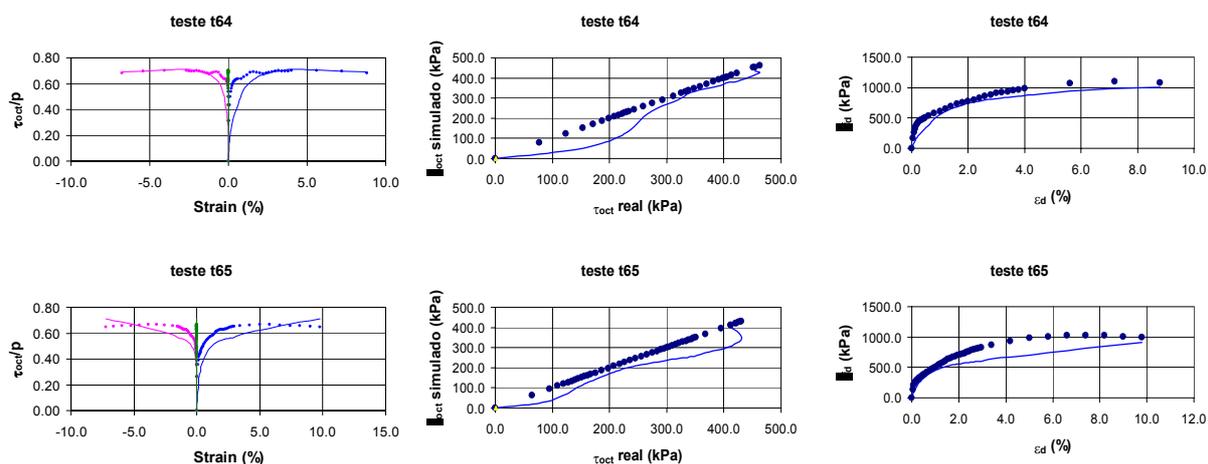


Figura 4.74. Gráficos tensão-deformação para a areia de Cambria. Testes de deformação plana

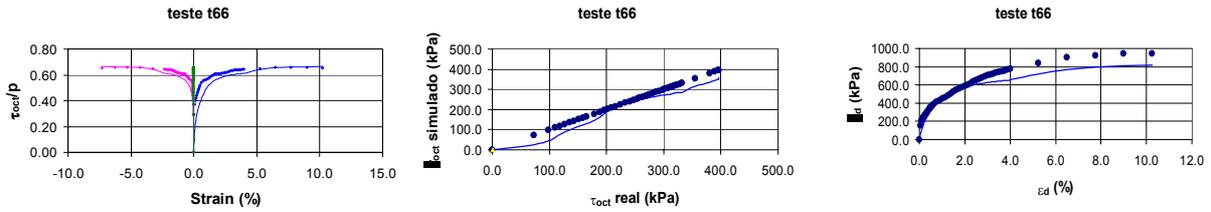


Figura 4.74. (Continuação) Gráficos tensão-deformação para a areia de Cambria. Testes de deformação plana.

A Figura 4.75 apresenta os gráficos tensão-deformação para os testes CTE e TE na areia de Cambria. Aqui se apresentou um problema típico de ajuste excessivo no teste 160. Observe-se o comportamento oscilatório da simulação para este teste que contrasta com as boas aproximações dos outros testes do mesmo tipo. Como o controle do erro de validação se faz de maneira global, é impossível evitar que para alguns casos apareça o fenômeno mencionado. O que se procura durante o processo de treinamento por lotes é a diminuição geral do erro de validação, não havendo controle individual deste erro para todos e cada um dos testes simulados.

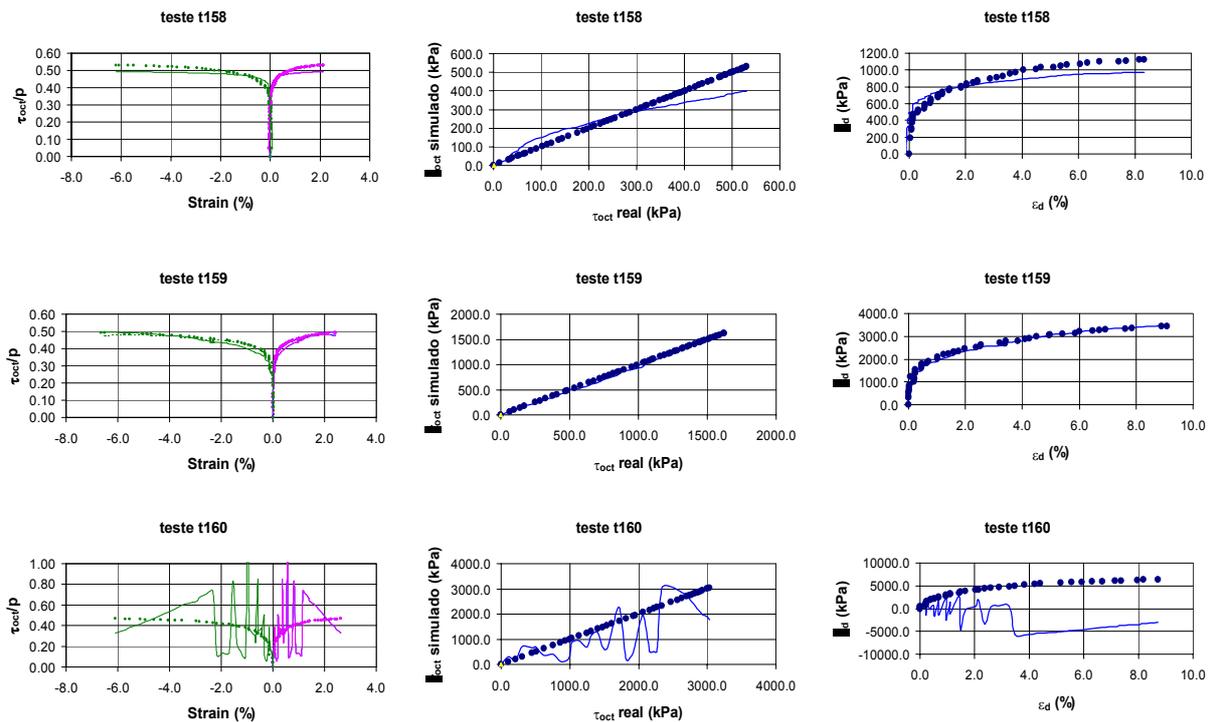


Figura 4.75. Gráficos tensão-deformação para a areia de Cambria. Testes CTE e TE.

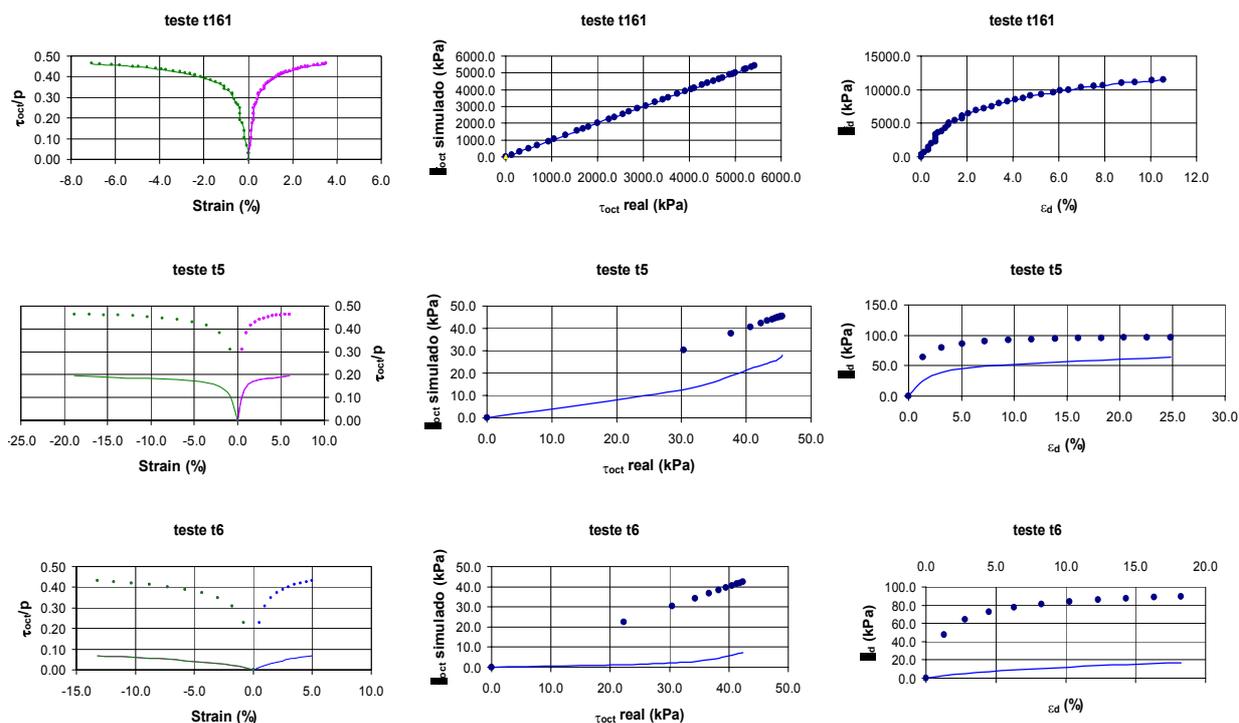


Figura 4.75. (Continuação) Gráficos tensão-deformação para a areia de Cambria. Testes CTE e TE.

Todos os teste simulados para a areia de Monterrey foram bem aproximados pelo modelo neural, razão pela qual são colocados juntos na Figura 4.76. As primeiras duas linhas correspondem aos testes TC, as duas seguintes aos testes TE e as restantes aos testes no plano octaédrico. Observe-se que só os testes 7, 18, 9, 12 e 17 apresentaram valores das tensões de ruptura claramente diferentes dos valores experimentais.

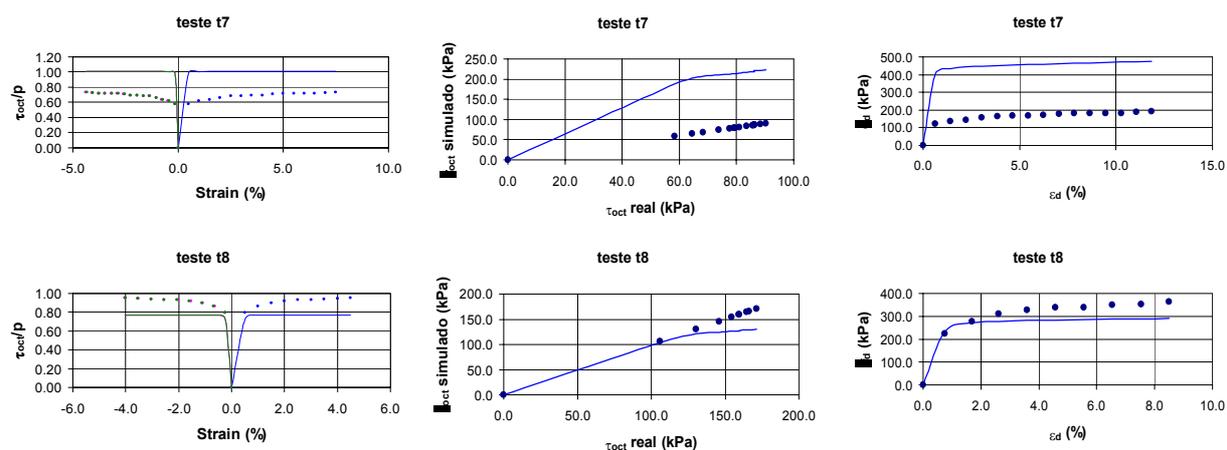


Figura 4.76. Gráficos tensão-deformação para a areia de Monterrey. Testes TC, TE e testes no plano octaédrico.

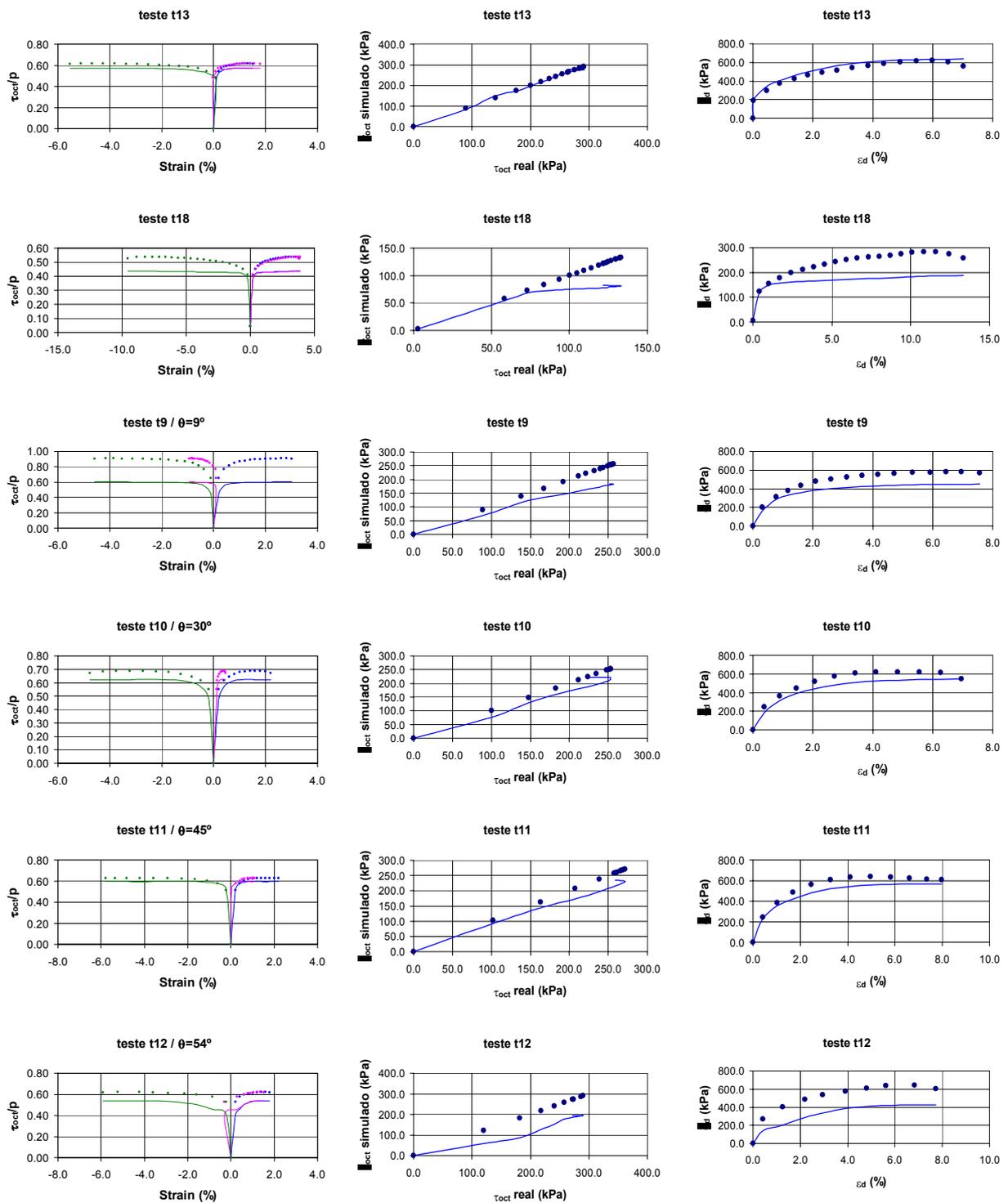


Figura 4.76. (Continuação) Gráficos tensão-deformação para a areia de Monterrey. Testes TC, TE e testes no plano octaédrico.

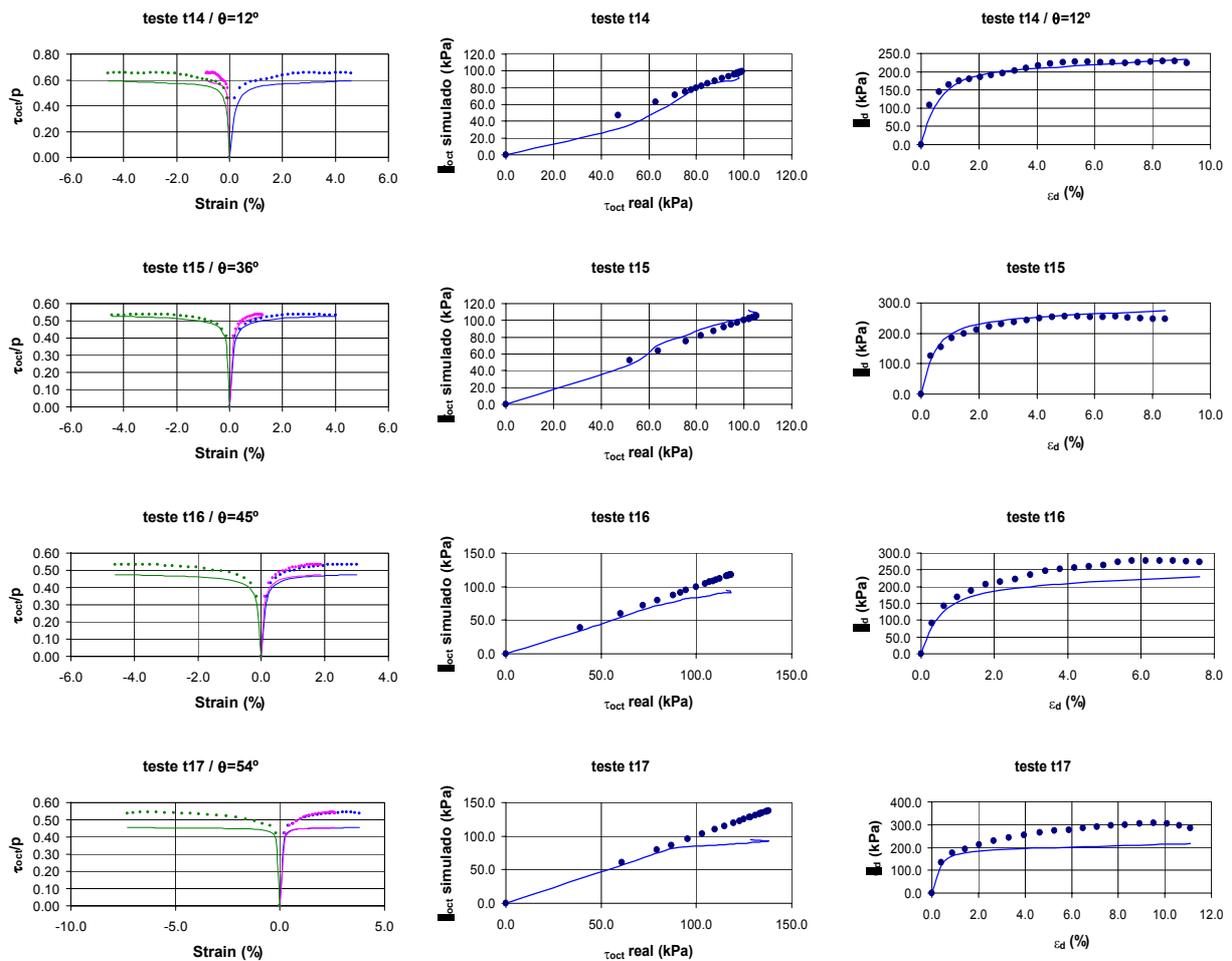


Figura 4.76. (Continuação) Gráficos tensão-deformação para a areia de Monterrey. Testes TC, TE e testes no plano octaédrico.

A partir da Figura 4.77 até a Figura 4.81 são apresentados os gráficos de resultados para os testes de tipo CTC das areias Calcária, Ottawa, Antioch, Chattahoochee e San Fernando. Observa-se em todas elas que o modelo neural aproxima satisfatoriamente a forma das curvas, e consegue uma boa previsão da tensão de ruptura. Em alguns dos testes simulados o modelo consegue ainda simular o fenômeno de amolecimento.

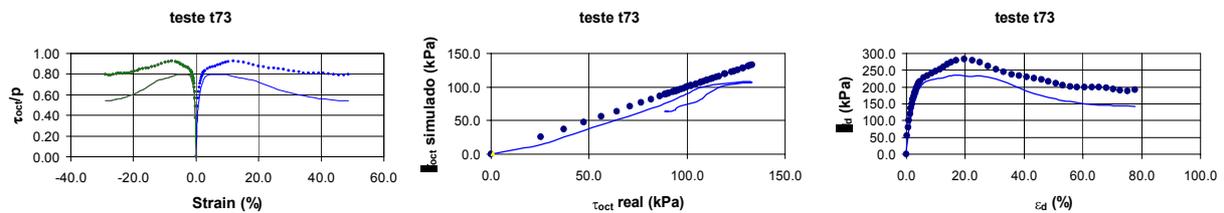


Figura 4.77. Gráficos tensão-deformação para a areia calcária. Testes CTC.

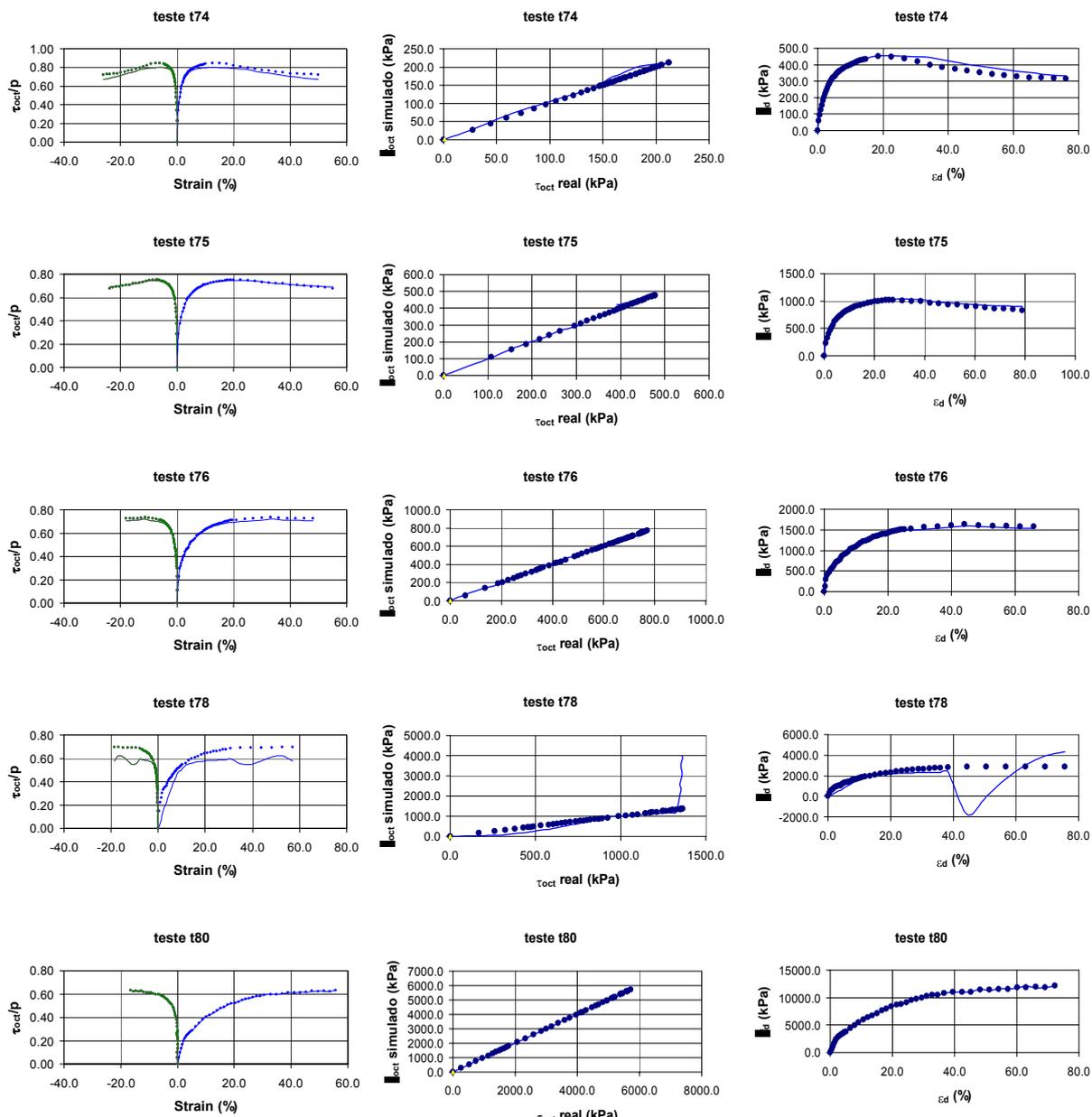


Figura 4.77. (Continuação) Gráficos tensão-deformação para a areia calcária. Testes CTC.

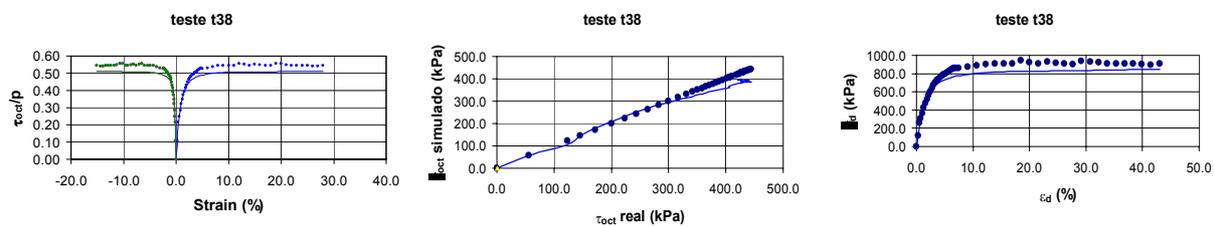


Figura 4.78. Gráficos tensão-deformação para a areia de Ottawa. Testes CTC.

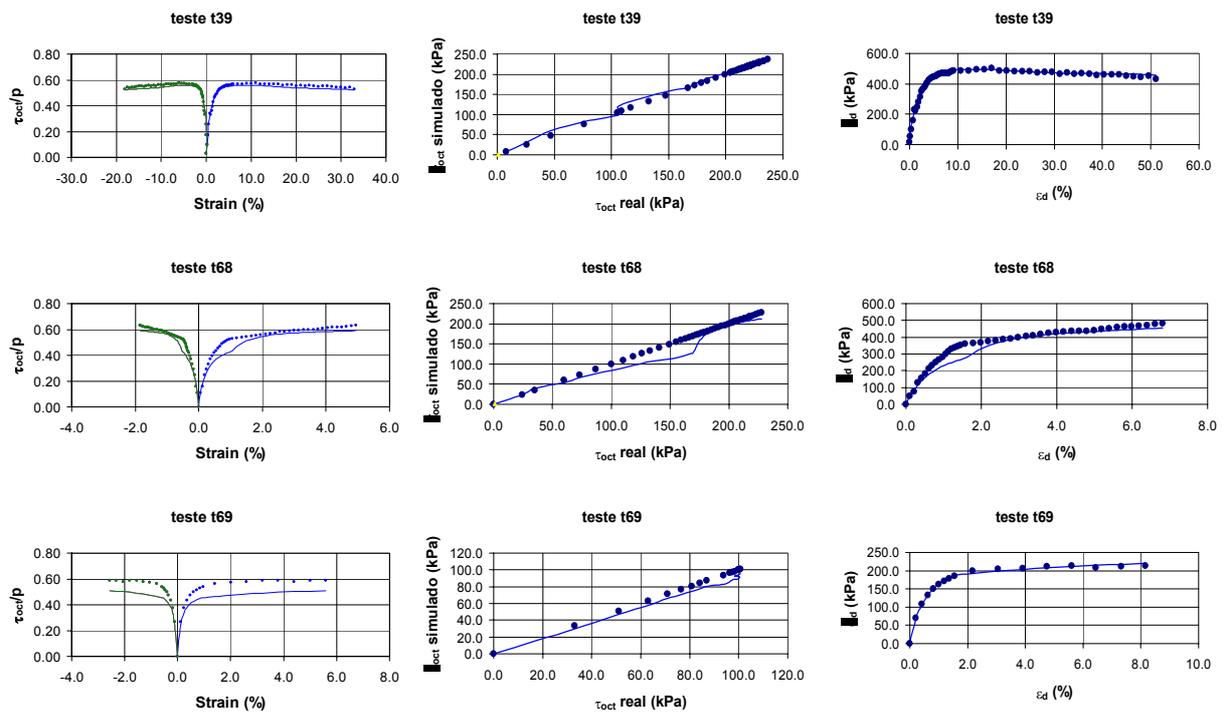


Figura 4.78. (Continuação) Gráficos tensão-deformação para a areia de Ottawa. Testes CTC.

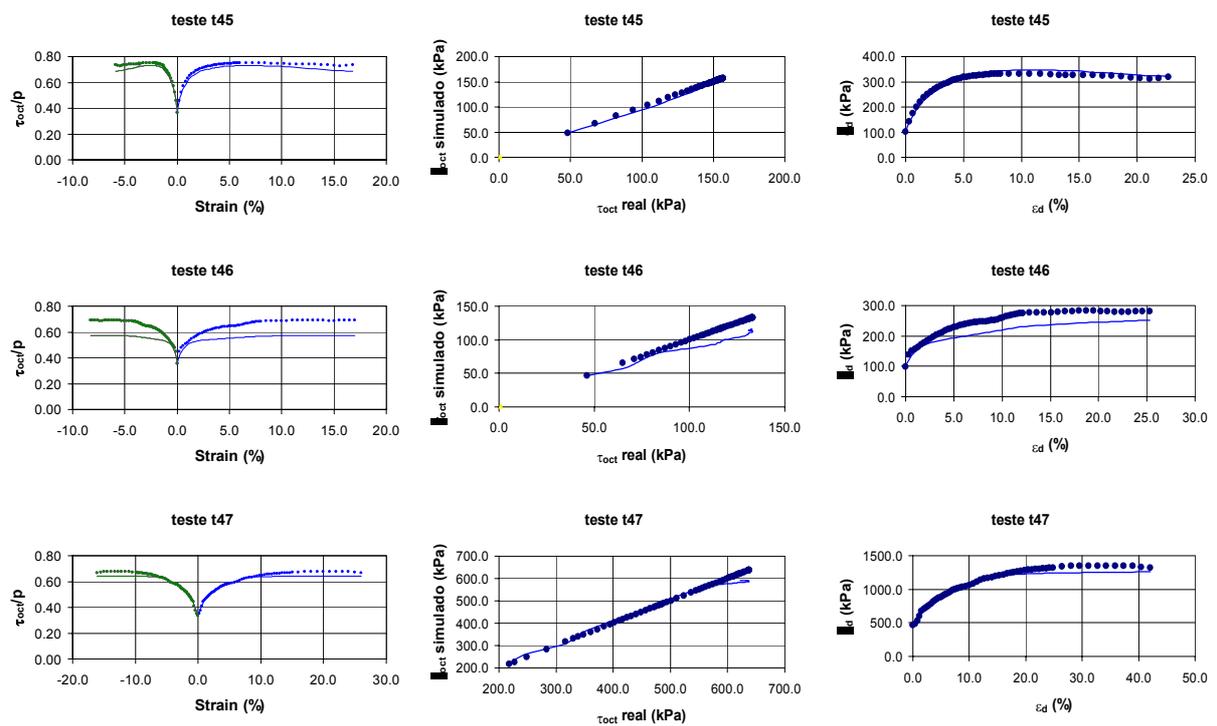


Figura 4.79. Gráficos tensão-deformação para a areia de Antioch. Testes CTC.

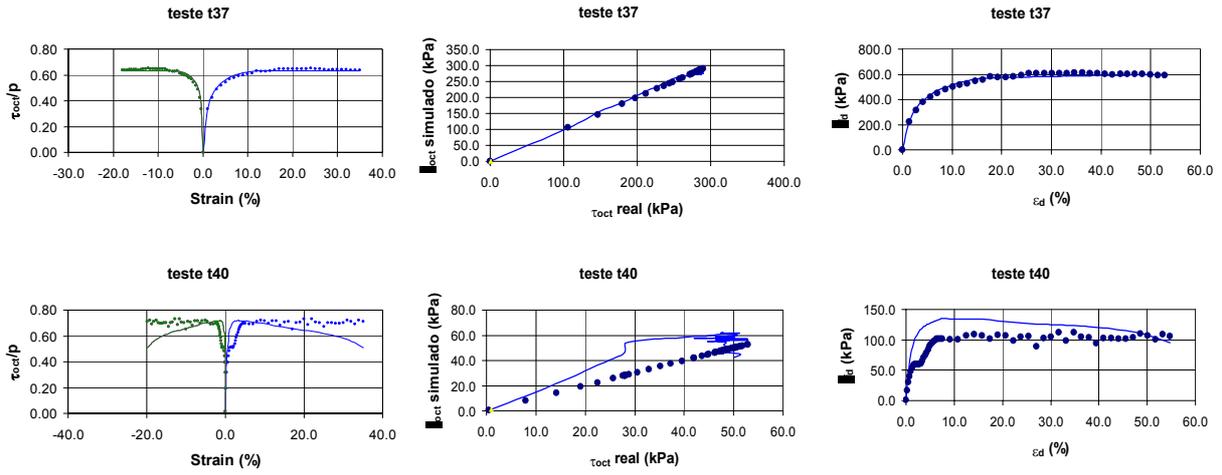


Figura 4.80. Gráficos tensão-deformação para a areia de Chatahoochee. Testes CTC.

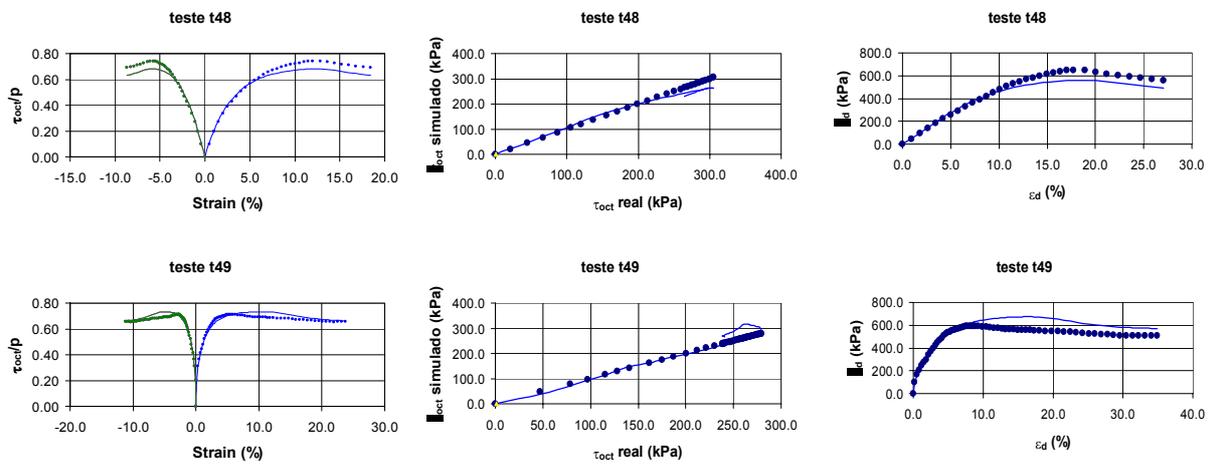


Figura 4.81. Gráficos tensão-deformação para a areia de San Fernando. Testes CTC.

A Figura 4.82 apresenta os resultados das simulações dos diferentes testes de compressão hidrostática para as areias de Toyoura, SLB, Ticino e Cambria. Todos eles razoavelmente bem aproximados pelo modelo neural. Nestes gráficos as ordenadas correspondem a tensão média (p) enquanto que as abscissas correspondem à deformação volumétrica (ϵ_v).

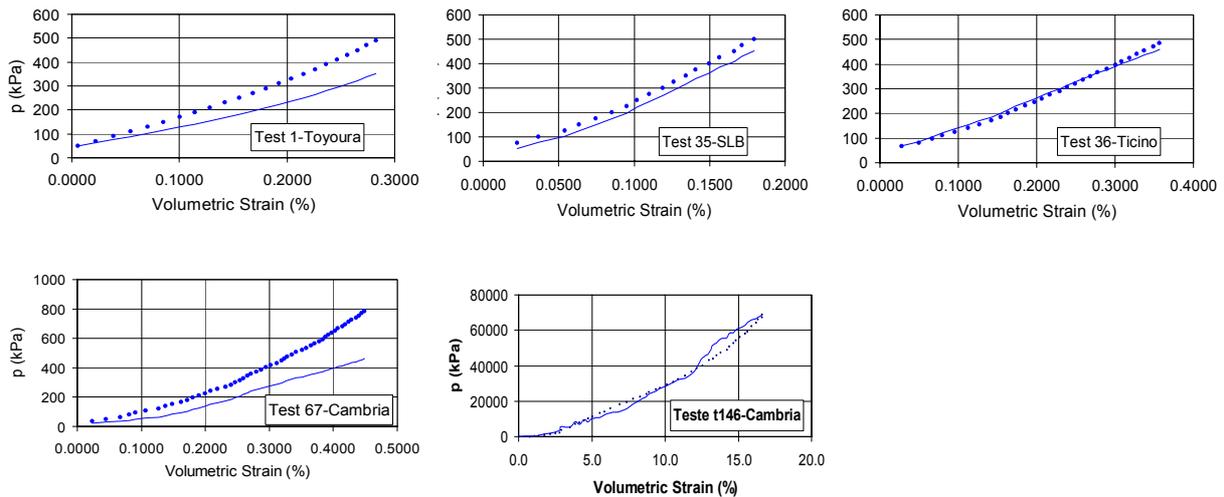


Figura 4.82. Gráficos tensão-deformação para testes de compressão hidrostática (HC) nas areias de Toyoura, SLB, Ticino e Cambria.

4.8.4 – DISCUSSÃO

O histograma da Figura 4.83 mostra a distribuição das freqüências dos erros de simulação das tensões para o modelo *Magola10*. Observa-se que para o 62,3% dos testes simulados o erro do modelo neural foi menor que 10%, e para 75,4% dos testes o erro foi menor que 20%. Estes resultados são tecnicamente iguais aos obtidos com o modelo *Magola9* (Figura 4.56).

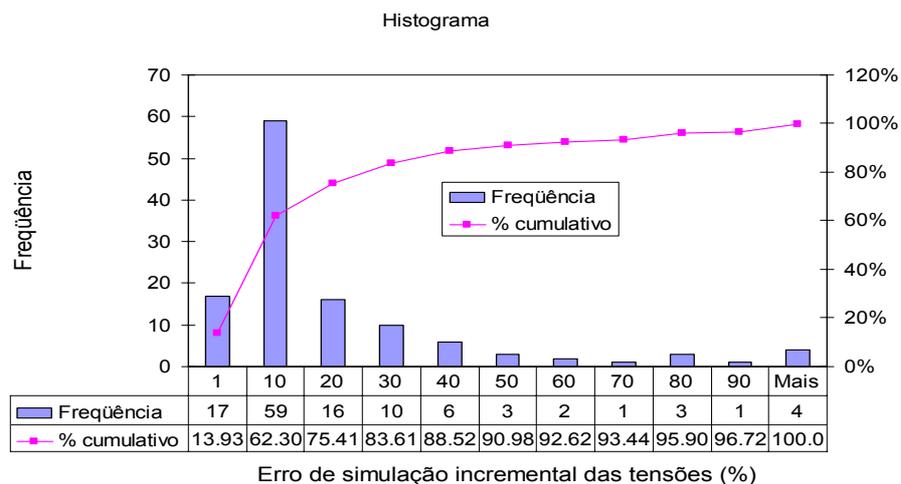


Figura 4.83. Histograma de freqüência para o erro de simulação incremental da rede *Magola10*.

Embora diversas modificações tenham sido aplicadas aos diferentes parâmetros de entrada da base de dados de treinamento, o resultado final do modelo não refletiu em uma melhora significativa do desempenho. A inspeção visual dos resultados a partir das curvas de simulação tensão-deformação que foram apresentadas permite chegar à mesma conclusão, ou seja, os dois modelos são tecnicamente iguais.

Adicionalmente, observa-se que existe um conjunto importante de testes para os quais o modelo neural consegue reproduzir adequadamente a forma da curva tensão-deformação, porém falha na previsão da tensão de ruptura. Alguns destes testes são: 2, 3, 5, 6, 7, 20, 22, 23, 24, 25, 26, 27, 28, 29, 30, 32, 33, 34, 63, 71, 103, 105, 106, 107, 114, 115, 117, 118, 119, 142 e 143.

Visando corrigir esta deficiência do modelo atual, foi introduzido à base de dados um parâmetro adicional que fornece informação explícita sobre a evolução do tamanho da superfície de plastificação, dando ao modelo a oportunidade de aproximar com maior precisão o momento da ruptura da areia para cada situação específica de carregamento.

4.9 – MODELO NEURAL 6

A previsão precisa da tensão de ruptura é uma característica importante de qualquer modelo constitutivo, paramétrico ou não. Em modelos do tipo elastoplástico, tal como já se comentou em seções anteriores, na medida que o solo vai sendo submetido a carregamentos, o tamanho da superfície de plastificação vai aumentando até um limite a partir do qual pode ser definida a tensão de ruptura do material. Além deste limite a superfície não aumenta de tamanho. As seqüências de valores de tensão, contidas na base de dados que foi usada para o treinamento do modelo neural, incluem informação relacionada com a evolução da superfície de plastificação de cada material. Esta informação não é explícita, ou seja, ela é inerente aos valores de tensão que cada ponto vai experimentando ao longo da evolução do ensaio, porém não existe nenhum parâmetro que esteja indicando diretamente a forma como evolui o tamanho da superfície de plastificação.

Nesse sentido, o modelo neural deve extrair da informação dada nas seqüências de tensões, apenas o conhecimento sobre a tensão de ruptura de cada material para cada situação particular de carregamento. Tal tarefa só foi realizada com sucesso para a maioria dos testes do tipo CTC e para alguns correspondentes a outras trajetórias. A dificuldade de extrair este tipo de conhecimento a partir das informações fornecidas no problema ora proposto, está não

só na complexidade do mapeamento pretendido, mas também na variedade de materiais envolvidos e nas suas diferentes características físicas.

Nas seções seguintes se apresentam algumas modificações que foram incluídas na base de dados de treinamento com o propósito de fornecer ao modelo informação relativa à evolução da superfície de plastificação do material e mais especificamente sobre a magnitude da tensão de ruptura para cada material.

4.9.1 – PRÉ-PROCESSAMENTO DA BASE DE DADOS

Existem várias maneiras de incorporar informação relacionada com a magnitude da tensão de ruptura em modelos constitutivos. A alternativa óbvia consiste em informar diretamente ao modelo o valor da tensão máxima que o material suporta antes de começar a experimentar processos de amolecimento, ou seja, indicar indiretamente o tamanho máximo da superfície de plastificação. Esta alternativa, embora direta e simples, tem como maior dificuldade a necessidade de conhecer “a priori” o valor da tensão de ruptura para cada possível condição de carregamento e para cada tipo de material, sob as diversas condições físicas e iniciais que possa experimentar.

Uma alternativa mais razoável consiste em usar grandezas matemáticas, como os invariantes de tensão, para indicar a evolução do tamanho da superfície de plastificação e conseqüentemente o ponto no qual se atinge a tensão de ruptura do material. Uma dessas grandezas é a tensão normal octaédrica (p), que pode ser calculada para qualquer estágio de carregamento do material. No entanto, ela apresenta duas características que dificultam sua aplicação direta em modelos neurais:

- Não fornece informação suficiente sobre a tensão de ruptura pois para certas trajetórias de tensão (trajetórias ao longo do plano octaédrico) o seu valor permanece constante para todos os estágios de carregamento.
- A sua relação com a tensão de confinamento inicial não é constante, ela depende da trajetória de tensões imposta ao material. Assim, para um certo valor da tensão confinante inicial (σ_{c0}) será possível obter diferentes valores da tensão normal octaédrica na ruptura (p_{rup}). Isto pode ser visto na Figura 4.84.

Desta forma, se faz necessário utilizar outra grandeza que seja independente da trajetória de tensão e que incorpore informação sobre a evolução da superfície de plastificação

e a magnitude da tensão de ruptura. Tal grandeza, chamada nesta tese de X , é uma combinação dos três invariantes de tensão tal como se apresenta nas equações seguintes:

$$I_1 = \text{tr}(\underline{\underline{\sigma}}) = \sigma_1 + \sigma_2 + \sigma_3 \quad (4.15)$$

$$I_2 = (\sigma_1 \times \sigma_2) + (\sigma_2 \times \sigma_3) + (\sigma_1 \times \sigma_3) \quad (4.16)$$

$$I_3 = \sigma_1 \times \sigma_2 \times \sigma_3 \quad (4.17)$$

$$X = \frac{I_1 \times I_2}{I_3} \quad (4.18)$$

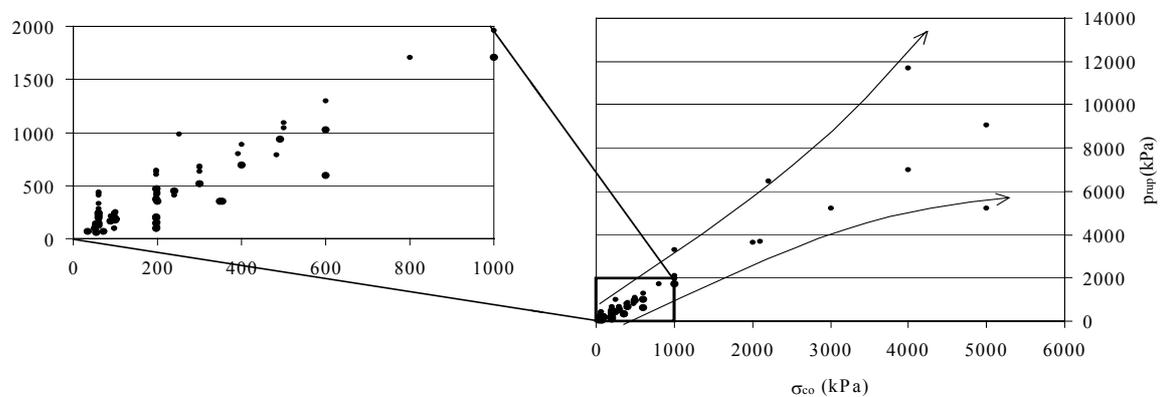


Figura 4.84. Relação entre a tensão de confinamento inicial (σ_{c0}) e a tensão octaédrica na ruptura (p_{rup}) para todos os testes da base de dados.

A grandeza X , que também é um invariante de tensão, é um indicador do tamanho da superfície de plastificação. Ela toma um valor mínimo de 9 quando $\sigma_1 = \sigma_2 = \sigma_3$, e vai aumentando na medida que o material é solicitado, até atingir um valor máximo em torno de 15 ou 20 na ruptura para os casos estudados. Posteriormente, diminui quando o material experimenta o fenômeno de amolecimento. Tal comportamento pode ser observado na Figura 4.85 na qual estão representados os pontos do teste e a evolução do valor do invariante X até o valor de ruptura. O valor de X na ruptura corresponde ao critério de Matsuoka-Nakai (1985).

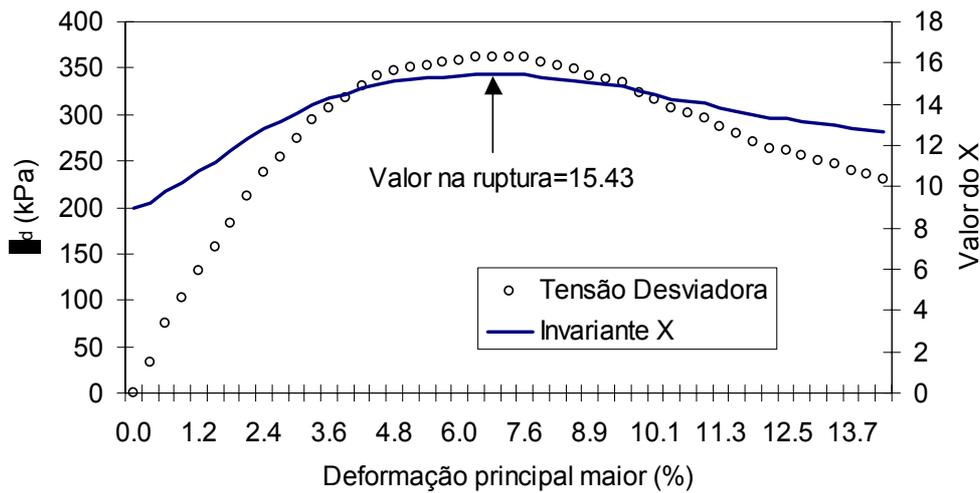
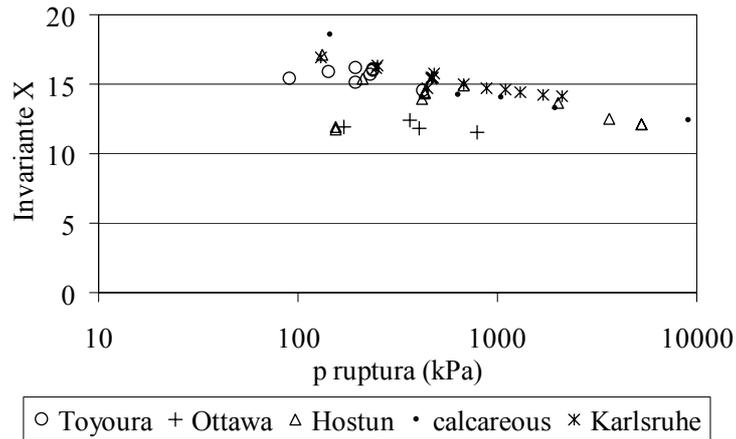
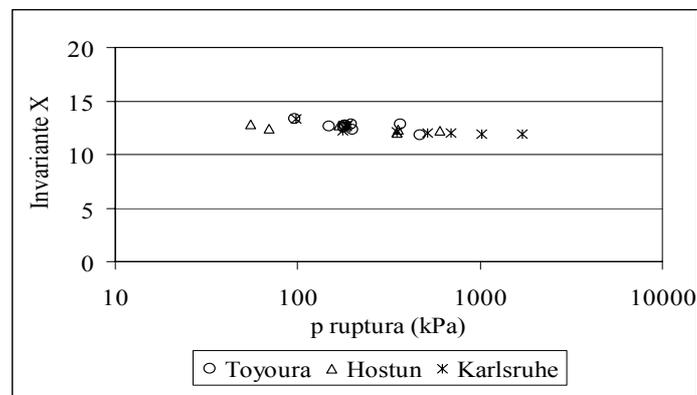


Figura 4.85. Evolução do invariante $X = \frac{I_1 \times I_2}{I_3}$ para o teste 70.

Uma outra característica interessante, e valiosa, do invariante X é que seu valor na ruptura independe da trajetória de tensão e da tensão octaédrica inicial e final, dependendo só do tipo de material e ligeiramente da sua condição física inicial. Na Figura 4.86a pode se observar que para as areias densas o valor do invariante apresenta uma tendência de redução a medida que aumenta a tensão normal octaédrica. No entanto, ao se observar independentemente cada tipo de areia, nota-se que essa tendência não é forte, mantendo os valores do invariante X na ruptura dentro de um intervalo de valores estreito. Isto decorre do fato de a envoltória de ruptura ser relativamente curva para materiais densos, o que não é considerado no critério de Matsuoka-Nakai (X). Já para areias na condição fofa (Na Figura 4.86b) a tendência é menor ainda, podendo-se considerar para fins práticos que o valor de X na ruptura é aproximadamente constante.



(a) Areias na condição densa



(b) Areias na condição fofa

Figura 4.86. Relação entre a tensão octaédrica na ruptura (p_{rup}) e o invariante X na ruptura para areias na condição densa (a) e na condição fofa (b).

Considerando as características apresentadas do invariante X, decidiu-se incorporá-lo na base de dados de treinamento mediante a criação de uma entrada adicional. As demais entradas da rede e as suas transformações antes da normalização ficaram iguais às do modelo anterior, com exceção da tensão confinante inicial que para este caso entrou de novo adotando seus valores originais. Na Figura 4.87 são apresentados os histogramas de frequência das primeiras quatro entradas da base de dados para os valores originais (linha superior), para os valores transformados (linha intermediária) e para os valores normalizados (linha inferior).

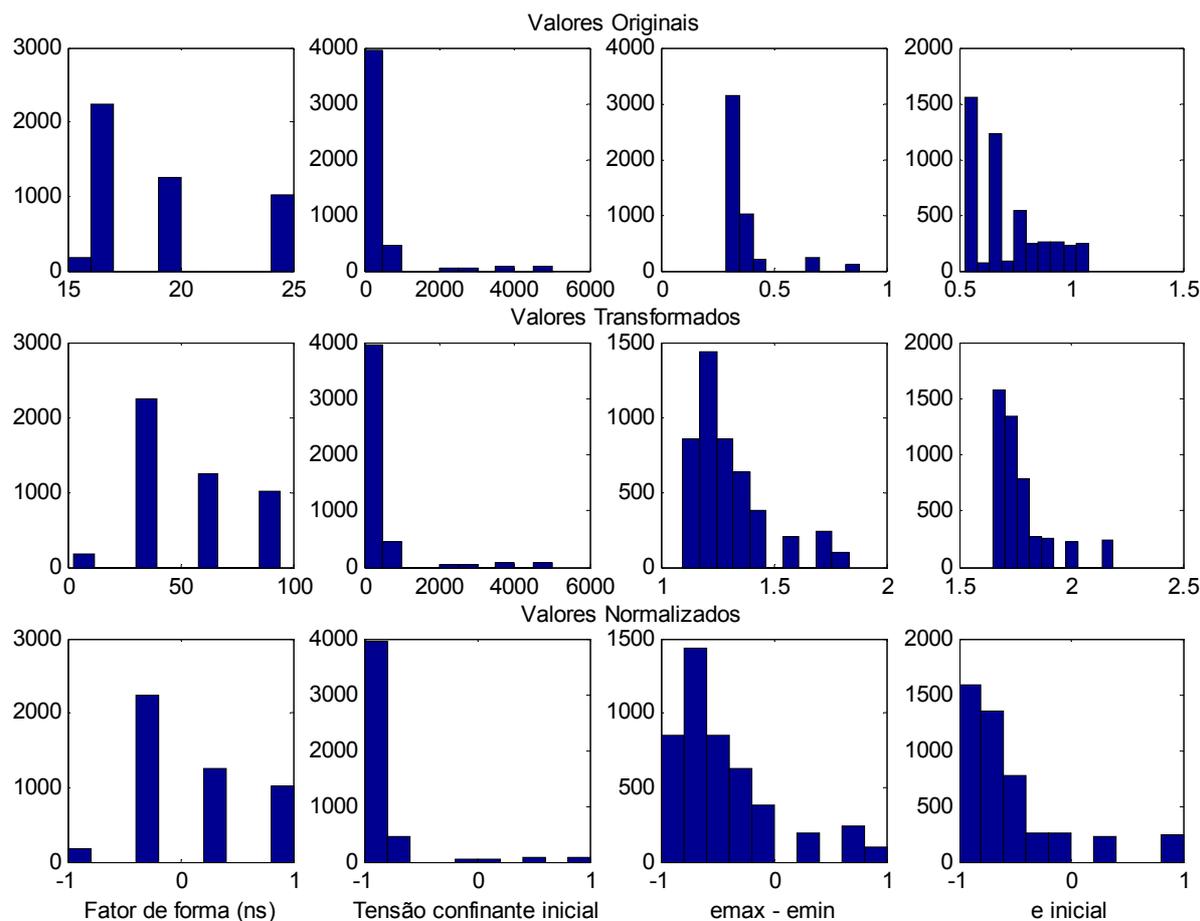


Figura 4.87. Histogramas de frequência das primeiras quatro entradas da base de dados do modelo neural 6.

4.9.2 – ARQUITETURA E TREINAMENTO

Incorporada uma entrada nova na base de dados, o vetor de treinamento modificado fica como apresentado na Eq.(4.19).

$$\{\Delta\sigma_i\} = f(n_s, \sigma_0, e_{\max} - e_{\min}, e_0, \{\varepsilon_i\}, \{\varepsilon_{i-1}\}, \{\sigma_i\}, \{\sigma_{i-1}\}, \{\Delta\varepsilon_i\}, X_i) \quad (4.19)$$

Uma vez que o número de entradas aumentou de 19 para 20, considerou-se adequado avaliar diferentes arquiteturas para determinar de novo o tamanho ótimo de rede. Foram criados cinco modelos diferentes com arquiteturas internas de 80x90, 100x100, 90x100, 120x100 e 80x80. Cada modelo foi inicializado cinco vezes e treinado durante 100 épocas no modo *por lote*. Os cinco melhores foram guardados para comparação, descartando os

restantes. A Figura 4.88 mostra a parte final do processo de treinamento, onde se aprecia que o melhor desempenho em termos de erro mínimo foi obtido com a rede de 120x100.

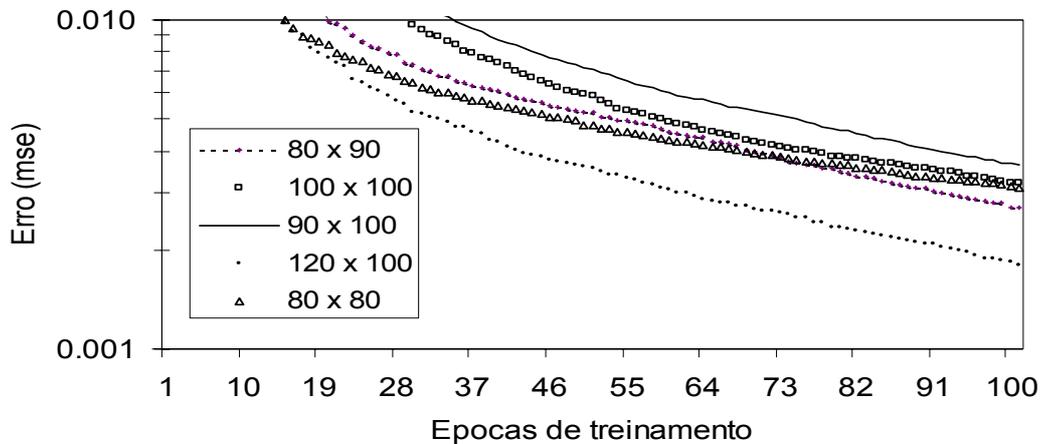


Figura 4.88. Evolução do erro de treinamento para cinco modelos neurais.

Embora 100 épocas de treinamento pareçam pouco representativas do desempenho ou convergência futura do modelo, a experiência prévia acumulada e de alguma maneira o impulso de explorar novos caminhos, motivaram a determinação de tentar um novo processo completo de treinamento com a arquitetura de 120x100. O novo modelo foi chamado de *Magola11*. O modelo *Magola11* foi treinado mediante a técnica de treinamento seletivo, dividindo a base de dados em 8 porções. A Figura 4.89 apresenta a evolução do erro de treinamento da rede *Magola11* para cada um dos lotes de informação apresentados para um total de 304100 épocas de treinamento. O tempo computacional de treinamento foi de 939147 segundos, equivalentes a 260 horas ou aproximadamente 11 dias. No entanto, devido à necessidade de interromper periodicamente o processo de treinamento para verificação do erro de simulação o tempo total de treinamento foi de aproximadamente 53 dias.

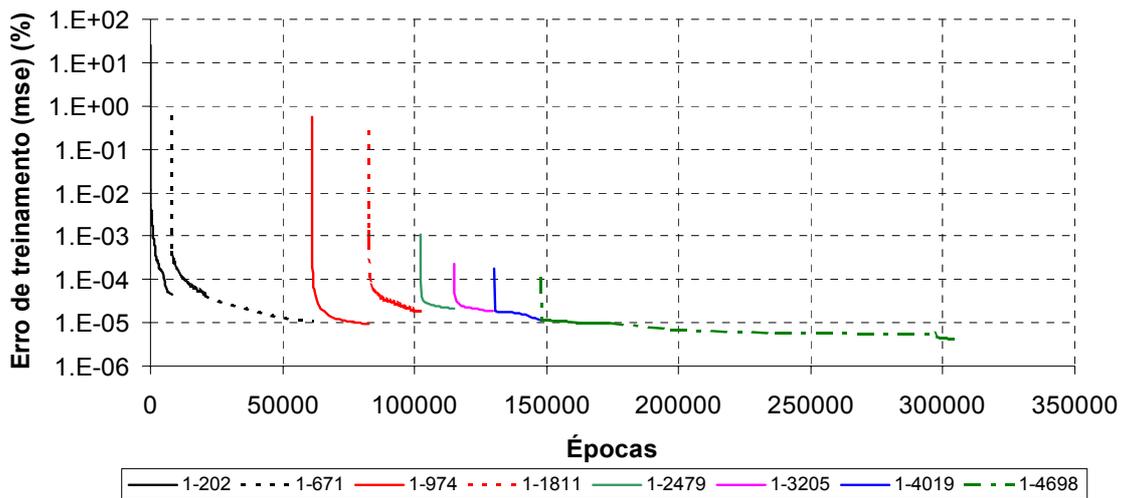


Figura 4.89. Evolução do erro de treinamento da rede *Magola11*.

4.9.3 – RESULTADOS DA SIMULAÇÃO

A rede *Magola11* foi simulada de maneira recorrente ou autônoma para todos os testes disponíveis. As tensões principais produzidas pela rede são comparadas com as tensões experimentais para determinar a precisão da aproximação dada pelo modelo. Tal comparação aparece na Figura 4.90. Os gráficos da linha superior da figura em questão, representam à totalidade dos testes disponíveis. Observa-se um coeficiente de correlação alto para os três componentes de tensão, não obstante a inspeção visual permite identificar alguns grupos de pontos que se encontram completamente afastados da tendência central. Esses pontos correspondem a cinco testes que por razões não explicadas não foram adequadamente reproduzidos pela rede neural, esses testes são: 125 (areia de Hostun, RTE), 142 (areia de Hostun, CTC), 118 (areia de Hostun, TC), 7 (areia de Monterrey, CTC) e 84 (areia de Hostun, CTC). Os pontos correspondentes a esses testes foram eliminados (só para efeito de apresentação da Figura 4.90) e novas correlações foram realizadas tal como aparece nos gráficos da linha inferior da mesma figura. Observa-se então uma melhora substancial nos valores dos coeficientes de correlação para os três componentes de tensão.

As curvas tensão-deformação para os cinco testes mencionados no parágrafo anterior são apresentados na Figura 4.91.

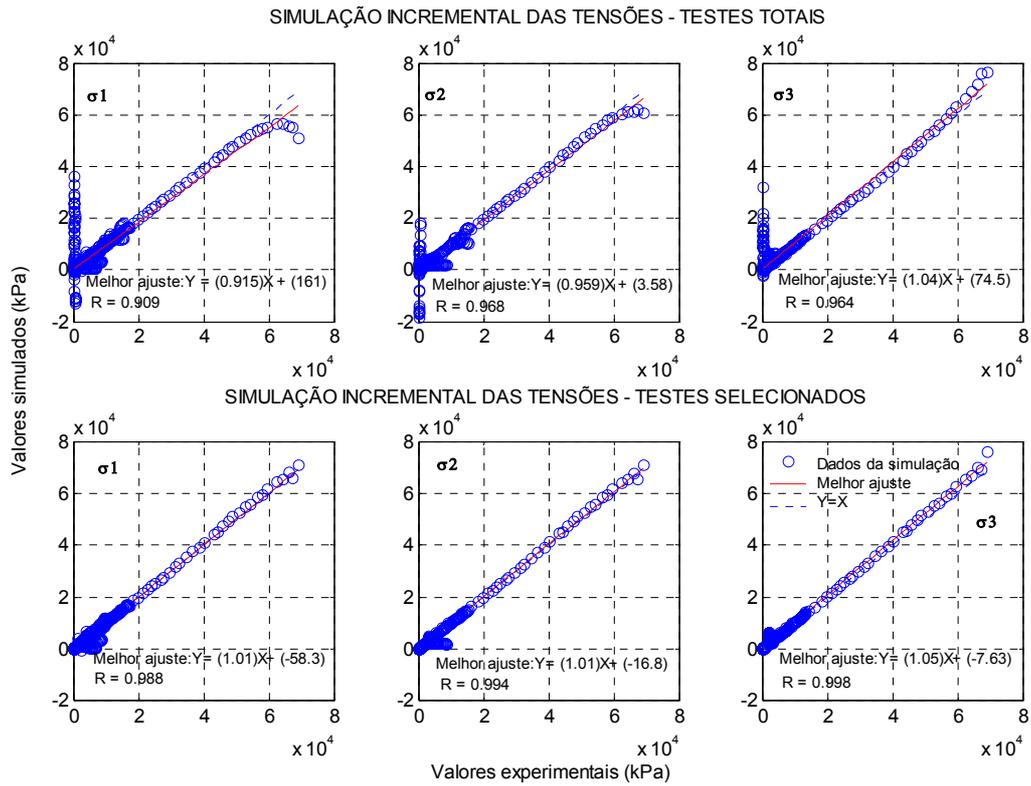


Figura 4.90. Simulação incremental das tensões para o modelo *Magola11*.

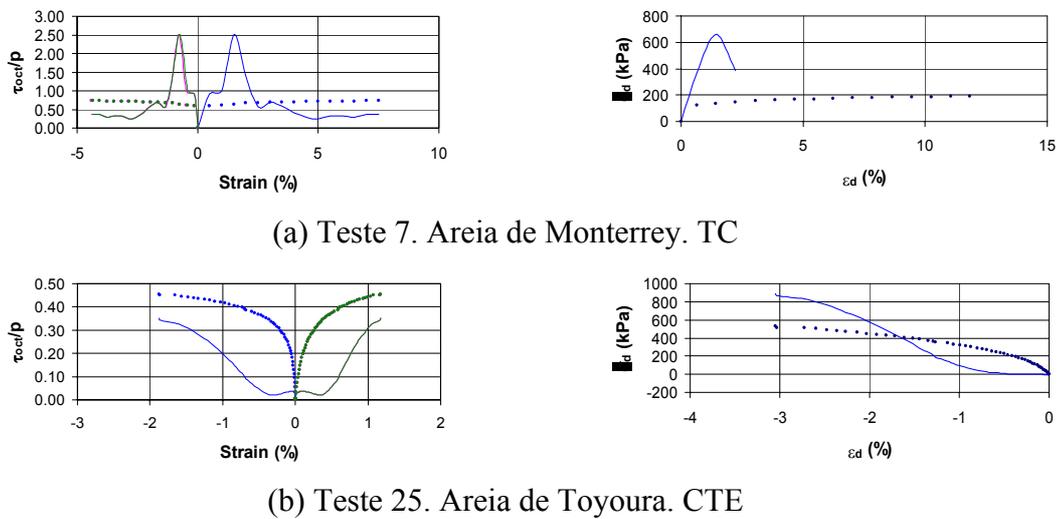
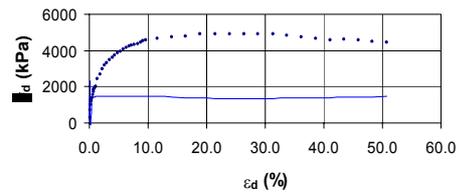
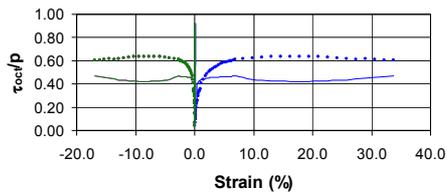
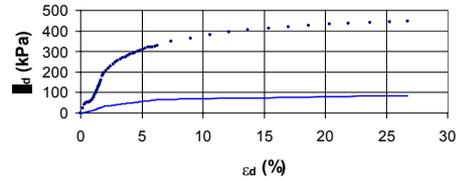
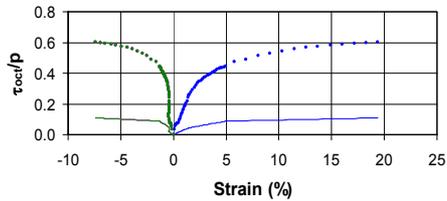


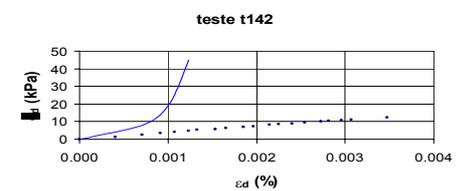
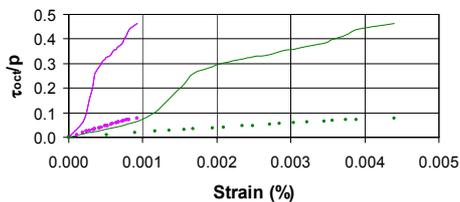
Figura 4.91. Gráficos tensão-deformação para cinco testes que não foram adequadamente simulados pelo modelo *Magola11*.



(c) Teste 84. Areia de Hostun. CTC



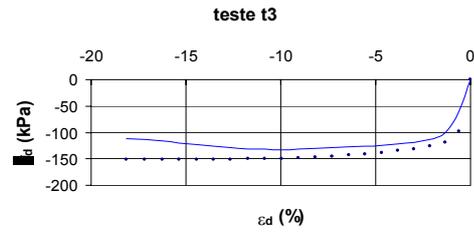
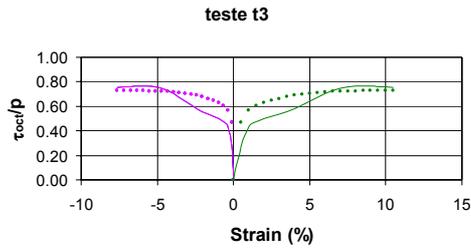
(d) Teste 118. Areia de Hostun. TC



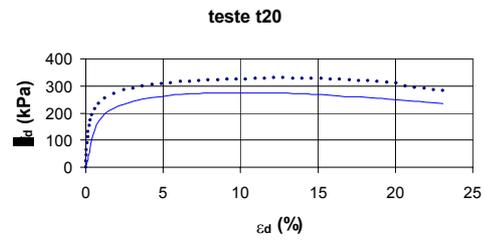
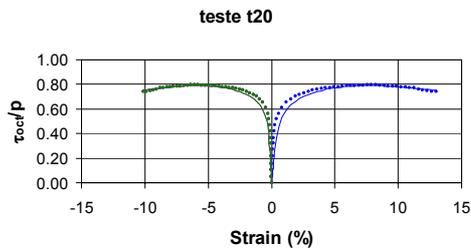
(e) Teste 142. Areia de Hostun. CTC

Figura 4.91. (Continuação) Gráficos tensão-deformação para cinco testes que não foram adequadamente simulados pelo modelo *Magola11*.

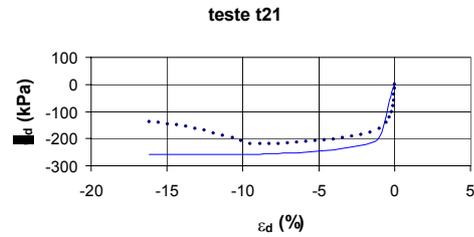
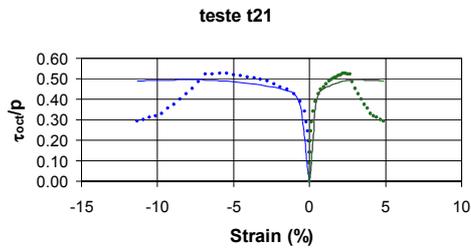
Já na Figura 4.92 são apresentadas as curvas tensão-deformação de 20 testes para os quais a rede apresentou uma melhora notável no seu desempenho em comparação com a rede anterior. As curvas dos testes restantes, para os quais o desempenho se manteve praticamente igual, podem ser observadas na seção de apresentação de resultados do capítulo seguinte, onde são comparadas com as curvas fornecidas pelo programa de elementos finitos ALLFINE-NN no qual a rede neural *Magola11* age como operador constitutivo, fazendo o mapeamento, ao nível de ponto, entre acréscimos de deformação e acréscimos de tensão.



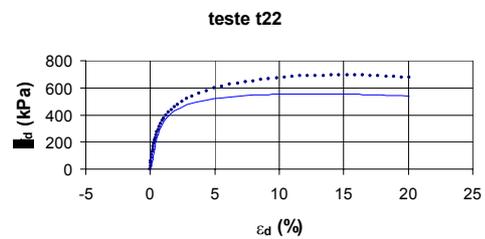
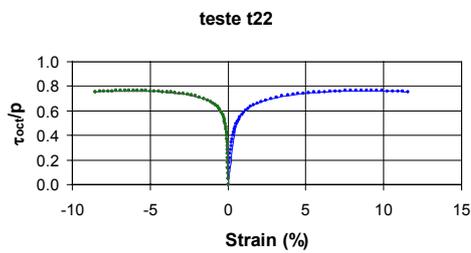
(a) Teste 3. Areia de Cambria. TC



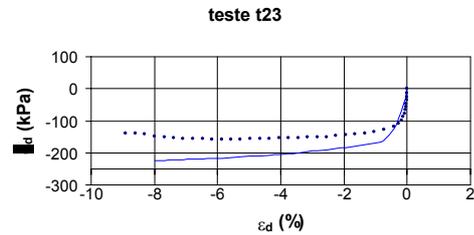
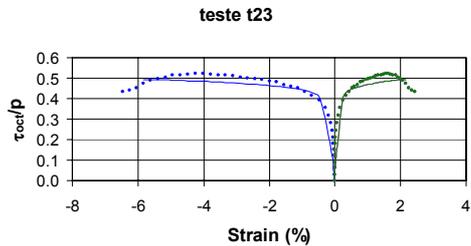
(b) Teste 20. Areia de Toyoura. TC



(c) Teste 21. Areia de Toyoura. TE

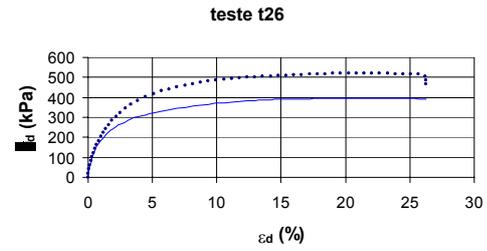
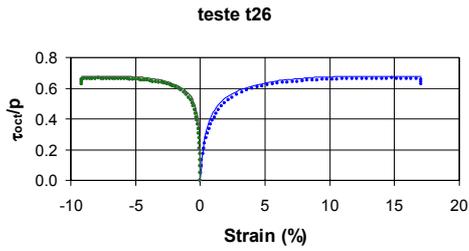


(d) Teste 22. Areia de Toyoura. CTC

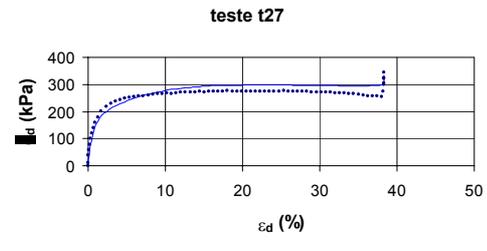
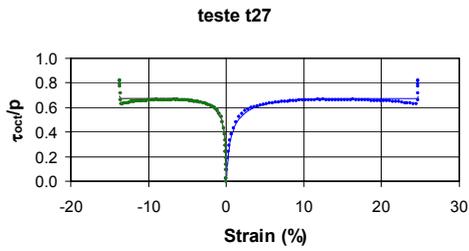


(e) Teste 23. Areia de Toyoura. RTE

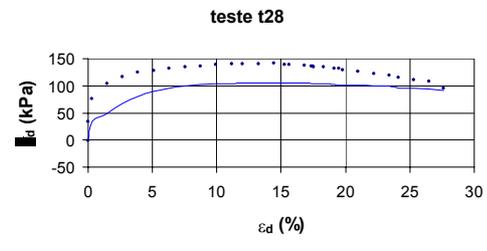
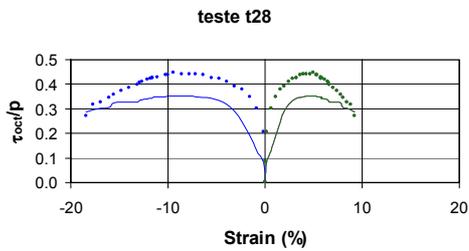
Figura 4.92. Curvas tensão-deformação dos testes para os quais a rede *Magola11* apresentou melhora significativa do desempenho em comparação com a rede *Magola10*



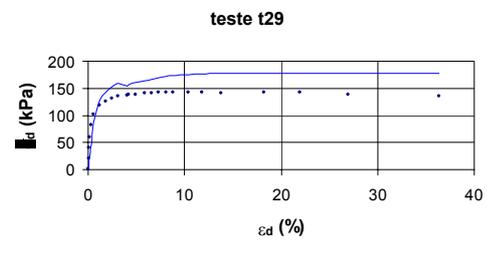
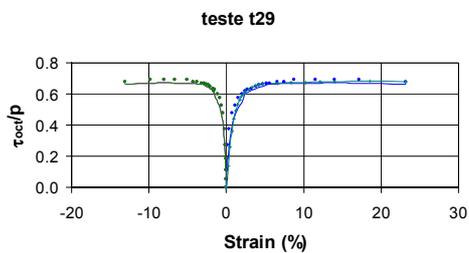
(f) Teste 26. Areia de Toyoura. CTC



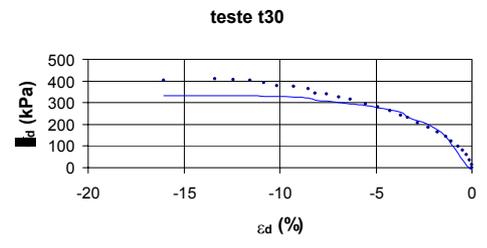
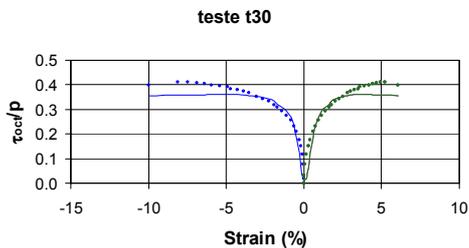
(g) Teste 27. Areia de Toyoura. TC



(h) Teste 28. Areia de Toyoura. RTE

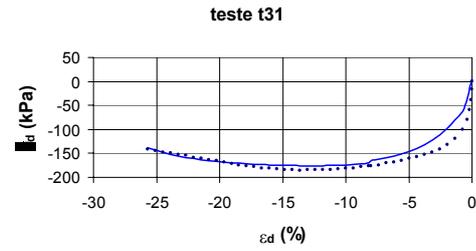
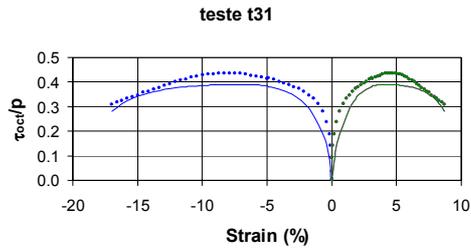


(i) Teste 29. Areia de Toyoura. RTC

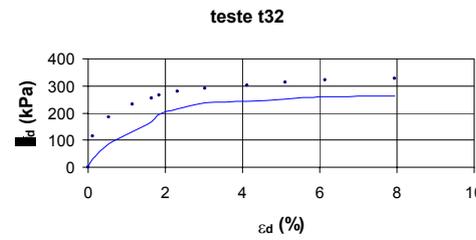
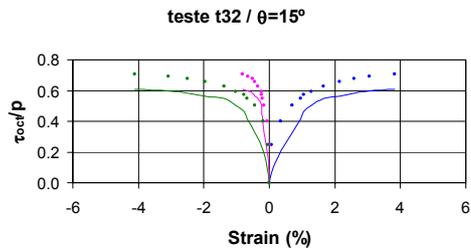


(j) Teste 30. Areia de Toyoura. CTE

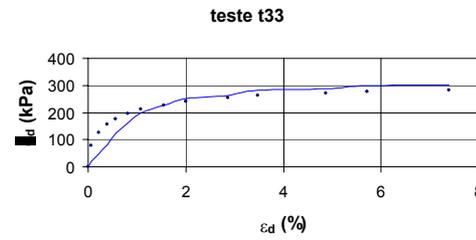
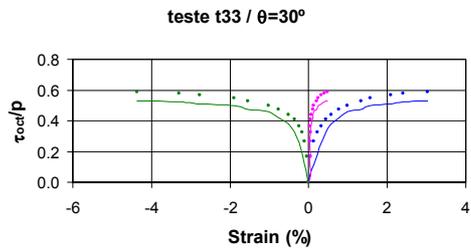
Figura 4.92. (Continuação) Curvas tensão-deformação dos testes para os quais a rede *Magola11* apresentou melhora significativa do desempenho em comparação com a rede *Magola10*



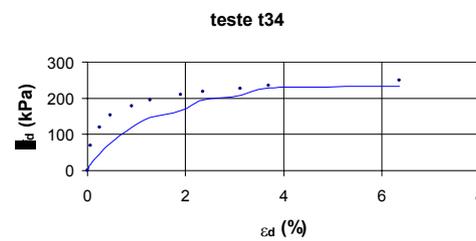
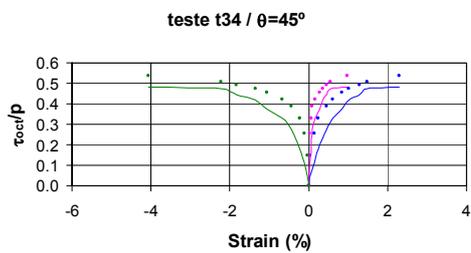
(k) Teste 31. Areia de Toyoura. TE



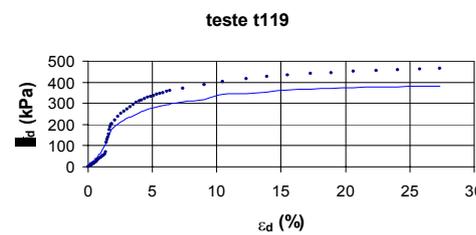
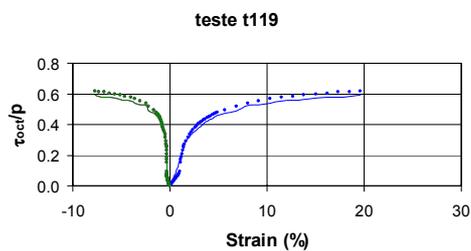
(l) Teste 32. Areia de Toyoura. Plano Octaédrico, $\theta=15^\circ$



(m) Teste 33. Areia de Toyoura. Plano Octaédrico, $\theta=33^\circ$

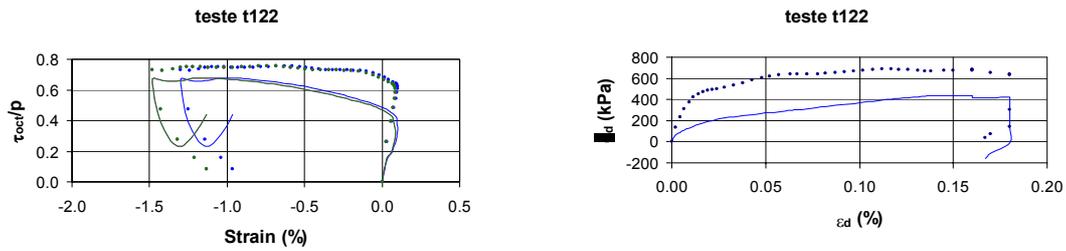


(n) Teste 34. Areia de Toyoura. Plano Octaédrico, $\theta=45^\circ$

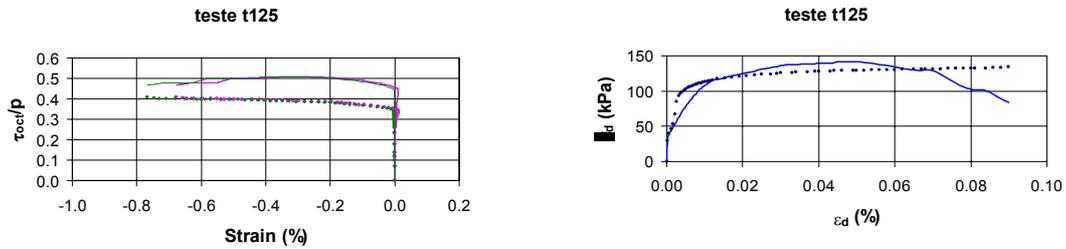


(o) Teste 119. Areia de Hostun. TC

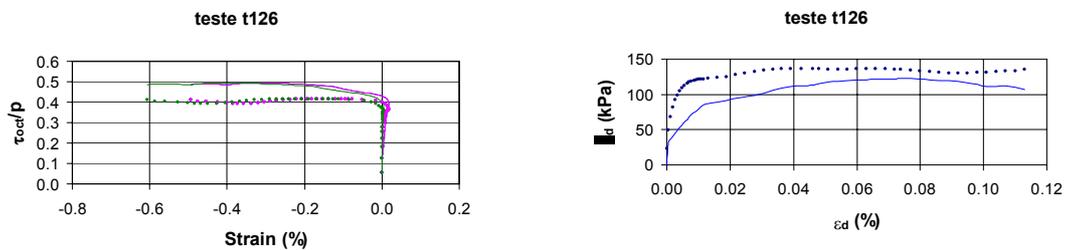
Figura 4.92. (Continuação) Curvas tensão-deformação dos testes para os quais a rede *Magola11* apresentou melhora significativa do desempenho em comparação com a rede *Magola10*



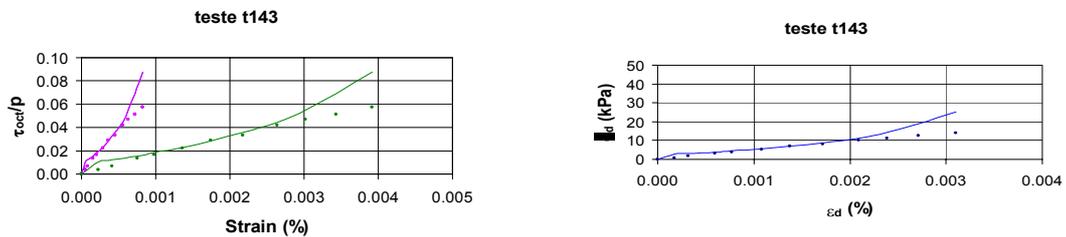
(p) Teste 122. Areia de Hostun. RTE



(q) Teste 125. Areia de Hostun. RTE



(r) Teste 126. Areia de Hostun. RTE



(s) Teste 143. Areia de Hostun. CTC

Figura 4.92. (Continuação) Curvas tensão-deformação dos testes para os quais a rede *Magola11* apresentou melhora significativa do desempenho em comparação com a rede *Magola10*.

4.9.4 – DISCUSSÃO

Em termos gerais o modelo neural *Magola11* apresentou erros de simulação menores que o modelo 5, tal como se pode apreciar no histograma da Figura 4.93. Nela é evidente a concentração ao redor do valor 10% e o aumento da proporção de testes (93,75%) com erro

menor do que 20% em comparação com o modelo *Magola10* para o qual apenas 75,4% dos testes apresentaram erros menores que 20%.

A introdução de uma entrada adicional para incorporar o invariante X, demonstrou ser conveniente para melhorar a previsão da tensão de ruptura. Isto é evidente nos testes da Figura 4.92 para os quais a rede *Magola10* tinha apresentado aproximações deficientes aos valores das tensões de ruptura. Vale a pena ressaltar que nenhum parâmetro de identificação do material relativo à ruptura, como por exemplo o ângulo de atrito, é exigido pela rede.

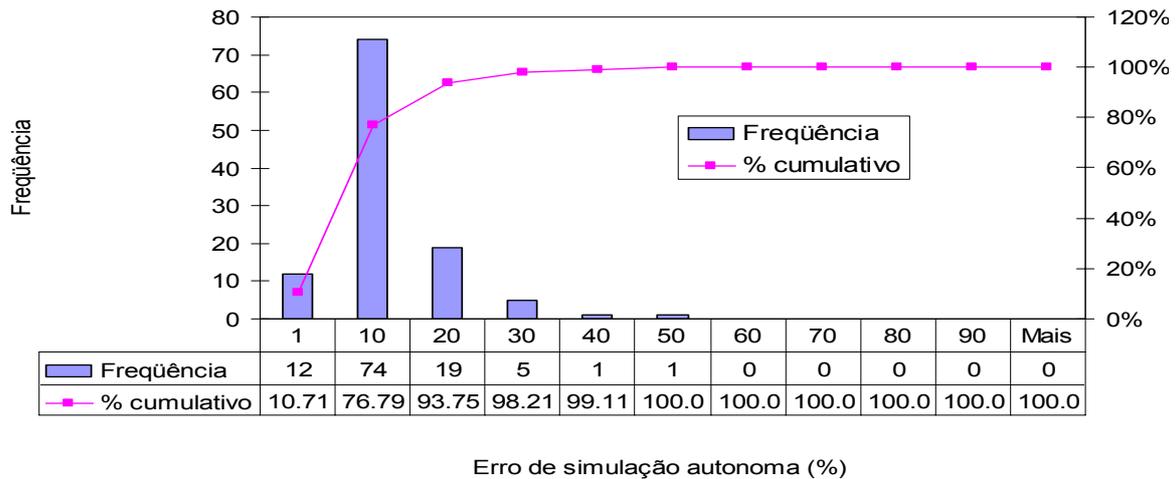


Figura 4.93. Histograma de frequência do erro de simulação autônoma da rede *Magola11*.

Nas tabelas do Apêndice D são apresentados todos os valores dos parâmetros livres da rede *Magola11* agrupados em três matrizes de pesos e três vetores de biases. Uma primeira matriz de 20x120 elementos que conecta as entradas da rede com a primeira camada oculta. Uma matriz de 120x100 que conecta a primeira camada oculta com a segunda camada oculta, uma matriz de 100x3 que conecta a segunda camada oculta com a camada de saída. O primeiro vetor de biases possui 120 elementos, o segundo 100 e o terceiro 3.

O nível de precisão alcançado pelo modelo *Magola11* permite concluir a fase de treinamento neural e passar à fase de implementação do modelo em um código de elementos finitos onde a rede neural age como operador constitutivo, substituindo a relação elasto-plástica convencional no processo de mapeamento entre acréscimos de deformação e acréscimos de tensão. O programa escolhido para tal implementação é o ALLFINE desenvolvido por Farias (1993) e modificado recentemente por Cordão Neto (2005).

IMPLEMENTAÇÃO

Nos capítulos anteriores foi mostrado em detalhe o processo por meio do qual uma rede neural do tipo PMC treinada com o algoritmo de retropropagação do erro conseguiu realizar um mapeamento satisfatório entre acréscimos de tensão e acréscimos de deformação, para um conjunto de areias submetidas a diversas condições de carregamento. O modelo neural, chamado *Magol11*, que funciona como um modelo do tipo deformação controlada deverá substituir uma relação constitutiva paramétrica convencional dentro de um programa de elementos finitos. Neste capítulo se descreve a forma como a rede neural constitutiva opera dentro do código de elementos finitos, e são apresentados alguns resultados obtidos com o sistema híbrido.

5.1 – INTRODUÇÃO

Os modelos constitutivos baseados em RNA estão sendo cada vez mais usados em aplicações híbridas com programas de elementos finitos na solução de problemas de contorno. Os modelos constitutivos neurais, em contraste com os modelos elastoplásticos convencionais, não requerem procedimentos especiais de integração para sua implementação em códigos de elementos finitos, e tampouco usam o conceito tradicional de matriz de rigidez (Hashash et al, 2004).

Um modelo constitutivo usado dentro de um código de elementos finitos desempenha duas funções. Primeiro, atualizar o estado de tensões dado um certo estado atual de tensão-deformação e um vetor de acréscimos de deformação. Segundo, calcula a matriz de rigidez ou matriz constitutiva do material para a relação constitutiva usada. Para o caso de modelos

constitutivos neurais, a matriz constitutiva não pode ser obtida de uma maneira explícita, sendo necessário o uso de aproximações alternativas (Hashash et al, 2004).

5.2 – IMPLEMENTAÇÃO NUMÉRICA EM UM CÓDIGO DE ELEMENTOS FINITOS

A relação entre um modelo constitutivo e a solução global das equações de equilíbrio numa análise de elementos finitos é apresentada na Figura 5.1. Nessa figura o gráfico $\{P\} - \{U\}$ (carga – deslocamento) representa o nível do elemento, enquanto que o gráfico $\{\sigma\} - \{\varepsilon\}$ (tensão – deformação) representa o nível do ponto em um material.

No n -ésimo acréscimo de carga é aplicado um ΔP_n , e o correspondente deslocamento ΔU_n é calculado. Inicialmente pode ser usada uma matriz de rigidez elástica $[k_e]$ tal como se apresenta na Figura 5.1. Numa primeira aproximação tem-se:

$$P_n = P_{n-1} + \Delta P_n \quad (5.1)$$

$$U_n = U_{n-1} + \Delta U_n \quad (5.2)$$

Na primeira iteração do processo (ver o ciclo representado com 1 na Figura 5.1) são atualizadas as deformações, as tensões e o correspondente vetor de forças internas I_n .

$$\Delta \varepsilon_n = [B] \Delta U_n \quad (5.3)$$

$$\Delta \sigma_n = \Delta \sigma_n NN(\Delta \varepsilon_n, \dots) \quad (5.4)$$

$$\sigma_n = \Delta \sigma_n + \sigma_{n-1} \quad (5.5)$$

$$I_n = \sum \iiint_V [B]^T \sigma_n dV \quad (5.6)$$

$$R_n = P_n - I_n \quad (5.7)$$

Na Eq. (5.3) $[B]$ representa a matriz que relaciona acréscimos de deformações ($\Delta \varepsilon_n$) com acréscimos de deslocamentos nodais (ΔU_n). Na Eq. (5.4) o operador NN representa o modelo constitutivo neural cuja saída é o vetor de acréscimos de tensão e cujas entradas são o vetor de acréscimos de deformação mais outros vetores que definem o tipo de solo e a sua condição física. Na Eq. (5.7) o vetor R_n representa o resíduo calculado como a diferença entre as forças externas (P) e as internas (I), cuja norma pode ser usada como critério de

convergência do algoritmo. Estas forças residuais representam as forças desbalanceadas que devem ser zeradas ou minimizadas até um valor aceitável após um processo iterativo. As forças residuais são reaplicadas, atualizando-se ou não a matriz de rigidez do sistema de acordo com o esquema de solução adotado para o problema não linear (ao nível global).

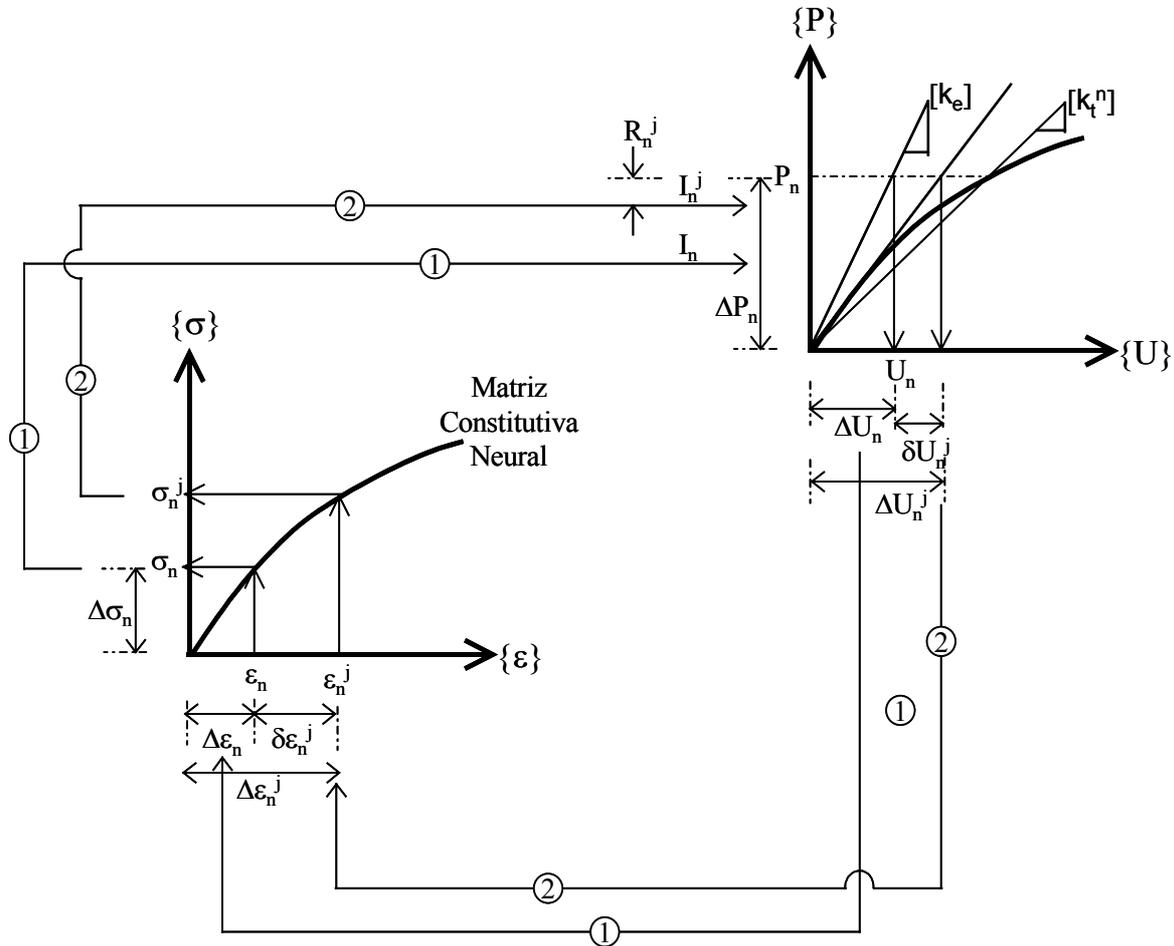


Figura 5.1. Representação gráfica do processo iterativo para solucionar o n-ésimo acréscimo de carga usando um modelo constitutivo neural dentro de um código de elementos finitos.

Na j-ésima iteração um novo acréscimo de deslocamento δU_n^j é usado para uma nova atualização das deformações, das tensões e das forças segundo as equações seguintes (ver o ciclo representado com 2 na Figura 5.1).

$$\Delta U_n^j = \Delta U_n^{j-1} + \delta U_n^j \quad (5.8)$$

$$\Delta \varepsilon_n^j = [B] \Delta U_n^j \quad (5.9)$$

$$\Delta \sigma_n^j = \Delta \sigma_n^j NN(\Delta \varepsilon_n^j, \dots) \quad (5.10)$$

$$\sigma_n^j = \Delta \sigma_n^j + \sigma_n^{j-1} \quad (5.11)$$

$$I_n^j = \sum \iiint_V [B]^T \sigma_n^j dV \quad (5.12)$$

$$R_n^j = P_n^j - I_n^j \quad (5.13)$$

O processo iterativo continua até satisfazer o critério de convergência. Neste processo não se faz necessária a determinação explícita da matriz de rigidez tangente $[k_t]$, podendo-se utilizar, por exemplo, o método de rigidez inicial. Neste caso deve-se fornecer valores típicos das constantes elásticas (E , ν). Caso se opte por utilizar uma matriz tangente, pode-se obter uma aproximação, a partir da estimativa da matriz constitutiva com a seguinte expressão:

$$D_{ij} = \frac{\Delta \sigma_i}{\Delta \varepsilon_j} \quad (5.14)$$

onde $\Delta \sigma_i$ representa os acréscimos de tensão calculados pela rede neural no intervalo anterior e $\Delta \varepsilon_j$ representa os acréscimos de deformações impostos.

A matriz de rigidez de um elemento seria então calculada de forma convencional:

$$K = \iiint_V [B]^T [D] [B] dV \quad (5.15)$$

O modelo constitutivo neural proposto foi implementado dentro do programa ALLFINE que é um código de elementos finitos de propósito geral para aplicações geotécnicas, com o intuito de possibilitar o seu uso na solução geral de problemas de contorno, envolvendo o tipo de areias para as quais a rede foi treinada.

Tal como foi visto no capítulo anterior, a RNA constitutiva realiza um mapeamento entre acréscimos de deformação e acréscimos de tensão a partir de um vetor de entradas que contém informação sobre o estado anterior tensão-deformação e sobre as características físicas da areia. Nesse mapeamento duas considerações importantes devem ser levadas em conta para a implementação no MEF:

- A RNA foi treinada com informação experimental na qual as deformações e as correspondentes tensões entram como vetores de três componentes, uma para cada direção principal.
- O vetor de acréscimos de deformação age, dentro da RNA, como uma variável independente. Sabe-se que a resposta de um modelo neural é boa para aqueles valores das variáveis independentes que se encontram dentro do intervalo de valores usados para treinamento. Em consequência, para valores de acréscimos

de deformação maiores ou menores que os usados para o treinamento, a resposta da rede poderá ser inadequada.

Para contornar o primeiro problema é necessário obter inicialmente o vetor de acréscimos de deformações principais, $\{\Delta\hat{\varepsilon}\}$, e as direções principais, $[Q]$, a partir do tensor de deformações, $\{\Delta\varepsilon\}$, referido ao sistema global (x,y,z). Assim, tem-se:

$$[\Delta\hat{\varepsilon}] = \text{autovalores } \{\Delta\varepsilon\} \quad (5.16)$$

$$[Q] = \text{autovetores } \{\Delta\varepsilon\} \quad (5.17)$$

onde “autovalores” e “autovetores” são funções que fornecem, respectivamente, os autovalores e os autovetores de um tensor. Estas funções estão codificadas numa sub-rotina específica do programa ALLFINE.

A matriz $[Q]$ permite a transformação das componentes de um tensor entre o sistema global e o sistema referido às direções principais de deformação, isto é:

$$[Q] = \begin{bmatrix} l_1 & m_1 & n_1 \\ l_2 & m_2 & n_2 \\ l_3 & m_3 & n_3 \end{bmatrix} \quad (5.18)$$

onde l_i, m_i, n_i são os co-senos diretores de direção principal i . Desta forma:

$$[\Delta\hat{\varepsilon}] = [Q][\Delta\varepsilon][Q]^T \quad (5.19)$$

Por hipótese assume-se neste trabalho que as direções principais de acréscimos de tensões coincidem com as direções principais de acréscimos de deformações, ou seja $[Q]_\sigma = [Q]_\varepsilon$. Esta é uma hipótese comum em modelos elasto-plásticos.

Assim sendo, tem-se:

$$[\Delta\hat{\sigma}] = [Q][\Delta\sigma][Q]^T \quad (5.20)$$

Considerando que a mudança de eixos é dada por uma transformação ortogonal, isto é $[Q]^T = [Q]^{-1}$, pode-se então calcular o acréscimo de tensões referido ao sistema global por transformação reversa da seguinte forma:

$$[\Delta\sigma] = [Q]^T [\Delta\hat{\sigma}] [Q] \quad (5.21)$$

Na implementação deste trabalho, os valores dos acréscimos de tensão principais, $[\Delta\hat{\sigma}]$, são obtidos pelo modelo constitutivo neural, tal como indica a Eq. (5.10).

O modelo constitutivo *Magola11* é composto de um vetor de entrada com 20 valores, duas camadas ocultas de 120 e 100 neurônios e uma camada de saída de 3 neurônios (os acréscimos de tensões principais) como ilustrado na Figura 5.2.

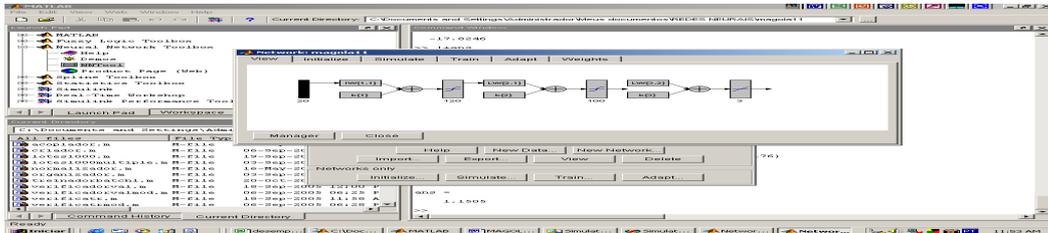


Figura 5.2. Arquitetura da rede *Magola11*.

As 20 componentes do vetor de entrada são determinadas a seguir:

1. Fator de forma dos grãos, n_s .
2. Tensão de confinamento inicial, σ_{c0}
3. $\epsilon_{\max} - \epsilon_{\min}$.
4. Índice de vazios inicial, e_0 .
5. Valor da deformação principal do estágio atual na direção 1, ϵ_{1i} .
6. Valor da deformação principal do estágio atual na direção 2, ϵ_{2i} .
7. Valor da deformação principal do estágio atual na direção 3, ϵ_{3i} .
8. Valor da deformação principal do estágio anterior na direção 1, ϵ_{1i-1} .
9. Valor da deformação principal do estágio anterior na direção 2, ϵ_{2i-1} .
10. Valor da deformação principal do estágio anterior na direção 3, ϵ_{3i-1} .
11. Valor do acréscimo de deformação principal do estágio atual na direção 1, $\Delta\epsilon_{1i}$.
12. Valor do acréscimo de deformação principal do estágio atual na direção 2, $\Delta\epsilon_{2i}$.
13. Valor do acréscimo de deformação principal do estágio atual na direção 3, $\Delta\epsilon_{3i}$.
14. Valor da tensão principal do estágio atual na direção 1, σ_{1i} .
15. Valor da tensão principal do estágio atual na direção 2, σ_{2i} .
16. Valor da tensão principal do estágio atual na direção 3, σ_{3i} .
17. Valor da tensão principal do estágio anterior na direção 1, σ_{1i-1} .
18. Valor da tensão principal do estágio anterior na direção 2, σ_{2i-1} .
19. Valor da tensão principal do estágio anterior na direção 3, σ_{3i-1} .

20. Valor do invariante de Matsuoka-Nakai do estágio atual, X_i .

As funções de transferência entre a camada de entrada e a primeira camada oculta, bem como entre as duas ocultas, são do tipo tangente hiperbólica. A função de transferência entre a segunda camada oculta e a camada de saída é do tipo limiar lógico.

Os pesos obtidos durante o processo de treinamento foram armazenados em três matrizes (ver Figura 5.2) $[IW1]_{20 \times 120}$, $[LW1]_{100 \times 120}$ e $[LW2]_{3 \times 100}$, correspondentes às ligações entrada-1ª camada oculta, 1ª camada oculta-2ª camada oculta, e 2ª camada oculta-saída, respectivamente. Os biases correspondentes foram armazenados em três vetores $\{b1\}_{120 \times 1}$, $\{b2\}_{100 \times 1}$ e $\{b3\}_{3 \times 1}$. Estas matrizes e vetores são apresentados no Apêndice D, e estão gravadas num arquivo específico que é chamado pelo programa.

O processo de implementação do modelo constitutivo neural é apresentado em forma de algoritmo na Figura 5.3. Nela são explicados os passos de programação que permitem à rede agir como operador constitutivo dentro de um código de elementos finitos. Nessa figura se observa como a RNA realiza o mapeamento entre acréscimos de deformação principal e acréscimos de tensão principal (passo 5) em substituição da relação elasto-plástica convencional. De posse dos valores do vetor de acréscimos de deformação principal, $[\Delta \hat{\epsilon}]$, e dos valores dos pesos e biases armazenados, assim como das entradas restantes, o programa calcula os acréscimos de tensões principais, para um dado estágio k , de acréscimo de deformações, de acordo com a seguinte expressão:

$$\Delta \sigma_k = \text{matmul}(LW2, (\tanh(\text{matmul}(LW1, (\tanh(\text{matmul}(IW, P) + b1)))) + b2))) + b3 \quad (5.22)$$

onde “matmul” é um operador tal que $\text{matmul}(A, B)$ retorna o produto entre duas matrizes A e B.

Em relação ao tamanho dos acréscimos de deformação que devem ser considerados na simulação do modelo, é necessário observar a distribuição dos valores que foram usados no treinamento da rede neural. A Figura 5.4 apresenta os histogramas de frequência dos acréscimos de deformação usados no treinamento da rede *Magola11* (gráficos da linha superior). Observam-se na mesma figura os valores médios e os correspondentes desvios padrões.

Nos histogramas da Figura 5.4 é clara a concentração de valores ao redor do valor médio para cada um dos três acréscimos de deformação. Isto indica que a rede deverá responder adequadamente para simulações nas quais o vetor de acréscimos de deformação esteja o mais perto possível dos valores médios usados no treinamento. Não é necessário, e de

fato seria improdutivo, usar os mesmos valores que foram usados no treinamento, porém a distribuições dos acréscimos de deformação usados nas simulações do modelo, deverão ser próxima às usadas no treinamento, tal como se apresenta nos gráficos da linha inferior da Figura 5.4, correspondente aos conjuntos de validação usados para a simulação final da rede.

PARÂMETROS	Parâmetro 1. Fator de forma, n_s Parâmetro 2. $e_{\max}-e_{\min}$
ENTRADA	Condição inicial 1. Tensão confinante inicial, σ_{c0} (kPa) Condição inicial 2. Índice de vazios inicial, e_0
SAIDA	Três componentes principais de acréscimos de tensão $\Delta\sigma$
Passo 1	Faça $k=1$. (primeiro ponto de carregamento) Faça $\epsilon_k=0$. (Entrada: vetor de deformações do estágio de carregamento atual) $\epsilon_{k-1}=0$. (Entrada: vetor de deformações do estágio de carregamento anterior) $\sigma_k=\sigma_{c0}$. (Entrada: vetor de tensões do estágio de carregamento atual) $\sigma_{k-1}=0$. (Entrada: vetor de tensões do estágio de carregamento anterior) Calcule $X_k=I1_k * I2_k / I3_k$ (Entrada)
Passo 2	Aplicar as transformações correspondentes ao n_s , (Eq.4.11), $e_{\max}-e_{\min}$, (Eq. 4.12) e_p , (Eq. 4.12) e ao σ_{k-1} (Eq. 4.14).
Passo 3	Imponha $\Delta\epsilon_k$ (Entrada: Primeiro vetor de acréscimos de deformação)
Passo 4	Normalizar todas as entradas para o intervalo [-1, 1]
Passo 5	Calcular $\Delta\sigma_k = \text{matmul}(\mathbf{LW2}, (\tanh(\text{matmul}(\mathbf{LW1}, (\tanh(\text{matmul}(\mathbf{IW}, \mathbf{P}) + \mathbf{b1})))) + \mathbf{b2}))) + \mathbf{b3}$ onde \mathbf{P} é o vetor de entrada para a primeira camada da rede (20 entradas)
Passo 6	Aplicar a normalização inversa aos valores do $\Delta\sigma_k$ e do $\Delta\epsilon_k$
Passo 7	Faça $k=k+1$ Faça $\epsilon_{k+1} = \epsilon_k + \Delta\epsilon_k$ (Atualização das deformações) $\sigma_{k+1} = \sigma_k + \Delta\sigma_{kd}$ (Atualização das tensões) $X_{k+1} = I1_{k+1} * I2_{k+1} / I3_{k+1}$ (Atualização do Invariante X)
Passo 8	Transforme o σ_{k+1} (Eq. 4.14)
Passo 9	Imponha $\Delta\epsilon_{k+1}$
Passo 10	Repetir os passos 4 a 9 até $k=n$, n é o número total de estágios de carregamento
Passo 11	PARE
Notas:	1. O operador $\text{matmul}(A,B)$ multiplica as matrizes A e B que devem ter as dimensões apropriadas. 2. Em negrito estão os vetores e as matrizes, em fonte normal os escalares. 3. As transformações a que se referem alguns dos passos do algoritmo estão explicadas no item de pre-processamento da base de dados do modelo neural 6 no capítulo anterior, as equações correspondentes são referidas entre parênteses.

Figura 5.3. Algoritmo de implementação da RN constitutiva *Magola11* no programa ALLFINE.

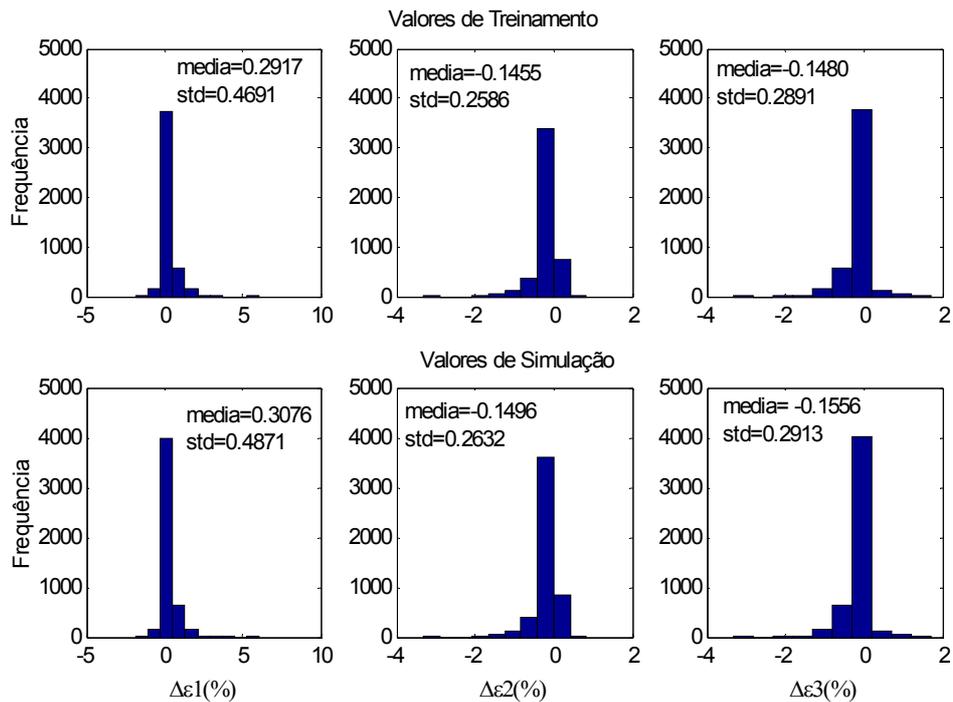


Figura 5.4. Histogramas de frequência das três componentes dos acréscimos de deformação usados no treinamento da rede *Magola11*.

Os resultados desta simulação, ainda sem ter sido implementada no MEF, são apresentados como linhas contínuas nos gráficos das figuras que aparecem no item seguinte. Nessas simulações os acréscimos de deformação usados foram definidos pelos valores médios de treinamento para cada uma das componentes principais.

5.3 – CASOS DE APLICAÇÃO

Uma vez implementado o modelo neural constitutivo dentro do programa de elementos finitos ALLFINE foram realizadas algumas simulações com o intuito de verificar o desempenho do conjunto.

5.3.1 – SIMULAÇÃO DE ENSAIOS DE LABORATÓRIO

Considerando que o modelo neural foi treinado a partir de informação experimental real e aceitando que o comportamento das amostras de areia nos diferentes testes corresponde ao comportamento da areia ao nível de ponto constitutivo, decidiu-se iniciar o processo de

validação do sistema híbrido (SH) mediante a simulação de diferentes trajetórias de tensão considerando um elemento cúbico simples com 8 nós, tal como se apresenta na Figura 5.5.

O elemento considerado apresenta o nó 1 completamente restrito na direção normal de coordenadas. Os 3 lados em contato com os eixos de coordenadas estão fixos, permitindo somente carregamentos nos 3 lados restantes. Só o nó da diagonal oposta à origem (nó 7) está completamente livre. Para compatibilizar o funcionamento do SH com a simulação desejada, foram impostos nos lados livres do elemento acréscimos de deformação compatíveis com os conjuntos de treinamento da RNA.

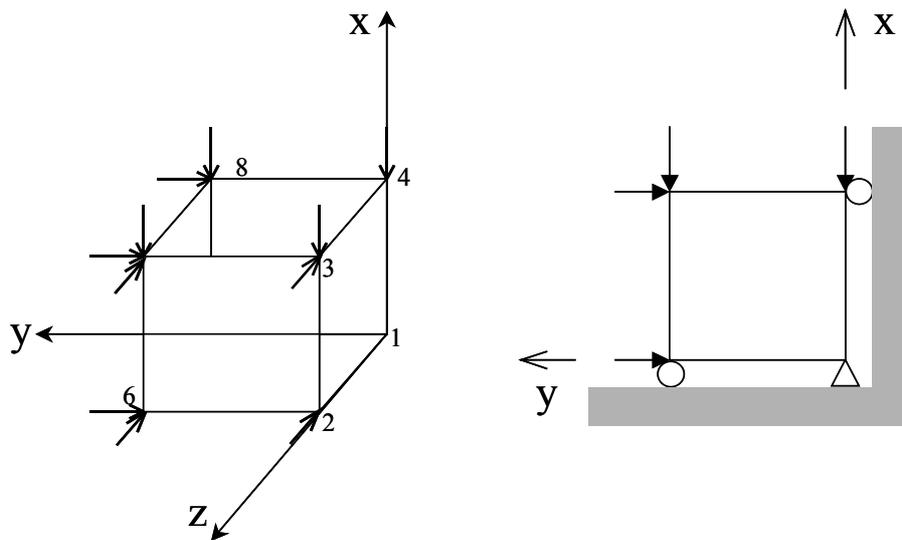


Figura 5.5. Diagrama mostrando um elemento cúbico de 8 nós. Nos lados livres são impostos acréscimos de deformação.

Tal como já foi mencionado anteriormente, o vetor de acréscimos de deformação imposto age como uma variável independente dentro da RN constitutiva. No entanto, deve-se destacar que existe uma relação entre o nível de deformação para um instante dado de carregamento e o tamanho ótimo dos acréscimos de deformação a serem impostos. Durante o processo de digitalização das curvas tensão-deformação experimentais e posteriormente no processo de interpolação que foi realizado para formar a base de dados original, foram impostos tamanhos de acréscimos de deformação pequenos naquelas regiões das curvas mais próximas à origem, e tamanhos maiores nas regiões ou porções finais das mesmas. Tal procedimento permitia manter certo controle sobre a distribuição do tamanho do vetor de acréscimos de tensão correspondente.

Quando simulada a rede neural para cada um dos testes considerados na base de dados, escolheu-se o tamanho médio de acréscimo de deformação de cada ensaio particular. No

entanto, para o SH isto é pouco prático, sendo ideal incorporar no código de elementos finitos informação que lhe permita definir o tamanho ótimo de acréscimo de deformação para o nível de deformação em qualquer estágio de carregamento. Na execução das simulações deste trabalho, tal implementação, por razões de tempo, não foi realizada, sendo necessário então testar diferentes valores de acréscimos de tensão ou deformação para cada caso simulado, procurando a combinação que fornecesse o mínimo valor de resíduo. Sendo que o autor, por causa do conhecimento adquirido sobre o comportamento do modelo, tinha desenvolvido um senso dos tamanhos ótimos de simulação, o processo de validação foi rápido. Isto foi válido ainda para a fase de análise do problema de contorno (sapata) que é apresentado em seções posteriores.

Nas simulações dos ensaios de laboratório observou-se que o tamanho ótimo do acréscimo de deformação ($\Delta\varepsilon_{ot}$) podia ser adequadamente estimado como o valor médio dos acréscimos usados durante o treinamento, reconhecendo cinco casos que dependem da magnitude da deformação principal maior alcançada no ensaio a simular, tal como se apresenta a seguir:

- Caso (a). $\varepsilon_{max}=2,5\%$, $\Delta\varepsilon_{ot}=0,01\%$. Neste caso o $\Delta\varepsilon_{ot}$ foi obtido como uma aproximação à média dos valores absolutos de todos os $\Delta\varepsilon$ do conjunto.
- Caso (b). $\varepsilon_{max}=5,0\%$, $\Delta\varepsilon_{ot}=0,15\%$.
- Caso (c). $\varepsilon_{max}=15,0\%$, $\Delta\varepsilon_{ot}=0,3\%$.
- Caso (d). $\varepsilon_{max}=30,0\%$, $\Delta\varepsilon_{ot}=0,6\%$.
- Caso (e). $\varepsilon_{max}=50,5\%$, $\Delta\varepsilon_{ot}=0,5\%$.

Os casos anteriores mostram que o valor ótimo do acréscimo de deformação depende do valor da deformação máxima alcançada no ensaio.

Definidos os tamanhos ótimos dos acréscimos de deformação é possível calcular o número de incrementos de carga de cada teste para as simulações com o SH. Foram escolhidos então alguns testes de dificuldade variável para simular as trajetórias de tensão correspondentes. Os resultados dessas simulações se apresentam nos gráficos da Figura 5.6. Cada teste foi simulado com um número variável de incrementos de carregamento para verificar a sensibilidade do modelo. Em cada gráfico foram incluídos, para efeito de comparação, os pontos experimentais e a simulação da rede *Magola11*. Os resultados destas simulações são analisados na Tabela 5.1.

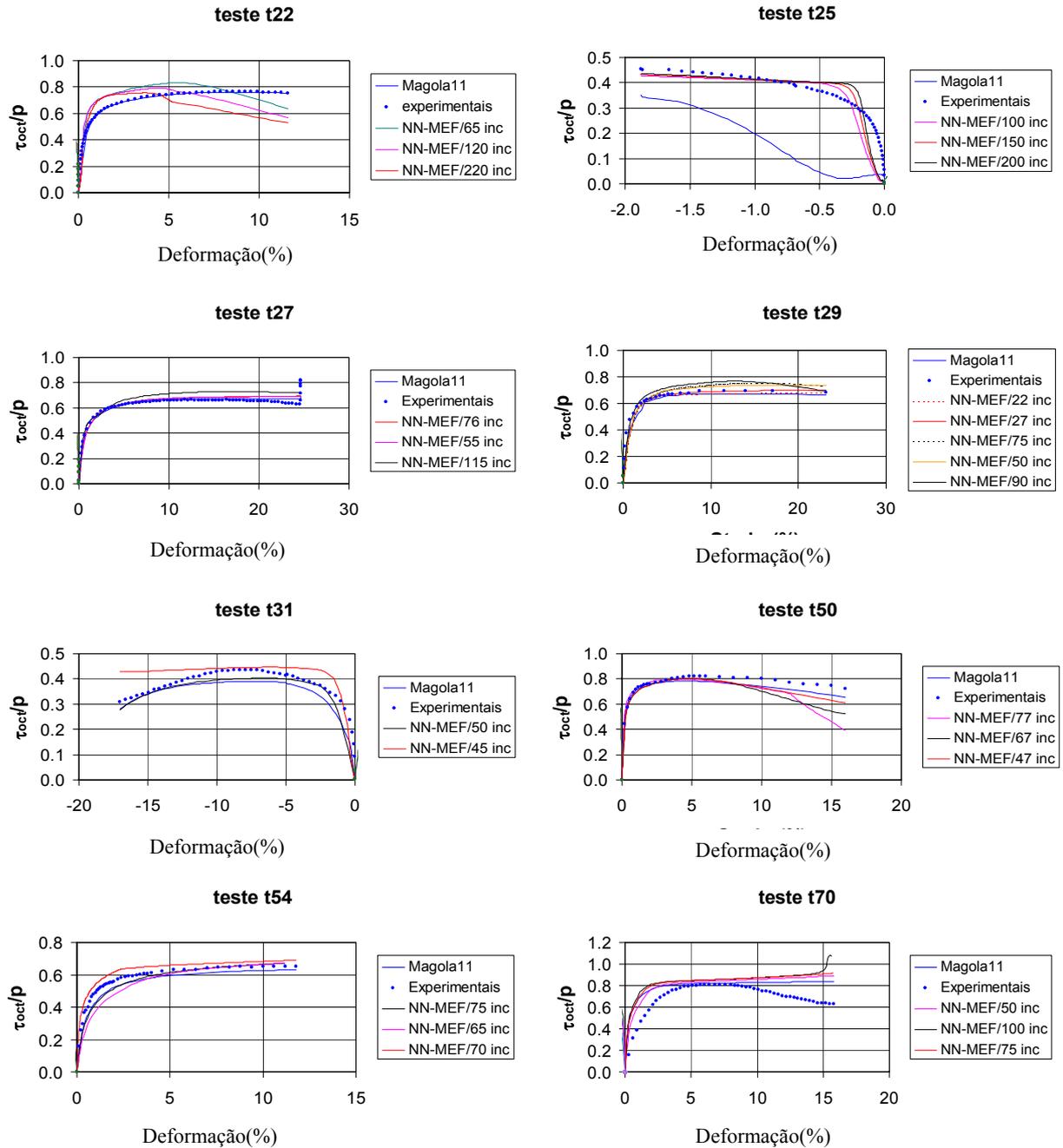


Figura 5.6. Comparação de simulações do sistema híbrido ALLFINE-RNA para diferentes incrementos de carga.

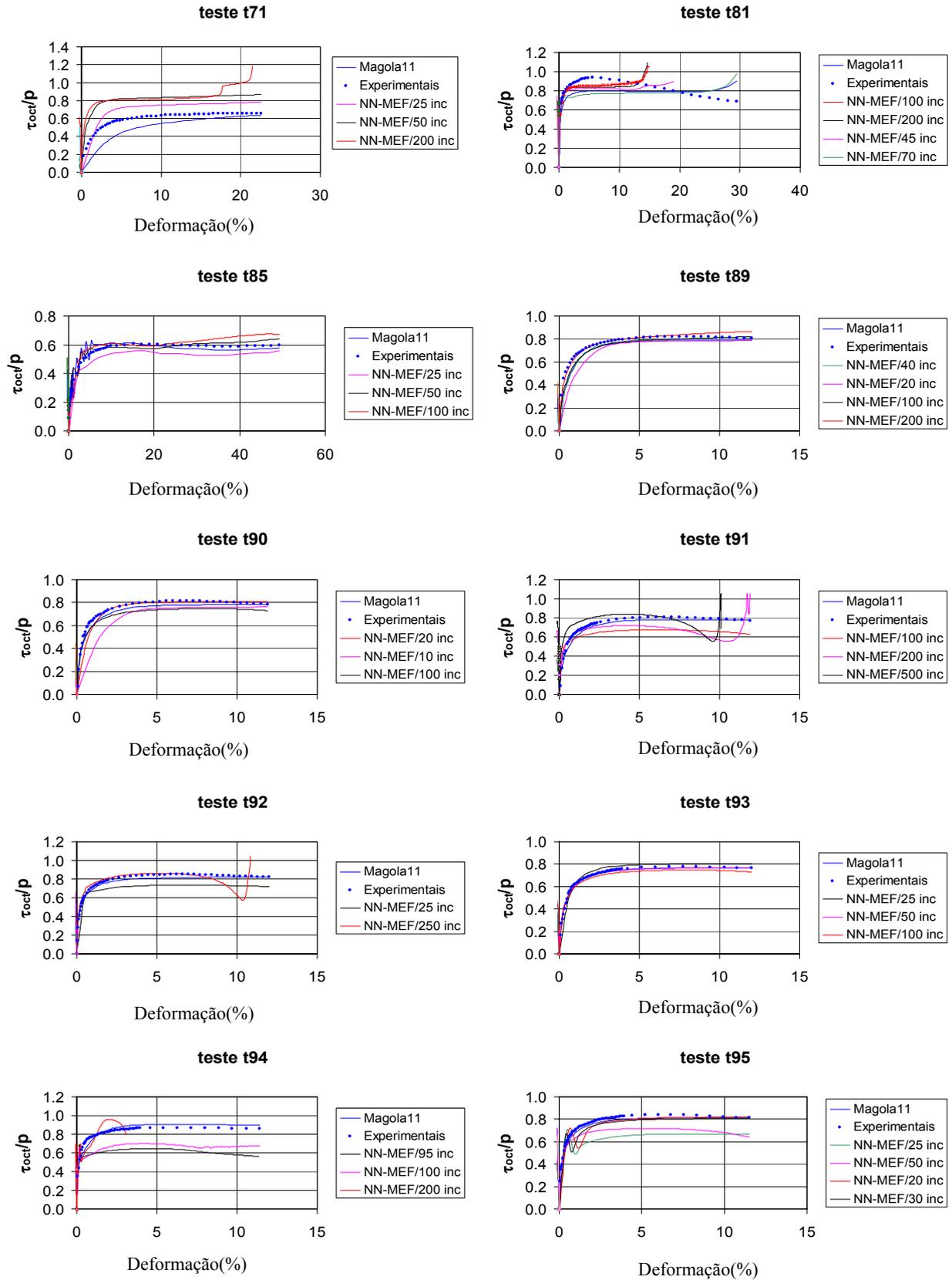


Figura 5.6. (Continuação) Comparação de simulações do sistema híbrido ALLFINE-RNA para diferentes incrementos de carga.

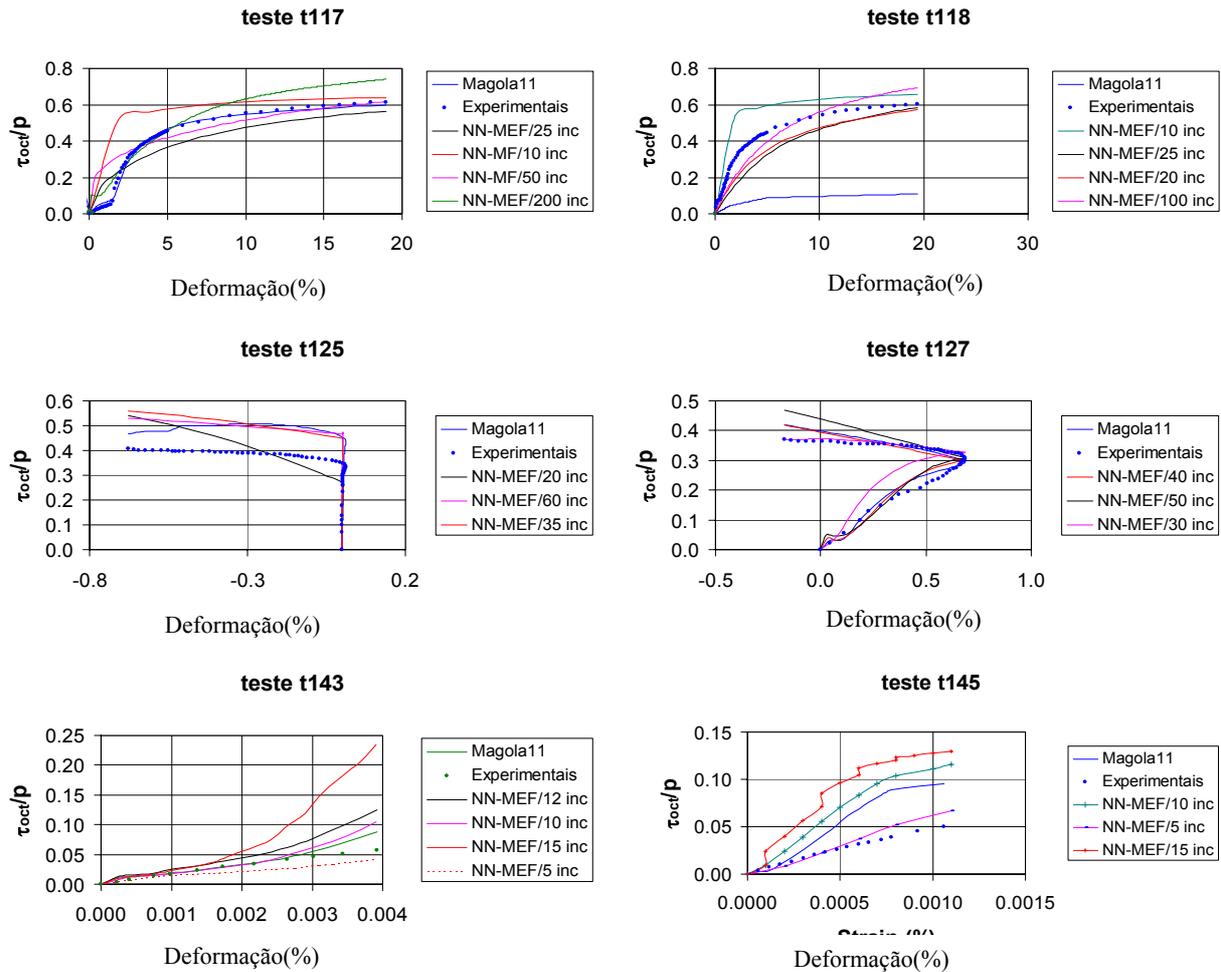


Figura 5.6. (Continuação) Comparação de simulações do sistema híbrido ALLFINE-RNA para diferentes incrementos de carga.

Para o caso do teste 25 a simulação com a RNA foi feita com um valor médio de acréscimo de deformação principal de 0,04% ($\Delta\varepsilon_{max}/N^{\circ}$ pontos do teste=1,8%/50). Para este teste em particular, a primeira parte da trajetória correspondente a deformações menores que 0,5% foi discretizada em acréscimos de 0,02% para compor o conjunto de treinamento, enquanto que a parte final do teste teve acréscimos da ordem de 0,03% até 0,06%. Isto explica a má resposta de simulação da RNA na primeira parte do teste e a sua tendência a corrigir a trajetória na parte final do mesmo. No início da trajetória a RNA simulada com acréscimos constantes de tamanho 0,04% não fornece respostas adequadas devido ao fato deste valor ser o dobro do valor ideal de simulação. Por outro lado, para a simulação do SH foi calculado o número ideal de incrementos de carga de acordo com o tamanho da deformação máxima do teste (1,8%). Considerando um valor de acréscimo ótimo de 0,01% se obtém um número ótimo de incrementos de carga de 180. Realizaram-se então três simulações, uma com 200

incrementos (valor próximo do ótimo), e mais duas com 100 e 150 incrementos correspondentes a tamanhos médios de acréscimos de deformação de 0,018% e 0,012%, respectivamente. As três simulações apresentaram desempenhos satisfatórios demonstrando a competência do SH e a sua independência do tamanho dos acréscimos de tensão, sempre que mantidos dentro de um intervalo de valores concordantes com os de treinamento mas não necessariamente coincidentes com eles.

Para os testes restantes apresentados na Figura 5.6 (com exceção dos testes 125, 127, 143 e 145), cujas deformações principais maiores atingiram valores superiores ou próximos a 15 %, a aplicação exata do critério de estimativa do número de incrementos de carga dado anteriormente gerava valores variáveis entre 270 e 300 em média. Sendo que 250 deles correspondiam à primeira parte das trajetórias (deformação até 2,5%), e considerando que este número poderia ser válido para o teste completo, decidiu-se verificar empiricamente a resposta do SH para valores de incrementos não só próximos aos valores ótimos como também para outros valores maiores e menores que este valor.

Na segunda coluna da Tabela 5.1 se apresenta o número de incrementos de carga que foi aplicado ao SH para a simulação de cada teste. A coluna seguinte mostra a deformação principal máxima aproximada do teste. A coluna de resultados inclui uma qualificação da simulação em termos de “bom” ou “ruim” realizada por simples observação entre a proximidade da curva de simulação com a curva experimental. Adicionalmente nessa mesma coluna se descreve se a resposta do SH se mostrou sensível ou não às variações do número de incrementos de carga. Finalmente a última coluna apresenta os valores médios de acréscimo de deformação principal usados para as simulações, calculados a partir do valor da deformação máxima e do número de incrementos de carregamento. Nos casos para os quais o modelo se mostrou sensível, só o valor do acréscimo ótimo foi calculado.

Com o intuito de encontrar alguma relação que permitisse determinar as condições que geram sensibilidade do SH para alguns dos casos analisados, foram analisadas as deformações usadas no treinamento de cada um dos testes considerados “sensíveis”, e comparadas com as deformações ou com os acréscimos de deformação usados nas simulações. Não foi possível encontrar uma relação que explicasse de maneira geral o comportamento “sensível” destes testes. Considerando que o modelo neural foi treinado para mapear um espaço de 20 dimensões em um espaço de três dimensões, acredita-se que não só a relação entre deformações e acréscimos de deformação determina a qualidade da resposta do modelo, mas uma combinação “certa” de todas as entradas para cada caso específico. No entanto, também

é válido afirmar que para a maior parte dos testes, a relação entre os níveis de deformação e os acréscimos de deformação não desempenha um papel preponderante na resposta do modelo, razão pela qual, muitas das simulações apresentadas mostraram pouca ou nenhuma sensibilidade ao número de incrementos de carregamento usado.

Tabela 5.1. Análise dos gráficos da Figura 5.6.

Teste	Número de Incrementos	ϵ_{\max} aprox (%)	Resultado	Tamanho do incremento de
22	65-120-220	11	Bom. Indepeinde dos incrementos	0,17-0,09-0,05
25	100-150-200	2	Bom. Indepeinde dos incrementos	0,01-0,013-0,02
27	55-76-115	25	Bom. Indepeinde dos incrementos	0,22-0,33-0,45
29	22-27-50-75-90	23	Bom. Indepeinde dos incrementos	0,26-0,31-0,46-0,85-1,04
31	45-50	17	Bom. N° Ot. Inc.=45. Sensível	0,38
50	47-67-77	17	Bom. Indepeinde dos incrementos	0,22-0,25-0,36
54	65-70-75	12	Bom. Indepeinde dos incrementos	0,16-0,17-0,18
70	50-75-100	15	Bom. Indepeinde dos incrementos	0,15-0,2-0,3
71	25-50-200	22	Bom. N° Ot. Inc.=25. Sensível	0,88
81	45-70-100-200	30	Bom. N° Ot. Inc.=70. Sensível	0,43
85	25-50-100	50	Bom. Indepeinde dos incrementos	0,5-1,0-2,0
89	20-40-100-200	12	Bom. Indepeinde dos incrementos	0,06-0,12-0,3-0,6
90	10-20-100	12	Bom. Indepeinde dos incrementos	0,12-0,6-1,2
91	100-200-500	13	Bom. N° Ot. Inc.=100. Sensível	0,13
92	25-250	13	Bom. Indepeinde dos incrementos	0,05-0,52

Teste	Número de Incrementos	ε_{\max} aprox (%)	Resultado	Tamanho do incremento de
93	25-50-100	13	Bom. Independe dos incrementos	0,13-0,26-0,52
94	95-100-200	13	Bom. N° Ot. Inc.=95. Sensível	0,14
95	20-25-30-50	12	Bom. N° Ot. Inc.=20-30. Sensível	0,48
117	10-25-50-200	19	Bom. N° Ot. Inc.=50. Sensível	0,38
118	10-20-25-100	20	Bom. Independe dos incrementos	0,2-0,8-1,0-2,0
125	20-35-60	0,7	Ruim. Independe dos incrementos	0,012-0,02-0,035
127	30-40-50	0,7	Bom. Independe dos incrementos	0,014-0,018-0,023
143	5-10-12-15	0,004	Bom. N° Ot. Inc.=10. Sensível	0,0004
145	5-10-15	0,0011	Bom. N° Ot. Inc.=5 Sensível	0,00022

A seguir são apresentados os resultados de simulação do SH usando o elemento cúbico de 8 nós (Figura 5.5), para todos os testes restantes classificados por tipo de trajetória e por tipo de areia. Para cada teste se apresenta o gráficos da relação τ_{oct}/p contra a deformação principal maior. Para todos os gráficos os pontos representam os dados experimentais, a linha contínua de cor azul representa a simulação da rede neural *Magola11*, e a linha contínua de cor vermelha representa a simulação do sistema híbrido ALLFINE-RNA.

Inicialmente são apresentados os resultados da areia de Karlsruhe para trajetórias CTC (Figura 5.7). Observa-se uma excelente aproximação entre os resultados tanto da rede *Magola11* como do sistema híbrido. Por se tratar de uma trajetória relativamente simples e levando em consideração a boa representação desta areia no conjunto de treinamento, era de se esperar uma resposta satisfatória.

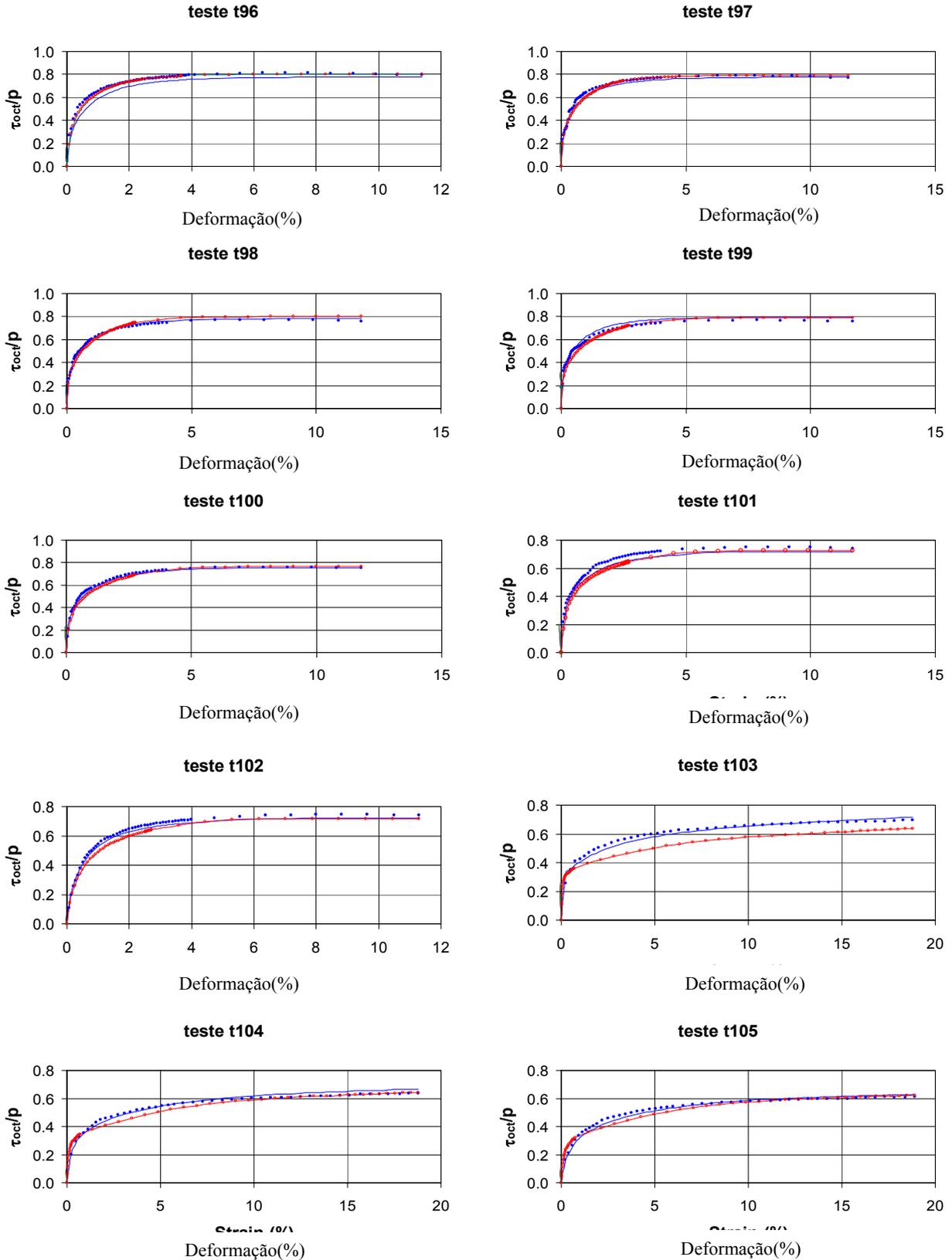


Figura 5.7. Simulações do modelo híbrido ALLFINE-NN. Areia Karlsruhe, trajetórias CTC

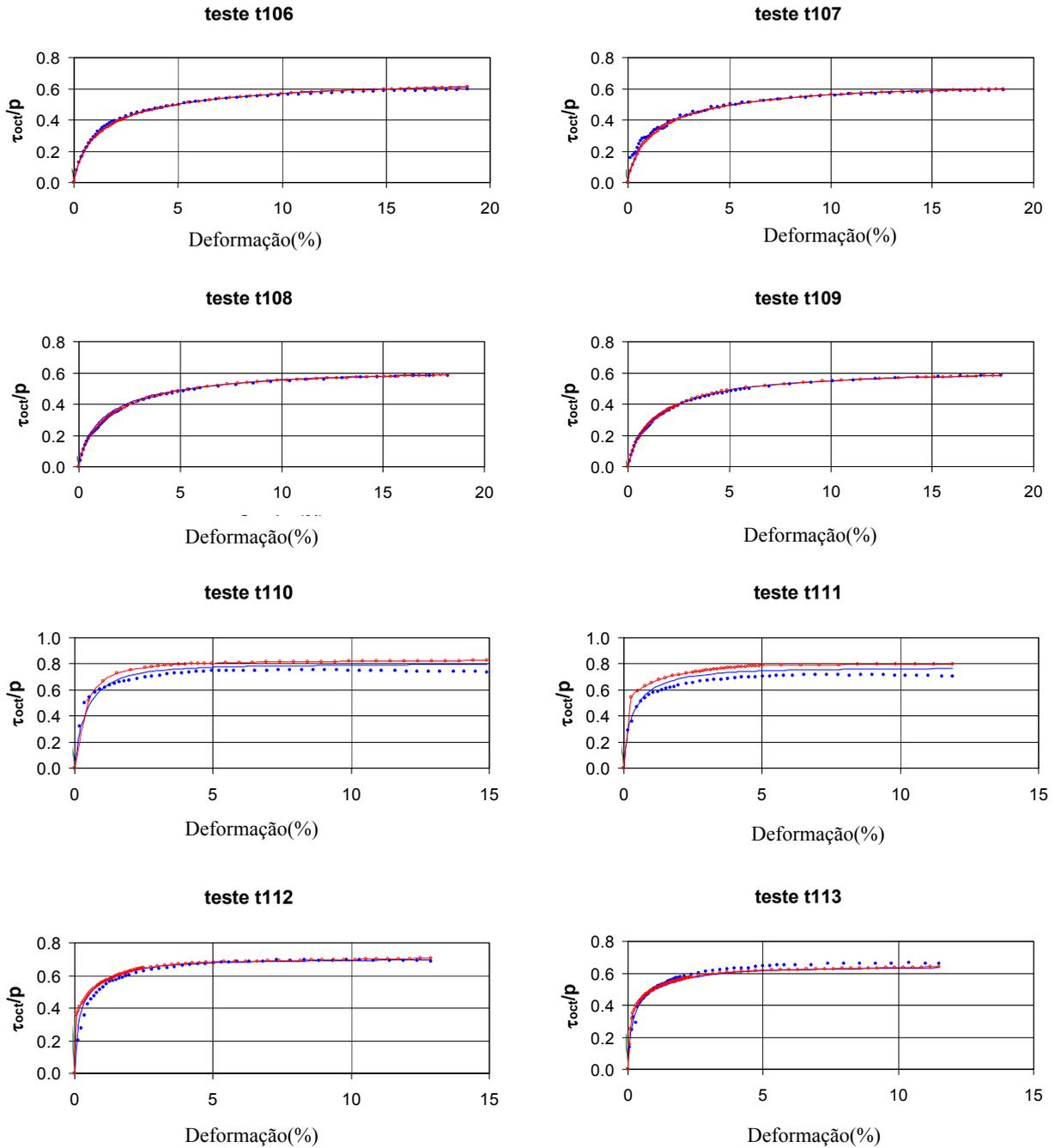


Figura 5.7. (Continuação) Simulações do modelo híbrido ALLFINE-NN. Areia Karlsruhe, trajetórias CTC

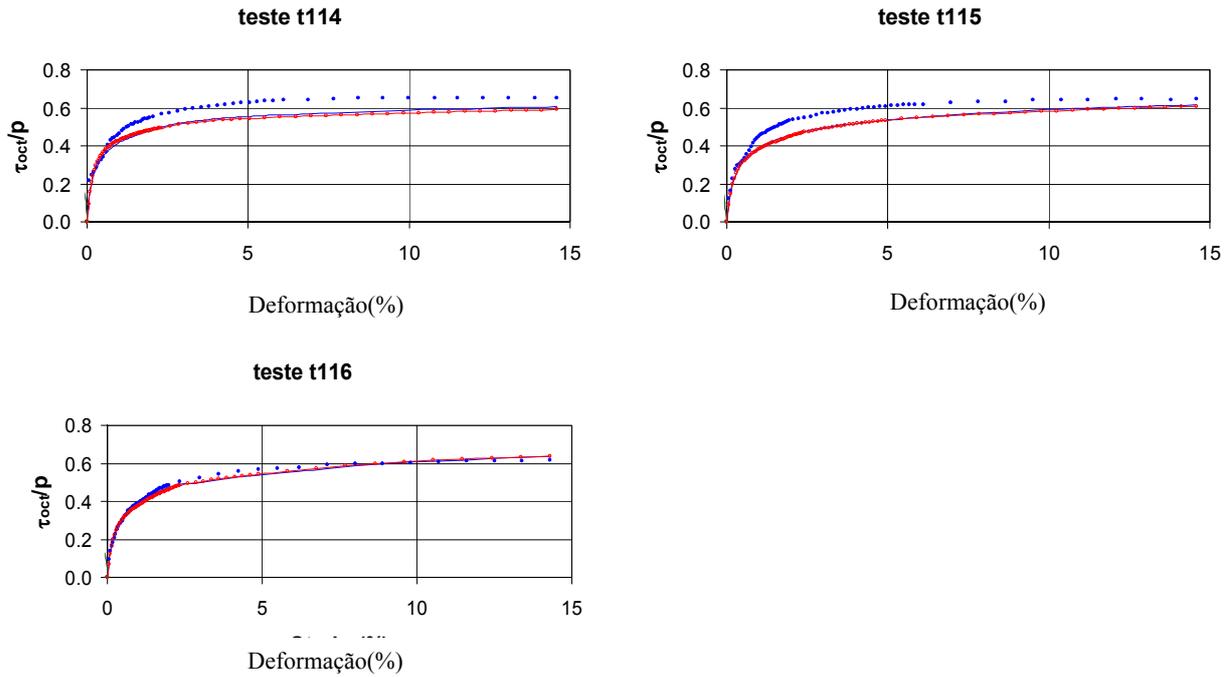


Figura 5.7. (Continuação) Simulações do modelo híbrido ALLFINE-NN. Areia Karlsruhe, trajetórias CTC

As simulações para a areia de Hostun são apresentadas a partir da Figura 5.8 até Figura 5.10, inicialmente para as trajetórias CTC, depois para as trajetórias TC e finalmente para as trajetórias RTE. Observa-se, no caso das trajetórias CTC, que para algumas simulações (testes 83, 84, 124, 142 e 144) o SH respondeu melhor que a RNA isolada. Nas duas trajetórias restantes, o desempenho de ambos sistemas foi semelhante, acompanhando de perto os pontos experimentais.

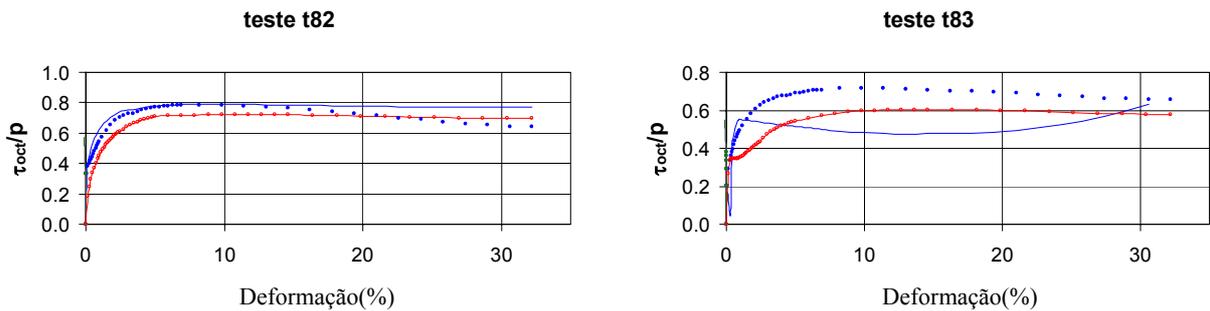


Figura 5.8. Simulações do modelo híbrido ALLFINE-NN. Areia Hostun, trajetórias CTC

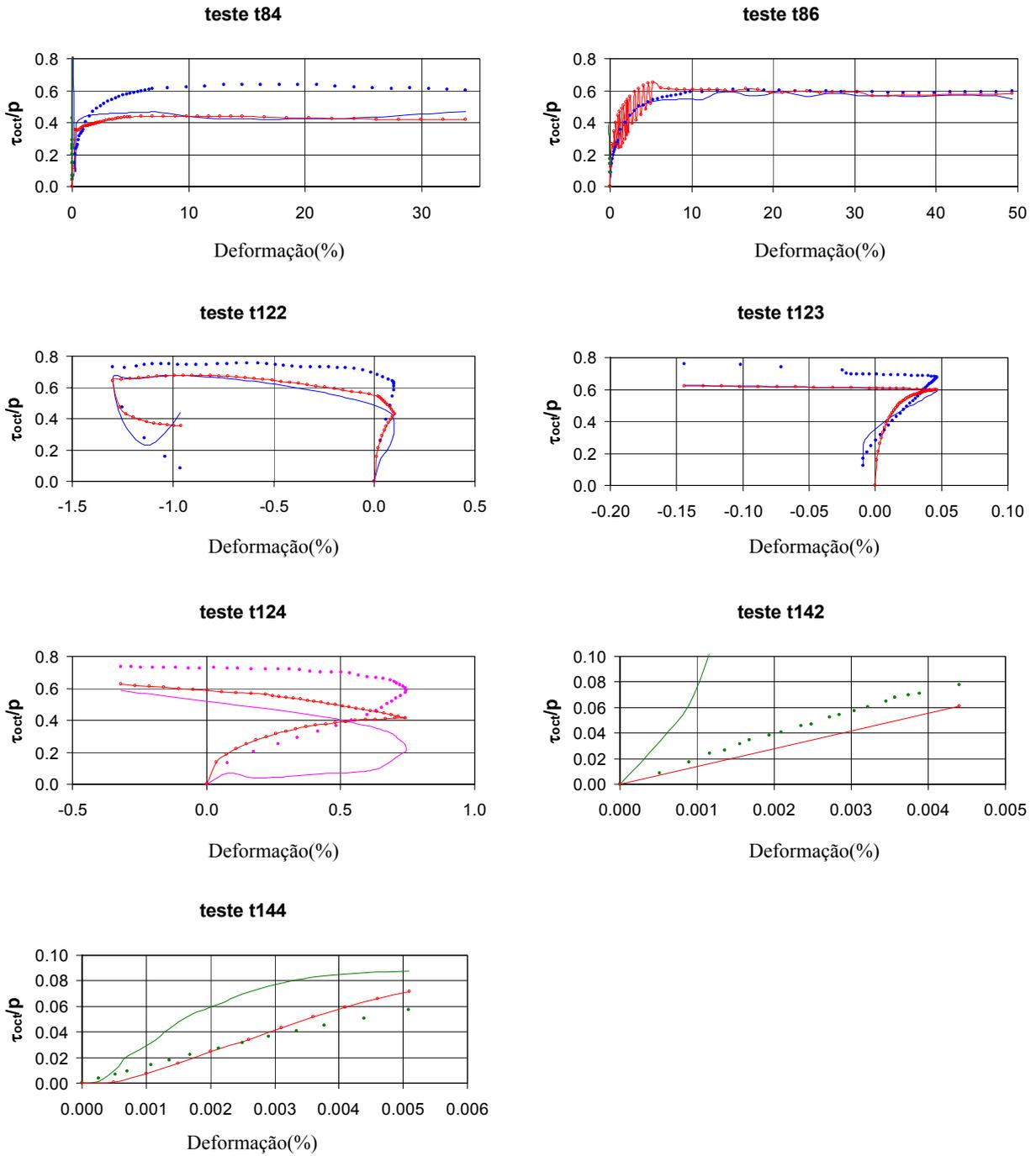


Figura 5.8. (Continuação) Simulações do modelo híbrido ALLFINE-NN. Areia Hostun, trajetórias CTC

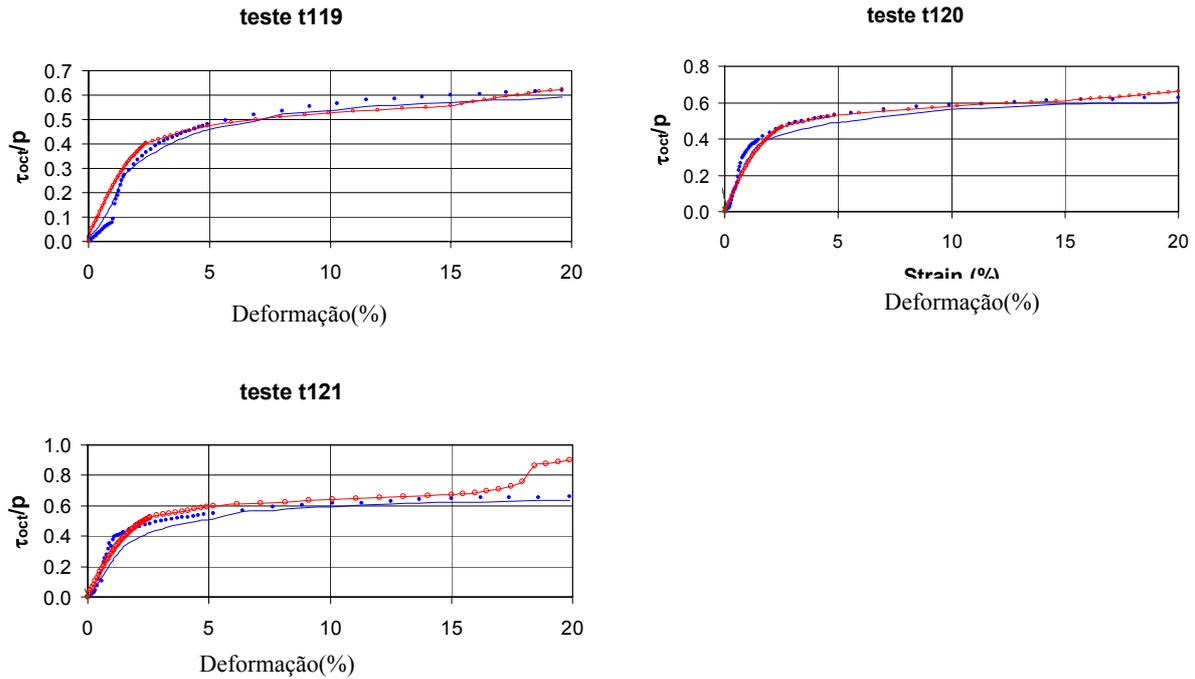


Figura 5.9. Simulações do modelo híbrido ALLFINE-NN. Areia Hostun, trajetórias TC

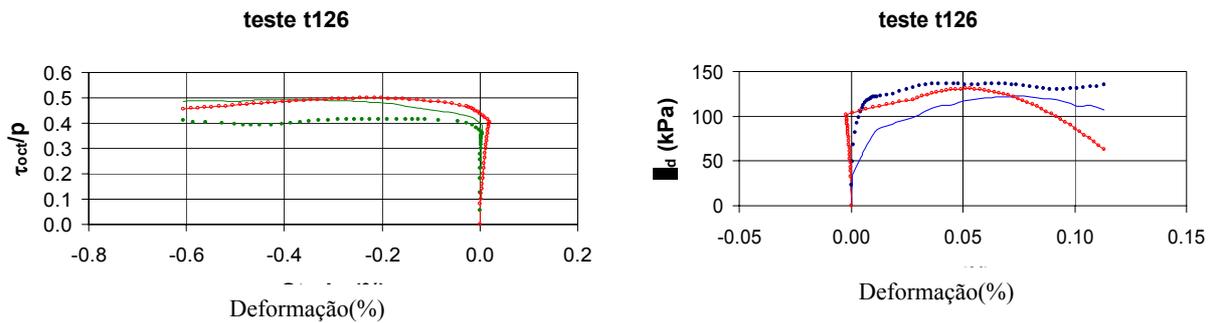


Figura 5.10. Simulações do modelo híbrido ALLFINE-NN. Areia Hostun, trajetórias RTE

A partir da Figura 5.11 até Figura 5.17 são apresentados de maneira contínua os gráficos das simulações da areia de Toyoura para as seguintes trajetórias: CTC, TC, CTE, TE, RTC, RTE e no plano octaédrico. Observa-se como nas trajetórias de extensão por compressão (Figura 5.13) o SH consegue não só identificar adequadamente a tensão de ruptura, mas também acompanhar a trajetória total. Já para os casos de trajetórias de redução (Figura 5.16), o SH apresenta maior erro na identificação da tensão de ruptura, porém acompanha adequadamente o sentido das trajetórias. As trajetórias correspondentes ao plano octaédrico foram bem simuladas pelo SH, que identificou corretamente as combinações de tensão para cada um dos testes simulados.

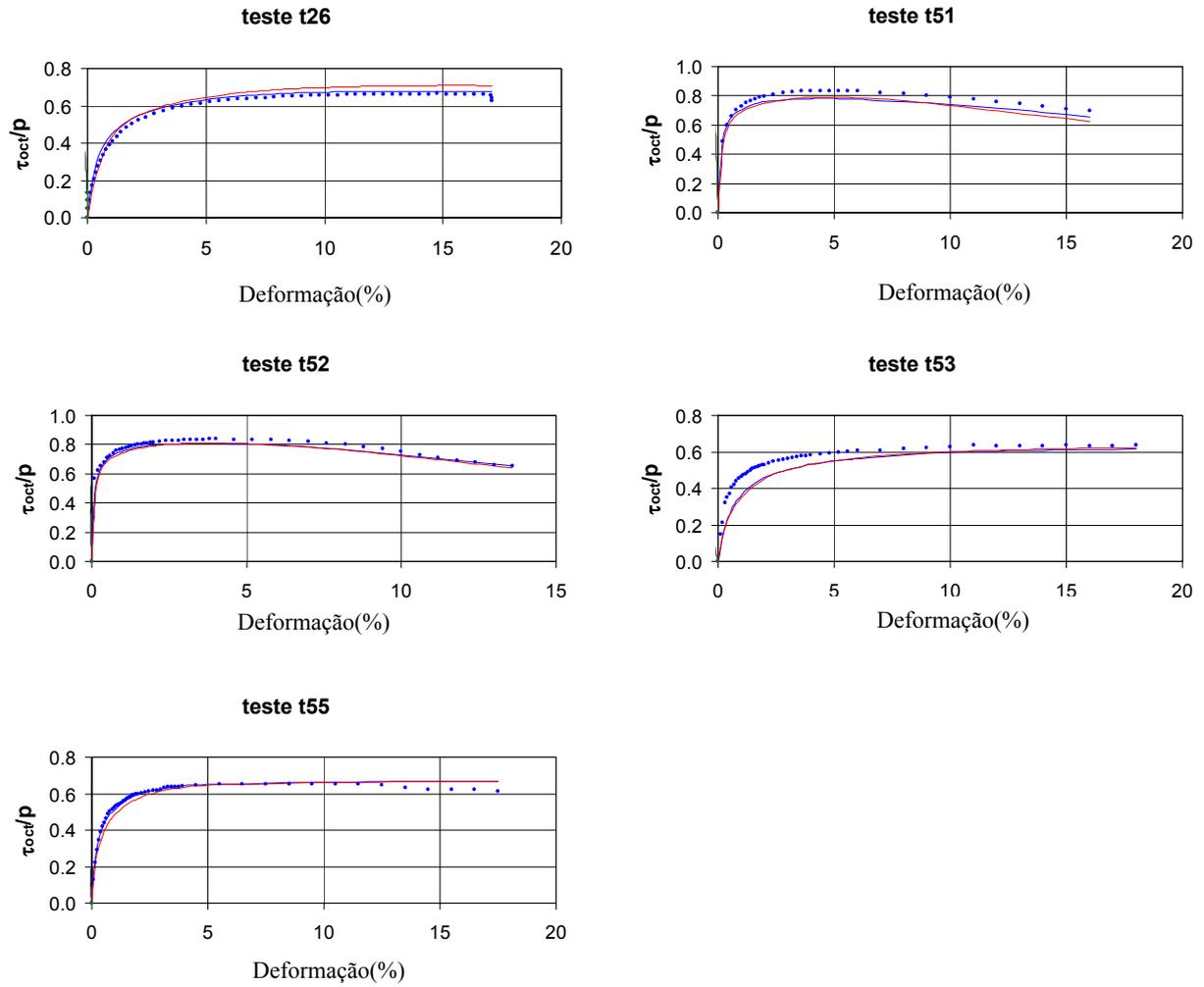


Figura 5.11. Simulações do modelo híbrido ALLFINE-NN. Areia Toyoura, trajetórias CTC

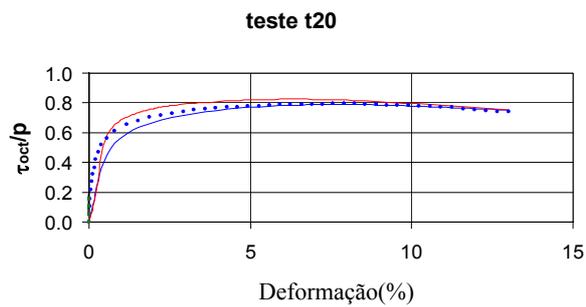


Figura 5.12. Simulações do modelo híbrido ALLFINE-NN. Areia Toyoura, trajetórias TC

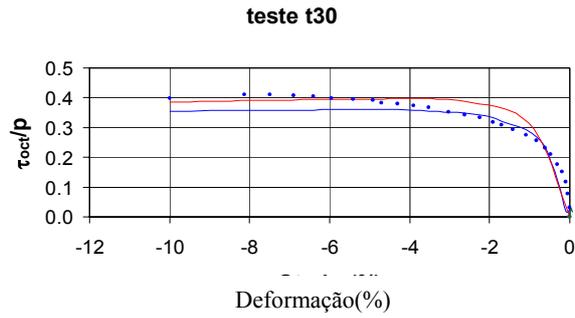


Figura 5.13. Simulações do modelo híbrido ALLFINE-NN. Areia Toyoura, trajetórias CTE

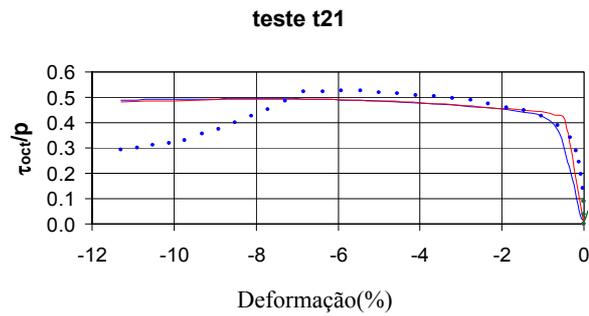


Figura 5.14. Simulações do modelo híbrido ALLFINE-NN. Areia Toyoura, trajetórias TE

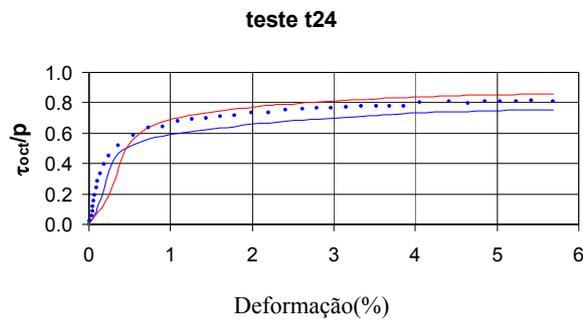


Figura 5.15. Simulações do modelo híbrido ALLFINE-NN. Areia Toyoura, trajetórias RTC

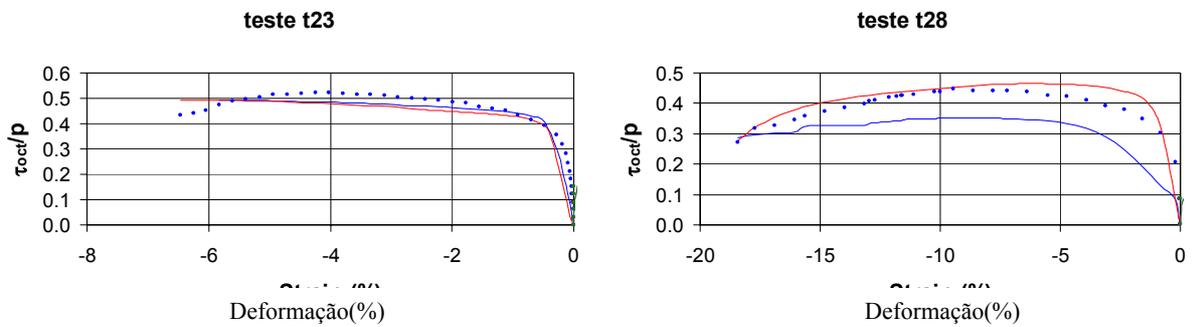


Figura 5.16. Simulações do modelo híbrido ALLFINE-NN. Areia Toyoura, trajetórias RTE

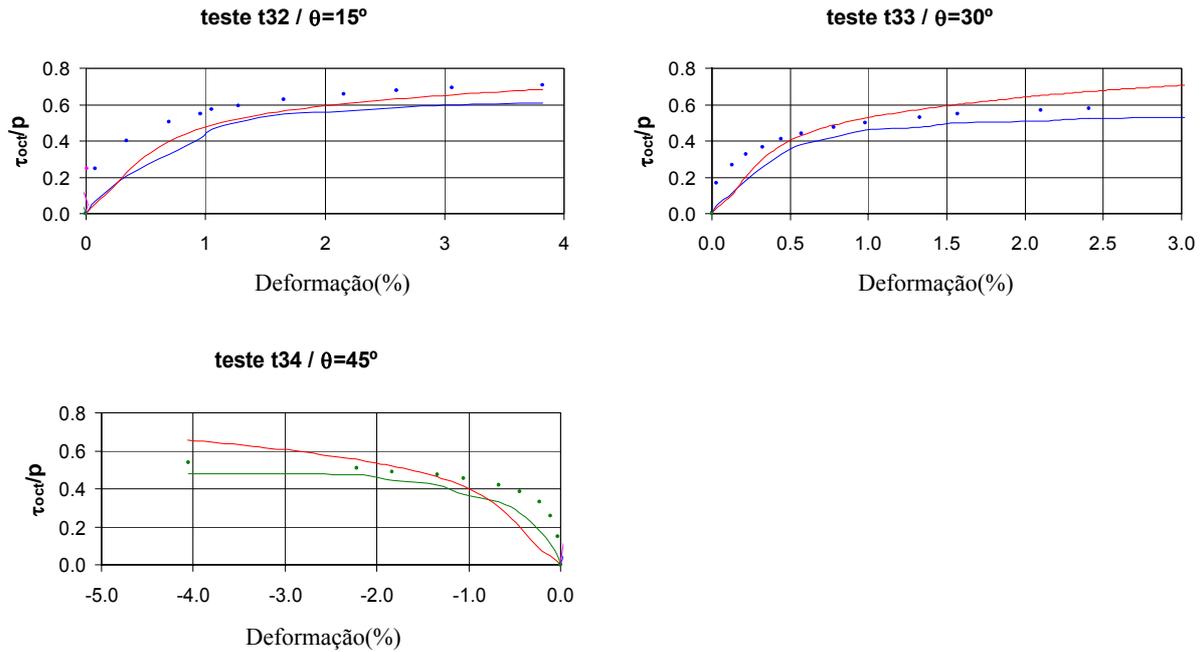


Figura 5.17. Simulações do modelo híbrido ALLFINE-NN. Areia Toyoura, trajetórias no plano octaédrico.

A seguir são apresentadas as simulações para a areia de Cambria. Inicialmente (Figura 5.18) são mostrados os gráficos das trajetórias TC, nos quais o desempenho do SH foi muito próximo dos pontos experimentais. Na Figura 5.19 são apresentados os testes correspondentes às trajetórias de deformação plana, todas elas adequadamente aproximadas pelo modelo. Finalmente são apresentadas as trajetórias de extensão (Figura 5.20 e Figura 5.21) nas quais é notável um comportamento oscilante nos testes 160 e 161. Uma explicação provável para este comportamento sobre-ajustado destes dois testes, está baseada no processo de treinamento seletivo no qual diferentes lotes de informação foram seqüencialmente apresentados à rede. Neste processo, o treinamento era suspenso periodicamente para avaliar o erro de validação do modelo. No entanto tal avaliação só mostra o comportamento do erro de uma forma global, ou seja, como uma média de todos os pontos simulados. Em consequência, algumas combinações das variáveis de entrada podem estar sendo apresentadas de maneira excessiva gerando posteriormente fenômenos locais de sobre-ajuste, que não interferem com o desempenho global do modelo. Esta hipótese é reforçada pelo fato de os testes 159 e 161 se encontrarem no primeiro lote de treinamento da rede, e em consequência foram apresentados insistentemente ao longo de todo o período de treinamento. O teste 160 difere dos dois anteriores só na tensão confinante inicial, sendo semelhantes em todas as demais variáveis. A

rede então memorizou o comportamento dos testes 159 e 161, e gerou resposta sobre-ajustada para o teste intermediário (160).

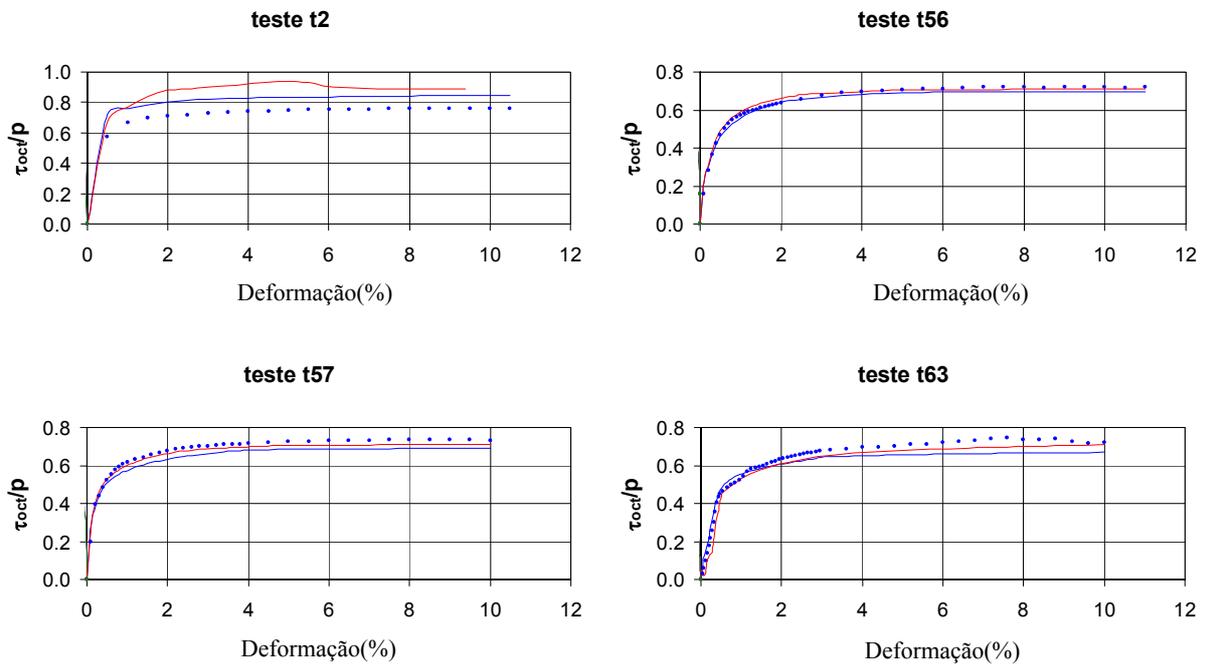


Figura 5.18. Simulações do modelo híbrido ALLFINE-NN. Areia Cambria TC.

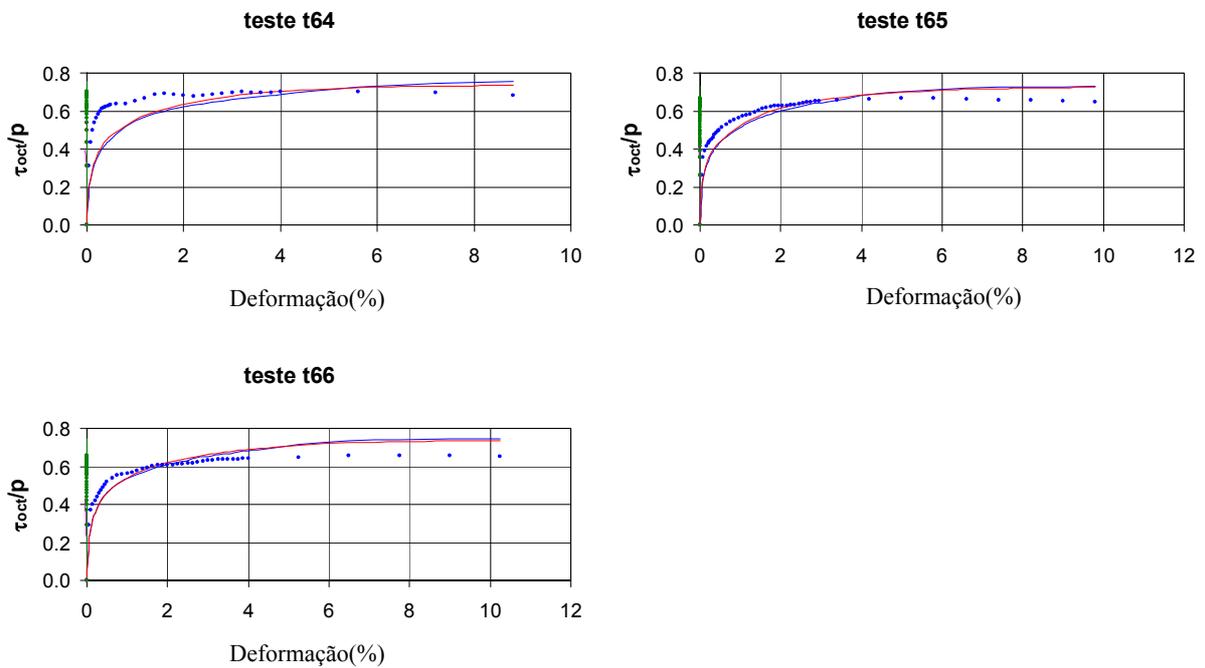


Figura 5.19. Simulações do modelo híbrido ALLFINE-NN. Areia Cambria deformação plana

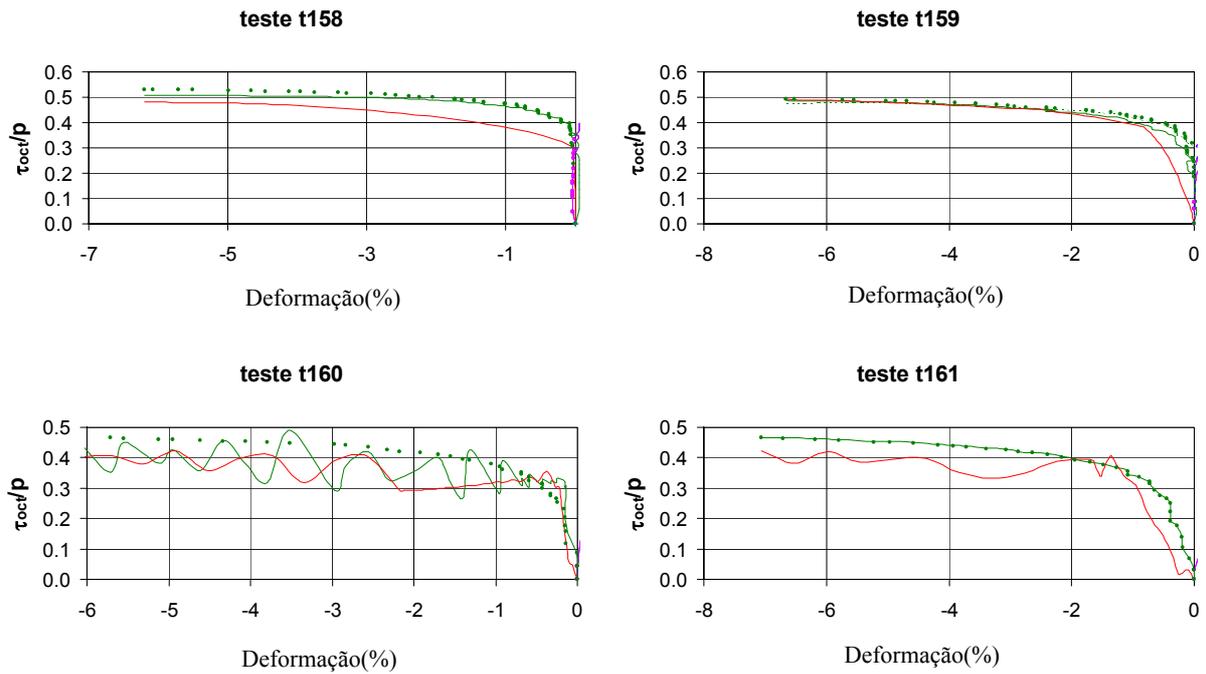


Figura 5.20. Simulações do modelo híbrido ALLFINE-NN. Areia Cambria CTE

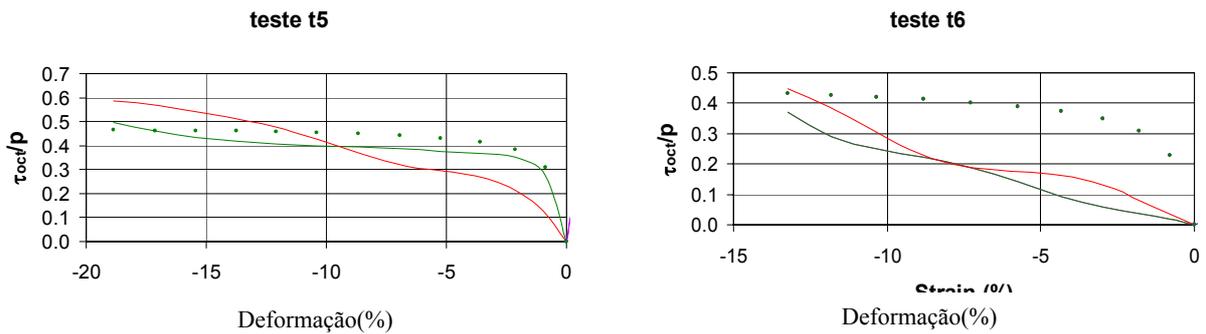


Figura 5.21. Simulações do modelo híbrido ALLFINE-NN. Areia Cambria TE

As simulações para a areia de Monterrey são apresentadas a partir da Figura 5.22 até a Figura 5.24. As piores simulações apareceram nos testes 7 e 8, correspondentes às trajetórias TC. A causa provável é uma escassa representação no conjunto de treinamento ou deficiência no número de apresentações à rede. Já os testes TE, e as simulações no plano octaédrico se mostraram satisfatórias.

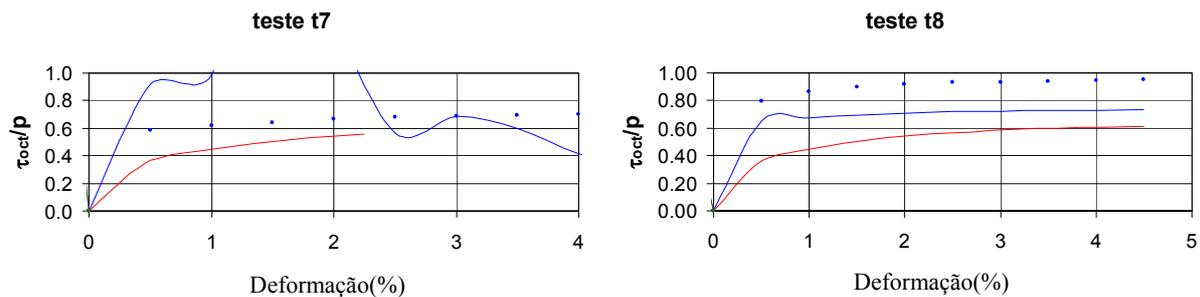


Figura 5.22. Simulações do modelo híbrido ALLFINE-NN. Areia Monterrey TC

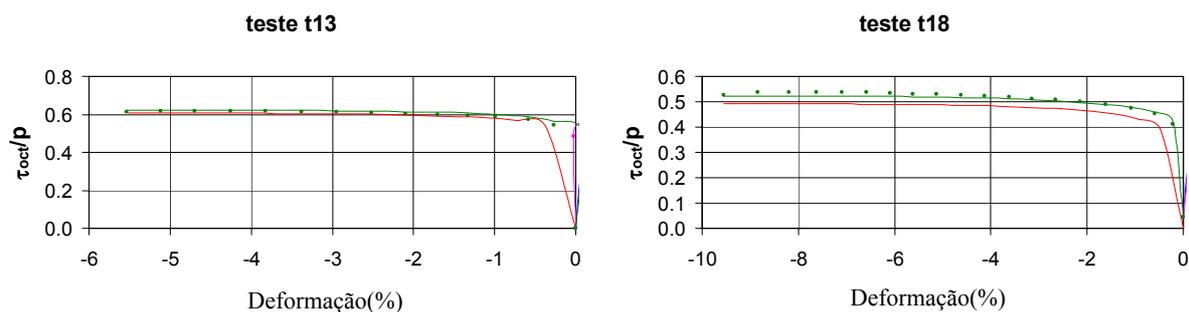


Figura 5.23. Simulações do modelo híbrido ALLFINE-NN. Areia Monterrey TE

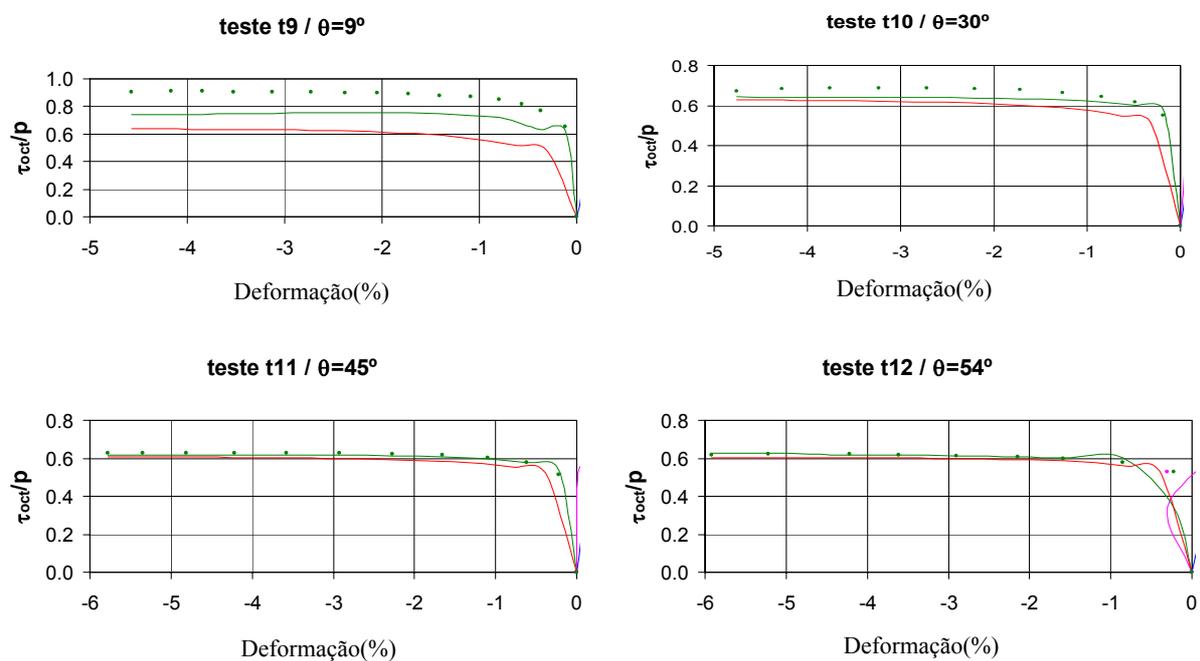


Figura 5.24. Simulações do modelo híbrido ALLFINE-NN. Areia Monterrey plano octaédrico

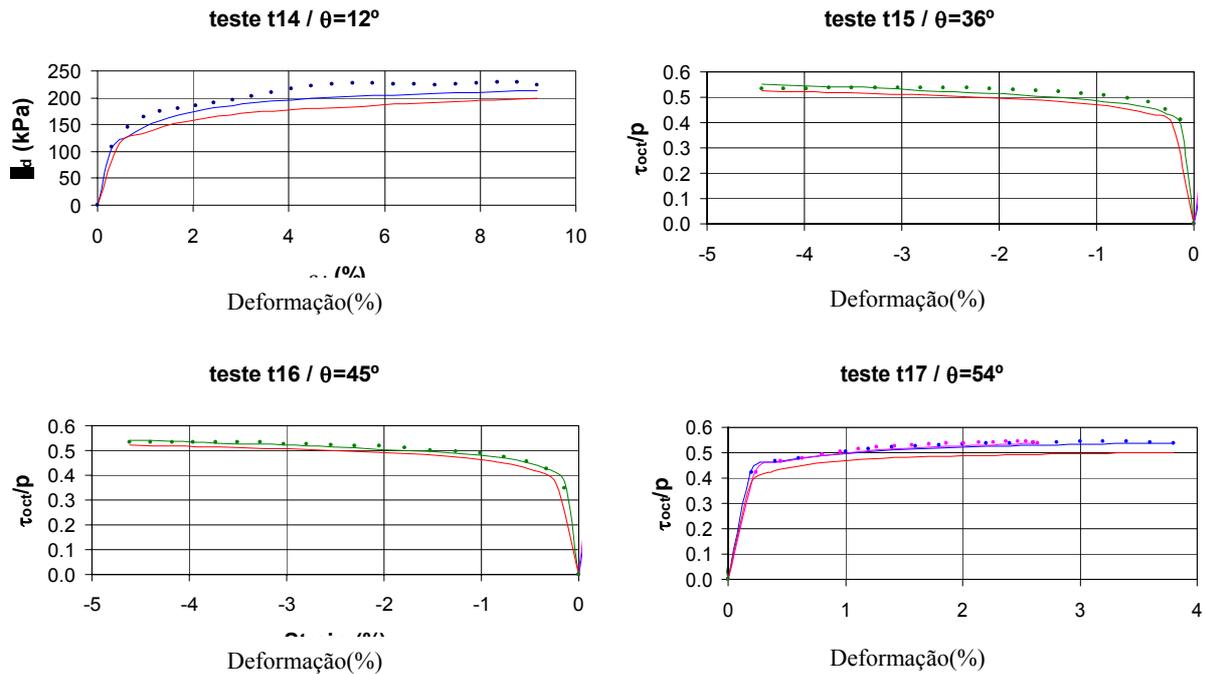


Figura 5.24. (Continuação) Simulações do modelo híbrido ALLFINE-NN. Areia Monterrey plano octaédrico.

As figuras subsequentes correspondem às simulações de trajetórias CTC para diferentes areias na ordem seguinte: Areia Calcária (Figura 5.25), Areia de Ottawa (Figura 5.26), Areia de Antioch (Figura 5.27), areia de Chattahoochee (Figura 5.28) e areia de San Fernando (Figura 5.29). Todas as trajetórias, para todos os materiais foram adequadamente acompanhados tanto pela rede neural como pelo SH.

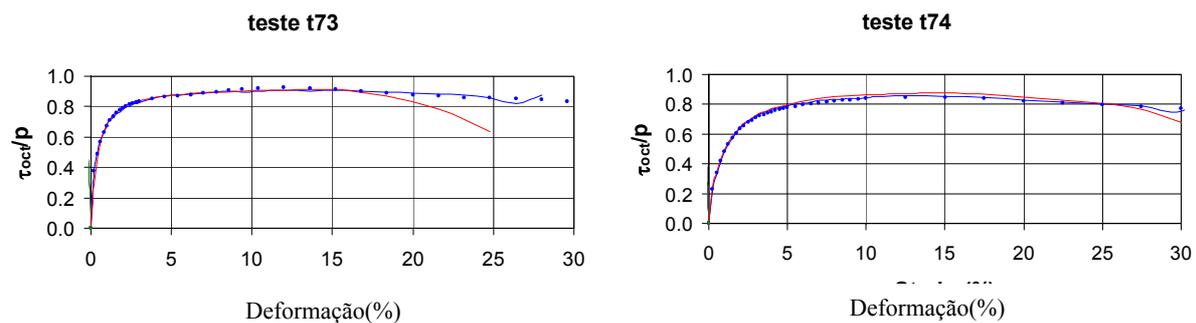


Figura 5.25. Simulações do modelo híbrido ALLFINE-NN. Areia Calcária CTC

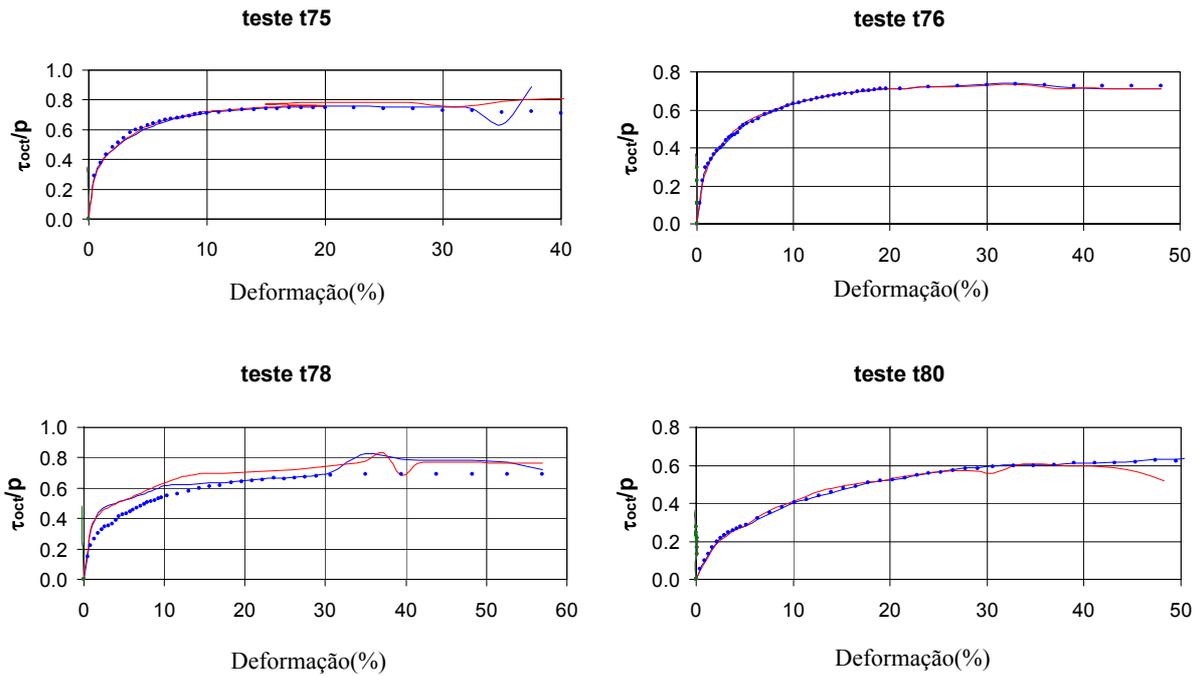


Figura 5.25 (Continuação) Simulações do modelo híbrido ALLFINE-NN. Areia Calcária CTC

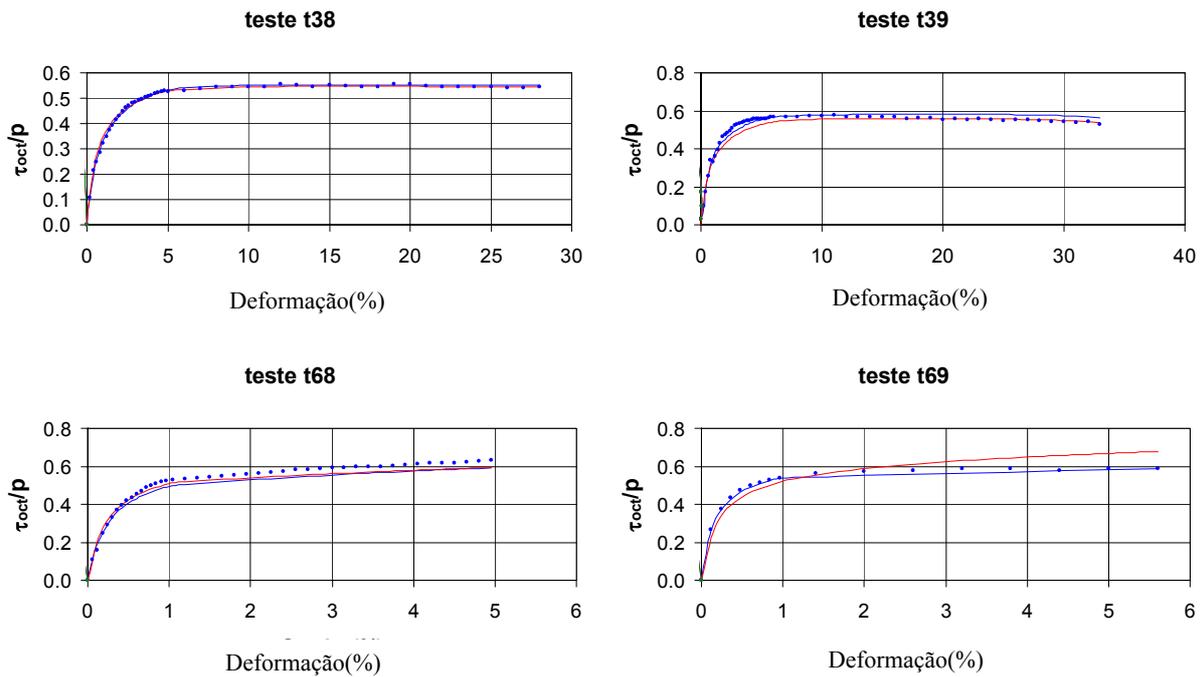


Figura 5.26. Simulações do modelo híbrido ALLFINE-NN. Areia Ottawa CTC

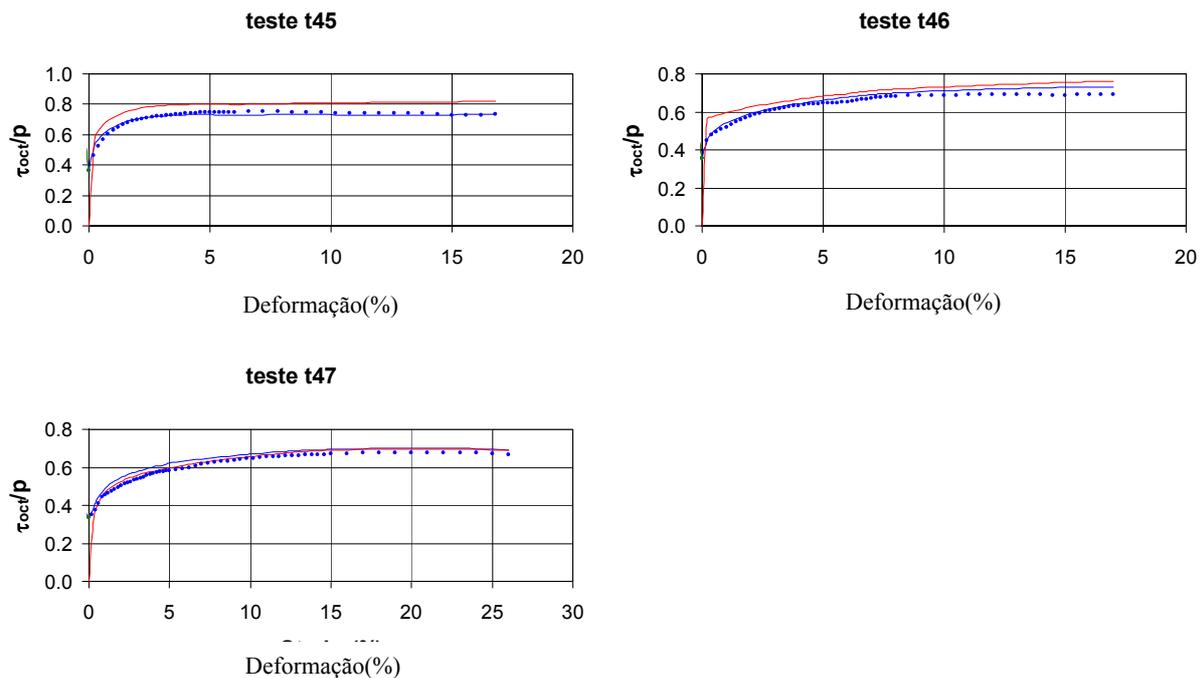


Figura 5.27. Simulações do modelo híbrido ALLFINE-NN. Areia Antioch CTC

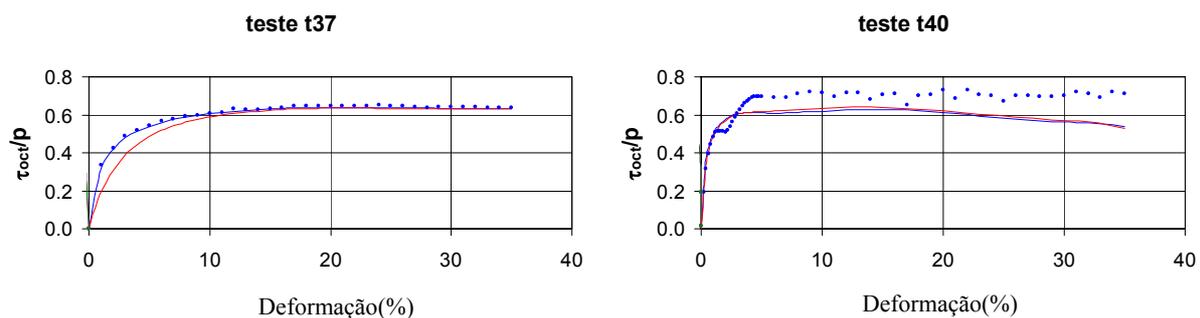


Figura 5.28. Simulações do modelo híbrido ALLFINE-NN. Areia Chattahoochee CTC

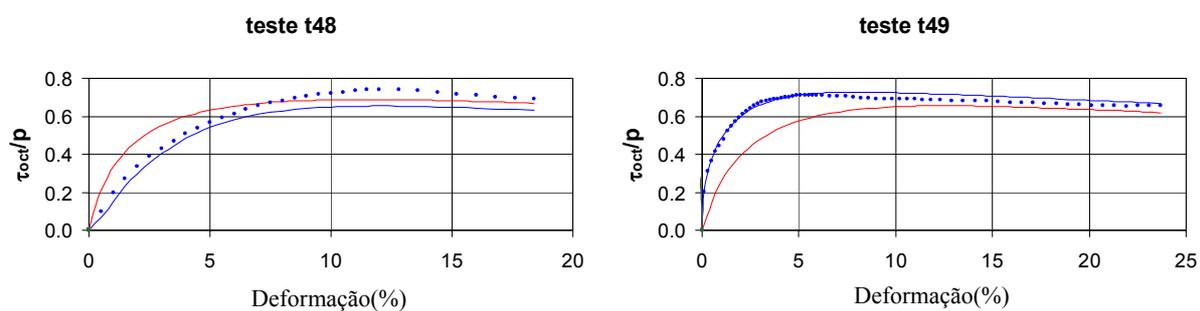


Figura 5.29. Simulações do modelo híbrido ALLFINE-NN. Areia San Fernando CTC

Finalmente são apresentadas as simulações para os testes de compressão hidrostática, para os quais, de maneira geral, o SH apresentou bom desempenho, com exceção do teste 146

no qual o modelo só acompanhou a trajetória até uma tensão confinante da ordem de 20000 kPa. Levando em consideração que a informação de treinamento para tensões de confinamento dessa magnitude foi escassa, se entende o porquê do fraco desempenho a partir do ponto mencionado.

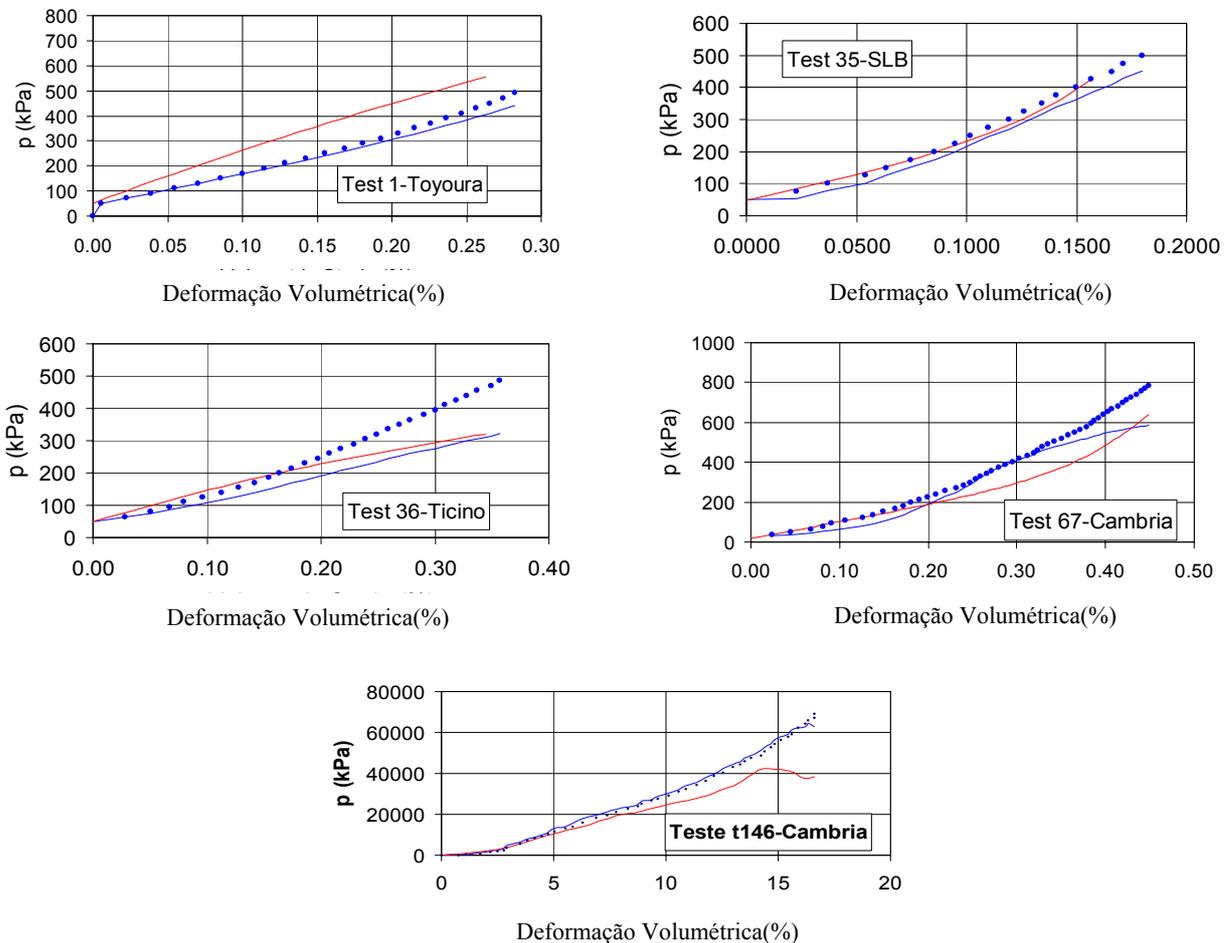


Figura 5.30. Simulações do modelo híbrido ALLFINE-NN. Ensaios HC

5.3.2 – SIMULAÇÃO DE UMA AMOSTRA CÚBICA

O passo seguinte no processo de validação da resposta do sistema híbrido ALLFINE-RN, consiste na simulação de amostras cúbicas submetidas à condição de carregamento triaxial. Para tal propósito a amostra de areia é representada por um volume cúbico que se discretiza com uma malha tridimensional de 1000 elementos de 8 nós cada um. A Figura 5.31 é uma representação esquemática da amostra simulada. Para cada caso analisado são impostas as deformações nas quatro faces laterais da amostra e na face superior. A face inferior está restrita unicamente para deslocamentos verticais.

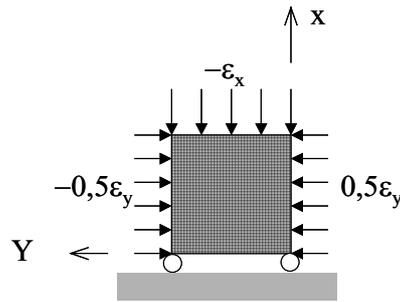


Figura 5.31. Esquema da geometria e condições de carregamento para a simulação de uma amostra cúbica num ensaio triaxial.

Foram escolhidos testes com diferentes trajetórias para algumas das areias mais bem representadas na base de dados. A Tabela 5.2 apresenta uma relação dos testes escolhidos e as suas características.

Tabela 5.2. Ensaio usado para simular o comportamento de uma amostra cúbica.

Número de teste	Areia	Trajectoria
56	Cambria	TC
33	Toyoura	Plano Octaédrico, $\theta=30$
10	Monterrey	TC
69	Ottawa	CTC
78	Calcareous	CTC

Os gráficos da Figura 5.32 apresentam os resultados das simulações realizadas para a amostra cúbica mencionada. Neles, os pontos representam os dados experimentais, os quais foram incluídos para efeito de comparação. As linhas pretas representam as simulações do SH para a amostra em questão. Observa-se em todos os testes um bom acompanhamento dos dados experimentais, uma boa previsão da tensão de ruptura do material e um adequado ajuste da trajetória de tensão de cada teste. As observações anteriores encorajam a utilização do modelo neural como substituto das relações paramétricas convencionais do tipo elasto-plástico, agindo como operador constitutivo dentro de um código de elementos finitos.

O passo seguinte no processo de validação do SH é a simulação de um problema de contorno simples representativo de uma situação real da prática geotécnica, o qual é apresentada na seção seguinte.

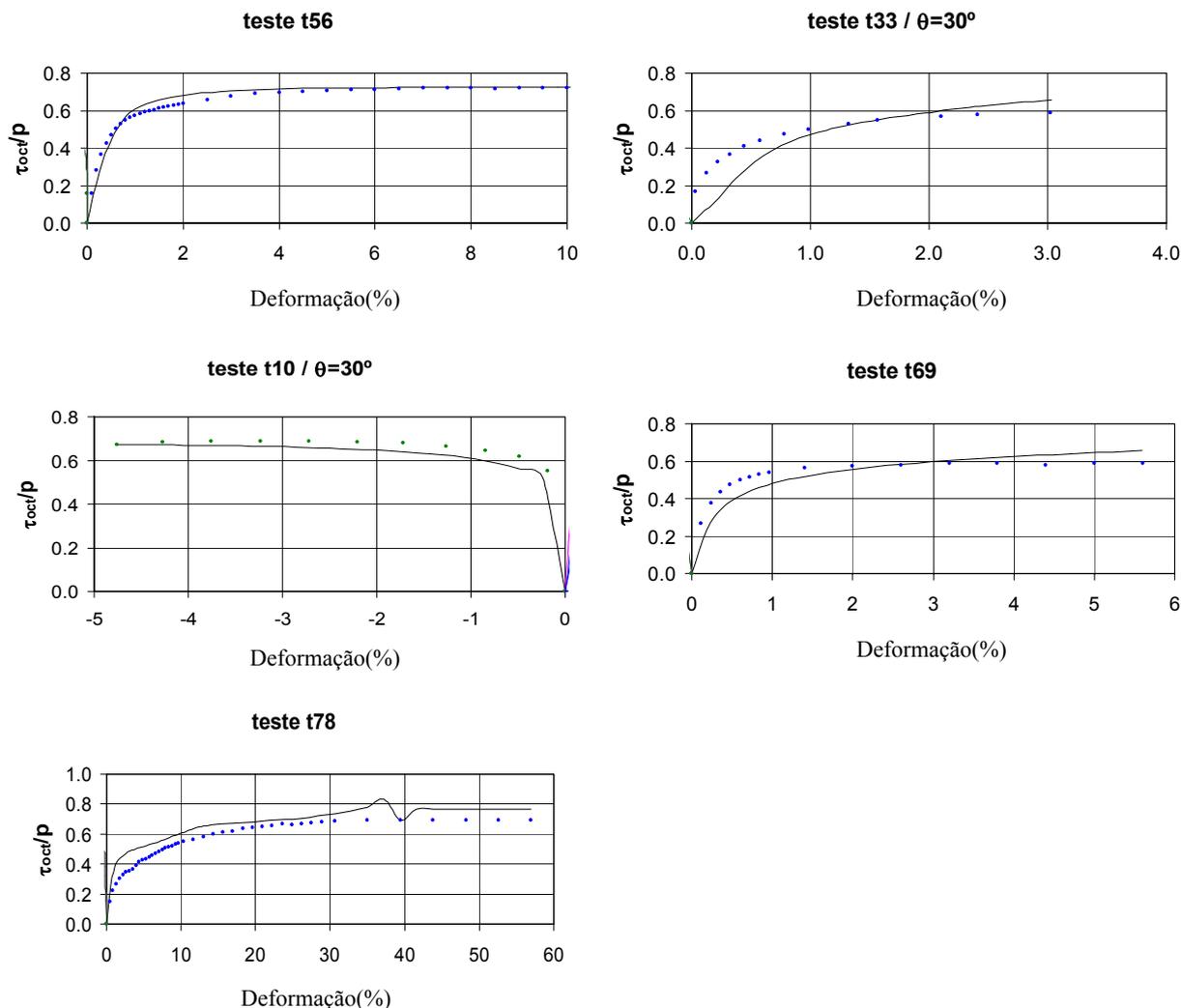


Figura 5.32. Resposta tensão – deformação de uma amostra cúbica simulada com o sistema híbrido ALLFINE-RN.

5.3.3 – ANÁLISE DE UMA SAPATA CORRIDA

Nesta seção é analisado o comportamento de uma sapata assente superficialmente em um depósito de areia. De modo a obter uma simulação o mais próximo possível da realidade optou-se por escolher um material que tivesse sido amplamente representado no processo de treinamento principalmente com relação às diversas trajetórias de tensão. Observando a base de dados, optou-se por fazer as simulações com a areia de Toyoura, dado que para este material se tinha a mais ampla variedade de trajetórias tanto de compressão como de extensão.

A malha de elementos finitos usada para a modelagem do problema proposto é mostrada na Figura 5.33, correspondente ao plano x-y. Na direção z, perpendicular ao plano

do papel, o modelo tem 10 m de comprimento. A simulação de fato foi feita em 3D, porém impondo condições que restringem o problema a uma condição de deformação plana.

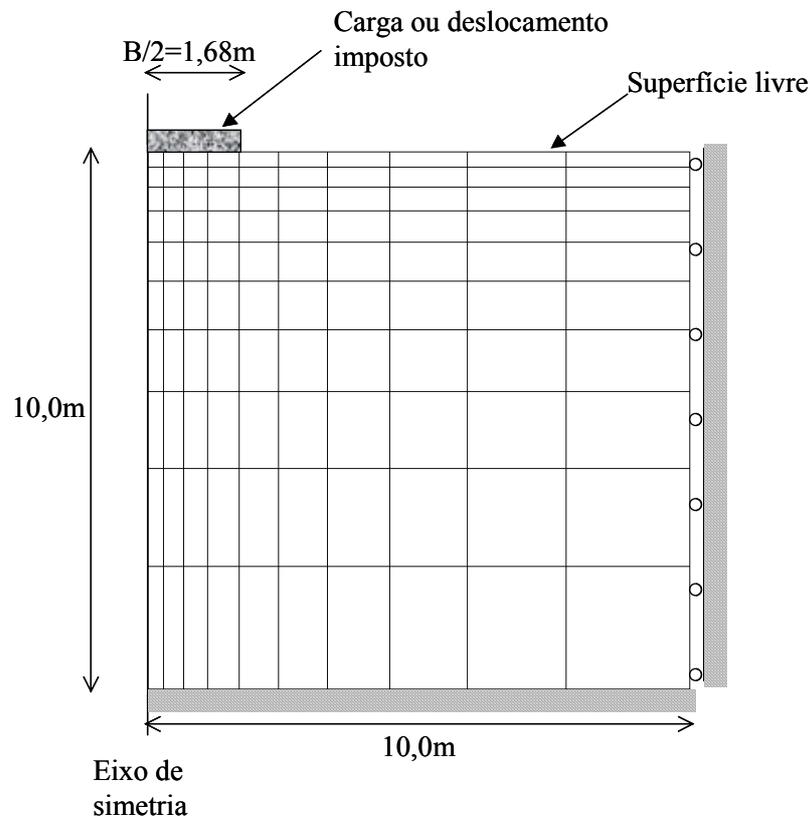


Figura 5.33. Malha de elementos finitos usada para a modelagem do comportamento de uma sapata assente em areia densa.

Foram analisadas duas situações diferentes: a primeira considerando um deslocamento imposto na sapata, e a segunda impondo o carregamento. Para todas as análises a condição inicial da areia foi definida pelo valor da tensão confinante inicial ($\sigma_{c0}=196$ kPa) aplicado em todos os pontos do maciço, e pelo índice de vazios inicial ($e_0=0.832$). Os valores do módulo de rigidez (E) e do coeficiente de Poisson (ν), necessários para a conformação da matriz de rigidez elástica inicial para o SH, são 100000 kPa e 0,35 respectivamente. Simulações mais avançadas que se encontram em processo de desenvolvimento, levarão em conta um estado inicial de tensão geostático no qual a tensão de confinamento média em cada ponto do maciço depende da profundidade e do peso volumétrico do material.

5.4.3.1 – ANÁLISE DE UMA SAPATA COM DESLOCAMENTO IMPOSTO.

Para este caso foi considerada inicialmente uma situação na qual o valor do deslocamento a impor na sapata era relativamente pequeno, da ordem de 0,01m final. O deslocamento é aplicado em estágios nos nós dos elementos correspondentes (elementos cúbicos tridimensionais de 8 nós), e a resposta do sistema é calculada em termos das tensões geradas e do campo de deslocamentos verticais induzidos no maciço.

Os resultados desta primeira simulação são apresentados nos gráficos da Figura 5.34. No gráfico correspondente ao campo de tensões pode ser apreciada a formação de zonas de igual tensão vertical imediatamente abaixo da sapata (bulbos de tensão), e a formação de uma região de concentração de tensões na quina externa da sapata. Esta região de concentração, aparece por causa de um efeito de punção que ocorre próximo à borda da sapata, região a qual é menos confinada.

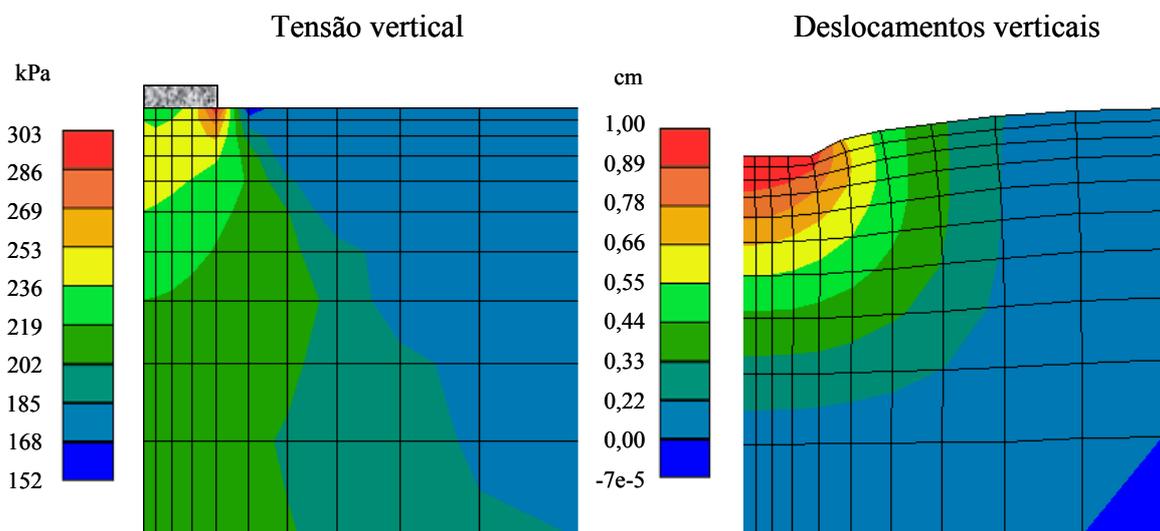


Figura 5.34. Gráficos de tensão vertical e de deslocamentos verticais para uma sapata com deslocamento pequeno imposto, analisada com o SH.

Visando testar a robustez do modelo, foi realizada uma segunda simulação com deslocamento imposto, considerando um valor maior que o do caso anterior. Foi aplicado então um deslocamento final de 0,04m na sapata, e foram plotados os resultados em termos das tensões e deslocamentos (Figura 5.35).

Para o caso apresentado na Figura 5.35 observa-se uma adequada distribuição de tensões na região abaixo da sapata, e um aumento nos valores das mesmas de acordo com o aumento no tamanho do deslocamento imposto. A distribuição dos deslocamentos no maciço, também se apresentou razoavelmente bem configurada.

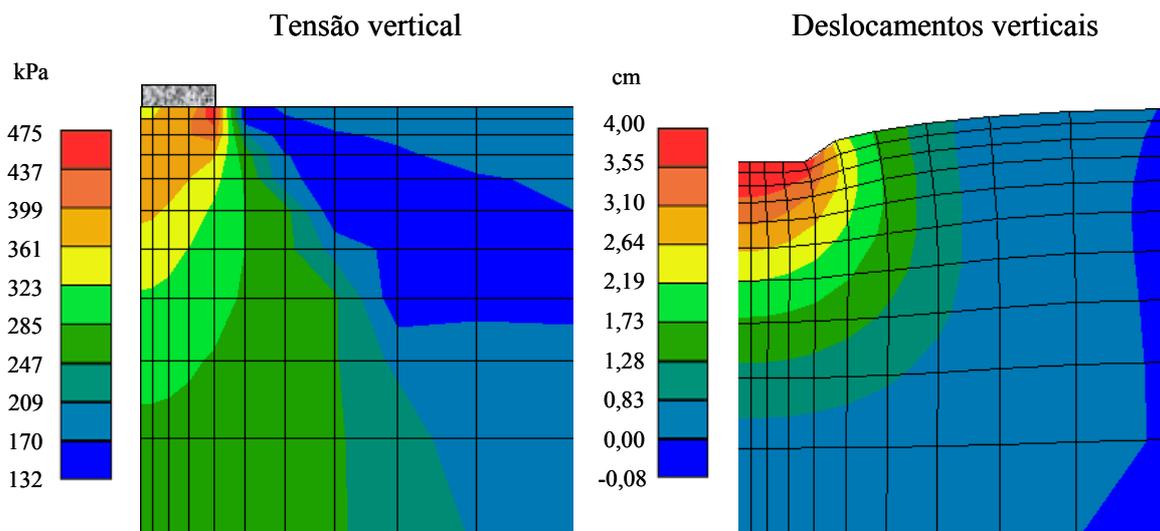


Figura 5.35. Gráficos de tensão vertical e de deslocamentos verticais para uma sapata com deslocamento grande imposto, analisada com o SH.

Para efeitos de comparação, o problema foi simulado usando o modelo Cam-Clay implementado igualmente no programa ALLFINE.

A determinação dos parâmetros necessários para a aplicação do modelo Cam-Clay foi realizada a partir dos resultados experimentais do teste 26. Partindo de um conjunto de parâmetros iniciais estimados dos ensaios disponíveis, foram realizadas simulações sucessivas da trajetória do teste 26 até conseguir uma aproximação aceitável entre o resultado da simulação e os valores experimentais. Os parâmetros finais são: $M=1,3$, $\lambda=0,0271$, $\kappa=0,0025$, $e_0=0,832$, $\nu=0,4$. A Figura 5.36 apresenta, em linha de cor preta contínua, os resultados da simulação final do teste 26 com o modelo Cam-Clay. Os resultados foram validados tanto para a relação τ_{oct}/p versus a deformação principal maior, como para a tensão de desvio versus a deformação de desvio. Para ambos gráficos foram incluídos, para efeito de comparação, os pontos experimentais do ensaio.

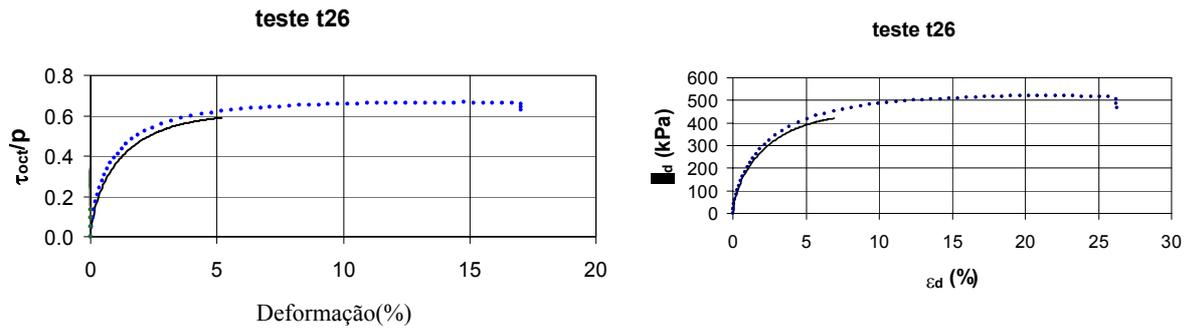


Figura 5.36. Simulação do teste 26 (Toyoura, CTC) realizada com o modelo Cam-Clay para calibração de parâmetros.

Na simulação do problema de contorno com o modelo Cam-Clay, inicialmente foi imposto um deslocamento de 0,01m na sapata, e as tensões e distribuição de deslocamentos no maciço foram calculados. Os resultados são apresentados na Figura 5.37 na qual pode-se observar um bulbo de tensões próximo daquele correspondente ao caso analisado com o SH, e uma distribuição de deslocamentos verticais igualmente semelhante.

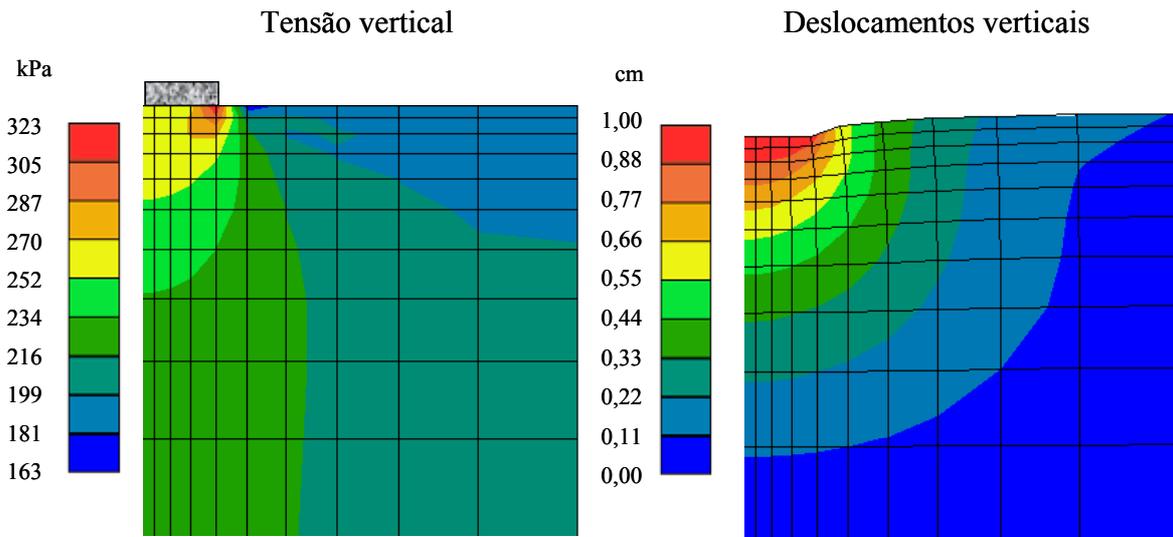


Figura 5.37. Gráficos de tensão vertical e de deslocamentos verticais para uma sapata com deslocamento pequeno imposto, analisada com o modelo Cam-Clay.

A Figura 5.38 mostra os gráficos de distribuição de acréscimos de tensão vertical e de deslocamentos verticais para a análise realizada como o modelo Cam-Clay a partir de um deslocamento de 0,04 m imposto na sapata.

Observa-se como o tamanho das tensões é maior que para o caso do SH, principalmente na região próxima a quina da sapata onde, tal como já foi mencionado, é normal a ocorrência de concentrações de tensão. Não obstante, para as regiões localizadas na porção média e no

extremo esquerdo da sapata (correspondente na verdade ao ponto central da mesma, pois o problema é simulado considerando um eixo de simetria tal como se apresentou na Figura 5.33), a magnitude das tensões verticais é muito semelhante à obtida com o sistema híbrido ALLFINE-RN (da ordem de 400 kPa). Por outro lado, com relação à distribuição dos deslocamentos verticais, observa-se como os dois modelos produzem respostas muito parecidas entre si, se diferenciando unicamente nas regiões mais profundas do maciço, onde os deslocamentos são muito pequenos.

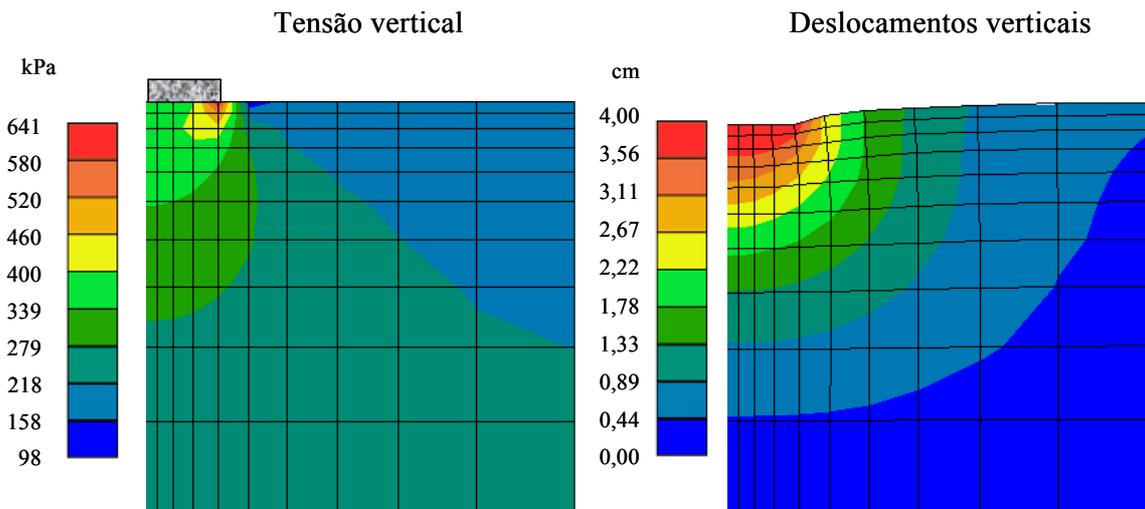


Figura 5.38. Gráficos de tensão vertical e de deslocamentos verticais para uma sapata com deslocamento grande imposto, analisada com o modelo Cam-Clay implementado no programa ALLFINE.

5.4.3.1 – ANÁLISE DE UMA SAPATA COM CARREGAMENTO IMPOSTO.

De igual forma que para os casos anteriores, foram realizadas algumas simulações na sapata para verificar o comportamento a partir de um carregamento prescrito. Neste caso foram consideradas duas situações: uma com uma carga de magnitude pequena e outra com uma carga maior. A primeira simulação corresponde ao caso no qual um acréscimo de tensão vertical de 100 kPa é imposto na sapata. Os resultados são apresentados na Figura 5.39, na qual se pode apreciar uma adequada configuração das regiões de igual valor de acréscimo de tensão abaixo da sapata, e uma configuração razoavelmente consistente de deslocamentos verticais.

Na segunda análise foi imposto um carregamento de 400 kPa e os resultados foram igualmente apresentados em forma de gráficos de iso-linhas de tensão vertical e

deslocamentos verticais. Esses gráficos são apresentados na Figura 5.40, observando-se um aumento correspondente no tamanho dos bulbos de tensão e na sua distribuição dentro do maciço. De igual forma, o deslocamento na sapata aumentou de acordo com a magnitude do carregamento imposto, e a distribuição de deslocamentos no maciço se mostrou bem configurada. No entanto, no gráfico de deslocamentos verticais apareceu uma região, na fronteira direita do maciço, com pequenos valores negativos, indicando deslocamentos no sentido ascendente.

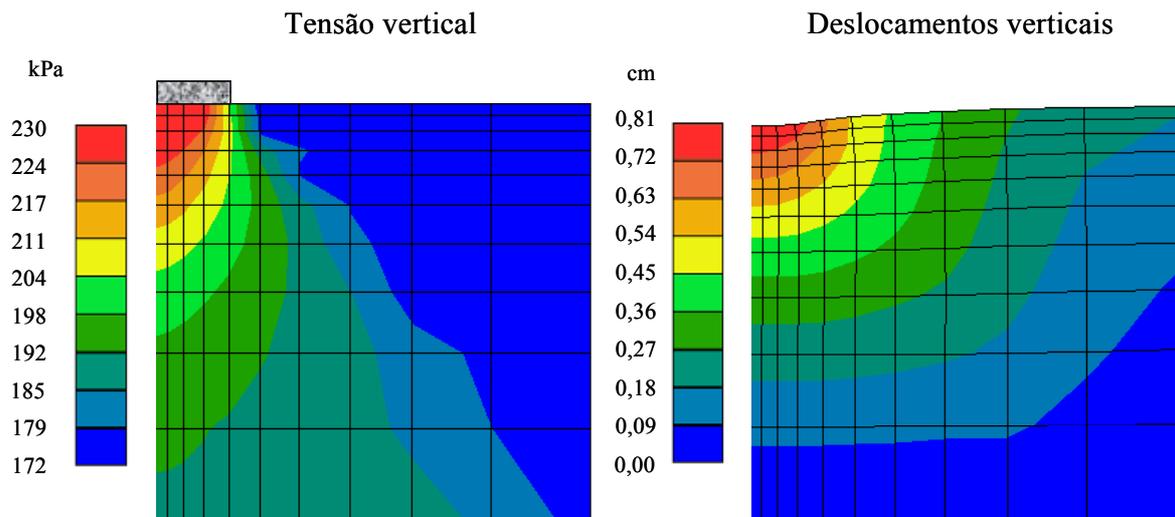


Figura 5.39. Gráficos de tensão vertical e de deslocamentos verticais para uma sapata com um carregamento prescrito de 100 kPa, analisada com o SH.

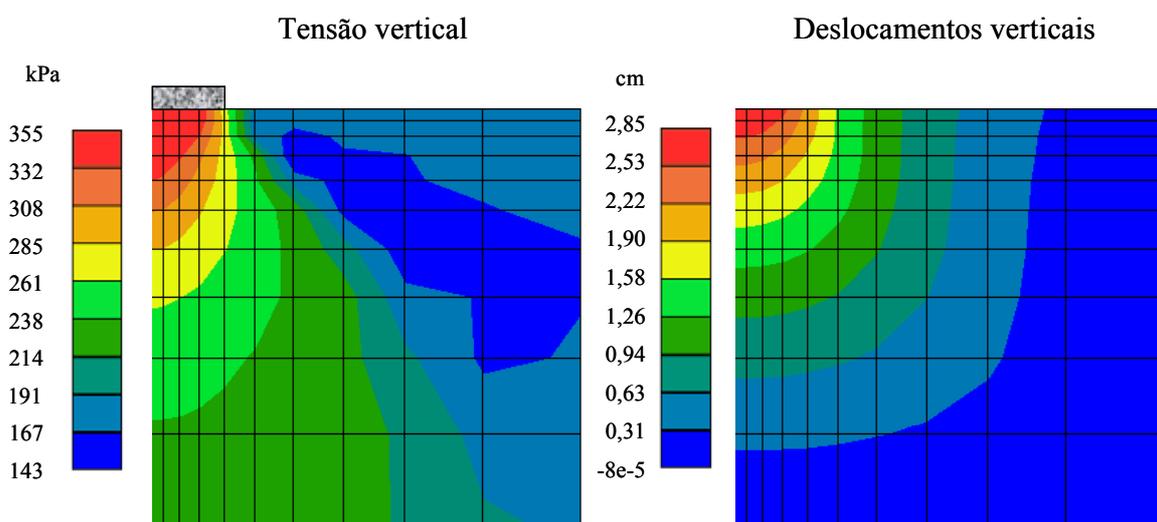


Figura 5.40. Gráficos de tensão vertical e de deslocamentos verticais para uma sapata com um carregamento prescrito de 400 kPa, analisada com o SH.

Para efeitos de melhorar a comparação entre os resultados dos modelos considerados achou-se adequado calcular a curva de carga contra deslocamento vertical para os casos analisados. Para tal efeito foram calculados os valores médios da carga e do deslocamento nos elementos localizados imediatamente abaixo da sapata, e os seus valores plotados tal como se apresenta na Figura 5.41.

Para o cálculo da tensão média ao longo da sapata, foram ponderados os valores das tensões finais em cada um dos pontos de Gauss localizados na parte superior dos elementos envolvidos, que são os quatro elementos do canto superior esquerdo do modelo. A ponderação é realizada levando em conta o comprimento de cada elemento e, obviamente o valor das tensões nos pontos de Gauss mencionados.

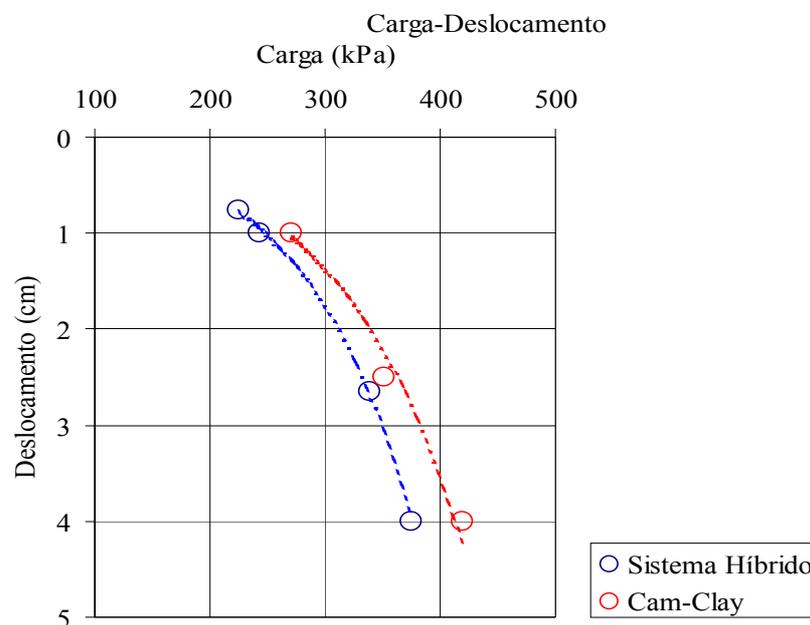


Figura 5.41. Gráfico carga-deslocamento para uma sapata simples analisada com o SH e o modelo Cam-Clay.

A partir dos resultados mostrados na Figura 5.41, conclui-se que os dois modelos apresentam respostas de simulação comparáveis. Com valores de deslocamentos verticais compatíveis com os níveis de tensão impostos. No entanto, deve-se destacar que o objetivo destas comparações não é validar os resultados de nenhum dos dois modelos, os quais, devido à forma como foram concebidos não são passíveis de se validar mutuamente. Enquanto que o modelo elasto-plástico adotado baseia seu desempenho na suposição prévia de um critério de ruptura e na consideração de trajetórias de tensão do tipo axis-simétrico; o modelo neural não

considera “a priori” nenhuma superfície de ruptura, e seu campo de simulação inclui trajetórias de tensão do tipo triaxial verdadeiro.

5.4 – DISCUSSÃO

As análises realizadas neste capítulo foram desenvolvidas em uma seqüência de etapas que permitiu implementar o modelo constitutivo neural dentro de um código de elementos finitos (programa ALLFINE). Inicialmente foi apresentado o algoritmo usado para a estruturação do modelo neural dentro do código do programa de elementos finitos. Simultaneamente foram feitas algumas observações relacionadas com as limitações e potencialidades do modelo neural agindo como operador constitutivo dentro do MEF. Uma vez terminada a fase de implementação foram realizadas simulações do comportamento das areias ao nível de ponto constitutivo. Para tanto achou-se adequado simular o comportamento de um elemento cúbico de 8 nós em todas as condições de carregamento impostas nos testes da base de dados. A seguir, o grau de complexidade das análises aumentou ao considerar a amostra de solo como um volume discretizado com uma malha de elementos finitos. As simulações realizadas mostraram um ótimo desempenho da RN como operador constitutivo dentro do MEF (o que foi chamado de sistema híbrido, SH).

Finalmente se passou a um nível maior de complexidade no qual foi analisado o comportamento de uma sapata retangular assente num depósito de areia homogênea. Para esta análise foi escolhida a areia de Toyoura e os resultados foram analisados em termos de distribuição das tensões no maciço de areia e de distribuição dos deslocamentos. Não tendo informação experimental para fazer as devidas comparações, considerou-se pertinente comparar contra os resultados obtidos por meio do modelo Cam-Clay.

O comportamento do SH nas simulações realizadas se mostrou satisfatório, o que é um alento para seguir pesquisando nesta área da geotecnia que se mostra bastante promissora.

CAPÍTULO 6

CONCLUSÕES

As RN's oferecem várias vantagens sobre os métodos computacionais convencionais. A vantagem mais freqüentemente citada é a sua capacidade de fornecer uma solução geral para um dado problema, a partir de um conjunto de exemplos, e continuar se adaptando às variações das características do problema mediante a simples apresentação de novos exemplos que descrevem tais variações. Sua capacidade de generalização lhes permite fornecer soluções adequadas ainda que existam erros dentro do conjunto de exemplos de treinamento. Todos estes fatores combinados fazem das RN's ferramentas poderosas para a modelagem de problemas nos quais as relações funcionais entre as variáveis dependentes e as independentes são pouco conhecidas, sujeitas a incertezas, ou apresentam certa variação com o tempo. Estes problemas aparecem comumente nas diferentes disciplinas da engenharia civil (Flood & Kartam, 1994).

As RN's têm se tornado cada vez mais populares devido a duas razões. Primeiro, está comprovado que sob determinadas condições, elas podem criar mapeamentos complexos não-lineares entre conjuntos de entradas e saídas. Segundo, esta propriedade tem sido amplamente demonstrada em múltiplas aplicações em todas as áreas da engenharia. A popularidade das RN's se deve também em grande parte, à crença errada que o seu uso para modelagem geral de problemas é direto ou muito fácil, o qual constitui ao mesmo tempo a maior causa de uso inadequado. Dois elementos são fundamentais para a aplicação de uma técnica de modelagem (Reich & Barai, 2000):

- Entendimento da teoria por trás do método de modelagem a ser usado, e também, conhecimento de métodos avançados para otimizar ou melhorar os resultados da modelagem.

- Entendimento do processo de aplicação do método a dados reais, incluindo validação do modelo e interpretação dos resultados.

Nesta tese foram mostrados de maneira detalhada, cada um dos passos seguidos no processo de criação, treinamento e validação de um modelo neural capaz de realizar um mapeamento preciso da relação geral tensão-deformação de um grupo amplo de areias, sob as mais diversas condições de carregamento triaxial verdadeiro. O modelo escolhido foi do tipo Perceptron de Múltiplas Camadas (PMC), cujo processo de treinamento, que foi chamado de “treinamento por lotes”. A seleção e depuração da base de dados foram os pontos chave na obtenção de resultados de simulação aceitáveis. Mostrou-se adicionalmente que o PMC adequadamente treinado consegue realizar tarefas complexas como, por exemplo, simular trajetórias completas de tensão a partir da apresentação exclusivamente do primeiro ponto experimental.

O vetor de treinamento usado é composto por dois tipos de variáveis. O primeiro tem a ver com variáveis usadas para identificar o tipo de material a considerar e a sua condição física inicial. Estas variáveis são: o fator de forma (n_s), a diferença entre o e_{max} e o e_{min} ($e_{max} - e_{min}$), o índice de vazios inicial (e_0), e a tensão confinante inicial (σ_{c0}). As três primeiras são apresentadas à rede após terem sido transformadas por meio de funções não lineares que destacam o contraste entre os valores pequenos, mantendo-o constante para os valores grandes. O segundo tipo de variáveis permite identificar o estado de tensão-deformação do solo tanto no estágio atual de simulação como no estágio imediatamente anterior. Igualmente neste grupo se encontra o vetor de acréscimos de deformação principal, que é imposto para que o modelo estime o vetor correspondente de acréscimos de tensão principal.

Visando otimizar a arquitetura neural para o problema proposto, foi testada uma grande quantidade de redes, variando tanto o número de neurônios como as funções de transferência e algoritmos de aprendizado. A base de dados inicialmente coletada, composta por 171 testes teve de ser reduzida a 109 testes para aumentar sua consistência e facilitar o aprendizado do modelo neural. Para isso foi usada uma rede competitiva do tipo Kohonen treinada com a informação fornecida pelo índice de vazios inicial (e_0), o $e_{max} - e_{min}$, o fator de forma (n_s) e a tensão de confinamento inicial (σ_{c0}). A pre-classificação dos testes da base de dados por meio da rede de Kohonen foi um fator fundamental para o desenvolvimento dos modelos constitutivos neurais que se apresentaram nesta tese. Tal processo melhorou consideravelmente o desempenho do modelo cujas simulações, avaliando a complexidade do problema analisado, podem ser consideradas adequadas para proceder a sua implementação

em um código de elementos finitos em substituição de uma relação constitutiva elasto-plástica convencional. Tal como já foi dito, o fator de forma (n_s) e o $e_{\max}-e_{\min}$ foram usados para identificar o tipo de areia. Em particular o $e_{\max}-e_{\min}$, parece ser um parâmetro de identificação poderoso dentro do qual está compreendida valiosa informação sobre o comportamento de deformação volumétrica da areia.

Numa fase adiantada de desenvolvimento do trabalho se percebeu que os modelos treinados, embora acompanhassem adequadamente o sentido das trajetórias solicitadas, não conseguiam identificar com precisão, em muitos dos casos, o valor das tensões de ruptura. Diante da intenção de manter o vetor de treinamento livre de parâmetros que informassem diretamente sobre a resistência da areia, como, por exemplo, o ângulo de atrito, achou-se adequado incluir uma entrada adicional que mesmo não informando diretamente os valores de ruptura do solo, fosse capaz de fornecer informação à rede sobre a evolução da superfície de plastificação desde o começo da trajetória até alcançar seu valor máximo justo antes de o solo romper. Esse parâmetro adicional foi o invariante X (que corresponde ao critério de Matsuoka-Nakai) o qual é calculado pela rede em cada estágio de carregamento.

A inclusão deste parâmetro permitiu desenvolver um modelo mais robusto cujas características de desempenho permitiram sua implementação dentro do código do programa de elementos finitos ALLFINE.

O sucesso na implementação de um modelo neural depende não só da qualidade dos dados usados para o treinamento do modelo, mas também do tipo e estrutura da rede neural adotada, o método de treinamento, e da maneira como os dados de entrada e saída são interpretados, processados e estruturados.

A implementação de modelos neurais dentro de programas de elementos finitos já foi descrita por alguns autores, no entanto, a simulação de trajetórias 3D reais de tensão é ainda hoje matéria de intensa pesquisa. Acredita-se que os resultados apresentados neste trabalho representam um avanço importante nesta área da geotecnia uma vez que um modelo neural único conseguiu simular ainda que de forma aproximada o comportamento mecânico de diversos materiais sob condições de carregamento complexas.

O modelo implementado no programa ALLFINE foi testado mediante três níveis de simulação diferentes, com grau de dificuldade crescente. Inicialmente foi simulado o comportamento da areia ao nível de ponto constitutivo, posteriormente foram simuladas amostras cúbicas similares às usadas no equipamento triaxial verdadeiro, finalmente foram simuladas duas condições de carregamento numa sapata simples assente num maciço de areia

fofa (areia de Toyoura). Os resultados foram comparados contra simulações semelhantes realizadas com o modelo Cam-Clay. Os resultados obtidos em todos os casos se mostraram satisfatórios e motivam a continuidade desta linha de pesquisa. Alguns problemas relacionados com a interação entre o código de elementos finitos e o modelo neural não foram analisados em profundidade por estarem fora do escopo desta tese.

6.1 – SUGESTÕES PARA FUTURAS PESQUISAS

As futuras pesquisas deverão ser encaminhadas em primeiro lugar para a melhoria da base de dados, mediante a incorporação de informação nova ou processamento da informação disponível para melhorar seu aproveitamento em etapas de treinamento de novos modelos. Em segundo lugar, será necessária uma fase de intenso estudo sobre a interação entre o MEF e a rede neural pois alguns aspectos triviais na modelagem com MEF como, por exemplo, a escolha do tamanho do passo adequado para a otimização dos processos de integração, não são necessariamente válidos quando se usam RNA's. Uma vez resolvidos estes aspectos será necessária a realização de análises de sensibilidade do Sistema Híbrido e simultaneamente a verificação de seu desempenho mediante a retroanálise com casos históricos.

Para melhor aproveitamento da base de dados já formada, pode-se, por exemplo, treinar modelos para conjuntos limitados de areias com vetores de treinamento nos quais, para cada teste, sejam incluídos valores dos incrementos de deformação com distribuições mais amplas que as apresentadas neste trabalho. Novas alternativas de treinamento poderão ser usadas como por exemplo, a técnica de regularização Bayessiana a qual permite a determinação automática dos parâmetros ótimos (pesos e biases) de desempenho da RNA, mediante uma análise estatística das variâncias associadas às distribuições desses parâmetros. Também está aberta a possibilidade de usar outros tipos de RNA's como as redes Elman e as redes de Hopfield, nas quais as camadas de saída estão ligadas mediante conexões de retroalimentação para as camadas de entrada, o que lhes permite reconhecer e aprender padrões ou seqüências temporais ou espaciais.

Finalmente, é importante ressaltar que o trabalho que está sendo desenvolvido constitui uma aplicação nova da técnica das redes neurais, cuja utilização na modelagem constitutiva de geomateriais é considerada fronteira do conhecimento geotécnico.

REFERÊNCIAS BIBLIOGRÁFICAS

- Adeli, H. (2001). Neural Networks in Civil Engineering:1989-2000. *Computer-Aided in Civil and Infrastructure Engineering*, 16, 126-142.
- Alí, M. & Chawathé, A. (2000). Using Artificial Intelligence to Predict Permeability from Petrographic data. *Computers and Geotechnics*, 26, 915-925.
- Basheer, I. A. (2000). Selection of Methodology for Neural Network Modeling of Constitutive Hystereses Behavior of Soils. *Computer-Aided Civil and Infrastructure Engineering*, 15, 440-458.
- Bellotti, R., Jamiolkowski, M., Lo Presti, D.C.F. & O’neill, D.A. (1996). Anisotropy of small strain stiffness in Ticino Sand. *Geotechnique* 46(1), 115-131.
- Braga, A de P.; Carvalho, A. P. & Ludermir, T. B. (2000). *Redes Neurais Artificiais: Teoria e Aplicações*. Editora LTC. Rio de Janeiro, RJ. p.262.
- Cambou, B. & Lanier, J. (1988). Induced Anisotropy in Cohesionless Soil: Experiments and Modelling. *Computers and Geotechnics*, 6, 291-311.
- Cassa, J. C.; Oliveira, R. T.; Cruz, M. F. & Floridia, G. (1997). Previsão da resistência mecânica do concreto usando redes neurais artificiais. *III Congresso Brasileiro de Redes Neurais*. Florianópolis, SC. 98-103.
- Chan, W.T.; Chow, Y.K. & Liu, L.F. (1995). Neural Networks: An Alternative to Pile Driving Formulas. *Computers and Geotechnics*, 17, 135-156.
- Colliat-Dangus, J. L., Desrues, J. & Foray, P. (1998). Triaxial Testing on Granular Soil Under Elevated Cell Pressure. *Advanced Triaxial Testing on Soil and Rock, ASTM STP 977*, Robert T. Donaghe, Ronald C. Chaney and Marshall L. Silver, Eds., ASTM, Philadelphia, 290-310.
- Cordão Neto, M. P. (2005). Análise Hidro-Mecânica de Barragens de Terra Construídas com Materiais Colapsíveis. Tese de Doutorado, Universidade de Brasília, 152p.
- Coronel-Ávila, J.A. & Parra-Grajales, W. (2003). Correlação de Registros de Poços Usando Redes Neurais Artificiais (em espanhol). Dissertação de Mestrado, Universidad Nacional de Colômbia – Facultad de Minas, Medellín, 123 p.
- Cubrinovski, M. & Ishihara, K. (2002). Maximum and minimum void ratio characteristics of sands. *Soils and Foundations*, 42(6), 65-78.
- Dantas-Neto, S. A. (2004). *Avaliação das Propriedades dos Ligantes e das Misturas Asfálticas Modificados com Borracha Granulada de Pneus Usados*. Tese Ph.D. Universidade de Brasília, Brasília, D.F., p.265.

- Demuth, H. & Beale, M. (2000). *Neural network Toolbox for Use with Matlab*. Version 4.
- Desai, C.S. & Siriwardane, H. J. (1984). *Constitutive Laws for Engineering Materials*. Prentice-Hall, New Jersey.
- Dyminski, A. S. (2000). *Análise de Problemas Geotécnicos Através de Redes Neurais*. Tese PhD. Pontifícia Universidade Católica do Rio de Janeiro. Rio de Janeiro, RJ. p.201.
- Ellis, G.W.; Yao, C.; Zhao, R. & Penumadu, D.(1995). Stress–Strain modeling of sands using artificial neural networks. *ASCE Journal of Geotechnical Engineering*. 121(5), 429-435.
- Evesque, P. (2001). P=Constant Compression on Loose Hostun Sand: The Case of an Anisotropic Response. *Poudres & Grains*, 12(2), 43-49.
- Farias, M. M. (1993). *Numerical Analysis of Clay Core Dams*. PhD Thesis, University College of Swansea, Swansea, United Kingdom, 159p.
- Felin, W. (2002). *Equations of the Hipoelastic Law*. Innsbruck University Na Internet: www2.uibk.ac.at/geotechnik/res/hypo_versions/hypo_www.html.
- Flood, I. & Kartam, N. (1994). Neural Networks in Civil Engineering I: Principles and Understanding. *ASCE Journal of Computing in Civil Engineering*, 2(8), 131-148.
- Garcia, S.R. , Romo, M. P. & Taboada, V.M. (2000). Knowledge-Based Modeling of Sand Behavior. *European Congress on Computational Methods in Applied Sciences and Engineering ECCOMAS2000*, Barcelona, 11-14 september, 2000.
- García, S. R. & Romo, M. P. (2004). Analytical Description of Complex Systems by Means of RP's and EF's. *European Congress on Computational Methods in Applied Sciences and Engineering ECCOMAS2004*, Jyvaskyla, 24-28 july, 2000.
- Ghaboussi, J., Garret, J.H. & Wu, X. (1991). Knowledge–based modeling of material behavior with neural networks. *ASCE Journal of Engineering Mechanics*, 117(1), 132-153.
- Ghaboussi, J. (1992). *Potential Applications of Neuro-Biological Computational Models in Geotechnical Engineering*. Numerical Models in Geomechanics – NUMOG I, Balkema, Rotterdam. pp.543-555.
- Ghaboussi, J. & Sidarta, D. E. (1998). New nested adaptive neural networks (NANN) for constitutive modeling. *Computers and Geotechnics*, 22(1),29-52.
- Goh, A.T. (1994). Seismic Liquefaction Potential Assessed by Neural Networks. *Journal of Geotechnical Engineering*, 120 (9), 1467-1480.
- Goh, A.T. (1995). Empirical Design in Geotechnics using Neural Networks. *Geotechnique*, 45 (4), 709-714.
- Goto, S. & Tatsuoka, F. (1988). Effects of End Conditions on Triaxial Compressive Strength for Cohesionless Soils. *Advanced Triaxial Testing on Soil and Rock, ASTM STP 977*, Robert T. Donaghe, Ronald C. Chaney and Marshall L. Silver, Eds., ASTM, Philadelphia, 692-705.
- Haruyama, M. (1987). Effect of Density on the Drained Deformation Behavior of Shirasu (Volcanic Sand Soil) under Three-Dimensional Stresses. *Soils and Foundations*, 27(1), 1-13.

- Hashash, Y. M. A.; Jung, S. & Ghaboussi, J. (2004). Numerical Implementation of a Neural Network Based Material Model in Finite Element Analysis. *International Journal for Numerical Methods in Engineering*, Vol. 59, 989-1005
- Haykin, S. (2001). *Redes Neurais: A comprehensive foundation*. Prentice-Hall, 2nd Edition.
- Hicher, P.-Y. (1996). Elastic Properties of Soils. *Journal of Geotechnical Engineering*, 122(8).
- Hoque, E. & Tatsuoka, F. (1998). Anisotropy in elastic deformation of granular materials. *Soils and Foundations*, 38(1), 163-179.
- Juang, C.H.; Chen, C.J. & Jiang, T. (2001). Probabilistic Framework for Liquefaction Potential by Shear Wave Velocity. *Journal of Geotechnical and Geoenvironmental Engineering*, 127(8), 670-678.
- Juang, C.H.; Lu, P. C. & Chen, C. J. (2002). Predicting Geotechnical Parameters of Sands from CPT Measurements Using Neural Networks. *Computer-Aided Civil and Infrastructure Engineering*, 17, 31-42.
- Kasabov, N. K. (1998). *Foundations of Neural Networks, Fuzzy Systems and Knowledge Engineering*. MIT Press.
- Kim, C.G.; Bae, G.J.; Hong, S.W.; Park, C.H.; Moon, H.K. & Shin, H.S. (2001). Neural Network Based Prediction of Ground Surface Settlement due to Tunneling. *Computers and Geotechnics*, 28, 517-547.
- Kolymbas, D., & Wu, W. (1990). Recent Results of Triaxial Tests with Granular Materials. *Powder Technology*, 60, 99-119.
- Lade, P. V. & Duncan, J. M. (1973). Cubical Triaxial Tests on Cohesionless Soil. *Journal of the Soil Mechanics and Foundation Division*, 99(SM10), 793-812.
- Lee, K. L. (1970). Comparisson of Plane Strain and Triaxial Test on Sand. *Journal of the Soil Mechanics and Foundation Division*, 96(SM3), 901-923.
- Lee, K. L., Seed, H. B. & Idriss, I. M. (1975). Properties of Soil in the San Fernando Hydraulic Fill Dams. *Journal of the Geotechnical Engineering Division*, 101, GT8, 801-821.
- Lee, I-M. & Lee, J-H. (1996). Prediction of Pile Bearing Capacity Using Artificial neural Networks. *Computers and Geotechnics*, 18, 189-200.
- Lefik, M. & Schrefler, B.A. (2003). Artificial Neural Network as an incremental non-linear constitutive model for a finite element code. *Computer Methods in Applied Mechanics and Engineering*, 192, 3265-3283.
- Licznar, P. & Nearing, M.A. (2003). Artificial Neural Networks of Soil Erosion and Runoff Prediction at Plot Scale. *Catena* 51, 89-114.
- Macêdo, F.N. (2003). *Retroanálise de Bacias de Deflexão Reais e Teóricas Obtidas por Métodos Estáticos e Dinâmicos*. Dissertação de Mestrado, Universidade de Brasília, Brasília D.F., p.221.
- Martínez-Carvajal, H. E. & Farias, M. M. (2004). Behavior of Sand Under True Triaxial Conditions Using Artificial Neural Networks, *European Congress on*

Computational Methods in Applied Sciences and Engineering, ECCOMAS 2004, Jyvaskyla, Finlândia, 24-28 julho, 2004.

- Marulanda, C. (2005). Integration of Numerical Modeling and Field Observations of Deep Excavations. PhD Thesis, University of Illinois, Urbana-Champaign, pp.285.
- Matsuoka, H. & Nakai, T. (1985). Relationship among Tresca, Mises, Mohr-Coulomb and Matsuoka-Nakai failure criteria. *Soils and Foundations*, 25 (4), 123-128.
- Mohamed, A.M.O. & Hawas, Y. (2004). Neuro-uzzy Logic Model for Evaluating Water Content of Sandy Soils. *Computer-Aided Civil and Infrastructure Engineering*, 19, 274-287.
- Najjar, Y.; Basheer, I.A. & Naouss, W.A. (1996). On the Identification of Compaction Characteristics by Neuronets. *Computers and Geotechnics*, 18 (3), 167-187.
- Najjar, Y.M.; Ali, H. E. & Basheer, I. A.(1999). On the use of neuronets for simulating the stress – strain behavior of soils. *Numerical Models in Geomechanics – NUMOG VII*, Balkema, Rotterdam. 657-662.
- Nakai, T. (1989). An Isotropic Hardening Elastoplastic Model for Sand Considering the Stress Path dependency in Three-Dimensional Stresses. *Soils and Foundations*, Japan, 29(1), 119-137.
- Nakai, T & Matsuoka, H. (1986). A Generalized Elastoplastic Constitutive Model for Clay in Three-Dimensional Stresses. *Soils and Foundations*, Japan, 23(2), 81-98
- NIT - Nagoya Institute of Technology (2000). Database of true triaxial test on Toyoura Sand.
- Oshiai, H. & Lade, P.V. (1983). Three-Dimensional Behavior of Sand with Anisotropic Fabric. *Journal of Geotechnical Engineering*, 109(10), 1313-1328.
- Pedroso, D. M. (2002). Modelos Elastoplásticos para Solos Argilosos: Capacidade de Previsão do Comportamento e Integração da Relação Constitutiva. Dissertação de Mestrado, Universidade de Brasília, Brasília, Brasil, 164 p.
- Penumadu, D. & Zhao, R. (1999). Triaxial compression behavior of sand and gravel using artificial neural networks. *Computers and Geotechnics*, 24, 207-230.
- Pernot, S. & Lamarque, C. H.(1999). Application of Neural Networks to the Modelling of Some Constitutive Laws. *Neural Networks*, 12, 371-392.
- Reich, Y. & Barai, S. V. (2000). A Methodology for Building Neural Networks Models from Empirical Engineering Data. *Engineering Applications of Artificial Intelligence*, 13, 685-694.
- Shahin, M.A.; Jaksa, M.B. & Maier, H.R. (2002). Predicting the Settlement of Shallow Foundations of Cohesionless Soils using Back-Propagation Neural Networks. *Research report N-R167*, Department of Civil and Environmental Engineering, University of Adelaide.
- Singh, V.K.; Singh, D. & Singh, T.N. (2001). Prediction of Strength Properties of some Schistose Rocks from Petrographic Properties using Artificial Neural Networks. *International Journal of Rock mechanics and Mining Sciences*, 38, 269-284.
- Sidarta, D.E. & Ghaboussi, J. (1998). Constitutive Modeling of Geomaterials from Non-Uniform Material Tests. *Computers and Geotechnics*, 22(1), 53-71

- Souza, R.O. (2002). *Influencia da Irregularidade Longitudinal na Avaliação de Pavimentos*. Dissertação de Mestrado, Universidade de Brasília, Brasília D.F. p.215.
- TRB/NRC-Transportation Research Board/National Research Council. (1999). *Use of artificial networks in geomechanical and pavement systems*. Circular Number E-C012.
- Theocaris, P. S. & Panagiotopoulos, P. D. (1995). Generalized hardening plasticity approximated via anisotropic elasticity: A neural network approach. *Computed Methods in Applied Mechanics and Engineering*, 125,123-139.
- Turk, G.; Logar, J. & Majes, B. (2001). Modeling Soil Behavior in Uniaxial Strain Conditions by Neural Networks. *Advances in Engineering Software*, 32, 805-812.
- Ueng, T.-S., Tzou, Y.-M. & Lee, C.-J. (1988). The Effect of End Restraint on Volume Change and Particle Breakage of Sands in Triaxial Tests. *Advanced Triaxial Testing on Soil and Rock, ASTM STP 977*, Robert T. Donaghe, Ronald C. Chaney and Marshall L. Silver, Eds., ASTM, Philadelphia, 679-691.
- Viggiani, G., Küntz, M. & Desrues, J. (2000). Does Shear banding in Sand Depend on Grain Size Distribution? *International Symposium on Continuous and Discontinuous Modelling of Cohesive Frictional Materials*, Stuttgart, 27-28 april.
- Yamamuro, J. A. & Lade, P. V. (1996). Drained Sand Behavior in Axisymmetric Tests at High Pressures. *Journal of Geotechnical Engineering*, 112(2).
- Yuanyou, X.; Yanming, X. & Ruigeng, Z. (1997). An Engineering Geology Evaluation Method Based on an Artificial Neural Network and its Application. *Engineering Geology*, 47, 149-156.
- Zhao, Z & Ren, L. (2002). Failure Criterion of Concrete Under Triaxial Stresses using Neural networks. *Computer-Aided Civil and Infrastructure Engineering*, 17, 68-73.
- Zhu, J.-H., Zaman, M.M. & Anderson, S.A. (1998). Modeling of Soil Behavior with a Recurrent Neural Network. *Canadian Geotechnical Journal*, 35, 858-872.

APÊNDICE A

BASE DE DADOS

Tabela A.1. Resumo das características gerais dos testes usados na base de dados.

<p>Nota: Não foi possível recuperar as deformações dos testes 4 e 19, em consequência foram descartados</p> <p>Códigos das Colunas:</p> <p>1: Número do teste</p> <p>2: Número da referência original.</p> <p>3: Nome da Areia.</p> <p>4: Número da Figura na referência original.</p> <p>5: Tipo de Ensaio:</p> <p>6: Diâmetro médio, D_{50}</p> <p>7: Coeficiente de Uniformidade, Cu</p> <p>8: Coeficiente de Curvatura, Cc</p>	<p>9: Índice de Vazios inicial, e_i.</p> <p>10: $e_{max}-e_{min}$.</p> <p>11: Tensão de confinamento inicial, σ_{c0} (kPa)</p> <p>12: Dureza do mineral predominante, h.</p> <p>13: Fator de forma, n_s.</p> <p>14: Teor de Finos (%).</p> <p>15: Observações: $b=(\sigma_2-\sigma_3)/(\sigma_1-\sigma_3)$</p> <p style="padding-left: 40px;">θ=ângulo no plano octaédrico</p> <p style="padding-left: 40px;">Dr =Densidade relativa.</p> <p style="padding-left: 40px;">H/D= Relação Altura/Diâmetro.</p>
---	--

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
1	Hoque & Tatsuoka, 1998	Toyoura	15-a TYK1A	HC	0.162	1.46	1.05	0.95	0.364	49	7	20	0	
2	Oshiai & Lade, 1983	Cambria	5-a	TC	1.32	1.4	0.9	0.535	0.29	98.1	7	17	0	$\theta=0, b=0,$ $\sigma_{oct}=cte$
3			5-a		1.32	1.4	0.9	0.535	0.29	98.1	7	17	0	$\theta=120$ $b=0$ $\sigma_{oct}=cte$
5			5-f	TE	1.32	1.4	0.9	0.535	0.29	98.1	7	17	0	$\theta=61$ $b=0.98$ $\sigma_{oct}=cte$

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	
6			5-f		1.32	1.4	0.9	0.535	0.29	98.1	7	17	0	$\theta=181$ $b=0.98$ $\sigma_{oct}=cte$	
7	Lade & Duncan, 1973	Monterrey	4	True	0.43	1.53	0.89	0.78	0.295	58.8	6	17	0	$b=0.00$	
8			3		0.43	1.53	0.89	0.57	0.295	58.8	6	17	0	$b=0.0$	
9			3		0.43	1.53	0.89	0.57	0.295	58.8	6	17	0	$b=0.15$	
10			3		0.43	1.53	0.89	0.57	0.295	58.8	6	17	0	$b=0.50$	
11			3		0.43	1.53	0.89	0.57	0.295	58.8	6	17	0	$b=0.75$	
12			3		0.43	1.53	0.89	0.57	0.295	58.8	6	17	0	$b=0.90$	
13			3		0.43	1.53	0.89	0.57	0.295	58.8	6	17	0	$b=1.00$	
14			4		0.43	1.53	0.89	0.78	0.295	58.8	6	17	0	$b=0.20$	
15			4		0.43	1.53	0.89	0.78	0.295	58.8	6	17	0	$b=0.60$	
16			4		0.43	1.53	0.89	0.78	0.295	58.8	6	17	0	$b=0.75$	
17			4		0.43	1.53	0.89	0.78	0.295	58.8	6	17	0	$b=0.90$	
18			4		0.43	1.53	0.89	0.78	0.295	58.8	6	17	0	$b=1.00$	
20	NTI, 2000	Toyoura	95ts14	TC	0.162	1.46	1.05	0.666	0.364	196	7	20	0	$\sigma_{oct}=cte$	
21			95ts15	TE	0.162	1.46	1.05	0.68	0.364	196	7	20	0	$\sigma_{oct}=cte.$	
22			95ts17	CTC	0.162	1.46	1.05	0.664	0.364	196	7	20	0	$\sigma_{rad}=cte.$	
23			95ts18	RTE	0.162	1.46	1.05	0.677	0.364	196	7	20	0	$\sigma_{rad}=cte$	
24			96ts10	RTC	0.162	1.46	1.05	0.659	0.364	196	7	20	0	$\sigma_a=cte.$	
25			97ts03	CTE	0.162	1.46	1.05	0.677	0.364	196	7	20	0	$\sigma_a=cte.$	
26			96ts12	CTC	0.162	1.46	1.05	0.832	0.364	196	7	20	0	$\sigma_{rad}=cte.$	
27			96ts14	TC	0.162	1.46	1.05	0.851	0.364	196	7	20	0	$\sigma_{oct}=cte.$	
28			97ts04	RTE	0.162	1.46	1.05	0.847	0.364	196	7	20	0	$\sigma_{rad}=cte.$	
29			97ts07	RTC	0.162	1.46	1.05	0.861	0.364	196	7	20	0	$\sigma_a=cte.$	
30			97ts08	CTE	0.162	1.46	1.05	0.84	0.364	196	7	20	0	$\sigma_a=cte.$	
31			97ts09	TE	0.162	1.46	1.05	0.866	0.364	196	7	20	0	$\sigma_{oct}=cte.$	
32			53-21	True		0.162	1.46	1.05	0.866	0.364	196	7	20	0	$\theta=15$
33			53-23			0.162	1.46	1.05	0.866	0.364	196	7	20	0	$\theta=30$
34			53-25			0.162	1.46	1.05	0.866	0.364	196	7	20	0	$\theta=45$
35	Hoque & Tatsuoka,	SLB	15-b SLK1V	HC	0.62	1.11	1.05	0.524	0.3	49	7	17	0		
36	1998	Ticino	15-d TCK1A		0.502	1.33	1.05	0.58	0.37	49	8	20	0		
37	Penumadu & Zhao, 1999	Chattahoochee	4-a R7T20	CTC	0.78	1.2-2.5	0.82	0.902	0.457	241	7	20	0		
38		Ottawa	5/R7T1		0.78	1.2	0.93	0.649	0.457	483	7	15	0		

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
39		Chatahoochee	6/R7T6		0.78	1.2	0.93	0.646	0.457	241	7	15	0	
40			6/R7T24		0.78	2.5	0.82	0.892	0.457	35	7	20	0	
41		Toyoura	7/R24T5		0.16	1.5	1.05	0.61	0.372	19613	7	20	0	
42			8/R24T1		0.16	1.5	1.05	0.61	0.372	9807	7	20	0	
43			8/R24T3		0.16	1.5	1.05	0.61	0.372	19613	7	20	0	
44			8/R24T12		0.16	1.5	1.05	0.83	0.372	29420	7	20	0	
45	Lee, 1970	Antioch	5-a	CTC	0.223	1.47	0.98	0.85	0.39	98.1	5	25	0	Dr=78%
46			5-b		0.223	1.47	0.98	1	0.39	98.1	5	25	0	Dr=38%
47			5-c		0.223	1.47	0.98	1	0.39	490	5	25	0	Dr=38%
48	Lee et al., 1983	Upper San Fdo Dam	10	CTC	0.09	5	0.8	0.99	0.88	196	6	20	40	Dr=55%
49		Lower S. Fdo Dam	11		0.09	3.6	1.3	0.99	0.88	196	6	20	40	Dr=55%
50	Goto & Tatsuoka, 1988	Toyoura	3	CTC	0.16	1.46	1.05	0.676	0.4	98.1	7	20	0	H/D=2,1
51			3		0.16	1.46	1.05	0.68	0.4	98.1	7	20	0	H/D=2
52			3		0.16	1.46	1.05	0.677	0.4	98.1	7	20	0	H/D=2
53			4		0.16	1.46	1.05	0.907	0.4	98.1	7	20	0	H/D=2,1
54			4		0.16	1.46	1.05	0.891	0.4	98.1	7	20	0	H/D=2
55			4		0.16	1.46	1.05	0.875	0.4	98.1	7	20	0	H/D=2
56	Oshiai & Lade, 1983	Cambria	4	TC	1.32	1.4	0.9	0.535	0.29	392.4	7	17	0	$\theta=0, b=0$
57			4		1.32	1.4	0.9	0.535	0.29	196.2	7	17	0	$\theta=0, b=0$
58			4		1.32	1.4	0.9	0.535	0.29	98.1	7	17	0	$\theta=0, b=0$
59			4		1.32	1.4	0.9	0.535	0.29	49.05	7	17	0	$\theta=0, b=0$
60			4		1.32	1.4	0.9	0.535	0.29	392.4	7	17	0	$\theta=120, b=0$
61			4		1.32	1.4	0.9	0.535	0.29	196.2	7	17	0	$\theta=120, b=0$
62			4		1.32	1.4	0.9	0.535	0.29	98.1	7	17	0	$\theta=120, b=0$
63			4		1.32	1.4	0.9	0.535	0.29	49.05	7	17	0	$\theta=120, b=0$
64			5-c		1.32	1.4	0.9	0.535	0.29	196.2	7	17	0	$b_{final}=0.31$
65			5-c	D.P.	1.32	1.4	0.9	0.535	0.29	196.2	7	17	0	$b_{final}=0.33$
66			5-c	1.32	1.4	0.9	0.535	0.29	196.2	7	17	0	$b_{final}=0.32$	
67	3	HC	1.32	1.4	0.9	0.535	0.29	19.62	7	17	0			
68	Ueng et al., 1988	Ottawa	4	CTC	0.78	1.2	0.93	0.65	0.3	200	7	15	0	Dr=50%
69			7		0.78	1.2	0.93	0.71	0.3	100	7	15	0	Dr=30%
70	Dangus et al., 1988	Hostun	2-a	CTC	0.32	1.7	0.94	0.689	0.343	90	7	25	3	Dr=90%
71			2-c		0.32	1.7	0.94	0.929	0.343	90	7	25	3	Dr=20%

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
72	Kolymbas & Wu, 1990	Calcareous Sand	4		0.32	1.7	0.94	0.689	0.343	10000	7	25	3	Dr=90%
73			11	CTC	0.17	2.8	0.77	1.076	0.649	50	3	20	9	Dr=90%
74			11		0.17	2.8	0.77	1.076	0.649	100	3	20	9	Dr=90%
75			11		0.17	2.8	0.77	1.076	0.649	300	3	20	9	Dr=90%
76			11		0.17	2.8	0.77	1.076	0.649	500	3	20	9	Dr=90%
77			11		0.17	2.8	0.77	1.076	0.649	15000	3	20	9	Dr=90%
78			11		0.17	2.8	0.77	1.076	0.649	1000	3	20	9	Dr=90%
79			11		0.17	2.8	0.77	1.076	0.649	10000	3	20	9	Dr=90%
80			11		0.17	2.8	0.77	1.076	0.649	5000	3	20	9	Dr=90%
81		Hostun	12		CTC	0.32	1.7	0.94	0.689	0.343	50	7	25	3
82			12	0.32		1.7	0.94	0.689	0.343	300	7	25	3	Dr=90%
83			12	0.32		1.7	0.94	0.689	0.343	1000	7	25	3	Dr=90%
84			12	0.32		1.7	0.94	0.689	0.343	2000	7	25	3	Dr=90%
85			12	0.32		1.7	0.94	0.689	0.343	3000	7	25	3	Dr=90%
86			12	0.32		1.7	0.94	0.689	0.343	5000	7	25	3	Dr=90%
87			12	0.32		1.7	0.94	0.689	0.343	10000	7	25	3	Dr=90%
88			12	0.32		1.7	0.94	0.689	0.343	15000	7	25	3	Dr=90%
89		Karlsruhe	CTC	8-a	0.33	2	0.8	0.535	0.314	200	6	17	0	Dr=98%/Upper
90	8-a			0.33	2	0.8	0.535	0.314	200	6	17	0	Dr=98%/Middle	
91	8-a			0.33	2	0.8	0.535	0.314	200	6	17	0	Dr=98%/Lower	
92	9			0.33	2	0.8	0.535	0.314	100	6	17	0	Dr=98%	
93	10			0.33	2	0.8	0.547	0.314	200	6	17	0	Dr=94.2%	
94	15-a			0.33	2	0.8	0.535	0.314	50	6	17	0	Dr=98%	
95	15-a			0.33	2	0.8	0.535	0.314	100	6	17	0	Dr=98%	
96	15-a			0.33	2	0.8	0.535	0.314	200	6	17	0	Dr=98%	
97	15-a			0.33	2	0.8	0.535	0.314	300	6	17	0	Dr=98%	
98	15-a			0.33	2	0.8	0.535	0.314	400	6	17	0	Dr=98%	
99	15-a			0.33	2	0.8	0.535	0.314	500	6	17	0	Dr=98%	
100	15-a			0.33	2	0.8	0.535	0.314	600	6	17	0	Dr=98%	
101	15-a			0.33	2	0.8	0.535	0.314	800	6	17	0	Dr=98%	
102	15-a			0.33	2	0.8	0.535	0.314	1000	6	17	0	Dr=98%	
103	15-b			0.33	2	0.8	0.792	0.314	50	6	17	0	Dr=16.2%	
104	15-b			0.33	2	0.8	0.792	0.314	100	6	17	0	Dr=16.2%	
105	15-b			0.33	2	0.8	0.792	0.314	200	6	17	0	Dr=16.2%	

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	
106			15-b		0.33	2	0.8	0.792	0.314	300	6	17	0	Dr=16.2%	
107			15-b		0.33	2	0.8	0.792	0.314	400	6	17	0	Dr=16.2%	
108			15-b		0.33	2	0.8	0.792	0.314	600	6	17	0	Dr=16.2%	
109			15-b		0.33	2	0.8	0.792	0.314	1000	6	17	0	Dr=16.2%	
110			21		0.33	2	0.8	0.547	0.314	100	6	17	0	Dr=94%	
111			21		0.33	2	0.8	0.558	0.314	100	6	17	0	Dr=90.6%	
112			21		0.33	2	0.8	0.602	0.314	100	6	17	0	Dr=76.6%	
113			21		0.33	2	0.8	0.635	0.314	100	6	17	0	Dr=66.2%	
114			21		0.33	2	0.8	0.677	0.314	100	6	17	0	Dr=52.8%	
115			21		0.33	2	0.8	0.71	0.314	100	6	17	0	Dr=42.2%	
116			21		0.33	2	0.8	0.755	0.314	100	6	17	0	Dr=27.9%	
117	Evesque, 2001	Hostun	1-a	TC	0.32	1.7	0.94	0.923	0.343	600	7	25	0	p=cte	
118			1-b		0.32	1.7	0.94	1.01	0.343	350	7	25	0	p=cte	
119			1-c		0.32	1.7	0.94	0.92	0.343	354	7	25	0	p=cte	
120			1-d		0.32	1.7	0.94	0.92	0.343	70	7	25	0	p=cte	
121			1-e		0.32	1.7	0.94	0.93	0.343	55	7	25	0	p=cte	
122	Cambou & Lanier, 1988	Hostun	3-'0'	CTC	0.32	1.7	0.94	0.684	0.343	200	7	25	0	b=0	
123			3-'1'		0.32	1.7	0.94	0.684	0.343	200	7	25	0	b=0	
124			3-'2'		0.32	1.7	0.94	0.684	0.343	200	7	25	0	b=0	
125			4-'0'	RTE	0.32	1.7	0.94	0.673	0.343	200	7	25	0	b=1	
126			4-'3'		0.32	1.7	0.94	0.673	0.343	200	7	25	0	b=1	
127			4-'4'		0.32	1.7	0.94	0.673	0.343	200	7	25	0	b=1	
128	Haruyama, 1987	Shirasu	3a	CTC				0.98	1.889	98		25		$\theta=0, \sigma_{oct}=cte$	
129			3a					1.272	1.889	98		25		$\theta=0, \sigma_{oct}=cte$	
130			3b	True					0.98	1.889	98		25		$\theta=30, \sigma_{oct}=cte$
131			3b					1.272	1.889	98		25		$\theta=30, \sigma_{oct}=cte$	
132			3c					0.98	1.889	98		25		$\theta=60, \sigma_{oct}=cte$	
133			3c					1.272	1.889	98		25		$\theta=60, \sigma_{oct}=cte$	
134			3d					0.98	1.889	98		25		$\theta=90, \sigma_{oct}=cte$	
135			3d					1.272	1.889	98		25		$\theta=90, \sigma_{oct}=cte$	

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	
136			3e					0.98	1.889	98		25		$\theta=120,$ $\sigma_{oct}=cte$	
137			3e					1.272	1.889	98		25		$\theta=120,$ $\sigma_{oct}=cte$	
138			3f					0.98	1.889	98		25		$\theta=150,$ $\sigma_{oct}=cte$	
139			3f					1.272	1.889	98		25		$\theta=150,$ $\sigma_{oct}=cte$	
140			3g					0.98	1.889	98		25		$\theta=180,$ $\sigma_{oct}=cte$	
141			3g					1.272	1.889	98		25		$\theta=180,$ $\sigma_{oct}=cte$	
142	Hicher, 1996	Hostun	3	CTC	1.95	1.4		0.55	0.343	70	7	25			
143			3		1.95	1.4		0.55	0.343	110	7	25			
144			3		1.95	1.4		0.55	0.343	160	7	25			
145			3		1.95	1.4		0.55	0.343	225	7	25			
146	Yamamuro & Lade, 1996	Cambria	1	HC	1.32	1.4	0.9	0.54	0.29	0	7	17	0		
147			1	HE	1.32	1.4	0.9	0.45	0.29	69000	7	17	0		
148			2	CTC	1.32	1.4	0.9	0.54	0.29	2100	7	17	0		
149			2		1.32	1.4	0.9	0.54	0.29	4000	7	17	0		
150			2		1.32	1.4	0.9	0.54	0.29	5800	7	17	0		
151			2		1.32	1.4	0.9	0.54	0.29	8000	7	17	0		
152			2		1.32	1.4	0.9	0.54	0.29	11500	7	17	0		
153			3		1.32	1.4	0.9	0.54	0.29	15000	7	17	0		
154			3		1.32	1.4	0.9	0.54	0.29	17200	7	17	0		
155			3		1.32	1.4	0.9	0.54	0.29	26000	7	17	0		
156			3		1.32	1.4	0.9	0.54	0.29	40000	7	17	0		
157			3		1.32	1.4	0.9	0.54	0.29	52000	7	17	0		
158			5		CTE	1.32	1.4	0.9	0.54	0.29	250	7	17	0	
159			5		1.32	1.4	0.9	0.54	0.29	1000	7	17	0		
160			5		1.32	1.4	0.9	0.54	0.29	2200	7	17	0		
161			5	1.32	1.4	0.9	0.54	0.29	4000	7	17	0			
162			5	1.32	1.4	0.9	0.54	0.29	6000	7	17	0			
163			5	1.32	1.4	0.9	0.54	0.29	8000	7	17	0			
164			5	1.32	1.4	0.9	0.54	0.29	10000	7	17	0			
165	5	1.32	1.4	0.9	0.54	0.29	12000	7	17	0					
166	6	1.32	1.4	0.9	0.54	0.29	14500	7	17	0					

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
167			6		1.32	1.4	0.9	0.54	0.29	17500	7	17	0	
168			6		1.32	1.4	0.9	0.54	0.29	22000	7	17	0	
169			6		1.32	1.4	0.9	0.54	0.29	24000	7	17	0	
170			6		1.32	1.4	0.9	0.54	0.29	26000	7	17	0	
171			6		1.32	1.4	0.9	0.54	0.29	35000	7	17	0	
172			6		1.32	1.4	0.9	0.54	0.29	42000	7	17	0	
173			6		1.32	1.4	0.9	0.54	0.29	52000	7	17	0	
<p>Nota: Não foi possível recuperar as deformações dos testes 4 e 19, em consequência foram descartados</p> <p>Códigos das Colunas:</p> <p>1: Número do teste 2: Número da referência original. 3: Nome da Areia. 4: Número da Figura na referência original. 5: Tipo de Ensaio: 6: Diâmetro médio, D_{50} 7: Coeficiente de Uniformidade, C_u 8: Coeficiente de Curvatura, C_c</p>					<p>9: Índice de Vazios inicial, e_i.</p> <p>10: $e_{max}-e_{min}$.</p> <p>11: Tensão de confinamento inicial, σ_{c0} (kPa)</p> <p>12: Dureza do mineral predominante, h.</p> <p>13: Fator de forma, n_s.</p> <p>14: Teor de Finos (%).</p> <p>15: Observações: $b=(\sigma_2-\sigma_3)/(\sigma_1-\sigma_3)$ θ=ângulo no plano octaédrico D_r =Densidade relativa. H/D= Relação Altura/Diâmetro.</p>									

APÊNDICE B

CLASSIFICAÇÃO DOS TESTES

Tabela B.1. Resultado da classificação realizada com três redes tipo Kohonen. Em preto e azul os testes agrupados na classe 1 usados para treinamento e generalização respectivamente, em vermelho os testes pertencentes às outras classes.

Número	Areia	e_0	$e_{\max}-e_{\min}$	σ_{c_0}	n_s	Número	Areia	e_0	$e_{\max}-e_{\min}$	σ_{c_0}	n_s
1	Toyoura	0.95	0.36	49	20	103	Karlsruhe	0.79	0.31	50	17
2	Cambria	0.54	0.29	98	17	104	Karlsruhe	0.79	0.31	100	17
3	Cambria	0.54	0.29	98	17	105	Karlsruhe	0.79	0.31	200	17
5	Cambria	0.54	0.29	98	17	106	Karlsruhe	0.79	0.31	300	17
6	Cambria	0.54	0.29	98	17	107	Karlsruhe	0.79	0.31	400	17
7	Monterrey	0.78	0.30	59	17	108	Karlsruhe	0.79	0.31	600	17
8	Monterrey	0.57	0.30	59	17	109	Karlsruhe	0.79	0.31	1000	17
9	Monterrey	0.57	0.30	59	17	110	Karlsruhe	0.55	0.31	100	17
10	Monterrey	0.57	0.30	59	17	111	Karlsruhe	0.56	0.31	100	17
11	Monterrey	0.57	0.30	59	17	112	Karlsruhe	0.60	0.31	100	17
12	Monterrey	0.57	0.30	59	17	113	Karlsruhe	0.64	0.31	100	17
13	Monterrey	0.57	0.30	59	17	114	Karlsruhe	0.68	0.31	100	17
14	Monterrey	0.78	0.30	59	17	115	Karlsruhe	0.71	0.31	100	17
15	Monterrey	0.78	0.30	59	17	116	Karlsruhe	0.76	0.31	100	17
16	Monterrey	0.78	0.30	59	17	117	Hostun	0.92	0.34	600	25
17	Monterrey	0.78	0.30	59	17	118	Hostun	1.01	0.34	350	25
18	Monterrey	0.78	0.30	59	17	119	Hostun	0.92	0.34	354	25
20	Toyoura	0.67	0.36	196	20	120	Hostun	0.92	0.34	70	25
21	Toyoura	0.68	0.36	196	20	121	Hostun	0.93	0.34	55	25
22	Toyoura	0.66	0.36	196	20	122	Hostun	0.68	0.34	200	25
23	Toyoura	0.68	0.36	196	20	123	Hostun	0.68	0.34	200	25
24	Toyoura	0.66	0.36	196	20	124	Hostun	0.68	0.34	200	25
25	Toyoura	0.68	0.36	196	20	125	Hostun	0.67	0.34	200	25
26	Toyoura	0.83	0.36	196	20	126	Hostun	0.67	0.34	200	25
27	Toyoura	0.85	0.36	196	20	127	Hostun	0.67	0.34	200	25
28	Toyoura	0.85	0.36	196	20	142	Hostun	0.55	0.34	70	25
29	Toyoura	0.86	0.36	196	20	143	Hostun	0.55	0.34	110	25
30	Toyoura	0.84	0.36	196	20	144	Hostun	0.55	0.34	160	25
31	Toyoura	0.87	0.36	196	20	145	Hostun	0.55	0.34	225	25
32	Toyoura	0.87	0.36	196	20	146	Cambria	0.54	0.29	0	17
33	Toyoura	0.87	0.36	196	20	148	Cambria	0.54	0.29	2100	17
34	Toyoura	0.87	0.36	196	20	149	Cambria	0.54	0.29	4000	17
35	SLB	0.52	0.30	49	17	158	Cambria	0.54	0.29	250	17
36	Ticino	0.58	0.37	49	20	159	Cambria	0.54	0.29	1000	17

Número	Areaia	e_0	$e_{\max}-e_{\min}$	σ_{C_0}	n_s	Número	Areaia	e_0	$e_{\max}-e_{\min}$	σ_{C_0}	n_s
37	Chattahoochee	0.90	0.46	241	20	160	Cambria	0.54	0.29	2200	17
38	Ottawa	0.65	0.46	483	15	161	Cambria	0.54	0.29	4000	17
39	Ottawa	0.65	0.46	241	15	Testes Rejeitados					
40	Chattahoochee	0.89	0.46	35	20	41	Toyoura	0.61	0.37	19613	20
45	Antioch	0.85	0.39	98	25	42	Toyoura	0.61	0.37	9807	20
46	Antioch	1.00	0.39	98	25	43	Toyoura	0.61	0.37	19613	20
47	Antioch	1.00	0.39	490	25	44	Toyoura	0.83	0.37	29420	20
48	U. S.Fdo Dam	0.99	0.88	196	20	58	Cambria	0.54	0.29	98.1	17
49	L. S.Fdo Dam	0.99	0.88	196	20	59	Cambria	0.54	0.29	49.05	17
50	Toyoura	0.68	0.40	98	20	60	Cambria	0.54	0.29	392.4	17
51	Toyoura	0.68	0.40	98	20	61	Cambria	0.54	0.29	196.2	17
52	Toyoura	0.68	0.40	98	20	62	Cambria	0.54	0.29	98.1	17
53	Toyoura	0.91	0.40	98	20	72	Hostun	0.69	0.34	10000	25
54	Toyoura	0.89	0.40	98	20	77	Calcareous	1.08	0.65	15000	20
55	Toyoura	0.88	0.40	98	20	79	Calcareous	1.08	0.65	10000	20
56	Cambria	0.54	0.29	392	17	87	Hostun	0.69	0.34	10000	25
57	Cambria	0.54	0.29	196	17	88	Hostun	0.69	0.34	15000	25
63	Cambria	0.54	0.29	49	17	128	Shirasu	0.98	1.89	98	25
64	Cambria	0.54	0.29	196	17	129	Shirasu	1.27	1.89	98	25
65	Cambria	0.54	0.29	196	17	130	Shirasu	0.98	1.89	98	25
66	Cambria	0.54	0.29	196	17	131	Shirasu	1.27	1.89	98	25
67	Cambria	0.54	0.29	20	17	132	Shirasu	0.98	1.89	98	25
68	Ottawa	0.65	0.30	200	15	133	Shirasu	1.27	1.89	98	25
69	Ottawa	0.71	0.30	100	15	134	Shirasu	0.98	1.89	98	25
70	Hostun	0.69	0.34	90	25	135	Shirasu	1.27	1.89	98	25
71	Hostun	0.93	0.34	90	25	136	Shirasu	0.98	1.89	98	25
73	Calcareous	1.08	0.65	50	20	137	Shirasu	1.27	1.89	98	25
74	Calcareous	1.08	0.65	100	20	138	Shirasu	0.98	1.89	98	25
75	Calcareous	1.08	0.65	300	20	139	Shirasu	1.27	1.89	98	25
76	Calcareous	1.08	0.65	500	20	140	Shirasu	0.98	1.89	98	25
78	Calcareous	1.08	0.65	1000	20	141	Shirasu	1.27	1.89	98	25
80	Calcareous	1.08	0.65	5000	20	147	Cambria	0.45	0.29	69000	17
81	Hostun	0.69	0.34	50	25	150	Cambria	0.54	0.29	5800	17
82	Hostun	0.69	0.34	300	25	151	Cambria	0.54	0.29	8000	17
83	Hostun	0.69	0.34	1000	25	152	Cambria	0.54	0.29	11500	17
84	Hostun	0.69	0.34	2000	25	153	Cambria	0.54	0.29	15000	17
85	Hostun	0.69	0.34	3000	25	154	Cambria	0.54	0.29	17200	17
86	Hostun	0.69	0.34	5000	25	155	Cambria	0.54	0.29	26000	17
89	Karlsruhe	0.54	0.31	200	17	156	Cambria	0.54	0.29	40000	17
90	Karlsruhe	0.54	0.31	200	17	157	Cambria	0.54	0.29	52000	17
91	Karlsruhe	0.54	0.31	200	17	162	Cambria	0.54	0.29	6000	17
92	Karlsruhe	0.54	0.31	100	17	163	Cambria	0.54	0.29	8000	17
93	Karlsruhe	0.55	0.31	200	17	164	Cambria	0.54	0.29	10000	17
94	Karlsruhe	0.54	0.31	50	17	165	Cambria	0.54	0.29	12000	17
95	Karlsruhe	0.54	0.31	100	17	166	Cambria	0.54	0.29	14500	17
96	Karlsruhe	0.54	0.31	200	17	167	Cambria	0.54	0.29	17500	17
97	Karlsruhe	0.54	0.31	300	17	168	Cambria	0.54	0.29	22000	17
98	Karlsruhe	0.54	0.31	400	17	169	Cambria	0.54	0.29	24000	17
99	Karlsruhe	0.54	0.31	500	17	170	Cambria	0.54	0.29	26000	17
100	Karlsruhe	0.54	0.31	600	17	171	Cambria	0.54	0.29	35000	17
101	Karlsruhe	0.54	0.31	800	17	172	Cambria	0.54	0.29	42000	17
102	Karlsruhe	0.54	0.31	1000	17	173	Cambria	0.54	0.29	52000	17

BASE DE DADOS – MODELO NEURAL 4

Tabela C.1. Intervalos de valores da base de dados usada para o treinamento dos modelos neurais PMC.

Entrada	max	min
n_s	25	15
σ_0 (kPa)	5000	0
$e_{\max}-e_{\min}$	0.88	0.29
e_i	1.08	0.52
$\varepsilon_1(\%)$	55.80	-21.33
$\varepsilon_2(\%)$	10.63	-28.85
$\varepsilon_3(\%)$	29.00	-28.85
$\Delta\varepsilon_1(\%)$	6.09	-1.87
$\Delta\varepsilon_2(\%)$	0.83	-3.32
$\Delta\varepsilon_3(\%)$	1.70	-3.32
σ_1 (kPa)	69052	0
σ_2 (kPa)	69052	0
σ_3 (kPa)	69052	0
Saída	max	min
$\Delta\sigma_1$ (kPa)	3035.4	-507
$\Delta\sigma_2$ (kPa)	3035.4	-100
$\Delta\sigma_3$ (kPa)	3035.4	-525

Tabela C.2. Síntese das diferentes trajetórias usadas no treinamento e validação da rede *Magola9*, para cada tipo de areia.

Tipo de Areia	Tipo de Teste															Total Global
	CTC	CTE	D.P.	HC	RTC	RTE	TC	TE	0=9	0=30	0=45	0=54	0=12	0=36	0=15	
Antioch	3															3
Calcareous	6															6
Cambria	2	4	3	2			5	2								18
Chattahoochee	2															2
Hostun	15					3	5									23
Karlsruhe	28															28
L. S.Fdo Dam	1															1
Monterrey							2	2	1	1	2	2	1	1		12
Ottawa	4															4
SLB				1												1
Ticino				1												1
Toyoura	8	2		1	2	2	2	2		1	1				1	22
U S.Fdo Dam	1															1
Total Global	70	6	3	5	2	5	14	6	1	2	3	2	1	1	1	122

APÊNDICE D

VALORES DOS PARÂMETROS LIVRES DA REDE *MAGOLA11*

Tabela D.1. Valores dos pesos das conexões entre as 10 primeiras entradas da camada de entrada e a primeira camada oculta da rede *Magola11*. Matriz de pesos IW.

	1	2	3	4	5	6	7	8	9	10
1	0.475450	-1.779800	0.977500	0.435670	0.502610	1.061000	-0.475160	0.752310	-0.495550	0.475450
2	-0.177420	-1.560500	0.288830	0.546710	0.764460	0.218760	0.351270	0.868550	-0.072750	-0.177420
3	2.319800	0.241280	1.279700	-0.050980	-1.149700	0.233740	-0.616900	-0.625130	-0.417500	2.319800
4	-2.290700	1.187200	-0.179330	0.119430	-0.790480	0.749570	-0.929310	0.962660	-0.258770	-2.290700
5	1.601500	2.083600	-0.352490	-1.543700	-0.667410	0.034830	1.067400	-0.097760	-0.126180	1.601500
6	1.644600	1.239100	0.956510	-0.743320	1.204500	-2.706700	-2.604700	0.935490	-1.178800	1.644600
7	0.941480	-0.274030	-0.725980	-0.596630	-0.646470	0.449130	-0.448460	-0.162110	1.109400	0.941480
8	0.772440	0.359940	0.423330	0.740100	0.249740	-1.008300	0.387230	0.989800	0.394310	0.772440
9	-2.131800	-0.845480	-0.077310	-0.383040	1.367400	0.330670	0.806280	0.458140	1.051500	-2.131800
10	0.747940	0.723450	0.165130	-0.572540	-0.646240	0.671920	-0.542780	-1.467200	0.445340	0.747940
11	-1.030300	-0.933530	1.322700	1.076400	-0.667880	0.489500	-0.811920	0.336910	0.560320	-1.030300
12	0.401150	-1.511800	-0.563470	-0.568260	-0.957780	0.671070	-0.669410	-0.397040	-0.003290	0.401150
13	0.321680	0.761630	1.229800	0.976700	0.822700	-0.702320	0.049470	-0.232040	1.269600	0.321680
14	1.003000	1.169300	0.890540	-0.088930	0.399430	0.763170	0.945560	0.572050	-0.751510	1.003000
15	1.340400	0.715140	1.628200	2.655500	0.898960	0.999760	-1.070400	-1.054300	-0.224070	1.340400
16	-0.013490	0.259970	0.482280	0.863770	0.930260	0.711840	0.253080	0.832150	0.020620	-0.013490
17	0.510770	1.137300	1.573900	0.210110	-0.388040	-0.080040	-1.180800	-1.322100	-1.094800	0.510770
18	-1.117100	-1.817100	1.429400	1.699700	-0.184250	0.019910	0.290240	-2.367700	-0.173060	-1.117100
19	-0.645860	0.399560	-0.592680	-2.796300	2.893900	-0.285870	1.667200	0.527830	-1.263700	-0.645860
20	0.530490	0.359980	-0.356580	-1.094400	0.350990	0.103130	-0.101160	-1.003400	0.293990	0.530490
21	-1.024200	-0.163110	-1.062800	-0.766510	-1.452600	0.083410	0.568780	-0.358930	-0.078220	-1.024200
22	0.144440	0.081490	-0.543410	-0.780730	-1.407000	0.774300	-0.457330	-0.886240	-0.839760	0.144440
23	1.681200	1.197400	-0.329950	1.039300	-0.707400	-1.142600	1.868500	-1.631100	-2.072100	1.681200
24	0.136710	0.280480	-0.118380	0.896270	-0.806390	0.022270	0.904740	-0.811280	-0.854470	0.136710
25	-0.673220	-0.368700	0.459050	-0.716530	-1.000400	0.569470	0.189670	-0.360860	0.813610	-0.673220
26	-2.996000	0.239010	-0.181450	1.143000	1.180600	-0.363300	-1.422600	1.083000	-0.817950	-2.996000
27	0.210510	1.238700	-0.447920	0.114300	-0.583500	-0.694880	0.444640	0.814790	0.844670	0.210510
28	0.364940	-2.491400	0.316700	-0.044690	-0.759710	-0.737340	-0.627030	-0.739870	-0.158000	0.364940
29	-0.657270	1.229300	-0.764060	-0.835530	0.982150	-0.515510	0.221060	0.782570	1.614900	-0.657270
30	0.749200	-0.525070	-0.310230	-0.106930	0.017920	0.634920	-0.596390	1.507500	-1.213700	0.749200
31	-0.124040	0.757910	-0.996690	0.702180	1.467400	1.192700	-0.034230	0.607600	-0.320680	-0.124040
32	-0.792410	0.932480	0.401340	0.728030	-0.348980	0.540210	-0.554830	0.834910	-0.270140	-0.792410
33	-0.870030	-0.194620	0.070370	0.375670	-0.927470	0.427480	0.523100	0.318950	-0.291080	-0.870030
34	-1.060600	0.331230	0.363410	-0.910950	0.872980	1.263500	0.318580	0.626110	0.057810	-1.060600
35	-3.309900	2.088000	-0.815820	1.305800	1.568900	2.287800	-1.923600	1.067300	-0.933800	-3.309900
36	-0.312330	1.224100	1.264400	0.734780	-0.263660	-0.211220	-0.859040	0.942370	-0.480660	-0.312330

	1	2	3	4	5	6	7	8	9	10
37	0.033200	1.848600	-0.393790	0.803490	-0.347790	-0.653140	0.390290	-0.917510	0.039710	0.033200
38	0.028780	0.344370	-0.949090	-6.172500	0.169250	0.034550	0.802220	-0.085730	-0.397000	0.028780
39	-0.552830	0.654900	-0.415230	0.185460	0.081930	-0.327180	1.705000	0.078740	0.147850	-0.552830
40	0.990360	0.347020	-0.334960	-1.223500	-0.509510	-0.114780	-0.428130	0.878810	-0.740840	0.990360
41	0.605550	0.609630	-0.600300	-0.263740	0.509100	-0.032800	0.957830	-0.567290	0.695990	0.605550
42	-0.238140	0.225070	-0.191790	0.490300	0.722660	0.496700	0.319590	0.416050	0.774940	-0.238140
43	-1.649900	1.408400	-0.931850	1.972700	-0.805990	0.156960	0.390340	-0.978620	0.548790	-1.649900
44	-0.711560	0.193800	0.523450	0.657070	1.313700	-0.783290	0.239040	0.211780	-1.018400	-0.711560
45	0.158730	-0.369050	-0.209640	0.830950	0.550020	-0.741630	-0.030680	0.694850	0.078970	0.158730
46	-0.780320	0.496610	-0.668050	-0.831680	-0.671300	-0.084110	-0.603270	-0.240300	-0.353320	-0.780320
47	-0.734070	3.109800	-0.961410	-4.008200	0.229220	0.337170	-0.230630	-0.375300	0.067400	-0.734070
48	0.216790	0.763650	-0.178710	0.692630	-0.817190	-0.624850	-0.679410	-0.508300	0.426210	0.216790
49	0.408810	0.529120	-0.913110	-0.396700	0.584490	-0.807850	-1.144700	0.913160	-0.444080	0.408810
50	0.979550	2.764800	1.368900	0.871460	0.746640	-0.659930	-0.485330	-0.130160	-0.785740	0.979550
51	-2.476800	-1.623600	0.665730	-0.106520	0.727590	-0.750750	-1.533400	0.171240	0.674230	-2.476800
52	-1.386200	1.374200	-1.352500	-2.029600	-0.572350	1.068500	1.001100	0.307880	1.101000	-1.386200
53	-1.060500	-0.590340	-1.123400	-1.109700	-0.205110	-0.856050	-1.983700	0.333980	0.613130	-1.060500
54	-0.254380	-0.389560	0.215580	-1.160300	-0.198380	0.658390	0.589400	0.034020	0.064650	-0.254380
55	-1.144500	1.882200	0.443610	1.691500	-0.648440	-2.144100	1.680600	-0.333950	-3.665700	-1.144500
56	0.336070	0.783380	1.027300	0.100950	-0.653610	-0.677510	-0.687840	0.002420	-0.503680	0.336070
57	3.140800	0.376220	-0.151340	0.181390	-0.468110	-0.380610	-1.236400	0.063320	0.399420	3.140800
58	0.633070	1.330900	0.913060	0.784550	0.260280	-0.526040	-0.170820	0.460690	-0.539320	0.633070
59	0.310640	2.311800	0.124480	0.145990	-0.176010	0.508370	-1.034000	0.800540	-0.011350	0.310640
60	-0.009600	-0.597470	0.229720	-0.374360	-0.779330	0.825290	-0.189890	-0.279950	0.618860	-0.009600
61	-3.056100	-1.598500	-0.748670	-1.313200	-1.071900	-2.595400	-1.834300	-0.468190	0.052570	-3.056100
62	0.090150	-0.526000	-0.743300	-0.779860	-0.616590	-0.180810	0.268220	0.245680	-0.711980	0.090150
63	0.620960	-0.092690	0.107390	0.008720	0.192860	-0.726130	0.715600	-0.597110	0.494000	0.620960
64	0.786940	0.696530	0.155350	-0.335180	-0.160050	-0.705130	-1.022600	-0.710830	-1.009800	0.786940
65	-0.501280	-0.678760	0.535000	0.677280	-0.746170	-0.636790	0.229010	0.118850	0.030700	-0.501280
66	-0.270710	-0.230070	0.030330	-0.649110	0.733550	0.112080	-0.778990	-1.100800	0.926110	-0.270710
67	0.686290	0.117290	-0.176720	0.161590	-0.746410	0.073890	-0.370060	-0.591400	-0.002530	0.686290
68	1.488900	0.796130	1.239900	1.860800	0.090600	1.013000	-0.305470	-0.371760	0.410780	1.488900
69	1.019900	0.879470	-0.244160	0.528640	-0.591640	-0.601840	0.136780	0.770950	0.900050	1.019900
70	0.325190	1.290800	0.158850	1.185300	-2.476100	0.648000	0.380560	-2.127000	1.485100	0.325190
71	-0.387860	0.365820	0.111830	1.011800	1.103000	-0.426080	-0.796570	-0.196570	-1.246300	-0.387860
72	-0.617960	-0.179040	-0.647490	0.553780	-0.577620	0.468470	-0.227790	-0.376410	0.083050	-0.617960
73	-2.077100	0.746010	1.382200	-1.160200	-1.152600	-0.933700	1.980000	0.720710	-0.619260	-2.077100
74	-3.226800	-1.972700	1.013300	-1.045900	-0.161090	1.190500	-0.730330	-0.972480	-1.015700	-3.226800
75	0.398660	1.872900	0.783630	0.668690	-0.635760	1.047800	0.248590	-0.225930	-0.310970	0.398660
76	-1.732300	3.111000	-1.792700	0.364070	-1.291900	-1.097500	-1.004200	-0.084890	0.272850	-1.732300
77	0.929420	0.931460	-1.169300	-1.058100	-0.033550	1.732500	-2.419000	0.108060	0.703930	0.929420
78	-0.728120	-1.946400	0.191290	1.026400	-0.798330	1.554200	-1.322700	-1.772100	-0.371750	-0.728120
79	-1.128300	1.166700	0.558240	-1.184300	0.411710	1.187800	1.030500	0.868700	1.842100	-1.128300
80	2.023600	-0.384640	0.660220	1.548700	-0.197460	1.162900	1.009500	1.129200	-1.804800	2.023600
81	-0.281240	0.271980	-0.611660	0.526570	1.028100	-0.884900	-0.466700	1.268400	-1.145800	-0.281240
82	-0.362460	-0.084110	0.279980	0.205690	0.569010	0.504630	0.986910	-0.158410	-0.173720	-0.362460
83	-0.740490	0.973690	-0.563400	0.769700	0.916640	1.237800	0.309690	1.532100	0.098110	-0.740490
84	-0.089280	-0.882610	1.051300	-0.171300	-0.093900	-0.445570	0.890600	-0.764400	0.838830	-0.089280
85	-0.177310	0.016310	0.300810	-0.238050	-0.862990	0.198060	-0.204470	-0.860110	0.017440	-0.177310
86	0.080070	-0.633860	-0.415170	-0.459050	0.777730	0.155260	-0.184190	-0.145350	-0.601340	0.080070
87	-0.882920	2.105200	-0.525770	-1.469900	-0.796130	-0.566540	0.426140	-0.377740	0.564530	-0.882920
88	-0.879740	-0.767410	-0.519320	-0.793770	-0.270170	1.057100	-0.649930	-0.191570	0.595500	-0.879740
89	0.900160	1.618100	0.982570	0.615580	-1.109500	0.804510	-0.255980	0.636700	-1.354000	0.900160
90	0.373750	-1.301800	-0.369940	-0.405080	-0.039760	0.244850	-0.745670	-0.039970	-0.438190	0.373750
91	1.553700	0.132590	-0.414510	0.153740	0.228660	-0.957050	0.539360	-1.013800	0.486560	1.553700
92	-0.559530	-0.038790	0.767590	1.256500	0.791080	0.549980	-0.591890	0.371220	-0.307580	-0.559530
93	-0.503670	0.310960	1.137600	-0.216610	-0.227960	-0.065100	-1.107900	0.167340	-0.889150	-0.503670
94	0.731500	-0.213880	-0.397140	0.817000	0.595860	-0.348210	-0.184170	-0.862370	0.388160	0.731500
95	-2.958600	-0.643800	-1.730300	-0.056900	-0.196150	0.677970	0.041120	0.459030	-0.135050	-2.958600
96	1.618900	-2.035800	0.815860	0.786530	0.452470	-0.747470	-0.134810	-0.502540	1.477400	1.618900
97	1.369000	1.084700	-0.747050	-0.483540	-0.651780	-0.993960	0.150570	0.416990	1.234300	1.369000
98	-0.240550	0.588850	1.773000	1.024800	0.596920	-1.086800	-0.705100	1.033400	0.497840	-0.240550
99	0.558250	0.953190	0.774460	0.217290	0.338440	-0.351950	-0.294800	0.876620	0.256580	0.558250

	1	2	3	4	5	6	7	8	9	10
100	-0.870970	1.059400	0.797030	0.134970	0.399040	-0.505600	-0.522360	0.288900	0.431630	-0.870970
101	-2.109500	-1.946400	-0.687770	-1.241700	-0.394040	-0.175970	-0.307710	0.255350	0.712130	-2.109500
102	0.645900	-0.615160	0.141420	0.135340	0.600600	0.082800	-0.342500	0.463690	-0.700620	0.645900
103	0.403000	-2.333100	-2.939900	-4.453300	1.345600	-1.632100	1.792600	-1.960700	-1.150400	0.403000
104	2.205800	0.373720	0.697600	2.083700	-0.154080	-0.335060	-0.514500	-1.245500	-0.148430	2.205800
105	-0.703750	0.902720	-0.959590	-0.123850	0.944100	-1.136000	1.548000	1.901100	0.555110	-0.703750
106	0.698470	-0.544290	-0.075300	1.172100	0.431340	-0.301750	0.670610	0.118060	-0.612430	0.698470
107	-1.373000	-1.233400	-0.797760	-0.605270	-0.130040	1.535400	0.742600	0.038830	-1.055800	-1.373000
108	-0.733720	-0.739260	0.599720	-0.938400	0.217200	0.704190	0.564100	0.628640	0.038600	-0.733720
109	-0.547050	-0.258630	-0.270870	0.001370	-0.937670	0.243960	-0.516960	-0.448380	-0.868470	-0.547050
110	-0.943780	-0.315770	-1.811700	1.535900	-1.811500	0.487800	-0.466340	-0.382750	0.402040	-0.943780
111	-0.085610	0.428790	-1.261000	-1.285100	-0.122600	-0.338440	-0.385980	-0.751380	0.879600	-0.085610
112	0.385850	-0.734660	0.380970	0.875450	0.429430	-0.069510	0.586930	-0.121310	-0.135340	0.385850
113	-0.054020	-0.444400	0.296250	-0.876520	-1.320300	1.033700	0.753150	-0.935430	0.418270	-0.054020
114	-0.058110	0.742210	-1.900900	-0.318930	0.405890	1.499900	-0.209160	-0.520740	-0.331530	-0.058110
115	0.368750	1.987400	0.060040	-0.461900	-0.320160	1.073300	-0.072860	0.873090	0.694810	0.368750
116	-0.397910	-1.945800	-2.065800	0.360520	1.470500	-0.275070	0.545970	0.088410	-1.356800	-0.397910
117	2.800800	3.920300	2.563100	-0.765110	-0.252100	-0.197380	0.426780	-0.588830	-0.418710	2.800800
118	-0.183090	0.738210	-0.098660	0.898010	-0.810000	-0.080060	0.382390	-0.819350	-0.554050	-0.183090
119	3.436400	-0.460940	1.338000	-1.880000	1.850500	-1.249100	0.253770	-0.587230	1.679000	3.436400
120	-0.953310	-2.097200	-0.783290	-1.210900	1.573000	-0.542170	1.710100	0.506830	0.436280	-0.953310

Tabela D.2. Valores dos pesos das conexões entre as 10 últimas entradas da camada de entrada e a primeira camada oculta da rede *Magola11*. Matriz de pesos IW.

	11	12	13	14	15	16	17	18	19	20
1	0.137850	0.066420	-0.137270	-0.869150	-0.435030	-0.873190	-0.529040	-0.713420	-0.497650	-0.047230
2	2.834600	0.387590	1.325500	0.464400	-0.194680	0.127550	-0.434700	0.355720	-0.284340	-1.785100
3	-1.916300	-1.161700	-0.259620	1.472000	-0.474050	-2.709900	-0.346580	1.045000	0.406300	-4.426800
4	-0.910870	-0.535290	1.098800	-0.341460	0.121880	0.368060	-1.033700	-0.524810	0.444560	-0.336600
5	-0.002080	0.037700	0.651580	0.318880	0.463840	-0.404990	-0.250400	-0.284000	0.236810	-0.994250
6	-0.857490	-1.156900	0.912720	-1.698700	-0.455410	0.631760	-1.521800	-0.924630	0.231560	-3.478100
7	0.522860	-0.573600	0.326210	-0.876850	-0.926980	-0.886490	0.747930	-0.995510	0.527600	-0.958800
8	0.202440	-0.228690	-0.456660	1.017700	0.143690	-0.483400	0.600940	-0.050930	-0.474420	0.860710
9	0.223650	-0.549560	-0.124060	-0.895880	0.876090	-0.582540	-0.179220	0.423850	0.234880	1.497300
10	-1.016700	-1.657600	-0.576950	0.003110	1.137600	0.023430	0.687240	0.440760	0.200920	0.302630
11	0.129320	0.304420	0.504860	0.648690	0.909120	0.190260	0.438020	-0.320970	1.081600	-0.194550
12	-1.394200	1.057900	0.436550	0.247360	0.682200	-1.180100	0.460380	0.550590	-1.178000	0.886350
13	-0.136340	-0.279530	-0.644430	-2.400200	-0.167600	-0.811470	-1.144600	-0.104680	1.013800	-0.030460
14	0.150020	-1.026400	0.488500	0.708510	-0.238510	0.450260	0.402360	0.082420	0.385800	0.028170
15	-0.517650	0.205490	0.353010	-1.142400	-1.429300	0.035840	0.359440	0.374180	-1.849900	-0.248070
16	0.111840	1.170800	0.951010	-1.385800	-1.466800	-0.076330	-0.906230	0.201620	-0.608940	-0.158470
17	0.521190	1.338200	-2.376100	1.475800	2.334300	0.320960	-0.097910	-0.817650	-2.458900	-1.179100
18	-0.795680	-0.343470	-1.687200	0.851910	0.201190	-0.056450	-1.972000	-0.902080	1.833800	1.543200
19	-0.861220	0.482020	0.755090	-1.324400	1.365900	0.466150	0.340510	-0.207230	0.512050	0.101670
20	-0.407860	-0.076490	0.491790	-0.102800	-0.469320	0.406920	-0.684400	-0.225880	-0.003880	0.758640
21	-0.545410	-1.060900	0.209860	0.462510	-0.049940	0.507240	-1.001900	0.480060	1.617200	2.596000
22	-1.009200	-0.267130	0.215320	-1.084100	-0.989150	0.292310	0.415150	-0.542490	0.022060	0.898010
23	-0.915970	0.245010	-1.221100	0.489660	1.082900	0.062060	-1.931700	-0.483720	-0.723630	1.035700
24	0.498120	0.259690	-0.810760	0.944120	0.837350	0.037540	0.543600	-0.697550	-0.096480	0.332590
25	0.191970	0.055790	0.558820	-1.257000	-0.815440	-0.083410	-1.204700	-1.339300	-0.969920	-0.292350
26	-1.161200	0.097680	-1.593900	-1.345400	-0.702440	-1.111000	1.030800	0.233110	0.295640	-1.494100
27	-0.253130	1.069800	0.284860	0.048520	1.134000	0.932730	0.536990	0.827500	-0.289320	0.098870
28	0.149980	0.919350	-0.124540	1.235600	-0.435540	-1.401400	0.880160	0.404150	-0.896590	0.468810
29	0.776500	-0.189910	-0.443940	0.396310	0.050860	-0.823980	0.049840	0.526770	-0.036040	-0.430540
30	0.740270	0.747470	-0.152700	0.659930	-0.441230	-0.128170	1.588300	-0.753000	-0.572470	2.514900
31	-0.324080	0.174670	1.177100	0.871400	2.177300	-0.757420	0.056760	0.255970	-3.017700	-4.637900
32	1.375800	0.831550	0.604420	0.794630	-0.193640	0.643970	0.972680	-0.474000	0.626440	0.489250
33	0.660240	-1.153300	-0.909420	0.700260	0.630640	0.110400	0.388690	0.301910	-0.154020	0.519230
34	-0.478770	-0.680220	-0.251830	0.672510	-0.612510	-0.168880	0.342090	-0.464930	-0.716240	1.670900

	11	12	13	14	15	16	17	18	19	20
35	-0.047370	-0.207690	-0.719020	-0.139590	2.655700	-2.683300	0.711940	-1.036200	-0.286650	-0.145330
36	-0.431940	0.046330	0.005990	-0.219430	0.977630	0.353470	-0.501710	0.305570	1.019300	-1.025200
37	1.000300	0.293180	0.032240	1.489600	1.950200	0.070440	0.855120	-0.372110	-0.370890	-1.492800
38	0.100440	0.330300	0.074800	0.134830	0.914740	2.686700	0.082730	0.130440	2.325700	-0.357550
39	-0.971270	-0.392480	-1.474000	0.317290	-0.399750	0.021700	0.049670	-0.421310	0.774730	-0.328620
40	-0.214410	-0.746920	-1.124600	0.313030	-0.550690	-1.083600	1.309200	0.736940	0.189910	-0.593730
41	0.035380	-1.052600	0.193350	-1.107200	-1.334500	-0.544490	-0.134250	-0.118130	1.173700	0.603470
42	-0.376170	-0.170400	-0.999740	1.188100	-0.541200	-1.299500	0.704400	-0.140190	-0.831200	-0.251260
43	-6.881200	1.127300	2.528300	-2.253900	1.947700	0.791980	0.162510	2.080900	2.490500	-0.351510
44	0.904620	0.862750	0.090260	-0.850460	1.143700	0.704530	0.821720	-0.123630	0.291540	-0.456210
45	0.120980	-0.277030	-0.826970	0.827530	-0.390250	1.111200	1.032200	0.998380	1.211100	0.525100
46	0.805890	0.659470	0.257300	-0.773480	0.000580	-0.839970	-1.076200	-1.484400	-0.653610	-0.227100
47	1.664800	0.040120	0.713190	-0.359390	-1.099400	-0.426570	-0.774900	-0.272520	-0.062160	-0.003990
48	0.173050	1.053400	0.893950	0.775540	1.318000	-0.104560	-0.524400	0.126300	0.719450	-0.444310
49	1.165500	-0.098940	0.968240	1.139200	-0.220020	-0.615590	0.444090	1.105400	0.845280	-0.356080
50	-0.277250	0.258650	1.537800	0.399740	-2.571200	-0.279670	-1.155600	-1.698700	-1.983600	2.535000
51	0.139090	-0.852750	0.269130	-1.766600	0.912320	0.287860	1.495300	-0.530180	-0.065080	-0.165040
52	-0.724940	0.141110	0.218690	0.461430	0.052780	-0.592540	0.771780	0.607020	0.030460	0.976020
53	-0.123840	-1.235000	-0.681190	-0.665320	1.085200	-0.824160	-0.208750	0.707080	-0.783540	0.806090
54	-0.305860	0.774770	-0.731770	-1.218100	-0.868750	-1.309400	0.890590	0.846500	-0.947490	0.130420
55	0.140660	-1.782700	-2.096500	-1.538200	-1.417900	-3.128900	0.687500	-1.755900	-0.294300	-1.714400
56	-0.951950	0.016630	-0.535960	-0.664160	-0.467170	0.059530	-1.150000	-0.313460	-1.188100	0.324950
57	0.009880	-0.235370	-0.392240	-1.121200	0.413840	-0.833040	0.932250	0.761790	-1.121500	-1.583000
58	-0.297760	-0.659590	-0.288580	-1.261800	-0.760120	-1.241500	-0.103660	-0.827820	-0.918530	-0.377590
59	0.999160	1.516200	1.123000	0.539880	0.594230	0.934870	0.161870	-0.067370	0.879640	0.382400
60	0.637920	1.108400	0.237940	-1.339100	-0.607420	-0.181900	0.555710	0.518900	-1.126700	-0.369720
61	0.019140	-0.066660	1.195700	0.468600	-0.530240	2.077300	-0.286100	1.361800	1.112300	2.674400
62	-0.322940	0.002990	0.851080	-0.831430	-0.381500	0.205610	-0.860280	0.506620	-0.110430	-0.404390
63	-0.670900	-0.406660	-0.044690	0.570010	0.702570	0.234770	1.362300	1.085100	0.045900	1.263800
64	-0.264760	-0.645290	0.533840	-0.601650	-1.213300	-0.730430	-1.070100	0.470900	-1.012700	0.331500
65	0.726860	-0.773090	-0.700690	0.602750	0.003560	0.973070	0.534090	0.897030	1.143000	0.950830
66	-0.376010	0.595100	0.679710	-0.802460	0.511470	-0.893190	-0.515570	-0.502060	0.573400	0.035500
67	0.066890	0.208110	-0.400920	1.025500	-0.261980	1.177000	0.891520	0.358240	1.334400	0.675150
68	-3.522000	-2.405200	0.326340	-2.881400	0.811170	-0.580700	0.767370	0.369100	0.873600	-4.466400
69	0.523280	-0.389310	-0.974490	-0.647000	-0.726880	-0.949870	-0.697760	-0.585270	-0.000660	0.137270
70	-0.675550	0.628740	1.583500	1.861300	1.173600	1.507600	0.395150	-0.493970	0.274580	-0.320730
71	-0.288130	0.015470	-0.968630	0.958120	-0.654550	0.702210	0.870500	0.701010	-0.544720	0.490590
72	0.593750	-0.325350	-0.856730	-1.179300	-0.433860	-0.152800	-1.248500	-0.673100	-0.788330	0.712450
73	0.273250	0.246780	-0.287240	2.022900	-0.270400	-2.161600	0.882480	-0.882020	-0.147090	-0.138650
74	-0.083790	-1.755400	-0.372460	1.220000	-0.437440	0.169530	0.799500	-0.869910	-0.021390	1.318800
75	0.518810	1.539600	1.502100	1.774200	1.427100	0.266980	-0.500200	-0.562970	0.451100	0.387260
76	0.404390	1.778800	-1.024700	0.463350	0.584450	1.139300	-1.392800	-1.976300	0.953100	-2.834300
77	0.368270	-0.698240	-0.274860	2.529500	1.342600	0.431850	1.042000	1.899400	-0.038980	0.476510
78	-1.469500	-0.218030	-0.152100	-0.479500	-0.651430	1.412600	-0.246620	-0.399160	1.381400	-0.593030
79	-1.550000	-0.969000	-0.568980	-0.506490	0.979890	-0.572260	0.777240	1.616300	-1.817400	-2.202700
80	0.441180	0.288230	-1.509000	-2.224900	-1.260200	-0.690020	0.340270	1.926600	-0.597060	1.563300
81	0.557920	-0.200460	-0.714090	0.071110	-0.574060	1.374900	0.423280	-0.154410	0.630930	0.099560
82	0.167550	-0.821370	0.687590	-0.536810	-0.878640	0.582930	-1.160900	-1.071900	-1.410200	0.087480
83	-0.538150	-1.488900	-1.096100	-1.768900	-1.536100	-1.297700	1.713700	0.811210	-0.135950	1.770600
84	-0.187670	0.353570	0.382100	-1.110100	0.361600	0.281950	-0.286540	0.128180	-0.890880	-0.481370
85	-0.113580	0.739810	-0.218560	-0.515260	-0.339290	-0.396830	-1.280100	-1.656400	-0.200840	-0.711100
86	0.005710	0.343720	-0.193650	-0.744520	0.577820	0.664800	-0.891980	0.187950	-0.848920	0.097260
87	-1.306400	-1.355200	2.868300	0.572550	1.090400	2.030300	1.296500	0.556170	1.374800	0.381600
88	-0.127830	0.146560	0.446730	0.350280	-0.440540	0.438160	-1.022000	0.128250	-0.866830	-0.271780
89	0.437780	0.422980	-0.001760	-1.052700	0.276010	-0.581310	1.849000	0.367090	0.671930	-1.409600
90	-1.016600	1.024500	0.990780	0.092610	-0.619570	0.773130	0.357100	0.154550	-0.248810	-0.046560
91	1.202300	1.314200	0.321480	0.029510	-0.784420	-0.858240	1.929800	0.464020	0.243480	0.556450
92	-1.277700	-0.760100	0.221940	-1.001500	-0.121460	0.768200	0.505290	0.739710	-0.361140	-0.030240
93	-0.493040	0.713880	-0.125100	-1.175100	-0.764400	0.190430	-1.617400	0.296970	-1.511000	0.684930
94	0.770170	-0.866370	-0.106220	-0.482820	0.134850	0.280480	-0.220140	0.879480	-0.405910	0.035600
95	-0.599920	-0.612790	0.154310	0.297300	1.260800	0.372050	0.121900	-0.952060	0.319630	0.346460
96	-0.243700	1.646400	-0.992310	1.019000	-0.246130	-0.298730	0.253610	2.103400	-1.834600	-1.950200
97	-0.504770	0.287410	0.466870	1.550400	0.631300	1.176000	0.280450	0.492740	-0.806800	0.190630

	11	12	13	14	15	16	17	18	19	20
98	0.438690	0.203680	-0.434620	0.323740	0.339950	1.394000	-0.506950	-1.220600	-0.296560	-0.794060
99	0.311880	-0.227210	-0.794610	1.347900	-0.210030	-0.447740	1.253700	-0.215920	-0.186150	0.658710
100	0.168540	0.680660	-0.769230	-0.360710	-0.083360	-0.755070	-0.643810	-0.979280	-1.047200	0.783880
101	-0.095220	-0.634690	0.022100	1.476700	1.094000	1.085800	-0.809460	0.726090	0.231650	1.841800
102	1.271200	-0.664880	0.678350	1.113000	0.081140	0.579820	0.452540	-0.732370	-0.385000	0.742540
103	-1.541900	2.121500	3.132000	-0.177870	0.747030	1.836000	1.479500	2.609400	1.942700	-1.161300
104	0.456840	-0.379810	-0.825220	1.340100	0.093060	0.260210	-2.368700	-1.257500	0.654630	0.227990
105	0.536320	-0.051260	-2.318000	0.359360	0.473250	-1.680800	0.349310	0.605720	-0.955000	1.785600
106	0.637520	1.182000	0.001990	-0.685010	0.300100	-1.061700	-0.070730	-1.296400	-1.452200	1.094500
107	-0.487290	1.300000	0.132030	-0.635060	1.480100	1.779100	1.237200	0.617780	1.340200	0.147130
108	0.084750	-0.265470	-0.212590	-0.973620	-0.881390	0.485220	-1.014700	0.271890	-0.624880	0.991620
109	0.655300	0.081680	0.848140	-0.545530	-0.135590	-1.218100	0.592490	-0.838220	-1.330900	0.279990
110	2.213600	1.048300	-0.672710	-0.541500	-0.121450	-0.126770	1.514000	0.060960	1.292100	-1.012700
111	-0.032870	-0.462260	0.049570	0.185770	-0.272740	-0.340200	-1.582200	-0.697790	-0.476170	-0.988140
112	-0.512180	1.329700	0.986550	-1.507900	-0.892740	1.056600	-0.633630	-0.205900	0.661250	0.966970
113	0.246940	0.672310	-0.792890	-0.811450	0.527750	-0.507100	0.118260	0.157840	-0.425360	0.404190
114	-0.399850	1.230600	0.380990	-0.423450	1.805000	0.219230	1.189700	1.055500	1.150500	-0.803060
115	0.895150	0.352470	0.486890	0.856520	0.108950	0.021300	1.738400	0.661140	-1.519600	-0.970870
116	1.563800	-2.093600	-0.984160	1.145700	0.598720	0.208470	-0.514270	-1.695300	-0.292200	-0.247190
117	-0.302040	0.694360	-0.791510	0.431940	0.467740	-3.273700	0.881620	0.434250	-1.734800	-3.631000
118	0.516020	-1.043000	-0.666610	0.283000	0.202100	-0.189440	0.913280	0.225900	0.365560	-0.244270
119	0.610900	0.428330	0.253310	-0.867140	1.315900	-0.474780	0.381050	-0.209700	-1.260900	-1.828300
120	-0.294840	-0.467630	-0.907170	-3.536700	-0.410140	0.894080	-0.565170	1.178200	-0.203970	0.906770

Tabela D.3. Valores dos pesos das conexões entre a primeira camada oculta (120 neurônios) e a segunda camada oculta (100 neurônios) da rede *Magola11*. Coluna 1 a 15. Matriz de Pesos LW1.

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
1	-0.0828	0.9263	0.2675	-1.2896	0.5818	0.0878	0.3093	-0.6394	-1.1057	-0.3172	0.5054	0.3270	-1.0090	-0.5766	0.4848
2	-0.7964	-0.9614	0.7321	0.3242	-0.2256	-0.0487	-0.1381	0.2185	-0.2753	0.4423	-0.5447	0.0271	-0.2042	-0.3491	0.4207
3	0.3987	0.7154	0.2560	-0.3616	0.2796	-0.7341	0.2011	-0.6096	0.2411	-0.7406	0.5246	-0.0453	0.1128	-0.3911	0.0598
4	-0.5735	1.1205	1.4208	-1.8003	-1.5795	0.2741	-0.2973	0.7542	-0.1853	0.5105	-0.5655	-0.7996	0.0607	-0.4137	-0.0877
5	-0.2070	-0.7748	0.6037	0.3482	-0.3801	0.3895	-0.3613	0.6056	0.0350	0.1457	0.9346	-0.3347	0.5907	0.4687	-0.2783
6	0.6952	0.8892	0.3775	0.5301	-1.2789	-0.3348	-0.6072	1.0065	0.3676	0.9689	-0.5979	-1.1942	-0.4407	0.8352	0.3331
7	-0.0673	-0.5888	-0.0937	-0.6851	0.2845	-0.0930	-0.7015	-0.5903	0.3082	0.0468	0.1043	-1.0282	-0.4891	-0.3897	0.5944
8	-0.3463	-0.4373	-0.0750	-0.3287	0.0325	0.0279	-0.8501	0.0736	-0.8408	1.0174	0.4664	-0.1357	-0.7345	0.8602	-0.8704
9	-1.0017	0.4355	0.7256	-0.2826	-0.9186	-0.8277	0.7604	-0.2628	-0.1227	-0.5713	0.6594	0.0300	-0.5665	0.6598	-0.2941
10	0.8477	0.2043	-0.1432	-0.0880	-0.7547	0.8401	0.2716	-0.5136	0.0718	-0.0019	0.0789	1.1939	0.5054	-0.7549	-0.7437
11	0.0782	0.2084	-1.1058	1.4160	0.4765	-0.3781	0.4870	-1.0062	1.3676	0.1082	-0.4000	0.8092	-0.0570	-0.4077	-0.0132
12	0.2938	-0.5650	-0.7503	0.8577	-0.3885	-0.7881	0.8309	-0.3336	-0.1200	0.0180	-0.2972	-0.3841	-0.1891	0.6462	0.7244
13	1.0165	-0.1399	-0.7336	-0.6039	-0.2394	0.7274	0.1662	-0.8646	-0.5345	-0.8530	0.1942	0.3545	0.1285	-0.3253	-0.5078
14	-0.1762	-0.8748	0.3561	-1.1389	1.1716	1.2055	-0.2433	-0.5967	0.5067	0.0733	0.8942	0.4892	0.1235	-0.4518	-1.1272
15	-0.9293	1.1694	0.5663	1.4594	-0.3270	-0.0788	-0.2035	-0.2414	-0.1355	-1.0405	0.4084	-0.8378	0.0853	0.5026	0.1114
16	0.9016	-0.5317	-0.5283	-0.1876	-0.7310	0.3961	-0.6187	0.1457	-0.0348	-0.7250	0.4031	0.3137	0.8352	0.2350	0.9374
17	-0.3384	0.1781	-0.8863	0.6322	0.5728	-0.5898	0.2680	0.4422	0.5979	-0.0931	0.3391	1.0015	0.0747	0.3656	-0.6504
18	0.0332	-0.5089	-0.6745	0.5427	1.2924	-0.0725	0.7859	0.0982	-0.4577	1.2041	0.8522	-0.3558	0.4690	-0.1458	0.0528
19	0.6554	-0.1093	-0.6819	-0.3569	-2.4433	-0.5690	0.0146	-0.8778	-1.3889	-1.2263	-0.6540	0.7529	1.7211	0.0170	0.0970
20	-0.9634	-0.6555	0.6456	-0.1764	0.4083	0.1252	-0.4563	-0.1887	-0.7646	-0.1437	-0.4293	0.2582	0.1604	-0.5969	0.1014
21	0.8292	-0.9742	0.5304	-0.0981	-0.0651	-0.0015	0.6478	0.9203	-0.3110	-0.6867	-0.6977	-0.7345	0.9862	-0.8197	0.0713
22	-1.0127	-0.2181	0.9533	-0.0994	0.1742	-0.3299	-0.7909	0.0570	0.0146	0.8823	0.4937	-0.9903	-0.5960	0.5663	0.2831

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
23	-0.8546	0.4212	1.1426	0.0590	-0.5880	-0.0603	-0.6079	0.6717	0.5083	-0.2464	0.7297	0.1066	-0.3808	-0.8440	-0.7549
24	0.5607	-0.3747	0.7730	0.3902	-0.2288	-0.1947	-0.6288	-0.7652	0.9459	0.5059	-0.7939	-0.0776	-0.0577	0.4018	-1.0044
25	0.7285	0.3501	-0.3510	0.3179	-0.0089	0.1525	0.7450	-0.0463	-0.1038	-0.4211	-0.1223	0.9003	-0.4016	-0.7777	0.3797
26	1.1630	-0.4129	1.3844	0.9979	-0.9438	-0.3904	0.6556	0.7328	-0.8638	0.9648	0.2739	0.3221	0.3112	-0.7099	-1.1172
27	0.7815	1.1406	-0.1889	-0.6416	1.0392	0.2377	-0.2873	1.0973	0.2022	-0.8679	-0.8937	0.7348	-0.7631	-0.2186	0.6723
28	0.1456	-0.8778	3.2490	0.3720	0.0576	0.1389	-0.6827	0.1070	-0.4182	0.8843	0.2714	-0.9847	0.2056	0.4691	-1.1523
29	-0.3211	0.0457	-0.6648	0.9904	-0.4282	0.8686	0.2906	-0.3650	-0.4945	1.2354	0.2173	-0.8773	-1.0253	-0.2602	-0.7294
30	0.5417	1.1743	-0.1747	-0.7167	0.1480	-0.8420	0.8797	-0.4956	0.5716	-0.2283	0.0171	-0.3345	-1.2410	0.3189	-0.9607
31	0.3523	0.4361	0.0504	-0.1053	-0.3746	0.7516	0.3298	-0.3971	-0.3046	-0.5612	-1.7737	1.0861	0.0742	-0.1384	-0.5638
32	0.0325	0.0508	0.0936	0.2838	-0.2323	0.5981	-0.6603	0.9178	-0.3486	0.2597	1.1333	0.7998	-0.8324	-0.0447	-0.2384
33	-0.9846	0.7847	-0.4718	-0.2834	-0.3387	0.0274	0.3673	-0.6672	0.1562	0.8798	-0.9474	-1.1015	-0.1270	0.5962	0.1216
34	-0.2877	-0.2914	0.3326	-0.2522	0.3106	-0.4529	-1.0810	-0.3988	-0.7299	-0.1107	-0.8650	-0.8729	1.3421	0.8552	-1.7835
35	0.3404	0.2859	-0.7515	-0.5860	-0.6235	0.9408	0.6711	-1.0638	-0.2658	0.4288	-0.6919	-0.8731	-0.4964	-0.5857	0.6380
36	-0.1802	1.0065	0.1997	-0.6458	-0.3307	-0.6699	-0.7182	1.1773	0.6848	0.6621	-0.2842	0.2384	-0.4506	-0.9772	0.6771
37	-0.9045	0.6050	-0.3788	-0.6626	0.5329	0.7227	-0.7442	-0.6932	0.2890	-0.0341	0.5257	0.7541	0.4309	0.3513	0.3997
38	0.4016	0.5589	0.1416	-0.6177	-0.7138	0.4279	1.0173	0.7763	-0.7765	-0.1321	-0.8070	-0.7970	-0.2024	0.5853	0.4902
39	-0.1303	1.5513	-0.5566	-0.0938	0.5095	0.1575	0.0134	-0.6881	-0.5960	-0.7821	0.3702	0.3157	-0.2269	0.8251	0.0811
40	0.5791	0.8231	0.5438	0.2632	-0.4979	-0.9161	-0.7647	-0.7490	-0.5929	0.0651	-0.8767	-0.2293	0.8325	-0.2078	0.5326
41	0.4017	-0.0198	-0.7941	0.6978	0.4815	0.0454	-0.6667	-0.7006	-0.4822	-0.3890	0.4471	0.8707	-0.9972	0.8131	-0.4809
42	0.8417	0.0952	0.5891	0.4420	-0.5125	0.7624	0.0563	0.4394	0.3193	-0.0614	-0.3872	0.6756	-0.1728	-0.4199	-0.7518
43	-0.1501	-0.9450	1.9789	-2.2787	0.0491	-2.7589	0.4852	0.1024	0.1086	-0.7877	-0.3084	-0.0315	0.8526	-0.2531	-1.7332
44	-0.4958	-0.8420	0.0110	0.7718	0.1621	-0.0423	-0.5680	0.2173	0.3217	0.2972	0.2849	0.0571	1.1451	0.8661	-0.6087
45	-0.1060	-0.2148	0.0727	-0.0065	0.4541	0.1583	-1.1196	0.9201	-1.0841	0.7670	-0.8475	0.0840	0.4855	0.0214	0.2793
46	0.9906	-0.1618	-0.6881	0.4736	0.0801	-0.3586	0.7832	0.1502	0.7324	-0.5750	0.6645	-0.2084	-0.5054	0.0290	0.2968
47	0.7976	0.5088	0.2227	0.6025	0.1140	-0.7404	-0.6648	1.0904	-0.3978	-0.1078	1.1130	-0.9644	-0.0940	-0.7357	-0.8774
48	0.1471	0.5697	0.9649	0.7534	-0.0604	0.6932	-0.3046	-0.3854	-0.4470	0.3504	-0.4722	0.1934	-0.7770	-0.0162	-0.7162
49	-0.8178	0.7546	0.1337	0.1517	-0.1157	0.0732	0.3434	0.5551	-0.8347	-0.2898	-0.7752	0.1143	0.1020	0.3318	0.8497
50	0.6004	-0.1630	-0.7976	-0.0137	0.8131	0.2306	1.0619	-0.7199	0.3768	0.5764	-0.7639	-0.1565	-0.7410	0.5121	0.6144
51	-0.1319	0.9350	0.8854	-1.0194	-0.6342	-0.6394	0.2789	0.3320	-0.2410	-0.5032	0.0958	-0.7631	0.1000	0.3764	0.4851
52	-0.6168	0.1095	-0.5070	-0.4830	-0.4557	-0.6704	-0.7798	0.3130	0.4275	-0.0833	0.8432	-0.4225	-0.3983	-0.2701	-1.0933
53	-0.4400	-0.5243	-0.4183	1.0675	0.3067	0.9312	0.2021	0.5792	-0.2012	-0.4754	-0.5303	0.0957	0.8659	0.8892	-0.8753
54	-0.7963	-0.6780	-0.5982	-0.3885	0.7634	-1.1208	0.3020	-0.2700	0.6999	-0.1054	0.0014	0.3981	-0.7793	0.1342	0.2407
55	-0.7890	-0.7929	0.2453	0.5401	0.5985	-0.6112	-0.2857	-0.3903	-0.0423	-0.4145	-0.1926	0.5402	-0.8124	0.5513	-0.5304
56	0.3956	-0.4422	-0.2928	0.0497	-0.6904	-1.3309	0.0305	-0.4657	-0.2669	-0.0572	0.2324	0.4113	-0.2512	-0.1749	0.5913
57	0.2575	1.1335	0.6857	-0.9354	-0.7503	-0.5209	-0.1533	0.8266	-0.8272	-0.3834	0.6279	-0.1712	0.0450	0.0746	0.6021
58	0.0247	-1.0745	-0.4971	0.1802	-0.3475	0.5592	0.4015	0.3731	0.9398	1.0745	0.1925	0.2627	0.3066	0.1342	0.2136
59	-0.0601	1.8129	-0.7926	0.7530	0.4056	-0.0572	-0.2708	0.4567	1.1182	-0.2946	-0.2326	-0.5237	-0.3714	0.3056	1.2117
60	-0.8674	-0.5765	0.1789	0.1723	-0.3651	-0.4019	0.2246	-0.6741	-0.4525	-0.7666	-0.6579	0.0431	0.3626	-0.6263	-0.3934
61	-0.0002	-0.0822	0.7481	0.2358	0.7385	-0.0037	0.8060	0.3463	0.2124	-1.1136	-0.8974	0.3071	-0.1870	-0.6049	0.9807
62	-0.7403	-0.3751	0.1381	-0.8991	0.5898	-0.6091	0.9103	-1.0709	0.1635	0.5410	-0.6981	1.0176	0.8139	-0.5409	1.2222
63	0.5920	0.5435	0.4093	0.0603	0.1761	0.9650	0.1581	-0.9930	1.0172	-0.1614	-0.3579	0.3114	0.8462	0.2505	-0.5892
64	-0.5236	-0.0375	0.3584	-0.5850	0.9065	-0.0896	1.0213	-0.0454	-0.8126	-0.8658	0.3802	1.1522	1.0359	0.2083	-0.2452
65	-0.1134	0.0888	0.2356	1.4049	-0.1492	-0.5958	0.0594	0.4971	-0.4194	-1.5212	-0.3929	0.8890	-0.4574	-0.9513	-1.2687
66	-0.1526	0.2959	0.2592	0.0349	0.3670	0.8946	-0.8797	-0.5673	-0.2789	-0.8258	0.1307	-0.9588	0.9697	0.8690	0.6377
67	-0.7689	-0.1831	-0.2193	0.0843	0.1134	-0.6508	-0.6904	-0.6522	-0.6757	0.5568	-0.1147	0.7947	-0.2469	-0.1517	0.5056
68	0.8214	0.9490	0.3158	-0.7698	0.6567	-0.6815	-0.4078	-0.2813	0.3330	0.6786	-0.6790	-0.7368	0.1794	-0.2167	0.2964

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
69	0.5210	-0.5561	0.8368	-0.3695	0.0313	-0.4126	0.3194	0.5982	0.3711	-0.7511	-0.7576	1.0617	0.0818	-0.5468	0.3584
70	0.0399	-0.2162	-0.1586	-0.0327	0.2111	0.4174	-0.7941	0.7868	-0.5951	0.2525	0.8196	0.0915	0.9795	-1.0313	0.7745
71	0.5788	-0.1466	0.7894	-0.6488	0.0189	0.0086	-0.7438	0.3584	-0.1045	0.3444	0.3038	0.3633	0.6353	0.2245	0.6476
72	0.5311	-0.1045	-0.0071	0.4577	0.9013	-0.1421	1.0791	0.6346	-0.4145	-0.6060	-0.9608	0.8974	0.2412	0.1651	0.7717
73	-0.0562	-0.8752	-2.1928	-2.3522	-0.5607	5.4667	-0.3207	1.0804	-1.2946	0.4239	-0.5239	-0.4263	0.7355	-0.0885	0.5190
74	0.3868	0.9157	-0.9998	-0.0240	-0.2670	0.4187	-0.1098	0.8383	-0.6067	0.8374	-0.7199	1.1656	-0.4579	-0.6472	-0.5192
75	-0.3191	-0.9492	0.6879	-0.8210	-0.7573	-0.8177	-0.6128	0.6865	0.2390	0.5290	-0.9687	0.6523	0.9621	-0.9373	-0.8192
76	-0.7748	-0.0392	0.0393	-0.2871	0.3825	0.6967	0.2045	-0.0966	0.8776	0.5139	0.4822	0.2600	-0.7741	0.0291	0.0858
77	0.6344	0.2326	1.2049	-1.1139	0.0956	0.3035	-0.7974	0.3184	0.6412	-0.1901	-0.2592	-0.3677	-0.4313	-0.4046	0.9925
78	-1.2582	0.4159	-0.3579	-0.3490	0.6221	-0.3066	-0.2045	-0.4739	0.3805	-0.7378	0.5796	-0.0183	0.4404	-0.8648	-0.1731
79	0.9943	-0.4981	-1.2915	1.1950	-0.4959	-1.1243	-0.7596	-0.3805	-0.5209	0.1346	0.0244	-0.7108	1.0467	0.2275	1.1494
80	0.2532	0.2922	-0.6159	-0.5240	-0.6954	-0.4423	0.3223	0.1169	0.1698	-0.4290	0.8064	0.0996	0.3253	0.6933	0.8220
81	0.5507	-0.3709	0.8976	-1.2979	-0.5192	-0.7300	-0.8814	-0.6051	0.6609	0.7094	0.8695	-0.3227	-0.6557	-0.0269	-0.1562
82	0.3954	-0.4058	-0.4867	-0.4384	0.7808	-0.1500	0.4103	-0.7279	0.4638	-0.2347	-0.9640	0.6950	0.4619	-0.2850	-0.2045
83	-0.0047	1.4294	-0.0323	-0.6700	-0.6347	0.3973	-0.3672	0.2500	-1.3741	0.6531	1.0949	0.8471	0.6614	1.2660	0.1160
84	-0.4536	0.2029	-0.4505	-0.1057	0.6993	-0.6887	-0.1946	-1.0437	-0.1598	0.8726	-0.5981	0.3297	0.0845	-0.0586	1.0268
85	-0.9856	-1.6877	0.2628	0.3995	0.5261	-0.7815	-0.2386	-0.9525	-0.5880	0.2492	-0.2582	0.6028	-0.2480	0.1927	1.4036
86	0.8855	-0.2586	-0.6394	-0.3437	-0.4043	-0.1986	0.1244	-0.0811	-0.4955	-0.7077	0.4749	-0.6531	1.0200	-0.0788	0.4023
87	0.7209	0.4976	0.6158	0.5091	0.9506	-0.1622	-0.0554	-0.1638	-0.8236	-0.7848	0.8770	-0.5156	1.1275	0.6050	-0.0676
88	0.7009	-0.6318	-0.0755	0.5159	-0.1527	0.8351	0.6723	-0.2467	-0.9430	0.5700	0.0488	-0.8443	0.2814	0.5731	0.8552
89	1.2155	-1.5105	0.5535	-0.3145	-0.7677	-0.7784	-0.1487	0.6937	-0.5471	-0.3173	-0.5980	0.8220	0.5003	-0.5670	0.2902
90	-0.1989	0.1341	1.0061	-0.3917	0.9559	-0.2198	0.2796	-0.8136	-0.2894	0.4094	-0.0775	-0.2788	-0.8660	-0.9686	0.7217
91	-0.0892	0.2314	-0.9002	-0.4444	0.4356	0.5534	-0.6706	0.4341	-0.7902	-0.4806	0.2135	0.6544	-0.5188	0.0057	-0.7656
92	-0.8176	-0.7426	0.6614	-0.7187	0.6005	0.4879	0.1095	0.8814	-0.2557	-0.2469	-0.5861	-0.3634	0.1180	0.6012	0.0129
93	-0.7530	0.4571	1.9567	-0.1905	-0.2012	0.1921	0.2117	-0.9450	-0.1415	0.9260	-0.4351	-0.1605	0.2718	-0.4877	1.2561
94	0.4174	0.9472	-0.8720	0.4120	0.1595	-0.9053	0.3212	0.7098	0.8106	0.7322	0.9621	-0.0862	0.4483	0.8206	0.5487
95	-0.8799	-0.8629	-0.0586	-0.5910	-0.1649	0.7355	0.0368	-0.0365	-1.0257	0.4541	0.9495	0.8213	-0.2968	0.0225	0.5156
96	0.5382	0.7324	-0.6295	0.2416	0.0412	-0.1338	0.7630	-0.3421	2.2710	-0.3672	-0.5843	0.3693	-0.0265	-0.2553	-0.8903
97	0.1812	-0.8571	-0.1501	0.0279	0.0658	0.4120	-0.7352	0.1588	1.2325	-0.6868	0.5229	-0.4936	0.2374	0.4779	1.8787
98	0.9050	0.1818	-0.3733	-0.1214	0.5484	0.0419	0.4620	0.5640	0.8832	-0.2419	-0.4062	0.0824	-0.6303	1.0848	-0.6147
99	1.0485	-0.7217	-0.7700	0.4152	0.4484	0.3096	1.0581	0.8811	0.6700	-0.6220	0.5350	1.0712	-0.0393	0.0087	0.7761
100	-1.4185	-0.4351	-1.4405	0.7175	0.8384	0.5401	0.2444	0.7549	-0.7695	0.5142	-0.0246	0.3965	-0.3405	0.3097	-0.5298

Tabela D.4. Valores dos pesos das conexões entre a primeira camada oculta (120 neurônios) e a segunda camada oculta (100 neurônios) da rede *Magola11*. Coluna 16 a 30. Matriz de pesos LW1.

	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30
1	-0.3958	-0.3704	0.6371	-0.6974	-0.8284	-0.6153	-0.1461	0.5136	0.6320	0.0964	-0.3754	0.4382	1.5829	-0.7937	0.6647
2	-0.4380	-0.1076	-0.5317	0.3261	0.8849	-0.5187	0.5749	-0.5567	0.8304	-0.7711	0.0986	0.7098	-0.9688	-0.3470	-0.0383
3	0.3771	0.8961	0.4106	-0.2904	-0.1707	-0.1348	-0.8044	-0.0234	0.1095	0.3711	-0.9858	-0.4733	-0.1010	0.8757	-0.2555
4	0.6168	0.1492	-0.4229	-0.5982	0.2485	1.2053	0.0734	-0.5877	-0.2486	0.7596	1.7145	0.7587	-0.8984	0.4213	1.1510
5	-0.0296	1.0912	0.2403	-0.6512	0.4404	-0.3207	0.1905	-0.3576	1.0130	0.7888	-0.0597	0.9562	-0.4658	0.8070	-0.4947
6	-0.6895	-0.3492	0.0705	-0.1511	0.3239	-0.7579	0.5104	0.5820	-0.3539	0.1473	-0.5957	0.5413	0.6143	0.4608	-0.7104

	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30
7	0.3551	-0.6057	-0.3507	-0.9266	-0.3796	-0.7611	0.3603	-0.4075	0.8693	-0.0422	0.3607	-0.4994	0.4518	0.0997	0.9342
8	-0.0640	1.2510	0.0214	-0.5068	-0.1264	0.5919	0.6829	0.3759	-0.5566	-0.1228	-0.7417	-0.8355	-0.3863	0.0272	-0.4738
9	-0.3332	1.0090	-1.0566	-0.2076	0.7696	0.4133	-0.0386	-0.5652	-0.1739	0.7171	0.6625	1.0261	0.7458	-0.2721	-0.5029
10	0.2582	0.1917	0.4421	0.6954	-0.1875	-1.1729	0.6746	-0.3912	-0.2687	0.9337	-0.2494	-0.7591	0.2301	-0.4574	-0.7995
11	0.1929	0.8807	0.2194	-0.1003	0.2636	0.8213	0.5767	-0.9702	0.0110	0.5279	-0.2199	-0.8326	-0.9032	-0.6132	0.6319
12	-0.9145	-0.7774	-0.5216	0.0851	0.9493	0.0893	-0.4982	1.2450	0.3841	-0.6380	-0.3008	-1.0552	-1.4525	0.4386	-0.6533
13	1.0066	-0.8929	-0.6389	0.6208	0.8485	0.2493	1.1665	-0.5280	-1.0302	1.0674	0.4754	-1.1620	0.5322	-0.5519	-0.1444
14	-0.6913	0.4289	-0.3619	0.0672	-0.4982	-1.3007	0.4288	-0.3986	-0.3383	0.8838	-1.2641	-0.2524	1.1131	0.3156	1.1161
15	0.0031	1.6043	0.1243	-1.4142	-0.1624	-0.7463	0.9421	0.5255	0.3802	0.6244	0.6025	-0.0780	0.3034	0.0904	-0.8122
16	1.4438	-0.9164	-0.5843	-0.6583	-0.6303	0.0170	-0.4870	0.0660	0.7737	-0.4259	0.5215	-0.7359	-0.2536	0.7025	-0.0445
17	-0.5069	-0.1193	1.1596	0.0871	0.5476	0.5178	0.4556	0.6730	-0.3881	0.8358	-0.8834	-0.2636	-0.3169	-1.0401	-0.7674
18	-1.1879	0.9043	-0.2998	-0.2671	0.6076	0.7574	-0.4662	0.5870	0.5581	0.2840	-1.0449	0.1121	1.3587	-1.0424	1.5856
19	0.1432	0.3477	-1.2582	0.0362	0.1812	-0.2953	0.4126	1.0356	-1.3127	-0.4282	1.8051	-0.6366	-0.6866	1.3732	0.6130
20	-0.7604	-0.7117	-0.9513	-0.0406	-0.1750	0.0905	-0.6186	-0.0451	0.6936	-0.5137	-0.1816	0.6592	0.0338	0.4960	-0.1712
21	-0.9116	0.3793	0.7858	-0.9352	-0.7125	-0.1048	0.2663	-1.3449	0.3789	-0.5784	0.1616	-0.8794	-0.1002	-1.1385	-0.6515
22	0.7922	0.7033	0.8134	0.5233	-0.1884	1.0077	0.7210	-0.5814	1.1230	-1.0014	-0.2385	0.2255	-0.9464	-0.7735	1.0464
23	0.9937	0.1849	-0.2110	0.2330	0.2590	-0.6121	0.0639	0.4883	-0.0988	-0.1743	-0.6131	0.1318	0.0964	-0.8784	0.1295
24	0.6571	0.6278	0.2800	-0.4061	0.5725	-0.2842	-0.8027	-0.7960	0.1410	0.6296	-0.3948	-1.0011	0.2533	-1.0039	-0.1098
25	-0.4460	-0.7970	-1.0329	-0.1907	-0.5387	0.6074	-0.0747	0.5949	0.9433	-0.3568	-0.9509	-0.0840	0.4132	-0.3076	0.1390
26	-0.2505	0.7719	-0.4812	-1.4389	-0.5751	-0.3648	0.0380	0.8325	-0.8191	1.0758	1.3328	-0.9137	0.2850	-0.7073	0.6702
27	0.2115	0.0428	-1.2406	-0.2474	0.3599	0.1608	0.4736	-0.2604	-0.4777	0.0140	0.0951	0.5309	0.0423	0.9715	-0.1774
28	-0.1341	0.0328	0.2055	-0.1436	-0.0450	0.7485	-0.1835	1.1829	0.6461	-0.2482	2.0235	0.1466	-0.1032	0.6440	0.0615
29	0.8449	-0.9185	-0.2508	-0.5808	0.1430	0.7073	-0.8188	0.2142	0.5897	0.2256	-0.0509	0.7833	-0.6288	0.5775	0.0531
30	-0.6583	0.8924	-0.1525	0.3401	-0.4189	0.7239	-0.1488	0.5728	-0.5491	0.7959	-1.0555	-0.4187	-0.2708	0.2598	-0.7508
31	1.0993	-1.4055	0.4553	-0.2645	0.5546	-0.6188	0.2364	-0.6841	0.6039	-0.4431	-0.8241	0.0321	0.1697	0.2009	1.8825
32	-0.2383	-0.4032	0.5612	-0.6002	0.3680	0.2974	0.6496	0.4498	-0.7839	0.1671	-0.6999	-0.0478	-0.2510	-0.6671	0.9860
33	0.2748	-0.8913	-0.5714	-0.1488	0.4897	-0.0512	-0.8519	-0.4337	0.0029	-0.2298	-0.8762	0.2632	0.0058	-0.1426	-0.6261
34	0.3860	-1.0272	1.1020	0.6418	-1.0490	0.3348	-0.0887	0.0752	0.9042	-0.3562	0.0120	-0.2753	-0.6506	0.8828	-0.9164
35	0.2769	-0.9937	-0.2505	0.2102	-0.0484	0.5487	0.2360	0.4454	-0.3578	-0.1055	0.5175	0.9620	0.1505	0.5346	-0.3626
36	0.4225	0.6110	0.4725	-0.6314	-1.0769	0.1687	-0.3564	-0.9259	0.9719	-0.5546	0.6252	-0.1407	0.2192	0.5219	-0.2949
37	0.1810	-0.2241	-0.4267	0.8789	-0.8850	0.7652	-0.0471	-1.3647	0.4828	-0.2773	-0.2674	-0.2471	-0.1419	0.0126	0.2705
38	0.4360	-0.0621	0.0674	-0.1693	0.1837	-0.0525	-0.2751	0.0706	0.8385	0.9185	1.0022	0.4850	-0.6492	-0.9022	0.1584
39	0.5101	0.9076	-1.3550	0.9385	-0.6900	-0.3484	-0.4836	0.7374	0.8717	-0.4715	-0.1504	0.8312	0.9039	-0.8223	-0.1255
40	1.0166	0.5510	-0.3431	0.3258	0.3680	-0.9066	0.1722	-0.4540	0.6292	0.7494	-0.1540	-0.5136	0.3982	-0.9299	-0.6493
41	0.6176	0.9207	-0.0578	0.4274	-0.5284	0.8483	-0.6813	0.1902	-0.0636	1.1423	-0.8359	-0.4554	-1.6017	-0.4280	-0.9250
42	0.8716	0.9163	0.7663	0.1696	-0.4991	-0.3439	1.1974	0.4322	-0.3233	0.5921	-0.9608	0.8335	0.9218	-0.8915	0.3921
43	0.3971	-0.1605	-1.6952	-0.8636	-0.2406	0.0853	-0.1366	1.0808	-0.6558	-0.8498	-0.8593	-0.4788	-0.3308	-0.4646	-0.4290
44	-0.7226	-1.8563	-0.6462	-0.5718	-0.8185	1.1133	0.9573	-1.8752	0.0877	0.0548	1.5854	0.1576	-1.2903	-0.9638	-0.0318
45	0.5959	-0.4162	-1.0520	-0.1862	0.7562	0.0319	0.3451	0.4310	0.9223	-0.2253	-0.7076	0.6743	-0.2688	-0.1775	0.2618
46	-0.0449	-0.6883	-0.5666	-0.6824	0.2185	-0.3151	-0.7784	-0.7968	0.2470	-0.4523	-0.9297	0.2189	0.6465	-1.0637	0.7116
47	0.3831	-0.2844	-1.4421	-2.3971	-1.1766	0.1207	0.2657	0.1824	0.5275	-0.4150	-1.4834	0.5858	-0.9535	0.2485	-1.1159
48	0.8593	1.0306	0.8072	0.4945	0.2698	0.7316	0.0146	1.2897	0.6083	-0.9436	-0.3227	-0.5797	0.6390	0.0257	-0.7484
49	0.5813	0.4250	0.1874	0.0577	0.6746	-0.8626	0.3497	1.1162	-0.8250	-0.4773	-0.4998	0.4178	-0.5919	-0.5677	1.1001
50	0.8376	-1.1529	0.5175	0.0257	0.6619	-0.4692	1.5857	-0.6256	-0.7191	-0.4761	-0.5909	-0.9231	-0.1240	1.2604	-0.4190
51	0.4300	-0.2913	-0.2322	-1.0402	-0.3729	-0.8660	-0.1749	0.7831	0.8935	0.8441	-0.0088	0.9175	0.3181	0.1795	0.9941
52	-0.1805	0.1486	-0.7040	1.0906	0.0913	-0.6338	0.5945	0.4073	0.6216	0.2324	-0.1034	-0.7239	0.1146	0.3195	0.1946

	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30
53	0.2650	0.6634	0.1495	0.3268	0.8173	-0.1609	0.6795	0.6733	-0.7275	-0.1067	-0.3697	-0.1865	0.4121	-0.2499	0.7074
54	-0.7259	0.2181	0.4789	-0.3097	-0.6854	1.0250	0.8350	0.2202	-0.6790	-0.1044	-0.6756	1.0114	0.7114	-0.1780	0.4153
55	-0.0722	0.0043	0.8917	-0.4568	-0.1763	-0.6433	-0.3253	0.5392	-0.6425	0.9646	0.8403	-0.9777	-0.2727	0.6187	-0.3177
56	-0.3550	-1.3465	-0.1506	0.0644	-0.1088	-0.6434	0.2872	1.6179	0.4550	0.0362	-0.5336	-0.0155	0.3488	-0.3845	1.2162
57	-0.1909	0.4123	-0.8497	0.9397	-0.3940	-0.4559	-0.3700	-0.1203	-0.2820	-0.3320	0.9231	1.0124	-0.8223	0.6382	-0.3197
58	0.2998	0.1387	1.4640	0.3264	-0.7707	-0.7109	-0.1040	0.3646	-0.6194	0.8546	0.2939	-0.9576	0.6762	-0.8083	0.5741
59	-0.7383	-1.7356	0.4196	-0.5234	-0.3641	-0.1640	-0.0087	0.5956	0.3313	0.0396	-0.0483	0.5327	0.3724	0.9871	-0.4961
60	-1.1610	-0.6418	-0.2591	-0.1626	-0.7464	0.8692	-0.1287	0.1293	-0.3316	0.3655	-0.7933	0.3762	0.5759	1.8687	-0.9573
61	-0.6727	0.7639	0.4954	0.8703	0.5604	0.1647	-0.8530	-0.9211	-0.1621	1.0940	-0.1152	-0.0993	0.0674	0.1661	0.0201
62	-1.2604	0.5297	-0.2863	1.0210	0.7179	0.0642	0.4867	1.0425	0.0741	1.0080	0.8731	0.2503	0.3632	-0.4183	0.5430
63	0.4873	-0.7887	0.3447	-0.6943	-0.5092	0.4111	-0.6663	0.2559	-0.7396	0.0982	-0.8711	-0.0152	0.5182	0.7608	0.4366
64	0.0762	-0.8562	0.3658	-0.7243	0.6501	-0.6089	-0.7479	0.0306	-0.2282	-0.2692	-0.3775	0.4863	-0.3045	-0.2570	-0.2581
65	0.4161	-0.4457	0.9057	0.1550	-0.4981	0.3667	-0.8141	0.2410	-0.0185	0.7675	0.9571	-0.3568	0.4597	-0.9470	0.3503
66	0.2378	-0.9381	0.1376	0.3884	0.6586	-0.3776	0.0242	0.3067	0.1950	-0.5134	-0.5371	-0.7845	-0.0881	-0.1774	-0.8428
67	-0.2934	0.7948	0.9086	0.2459	-0.2924	0.0938	0.7542	-0.5679	-0.1160	0.6194	0.6705	0.6495	-0.2788	-0.2774	0.1988
68	0.2975	0.2894	-1.0922	-0.1118	-0.3636	-0.2131	0.3265	-0.5577	0.0364	0.3059	-0.4642	1.0263	-0.1131	-0.6398	0.7738
69	0.1334	0.6473	-0.0011	0.5795	-0.1632	0.3581	0.6142	-0.5441	0.2787	-0.3108	0.7933	0.5011	-0.4831	-0.1209	-0.1952
70	0.4979	1.0225	0.7159	-0.0596	0.0253	0.9580	-0.6460	-0.9954	0.8888	0.5209	0.8100	0.2072	0.7085	0.4002	0.1990
71	-1.0863	-0.3485	0.5939	0.7550	-0.4677	-0.2116	-0.3816	-0.3236	-0.7949	-0.0236	-0.8223	0.7374	0.4834	0.6282	0.5989
72	-0.6034	0.3680	-0.2241	0.0052	0.0438	-0.6568	0.7974	0.5418	0.7340	-0.2360	-0.7741	0.1062	-0.5594	-0.8658	-0.3082
73	0.3478	1.3857	3.8445	2.3933	0.2436	-2.8353	0.1661	0.2454	-0.1495	-0.0725	-0.5612	-0.2479	-0.2891	-0.2613	-2.2582
74	-0.2998	-0.8899	0.1810	-0.1857	0.9936	0.2895	-0.7450	-0.7107	-1.0450	-0.1772	0.7199	0.4695	1.3768	-0.2398	0.0546
75	0.3448	0.3140	0.9104	-0.5405	-0.0486	0.5481	0.4249	0.4719	-0.4154	-0.2687	0.2462	-0.9248	-0.9005	-0.9090	0.2397
76	0.7958	0.5142	0.0510	0.6336	0.8944	-0.9255	-0.1468	0.8354	-0.7280	-1.0173	-0.7345	-0.5658	-0.7790	-0.9126	0.1254
77	0.2148	0.2747	-0.2248	0.0566	-0.0436	-0.2206	0.9466	0.3446	0.1792	-0.1473	-0.8615	-0.6272	-0.0443	0.2039	-0.2796
78	-1.1636	-0.8452	-0.5998	0.1646	0.8777	-0.7121	0.9969	1.0385	0.6287	0.6253	-0.9513	0.1226	-0.9461	-0.2687	-0.4174
79	-0.2352	-0.9232	-0.1999	1.4144	0.1930	0.2509	-0.2949	-0.6797	-0.5256	-0.0208	0.9925	0.4497	0.6339	0.4505	-0.5494
80	0.4331	0.5354	0.6442	0.4245	-0.0108	-1.0960	0.5468	-0.7086	-0.7489	0.2665	0.4862	0.5875	0.9850	0.9924	-0.5462
81	-0.5028	0.6979	0.0759	0.1684	-0.7224	-0.8370	0.7678	-0.8782	-0.0238	-0.9000	-0.4082	0.7719	0.0650	-0.5970	0.8861
82	0.4573	-0.6502	0.4628	-0.5930	-0.3278	-0.4356	-0.7217	0.1368	0.6085	-0.2746	-0.8212	0.4981	-0.0964	-0.4331	-0.8610
83	0.4038	0.7349	0.1812	-0.5826	0.3969	-0.0738	0.7518	0.8709	0.1948	-1.1418	0.3605	-0.1318	0.7939	-0.3833	0.8098
84	0.7544	-0.3074	-0.7866	0.5460	0.0510	0.5177	-0.2620	0.7809	0.2280	1.0346	-0.7599	-0.6380	-0.3539	0.1603	-0.8008
85	-0.5900	1.2178	0.6026	0.6790	0.5829	0.1519	0.4958	-0.1825	-0.5968	1.1269	-0.8138	-0.8344	-0.2801	-0.1153	-0.4194
86	-0.6385	0.1063	-0.1321	-0.3627	-0.4746	-0.8481	-0.6578	0.2504	-0.3157	0.9423	0.9466	-0.7600	0.6656	-0.5381	0.1510
87	0.9570	-1.9275	-0.3380	0.0965	-0.7963	0.5768	-0.4785	-1.0944	-0.6445	0.2830	0.2172	-0.7964	-0.1906	0.3582	0.4852
88	0.7499	0.8016	0.2882	-0.1028	-0.4536	-0.1201	-0.2036	1.1822	0.6062	-0.7498	-0.2183	0.2384	-0.9359	-0.5536	0.1946
89	-0.2848	1.6968	-0.0910	1.2079	0.5634	0.3422	0.4978	-0.1715	0.7664	0.9114	-0.4819	0.8938	1.1908	-0.7449	0.0703
90	-0.6370	0.3717	0.2817	-0.6908	-0.0184	0.3670	0.9591	-0.6686	0.0316	-0.4724	-0.4955	-0.8323	-0.2833	-0.5432	-0.6759
91	0.3733	-0.9305	0.9292	0.4759	-0.6223	-0.4530	-0.1559	-0.2552	-0.6791	-1.0731	-0.1500	0.1385	0.0670	0.7867	0.4992
92	0.0773	0.7724	-0.0985	-0.7083	-0.8643	-0.9413	-0.3568	1.2177	0.2764	0.9195	-1.1891	-0.4892	-1.6149	-1.0643	0.5643
93	0.8273	0.8981	0.9222	1.6711	-0.6196	-0.1346	-0.5837	1.0655	-0.1625	0.3918	-1.0214	0.4815	0.1494	1.4063	0.8307
94	-0.5020	-0.1165	-0.6833	0.8182	0.9821	0.9883	-0.8329	-0.2891	-0.1916	-0.4612	0.6746	-0.9437	-0.6826	0.7520	-0.8878
95	0.0775	0.1019	0.4657	0.7543	-0.1566	0.9093	-1.0092	0.1520	0.3866	-0.8099	-0.0863	0.0097	-0.7114	-0.3554	-0.8515
96	-0.5101	-0.6018	1.3319	1.5059	1.1129	-0.4897	0.9900	-0.5111	-0.0810	0.1944	-0.3884	-0.8526	-0.4042	-0.6460	0.6129
97	-0.2127	-0.2547	-0.9901	2.1339	-0.7459	-0.6724	-0.0574	0.3446	-0.1847	0.1578	2.5184	0.8418	0.7100	0.7042	-0.1380
98	-0.6074	-0.5809	-0.0758	0.2669	0.4213	1.0274	0.8183	-0.2216	-0.4008	0.5177	0.1283	0.3561	-1.2070	0.8243	0.9878

	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30
99	0.6321	0.6582	-0.8731	-0.1865	0.6749	-0.0284	-0.2122	-0.5469	-0.2329	-0.1650	-0.1077	-0.1114	-0.7894	-0.1617	-0.8908
100	-0.8666	1.9769	0.6422	0.0665	-0.0787	-0.7607	-0.2226	0.7155	-0.5279	-0.3043	0.2153	0.6662	-1.7969	-0.3300	0.3722

Tabela D.5. Valores dos pesos das conexões entre a primeira camada oculta (120 neurônios) e a segunda camada oculta (100 neurônios) da rede *Magola11*. Coluna 31 a 45. matriz de pesos LW1.

	31	32	33	34	35	36	37	38	39	40	41	42	43	44	45
1	0.5139	0.3063	-0.4786	0.1294	-1.4393	-0.8853	-0.5192	-0.4999	0.4260	0.0795	-0.4966	0.4695	0.2868	-0.7748	1.1074
2	0.4210	0.1471	0.6133	-0.3349	-0.9461	-0.6401	-0.4776	0.7130	1.0428	0.1402	0.0032	0.0608	-0.3225	0.9408	-0.4442
3	0.0820	0.3206	-0.6132	0.0593	0.2641	0.2857	0.7672	0.0288	0.8805	0.0852	-0.1628	-0.4512	0.4899	0.2968	0.9521
4	-1.3546	0.1449	-0.9909	0.7695	-1.4397	-0.7714	-0.2625	-0.9972	1.2947	-0.2119	0.5113	0.4395	0.3841	-0.5511	-0.9004
5	0.7207	0.4693	0.7037	0.4330	-0.8309	0.5195	0.7762	0.1164	-0.4625	-1.0364	0.3204	-0.1994	-0.3755	0.7333	-0.6276
6	0.5565	0.7029	-0.6975	0.5834	-0.1630	0.8525	0.2865	-0.5657	0.0142	0.5590	0.5080	0.2403	-0.5448	0.5603	0.1199
7	0.3561	0.0525	-0.5947	-0.1128	-0.0309	-0.5784	0.2332	0.6152	-0.1037	0.7638	-0.2860	0.2825	0.8963	0.5918	0.6751
8	0.5896	-0.0880	0.8350	-0.6172	-0.0478	0.0908	-0.4558	0.2113	0.1357	-0.9577	-0.8679	-0.4236	-0.4000	-0.0717	1.0861
9	0.6271	-0.7173	0.9202	-0.9059	0.1173	0.6679	1.1054	0.5755	-0.9590	0.1746	-0.2942	0.2644	-0.3495	0.3391	0.4349
10	-0.3249	0.9141	0.0495	0.9095	-0.6436	-0.1422	0.2900	-0.8785	-0.9591	0.4802	0.3914	0.4154	0.7834	-0.5662	-0.8165
11	-0.6723	-0.1475	-0.3690	-0.1911	0.9522	0.5422	0.9598	-0.1814	-0.7513	-0.3299	0.5359	-0.7887	-0.2713	0.6751	-0.2067
12	-0.9787	0.7840	0.4194	1.6638	-0.9292	-0.9799	-0.3687	0.6671	0.8907	-0.5475	-0.9943	-0.3043	0.5000	0.0877	-0.3189
13	-0.4095	-1.0607	-0.5426	-0.2405	0.4095	-0.2721	-0.2219	0.7218	-1.2340	0.2154	-0.0918	-0.7283	-0.7387	0.1780	-1.3597
14	-1.1156	0.4281	0.7116	-0.9882	-2.0430	0.9765	-0.4734	0.2555	-0.3914	-0.4212	0.1684	-0.0028	1.0165	0.8543	0.4524
15	0.9935	0.7037	0.3118	0.9882	-0.4500	-0.0717	-0.5188	-0.1534	0.2181	-0.5127	0.1370	-0.7757	-0.7464	1.0573	1.0155
16	0.2578	-0.6582	-0.2413	0.6011	1.1184	-0.3709	0.2203	-1.0952	-0.9282	0.3557	0.1783	-0.0575	0.4918	0.3148	0.6628
17	-0.7036	-0.7876	0.3270	0.4322	-0.0598	-0.0176	-0.4126	-1.0252	0.1439	-0.7854	-0.5413	-0.2365	-0.5857	-0.2151	-0.5304
18	-0.3938	0.9591	0.4487	-0.0232	0.0263	0.9478	0.0153	-0.6190	0.4657	1.1392	0.5798	0.5133	2.2362	0.3477	-0.6206
19	-2.7476	-0.7169	-0.6354	-1.0190	2.1257	0.8070	0.5024	-0.7467	-0.0142	-1.1962	0.7432	-0.5330	5.2930	-0.3134	-0.9181
20	0.8692	0.5572	-0.8336	-0.2779	0.1900	0.3335	-0.0483	0.7092	0.3791	-0.1742	-0.8588	0.0257	-0.1810	-0.0405	-0.2426
21	-0.0390	-0.8614	0.5355	-0.0380	1.4800	0.8032	-1.1910	-0.4766	-1.4071	-0.6827	0.3650	-0.4250	-0.6430	0.1386	0.6233
22	-0.6195	-0.3462	1.3148	-0.4560	-0.8721	-0.5082	-0.1351	0.7976	1.0340	0.8847	0.0863	-0.8492	-0.8245	0.5800	-0.5596
23	0.2647	0.2736	-0.9379	0.4701	1.2428	-0.0171	0.6453	-1.5166	-0.5992	-0.7073	-0.7448	-0.6077	0.0507	0.7910	0.7089
24	-0.4507	-0.4729	-0.8274	0.1185	0.2084	-0.1819	0.5948	0.9006	-0.5968	0.1490	0.0167	-0.9993	0.5630	-0.4430	-0.6569
25	-0.1572	0.0882	0.6148	-0.5165	-0.3862	0.6057	-0.3221	0.5065	-0.7543	-0.5106	-0.9762	-0.8529	-0.1694	0.7535	0.1151
26	-0.6622	0.1115	-1.0279	-0.2495	-1.8193	-0.7809	0.3098	-1.1420	0.1482	-0.0832	0.1900	0.8778	-0.4957	0.5743	0.6662
27	1.1817	0.2136	0.3641	0.3045	-0.5070	1.0289	-0.7792	0.0456	-0.6699	-0.5047	-0.5041	-0.4816	-0.0380	0.4769	0.3717
28	1.0015	1.1891	0.0087	0.4151	2.1568	0.0597	1.2970	0.7575	0.3278	0.5810	-0.8173	0.5203	0.1127	0.8968	-0.5509
29	-0.0279	-0.4567	-0.6865	-0.0256	-0.1092	0.4914	0.4389	-0.2218	0.5523	0.0132	0.9291	0.0978	-0.7128	0.1991	-0.9328
30	0.8899	0.9090	0.0974	0.5065	-0.4110	0.6555	0.1295	0.3606	0.6000	0.8278	-0.5570	0.2417	-0.4968	-0.5692	0.5954
31	-2.7348	0.1270	-0.4648	-0.4561	1.0516	0.3106	0.4956	-1.2119	-0.2247	1.1065	0.3647	-0.4881	0.5396	-0.9544	-0.9587
32	-0.4682	0.7943	-0.0363	0.0392	0.0224	-0.2692	0.6070	-0.3186	0.8119	-0.6245	0.6928	0.5501	1.0014	-0.1476	-0.6728
33	0.8496	-0.1827	-0.2381	-0.3890	-0.2337	-0.6117	-0.0851	-0.4867	0.0830	0.0131	0.3402	0.4230	0.2727	-0.5534	1.0841
34	-1.5207	-0.1642	0.8591	0.0295	-0.1478	0.4374	0.3685	-1.2869	0.2557	0.4444	-0.4070	-0.9298	0.9254	-0.8876	0.1171
35	1.0991	0.3246	0.3323	-0.3267	-1.1828	-0.5086	-0.2467	-0.2764	-0.0172	0.2460	0.6039	-0.9863	0.4715	-0.8738	-0.9810
36	-0.3327	-0.1052	-0.2482	0.0763	-0.5598	0.8600	-0.9427	0.1284	-0.9142	-0.1175	-0.6197	0.1036	-0.6283	0.5024	0.7997

	31	32	33	34	35	36	37	38	39	40	41	42	43	44	45
37	-0.2346	0.7820	0.6259	0.6204	-0.2606	-0.3863	0.7189	1.2527	1.4231	0.1196	-0.1105	-0.2140	1.0054	-0.1089	0.4666
38	0.6616	0.2751	0.2369	-0.7424	-0.5676	0.0098	0.3449	0.3286	0.4303	-0.7930	0.9734	-0.3904	0.9456	-0.6306	-0.7264
39	1.1961	1.3408	-0.8117	-1.4403	-0.0874	0.9698	0.1964	0.9412	-0.8158	-1.1661	-1.2576	-0.7204	0.7809	0.8596	1.1501
40	-0.4073	-0.6414	0.5080	0.2490	-0.4481	0.4101	-0.6633	-0.4627	0.7756	-0.5941	0.9827	0.7189	0.2791	0.1957	-0.8232
41	-0.0464	0.4335	-0.4588	-0.4507	0.5488	0.8792	1.3156	0.8913	-0.9727	0.8958	0.8181	0.2502	-0.3181	0.1631	-0.6858
42	-1.4190	0.0028	1.0367	0.4993	-1.1171	-0.7896	-0.4958	-0.3412	-0.6338	0.1996	1.0134	-0.3487	0.3501	-0.4248	0.3865
43	1.4717	0.2182	0.3217	0.0588	-0.7204	-0.5176	0.3315	-3.3398	-1.7454	-0.6507	-0.6055	0.1491	-0.3714	-0.2304	-0.9252
44	-0.2417	0.2678	-0.3342	0.4064	-0.0114	0.2345	-0.7040	0.1873	0.2116	-0.9347	-0.1675	-0.5302	-0.1515	-1.0412	0.1797
45	0.7577	0.9743	0.6543	-0.8903	0.8659	0.6312	-1.0515	-0.8048	0.8021	0.8036	-0.0997	0.8037	0.0396	-0.0194	0.6473
46	-0.6959	0.8061	-0.1224	0.3179	0.4547	-0.6830	-0.6736	-0.8874	-0.2259	0.1475	0.6898	-1.0326	0.4248	0.0439	-0.1280
47	-0.2176	0.4462	-0.0571	-1.1669	0.2709	0.1568	0.7267	-1.5358	-0.6188	0.2869	-0.3985	0.2336	-1.2525	0.6425	0.7744
48	0.6170	-0.4785	0.3287	0.0843	0.6245	0.4148	1.5787	-0.1291	0.2766	-0.1465	0.7791	-0.8748	0.7117	0.2830	1.0702
49	0.1863	-0.6477	-0.9416	-0.0453	-0.6115	0.1133	-0.5303	-0.1435	-0.7246	0.2256	0.6491	-0.4245	-0.2456	0.9015	-0.1909
50	-0.1170	0.2073	0.1129	0.1639	-0.3873	0.0840	0.5187	0.0470	0.8486	0.5176	1.5415	-0.8026	0.4593	-0.4250	0.1724
51	-0.6699	-0.0962	0.0478	-0.6617	-0.9420	-0.2108	0.5690	-1.0639	-0.9195	-0.1015	0.7056	-0.4590	0.3936	-0.5812	-0.1554
52	-0.3864	0.3724	1.0234	-0.3998	0.4640	-0.4031	0.9811	-0.0017	-0.3008	-0.3079	0.0808	0.6064	-0.3620	-0.5102	0.0169
53	-0.3419	0.6180	-0.7015	-0.4521	0.9409	-0.3761	-0.6183	0.3754	-0.4148	0.0639	0.0756	0.0496	-0.2972	-0.7824	-0.0272
54	-0.4702	0.7248	-0.7548	-0.5045	-0.5946	0.4729	-0.7677	0.5141	-0.7282	-0.8148	0.4743	-0.9412	0.0473	0.3057	0.9361
55	0.3134	-0.4806	0.0424	-0.3361	-0.5591	0.5219	-0.3884	-0.7244	0.7985	-0.6424	-0.6748	0.1118	-0.6143	0.5029	0.5738
56	-0.3066	-0.0814	-0.3492	-0.2597	1.0744	0.3807	-0.0867	0.6561	0.6754	0.1343	0.5424	0.7229	-0.2279	-0.2666	0.7704
57	-0.2690	0.8761	-0.6591	-0.2731	-0.1843	0.6480	-0.0862	-0.2534	-0.8301	0.9283	-0.5489	-0.6170	0.1904	0.0839	0.1147
58	-1.0690	-0.5798	-1.1682	0.1241	0.9096	0.5222	0.7820	0.7650	0.6381	1.0048	1.2441	-0.0161	0.6515	-0.0307	-0.3289
59	1.8797	0.9725	-0.7080	-0.2352	-1.5623	0.9338	-0.7218	-0.5415	0.5075	-0.2021	1.0520	0.8990	-1.0948	0.7955	0.8474
60	0.1311	0.7704	-0.1338	-0.6348	-0.3451	0.9652	1.4281	-0.6303	-0.6397	1.3964	0.2692	-0.4326	0.1071	0.6166	0.8656
61	-0.0297	0.4200	-0.7482	1.0102	-0.4065	-1.0598	0.7195	0.1757	0.3351	0.4896	0.0820	-0.1412	0.3663	0.0028	0.3264
62	0.7301	-1.1806	-1.0053	1.0248	-0.0221	-0.1555	0.1624	-1.9328	-0.1305	-0.0014	0.2030	0.1656	-0.8345	-0.8981	0.2113
63	0.8312	-0.9094	-0.3677	-0.0976	-0.7877	0.1788	0.7741	-0.6695	-1.0243	-0.7261	-0.1798	0.7871	0.9354	-0.8405	0.3260
64	-0.7020	-0.5169	-1.0150	0.4282	0.9206	-0.8377	-0.8854	0.4610	0.8726	0.3646	-0.5628	-0.8906	0.8087	0.3850	0.4386
65	-0.4603	-0.3069	-0.4960	-0.6303	0.9176	0.5851	0.0905	-0.5206	1.5402	0.3133	0.4209	0.0178	1.6511	1.0358	-0.2611
66	-0.0332	-1.0580	-1.2584	-0.4629	-0.8533	0.0094	-0.0556	0.5468	0.4994	-1.0266	0.8514	0.7425	-0.0306	-0.8762	-1.1988
67	-0.1618	0.6603	0.2437	0.3335	0.4355	-0.9376	-0.8883	0.5024	-0.9049	0.2345	-0.3430	-0.6104	-0.1524	0.5897	1.0179
68	-0.6921	0.9387	-0.7617	0.2803	0.8798	-0.0273	-0.1391	-0.4988	1.0578	0.0905	-0.0909	1.2399	-0.4223	-0.0011	0.1113
69	-0.5031	-0.4875	-0.8645	1.2498	-0.6254	-0.7764	-0.7054	-0.9503	0.2453	-0.3049	0.7949	0.3119	-0.3796	-0.9701	0.0959
70	-0.1620	-0.6938	-0.7612	-1.1424	0.8316	1.0152	0.7133	0.8244	0.4175	0.8245	0.3794	-0.6728	0.3759	-0.5438	-0.2342
71	-0.9125	-0.8380	0.2639	-0.3810	-0.9083	0.1924	0.6249	1.0800	0.8885	-0.1966	0.1394	-0.3287	-0.6583	0.6242	0.3557
72	-0.8482	-0.8320	0.3890	-0.6519	-0.0663	-1.0137	0.0418	-0.0416	-0.9193	-0.9931	0.8348	-0.6677	-1.0119	-0.6524	0.8153
73	-2.1228	0.7859	0.4325	0.6041	-3.4294	0.6357	0.5557	-0.6421	0.2129	-0.8709	1.0152	-0.8610	-1.1323	0.2324	-0.4267
74	0.1900	-0.4872	0.6581	0.7910	0.6538	-0.3664	0.1160	0.5270	0.2385	0.0346	-0.4465	0.4106	-0.0227	-0.0502	0.1555
75	0.7188	0.5955	0.0923	1.0183	0.8972	-0.9863	0.7732	0.8658	-0.6052	0.5373	0.4008	0.5246	0.9836	-0.2058	-0.2161
76	-0.4340	-0.4168	-0.9283	-0.7385	0.7777	-0.9290	0.6172	0.4244	-0.1657	0.2322	-0.0069	0.0496	-0.0028	-0.1802	-0.6192
77	1.1805	0.7482	0.3619	-0.1471	-1.5226	0.1240	0.6585	0.3802	-0.1426	0.5327	-0.4581	-0.6907	-0.1770	-1.0146	0.2768
78	-0.2834	0.7509	-0.6453	-0.9680	0.6350	-0.4617	0.2589	0.5653	0.4134	0.1542	-0.8538	0.5731	0.4223	0.8756	0.2841
79	-1.1660	-0.0518	-0.0237	1.0426	-0.7418	-0.1332	-1.0979	0.7347	0.1819	0.6020	0.2386	-0.0344	1.4875	0.5941	-0.4005
80	0.2333	0.6250	0.5259	0.7443	-0.7500	-0.2553	-0.4410	-0.2659	-0.6481	0.8000	0.1903	-0.9979	0.6832	-0.2099	0.6823
81	-0.8391	0.9194	-1.0902	-0.2806	0.2799	0.5380	-0.5965	-0.0754	0.6610	-0.0061	0.7181	0.3805	-0.3222	-0.4006	-0.1082
82	0.1097	0.3933	-0.4401	0.9015	0.9007	0.2222	-0.1248	0.1214	-0.5125	-0.2599	0.1714	-0.7610	0.1452	0.1702	-0.4641

	31	32	33	34	35	36	37	38	39	40	41	42	43	44	45
83	0.6412	1.1342	-0.2584	-1.4019	-0.9138	-0.5513	0.5596	0.6532	0.3024	0.7868	0.1833	0.6011	-0.0855	-0.5204	-0.2359
84	-0.0701	0.3414	-0.4867	0.2754	-0.7591	-0.6766	-0.0146	0.6076	0.0416	-0.5174	0.0273	-0.8574	-0.4349	0.4481	-0.0610
85	-1.2928	0.1809	0.6757	-0.3146	1.1385	-1.1496	0.9486	0.6093	0.7641	-0.3263	0.2143	-0.1187	-0.6991	-0.2669	0.3373
86	0.7021	-0.0196	0.7781	0.6581	0.9866	-0.2874	0.3604	-0.0101	-0.6615	0.2182	0.8906	0.7224	-0.4444	0.2888	-0.4555
87	-1.7049	-1.0190	-0.2209	0.0516	-1.1486	-0.0312	-0.9955	-0.6772	-0.6016	-0.8874	-0.9137	-0.3720	0.7738	0.5551	0.7406
88	-0.7517	0.3850	0.8061	0.7714	-1.0576	-0.9188	-0.9426	-1.1403	-0.0641	0.5498	0.2390	-0.7914	1.1948	0.8902	-0.3680
89	-0.5968	0.4883	0.7072	0.5552	2.0475	-0.3850	-0.8020	-1.3238	-0.4775	0.0142	-0.8553	0.5282	0.8717	-0.7662	-0.6338
90	0.3050	0.6128	0.5417	0.3163	0.8671	-0.2501	0.2400	-0.4525	0.4406	0.3059	1.0170	0.7260	0.6189	0.4596	0.6106
91	-0.7759	-0.2390	0.2403	-1.0860	-0.1397	0.6053	-0.6746	0.5385	-0.8010	-0.7597	0.8590	-0.2802	-0.8812	0.8493	0.4306
92	-0.5119	0.0658	-0.0881	0.4945	-0.7588	0.0437	0.7483	-0.6745	0.4947	-0.0037	0.8724	0.8194	-0.5866	-0.1370	-0.1985
93	1.5275	-0.3593	0.7409	-0.4549	-0.0210	-0.7560	2.3335	0.8752	0.6894	-0.1322	-0.0210	-0.7935	-0.6849	-0.7539	0.6094
94	-0.8078	-0.8090	0.4861	0.4082	0.4187	0.5584	0.1515	1.0492	-0.3244	0.1329	-0.5495	-1.0529	0.0239	0.2503	0.5107
95	0.9473	0.9585	-0.0290	-0.8923	-0.0732	0.0330	-0.8151	-0.0505	0.5039	-0.7247	0.6638	0.9196	0.2732	0.7493	0.6110
96	-2.0025	-0.6241	-0.5126	-0.1969	1.1218	-0.7692	-0.4849	0.9553	-1.1189	1.0186	-0.7250	-1.2478	1.2171	0.4279	-0.0198
97	0.1998	0.5507	0.8431	-0.7693	1.8543	-0.6710	-0.1364	2.8592	0.1640	0.0980	-0.5084	0.7088	0.7566	-0.5130	0.1649
98	0.1606	0.4045	0.9153	-0.6543	0.3831	-0.8624	0.4596	0.3303	-0.0983	0.0593	-0.5824	0.6629	-0.5248	-0.1701	-0.1613
99	0.7377	0.8539	0.5327	-0.4001	0.7475	-0.3382	0.2399	0.1087	-0.8226	-1.0674	-0.8979	-0.8139	0.3203	-0.5397	-1.1304
100	0.6549	1.1069	-0.6159	-0.0222	0.8136	0.0327	2.0573	0.5792	0.0452	0.6069	-0.4012	0.1165	-0.4516	0.2619	-0.9244

Tabela D.6. Valores dos pesos das conexões entre a primeira camada oculta (120 neurônios) e a segunda camada oculta (100 neurônios) da rede *Magola11*. Coluna 46 a 60. Matriz de pesos LW1.

	46	47	48	49	50	51	52	53	54	55	56	57	58	59	60
1	-0.8708	-0.1973	0.1238	-0.2183	0.5414	0.0066	-1.4163	0.3569	-0.9045	-1.3365	-0.5633	0.4305	-0.5562	-0.7640	-0.1114
2	-0.1562	-0.1821	-0.4283	0.2232	-0.4867	0.0004	-1.0331	0.3199	-0.2202	-0.8836	0.0361	0.2135	-0.5467	-0.1527	0.2050
3	-0.4015	0.5444	-0.5040	-0.6894	-0.6039	0.8129	0.2515	0.2677	-0.6593	0.5119	-0.0498	-0.5532	-0.7565	-0.0233	-0.7271
4	1.0002	-0.8483	0.8146	-0.6878	1.6652	-0.1258	-0.7969	0.0202	1.0875	-0.6450	0.1614	-1.2979	1.0984	1.6195	0.1245
5	-0.8931	0.0468	-0.2637	0.1526	-0.1022	1.1196	0.0604	-0.6569	0.3873	-0.0789	-0.7828	0.4754	0.2864	0.1538	0.1701
6	-0.3737	-0.4075	0.4387	0.4359	0.4088	0.4714	-0.0669	-0.5092	-0.5360	0.2358	0.9714	0.4892	-0.5756	-0.0651	0.0334
7	-1.1547	-0.2616	-0.9878	0.2241	0.3651	-0.0778	0.6183	-0.6643	-0.6257	0.0028	0.0473	1.0123	-0.3643	0.5169	0.3681
8	0.2922	-1.0585	0.5548	-0.8172	0.3532	0.9735	-0.3891	0.9362	-0.8949	-0.8934	0.6611	-0.6635	0.5703	-0.5383	-0.3883
9	-0.8838	0.1458	0.5216	0.7433	-0.7863	0.7617	-0.7746	-0.3811	0.3558	-0.1712	-0.9868	0.7488	0.7089	-0.2998	0.4242
10	-0.7943	-0.5502	0.6160	0.6734	-0.0980	-0.3080	0.0860	0.2380	-0.6029	-1.1964	0.1694	-0.3016	-0.6340	0.2990	-0.3423
11	0.2863	0.6238	0.4741	0.8216	-0.3640	0.2327	-0.0467	-0.4154	-0.8019	1.6581	-0.0424	-0.9948	0.4476	1.0972	0.2884
12	-0.7259	1.3290	-0.8860	0.2143	-0.8142	-0.1027	0.0242	-0.9179	-0.0815	-0.3944	0.7720	-0.1263	-1.0420	0.6356	0.7647
13	1.3082	-0.3435	-0.5199	-0.1073	0.2772	0.1591	0.1998	0.0372	-0.5808	-0.6123	-0.6747	-0.6623	0.3548	-0.0661	-0.4066
14	0.3125	-0.5163	0.4766	0.3690	-0.1011	0.0319	-0.7853	-0.5710	0.2071	-1.2839	-0.7548	0.9208	0.9143	0.4441	-0.1522
15	-0.9813	0.8888	0.8925	-0.8469	0.4544	0.5106	-0.5032	-0.4825	-1.0651	0.9684	-0.2004	1.1515	-1.1064	1.0786	0.5902
16	1.0900	1.2084	0.5183	-0.1074	0.4481	-0.8665	0.9655	-0.5370	1.0953	0.7499	0.3999	0.8621	0.8979	0.4760	0.8216
17	-0.3781	-1.1511	-0.7402	-0.2997	0.0413	-0.7449	-0.4709	-0.8969	0.1172	0.9435	0.2814	0.7668	-0.4023	0.3126	0.7107
18	0.1814	-0.0001	0.5958	0.0527	-1.0381	0.3509	0.7311	0.6274	0.0254	1.6151	-0.9542	-0.2748	0.5316	0.6521	-0.8080
19	1.3561	-0.4354	0.6884	-1.0580	-2.4501	1.5672	0.7437	1.4827	-0.4699	0.2635	0.8479	0.1301	0.8809	-0.8143	-0.5016
20	-0.9364	-0.2465	0.0324	-0.5086	-0.8860	0.4932	0.9958	0.3446	0.1303	-0.6771	-0.7337	-0.6893	-0.4893	1.2335	-0.9747

	46	47	48	49	50	51	52	53	54	55	56	57	58	59	60
21	-0.3597	-0.7842	0.1870	-0.0997	-0.6930	-0.9014	0.2123	-0.3587	0.5676	0.0545	0.7966	0.8194	0.2377	-0.0463	0.6259
22	-0.8005	-0.7347	0.2149	0.9145	-0.3992	1.1345	-0.0719	1.1457	-0.3298	0.4602	0.2828	-0.5080	-0.5830	0.5513	0.1707
23	0.8837	-0.2662	-0.3272	0.3377	-1.3178	0.9327	-0.8263	0.3612	0.3486	0.4259	0.7187	0.8950	0.5768	-0.3569	0.3916
24	1.0199	0.9400	-0.7875	0.6145	0.5642	-0.1094	-0.4630	-0.0433	0.6908	-0.3383	-0.3582	0.2072	0.1344	0.0847	0.1561
25	-0.0957	-0.7264	0.4138	-0.1819	-0.6960	-0.7993	0.0695	-0.8750	0.4610	0.3712	0.7316	-0.8770	-0.7504	0.5958	-0.3642
26	1.0092	-0.0267	-0.5129	0.6583	-0.3983	0.1242	-0.9012	1.1604	-0.2282	0.7447	0.3044	0.8444	0.0903	-0.2101	-0.3356
27	-0.8966	-0.4824	-0.1023	0.0810	0.5893	-0.6596	-0.0446	0.7153	-0.7710	-1.3853	0.3443	0.3262	-0.5153	-0.8823	-0.4334
28	-0.3191	0.7610	-0.5625	1.0027	-0.9249	1.2922	1.4270	1.6491	0.7448	2.2815	-0.3706	1.8925	-0.3048	0.5194	0.1934
29	-0.7443	0.6970	0.4555	0.9327	-0.9395	0.2872	-0.6408	-0.6856	1.0143	0.1613	-0.4051	0.9039	-0.6367	-0.6948	0.8842
30	-0.8814	-0.8036	0.1869	0.0696	-0.2104	0.8681	0.2733	-0.6902	0.8263	-0.8098	-0.9100	0.6434	-1.0256	0.5216	-0.8625
31	0.4779	-2.9491	1.2936	-0.9174	0.1589	-0.2390	1.4888	0.9544	0.1472	2.0132	0.1742	-0.8112	-0.6325	-0.5151	-0.1726
32	0.1896	0.8069	0.1953	1.0146	0.1563	0.9957	-0.8802	-0.1384	0.1485	0.3333	-1.0922	0.5874	-0.5734	0.4491	-0.5249
33	-0.9328	-0.0010	-0.3415	0.5037	-0.3284	-0.0046	0.7210	-0.7760	-0.4461	-0.3215	-0.3720	-0.6302	0.9204	0.5398	0.5579
34	-0.6230	0.2365	0.4347	-0.5402	0.6655	0.6583	0.3840	0.4450	-0.8293	0.8177	0.4442	-1.4079	1.5141	0.9534	0.4678
35	0.8908	-0.4337	1.0791	0.4145	-0.7068	-0.4298	0.1545	-0.0184	-0.0932	0.1291	0.9391	0.7781	-0.0483	-1.0217	0.5494
36	0.1899	0.4642	0.5708	-0.0168	0.4458	-0.0699	-0.8485	-0.1759	-0.9031	0.6100	-0.5520	-0.6868	0.0102	-0.4349	-0.8741
37	-0.5324	0.3931	0.4994	0.0828	0.4719	-0.2070	-0.1615	-0.9403	-0.2102	0.5785	0.6446	0.3441	-1.1291	0.9052	0.5089
38	0.3361	0.2814	0.3421	-0.5728	-0.2122	0.0385	0.1036	0.3948	-0.3397	-0.1594	-0.1674	0.2935	0.1153	-0.1045	0.1315
39	-0.1961	-0.1243	-0.6829	1.5747	-0.8038	-0.3657	-0.4557	-1.1822	0.2834	-0.0010	0.5757	0.6876	-0.4527	0.6146	0.8013
40	0.3796	0.9887	0.6394	0.1212	-0.0006	0.1768	0.5018	0.5623	0.0865	-0.3923	-0.6890	0.7010	0.1899	-0.7838	0.0083
41	0.2479	0.7444	-0.1190	0.4983	-1.3073	-0.4572	0.0851	0.2490	-0.6402	0.2488	-1.0573	0.2031	-0.9798	1.2713	0.2067
42	0.8142	-0.7752	-0.4698	-0.2359	-0.3759	-0.9403	0.7446	0.8339	0.4967	0.0116	0.1414	0.1561	-0.3323	0.6077	0.3985
43	-0.2267	0.4039	0.4396	-0.9403	0.8376	-0.8176	0.6324	0.6937	-0.3560	-0.5133	0.3593	1.2369	-0.6602	1.8101	0.7765
44	0.2175	0.5965	0.6674	-0.9533	0.2427	0.6553	0.6290	-0.1748	0.6617	-0.1263	-0.7642	-1.0104	-0.7273	-0.2757	0.8759
45	0.2955	-0.1555	-0.2179	-0.1661	-0.2295	0.2468	-0.7960	-0.6101	-0.3876	-0.5525	-0.2671	-0.8621	-0.7821	0.7356	-0.3558
46	0.7788	-0.8788	-0.3508	-0.9203	0.9316	0.0806	0.2524	0.2891	-0.8078	0.8024	-0.5816	0.2567	0.3605	0.2596	0.4572
47	0.3794	0.5233	0.9684	-0.0952	1.2291	-0.6710	0.8853	-0.5337	0.4685	-0.0533	-0.2876	-0.4731	0.4659	-0.3111	-0.2244
48	-0.7961	0.0996	0.8729	0.3538	0.7944	0.4668	0.4345	0.1744	-0.2111	1.0797	-0.4299	-0.0961	-0.1301	1.0426	0.1478
49	-0.7095	1.0236	-0.0551	-0.8946	0.1322	0.9111	0.5528	0.9951	0.3980	0.9350	-0.2493	0.7163	0.5434	-1.0197	-0.4365
50	-1.1531	1.3849	-1.0233	0.2445	-0.6492	0.7558	0.6764	-0.7465	0.9487	0.3761	0.2641	1.0801	0.4754	-0.7482	0.4858
51	-0.4798	0.5701	0.8346	1.0307	-0.0878	0.5388	0.4010	0.1214	0.3670	-0.1704	0.6727	0.7001	-0.3043	0.5516	0.1988
52	-0.6078	0.1838	-0.3252	-0.0726	-0.8718	0.3368	0.9261	0.7672	-0.8885	0.2594	-0.5828	0.6276	-0.7722	-0.2185	0.6998
53	-0.8573	-0.6608	-0.8249	-0.5862	-0.3720	0.5228	-0.1591	-0.8631	0.0975	-0.4257	0.0927	0.3257	-0.7826	0.0490	-0.3623
54	0.4326	-0.0199	0.2643	0.6068	-0.3021	0.5370	0.8781	-0.2246	0.1264	-0.3226	-1.0371	0.0787	0.5091	0.8743	-0.9118
55	-0.8317	0.2157	0.1358	-0.3250	0.6742	-0.0391	-0.2934	0.4005	0.9138	0.7041	-0.1558	1.0037	0.8901	0.2786	-0.8496
56	-0.1103	0.5556	0.9445	0.0427	0.4341	-0.9966	0.5756	-0.1592	0.7787	-0.5729	0.6875	-1.1232	-0.1679	-0.5906	0.9083
57	-0.5349	0.7230	0.4976	0.5036	0.4855	-0.8097	-0.1922	-1.0708	0.4020	0.1461	-0.8029	-0.4866	0.5374	-0.4475	-0.2647
58	0.9980	-0.4420	0.2408	-1.2419	-0.8205	-0.7354	-0.5028	0.6544	1.3967	-0.8515	0.9879	-0.4429	0.7745	-0.2902	-0.0946
59	0.8656	-0.4753	0.6785	1.0967	-0.6092	0.5845	0.8762	-0.2101	0.8083	-0.2072	-0.0461	0.8709	-0.6126	1.0028	-0.8525
60	-1.1281	-0.6084	-0.3107	-0.3475	-0.7879	-0.3632	0.4873	0.6829	0.8061	-0.3156	0.0547	-1.0394	0.2902	-0.1426	0.8839
61	0.5682	-0.6656	0.8198	0.3238	-0.0804	-0.4043	0.0334	-0.5825	0.9466	0.0861	-0.3940	-0.6543	0.4469	-1.1141	0.8641
62	-0.3807	-0.3720	-0.1848	0.3240	2.4648	-0.1577	0.5981	-0.6481	-0.6032	0.8351	0.2837	-0.3770	0.1771	-0.8739	0.0373
63	0.5922	0.8700	-1.0300	-0.3237	0.2071	-0.7942	0.3436	0.6408	0.1154	0.2689	0.2755	0.5196	-0.8488	-0.1838	0.5513
64	1.0527	0.2348	0.2801	0.4922	-0.6791	0.2354	0.5507	-0.8467	0.1520	-0.5416	-0.1367	0.8069	-0.4034	0.5053	0.5980
65	-0.5397	-0.2361	0.3886	-1.0393	-0.0229	-0.4539	1.1290	-0.5979	-0.5180	1.1463	-0.5317	-1.1152	0.9670	-0.9005	0.1820
66	0.6297	-0.0391	0.0649	-0.8961	0.1768	0.0574	0.9177	-0.4278	0.5554	0.3417	-0.6729	-0.5049	0.8602	-0.8170	0.8270

	46	47	48	49	50	51	52	53	54	55	56	57	58	59	60
67	0.4706	0.1374	0.5889	0.3694	-0.7256	0.2448	0.7242	-0.6632	0.7377	-0.2287	-0.9360	-0.9210	-0.3446	0.1407	0.7426
68	0.1853	-0.1353	-0.1497	0.0090	-0.2497	0.1496	-0.7370	0.0791	-0.6282	-0.5131	0.7506	0.3767	-0.4504	-0.8468	-1.0406
69	-0.1727	-0.4585	0.0470	-0.0570	-0.2815	-0.9018	1.0282	-0.6231	-0.8405	-0.4170	-0.5743	-0.1959	-0.5983	-0.7158	1.0106
70	-0.2044	-0.2065	0.6237	0.0353	-0.0835	0.7853	-0.5848	-0.3013	-0.1995	-0.4448	-0.8208	0.0442	-0.2169	0.7911	0.1603
71	-0.4259	0.4855	0.4100	0.3472	0.2798	0.8485	0.1451	0.1134	-0.6527	0.2257	-0.3996	-0.0253	-0.2826	-0.1565	-0.6773
72	0.8002	-0.0905	-1.0820	0.3325	0.0419	-0.2599	-0.4157	-0.1980	-0.2647	0.1422	0.9793	0.2109	-0.1536	-0.2830	0.4911
73	0.0174	-1.1751	1.0292	1.6908	0.3468	2.1451	-0.8249	-0.4299	0.8498	-3.7249	-0.4246	1.3015	1.0753	0.2787	-0.0848
74	-0.2713	0.9831	0.2159	-0.1916	-0.6254	0.9314	0.5589	0.3502	0.1629	-0.5191	-0.5416	-0.6770	-0.4424	-0.0601	0.0456
75	0.0995	0.5423	-0.6962	-0.1203	-0.4667	-0.2216	-0.6428	-0.5460	0.4387	0.2011	-0.5690	0.1134	0.1157	0.2299	-0.8041
76	0.0717	0.6594	0.4540	-0.4573	0.5813	-0.6032	1.0723	-0.2720	0.3848	0.5507	-0.7405	-0.6104	0.7880	0.0563	0.2900
77	-0.9855	0.3039	0.1690	1.0733	-1.0473	-0.6461	-1.1649	-1.3307	0.5896	-1.2389	0.7892	-0.1063	-1.1948	-0.8633	0.8255
78	0.3677	0.2002	-0.4385	0.0144	-1.0686	0.7382	0.7565	-0.3685	0.0591	0.8530	-0.4105	-0.5656	-1.4399	0.8758	-0.0163
79	-0.0911	-0.4832	0.1979	-0.7814	-1.7010	0.0958	0.7639	0.7058	0.6752	1.2308	0.2213	-0.1570	-0.5662	-0.7938	0.8656
80	0.4969	0.4243	-0.8383	-0.4972	0.6653	-0.9941	0.4513	0.9585	1.1097	0.5226	-0.5899	0.7759	-0.3080	0.5949	0.8324
81	1.0561	0.1640	0.3451	0.1527	0.3606	-1.0631	0.1668	0.3609	0.7153	-0.1478	-0.8589	0.0490	-0.8317	0.9141	0.2636
82	0.2101	0.3986	-0.8075	0.8128	0.3225	-0.1557	0.1684	-0.3707	0.6009	0.3504	0.0800	0.1751	-0.3733	0.2393	1.0342
83	0.1719	0.5343	-0.7430	-0.1734	0.1335	0.7820	-0.1728	0.0033	0.8094	0.0876	-0.1736	0.2595	0.8179	-0.3124	0.1143
84	-0.7065	0.0064	-0.1899	0.2732	-0.5340	-0.8360	-0.8488	-0.0837	0.8206	-0.7004	0.2860	0.6652	0.8508	-0.8848	-0.2621
85	0.6262	-0.7655	0.4364	0.4992	0.7237	-1.2547	-0.6610	0.4416	0.5552	0.5859	-0.1869	-0.0941	-0.7278	0.1793	0.1555
86	-0.3555	-0.2660	-0.3516	-0.1898	0.0768	-0.8343	0.6262	-0.7238	1.0138	-0.7870	0.6322	-0.4706	0.5956	-0.6256	1.0067
87	0.6785	0.4645	-1.0468	-0.4755	1.2356	-0.5091	-0.2572	-0.3277	0.1687	0.2548	-0.2038	0.5571	0.2210	-0.5467	-0.7355
88	0.6406	-0.5405	-0.9020	0.0001	0.0572	0.5213	-0.2879	-0.6251	0.8778	0.4552	1.0868	-0.6847	0.5602	0.0138	-0.9228
89	0.5665	-0.9851	-0.5459	-0.0100	0.8595	0.3287	0.5620	-1.8492	0.2062	2.2388	0.4850	-1.8095	0.5389	-0.1414	-0.3145
90	1.1794	-0.4622	-0.6036	0.1622	0.2964	-0.1703	-0.6934	0.4344	0.8912	-0.6793	1.0119	0.0215	-0.7976	-0.8501	-0.0226
91	0.6453	0.2035	-0.7970	0.1213	0.8669	0.9017	-0.1503	0.1161	-0.6941	-0.2726	0.3281	-0.1076	0.6835	0.7918	-0.9097
92	0.5909	0.0290	-0.2217	-0.1855	0.7823	0.5157	-0.7557	-0.8345	0.4991	1.0018	-0.9359	-0.1808	0.4678	0.9027	-0.4580
93	0.9463	-0.8249	0.7253	-0.0302	0.3872	-1.8193	-1.3613	-1.3141	0.4440	2.0390	0.5721	0.5480	0.8161	-0.8820	-0.4618
94	0.1558	0.1731	-0.5881	0.6153	-0.1898	-0.0141	-0.7958	-1.0467	0.5743	-1.0162	-0.0608	0.6398	-0.0978	-0.2001	-0.6927
95	-0.9032	0.2778	0.8262	0.4593	-0.2199	0.7986	0.2099	0.4972	-0.8602	-0.0679	-0.9140	0.9537	-0.6714	0.5509	-0.5007
96	0.5180	-0.8376	-0.0525	-0.4530	-0.9328	-0.6519	1.2350	-0.1730	0.5537	-0.9242	0.0240	-0.7798	-0.4220	0.9237	0.3096
97	-0.4859	1.1343	0.6902	0.9340	-0.4905	1.6344	-0.2929	0.4281	-0.2192	1.8165	-0.3844	-0.6500	0.1129	0.3954	0.5484
98	-0.9924	0.3382	0.2440	0.5790	0.3982	-0.4696	0.5239	-0.1586	-0.0576	0.0531	-0.9905	0.4321	0.4873	0.3627	-0.4611
99	1.0896	-0.5448	-0.9902	-0.5418	-0.7513	0.8363	0.6875	-0.0775	-0.2401	-0.0608	0.3416	-0.1408	-0.3935	0.2220	0.6210
100	0.4006	-0.3391	1.0614	-0.8144	0.5837	-0.9920	1.2159	-0.5480	-0.5394	1.4044	0.4804	0.6659	-0.0512	-0.1553	-0.4878

Tabela D7. Valores dos pesos das conexões entre a primeira camada oculta (120 neurônios) e a segunda camada oculta (100 neurônios) da rede *Magola11*. Coluna 61 a 75. Matriz de pesos LW1.

	61	62	63	64	65	66	67	68	69	70	71	72	73	74	75
1	0.4287	0.0785	-0.6781	-1.0366	0.6308	0.5440	-0.3693	-0.1207	-0.2148	0.1818	0.5150	0.3806	0.1774	-0.3303	-0.8781
2	-0.6403	-0.4617	-0.5280	0.4647	-0.0717	-0.3683	0.9507	0.5998	-0.9193	-0.3930	1.0350	-1.0284	0.7502	0.7909	0.3714
3	0.8852	-0.3765	0.2892	-0.6894	-0.7329	-0.3617	0.6180	-0.6358	0.0613	-0.2383	0.4377	0.2872	0.6318	-0.6680	-0.3261
4	1.4917	0.7328	-0.5356	1.0874	0.4039	1.0135	-0.1433	-1.1978	1.1157	2.3084	0.0239	-0.5457	0.8317	0.2307	-0.7274

	61	62	63	64	65	66	67	68	69	70	71	72	73	74	75
5	-0.1230	0.3253	-0.2556	-0.0605	0.6968	-0.5677	-0.8086	0.0584	-0.6856	-0.8499	-0.1106	0.2817	0.4812	-0.5930	1.0010
6	-0.5293	0.0953	0.4010	0.2543	0.4476	0.2807	0.1648	0.0649	-0.0555	-0.0196	0.4342	0.6418	-0.6160	0.8704	0.1951
7	0.8516	-0.5995	-0.2216	-0.6913	0.9020	-1.0193	0.4632	0.7486	-0.9659	-0.1432	0.7941	0.5181	0.6299	0.9195	-0.6762
8	0.5168	-0.1994	0.5315	0.7280	-0.7254	0.2801	-0.6113	-0.7303	-1.0320	0.3232	-0.1388	0.6767	0.8834	-0.9226	0.0842
9	-0.0469	0.1093	0.3782	-0.5908	0.4501	0.7589	-0.2284	-1.0227	0.8161	0.4455	1.0353	-0.4723	0.9448	-0.8617	-0.7229
10	0.0583	0.3982	0.1130	0.4136	0.6448	0.8671	0.5303	0.0837	0.2361	0.1578	0.1266	0.0670	0.8115	0.3734	0.1502
11	-0.0995	0.3006	0.1904	-0.2195	0.6515	0.3585	-0.6338	0.5524	0.4591	0.7057	0.6076	-0.9305	0.9186	1.3451	0.0874
12	0.5820	-0.5799	0.1485	-0.3964	-0.4544	0.1412	0.6660	-2.6044	0.2556	-0.7191	0.3180	-0.0290	0.0175	-0.4178	0.7365
13	0.2244	0.3758	-0.1033	0.8440	-1.1216	-0.8482	0.1489	0.7895	-0.6408	0.3104	0.2414	-0.0262	-0.4713	-0.8469	-0.1293
14	-2.1313	0.8517	0.5383	0.1968	0.3582	0.0217	-0.0941	0.0541	0.9553	-0.1525	0.2655	0.5372	-1.6189	-2.1082	-0.3702
15	0.6742	-0.7567	0.0337	-0.6526	0.8313	-0.9731	0.0766	0.1834	0.1043	-0.9514	-0.0285	0.0818	-0.4903	-0.2146	-1.3605
16	-0.8397	-0.1380	-0.4451	0.9004	0.6283	-0.7273	-0.1924	0.7876	0.9296	-0.4841	0.7358	-0.4852	-0.3154	-0.3682	-0.6321
17	-0.6690	-0.2003	-0.9194	0.4435	-0.5855	-0.1909	0.1332	0.5010	0.5741	-0.6110	-0.3934	0.5765	0.6011	-0.3032	-1.2075
18	-0.8479	0.3205	0.4975	0.6461	0.3960	-0.5362	-0.2705	-2.1474	0.1762	0.7530	-0.0093	-0.8434	-0.4036	-0.3638	1.1148
19	-0.6583	-0.6575	-1.0180	1.3250	0.3127	1.3057	0.0087	-0.3509	0.0627	1.4987	0.7540	0.9232	-1.0158	-0.8365	-1.1127
20	0.0166	-0.0538	-0.3156	-0.8128	-0.7319	0.6049	-0.0447	0.8512	-0.8640	0.4791	-0.7242	0.5813	-0.4662	-0.4117	0.5497
21	0.7257	-0.4907	-0.9093	1.0252	0.8357	0.5270	0.8145	-0.0799	-0.5391	-1.2790	0.0900	-0.0450	0.1626	0.9520	0.0279
22	0.5360	0.5222	-0.5828	-0.2837	-0.3572	-1.0132	0.0624	-0.3409	0.7743	-0.2963	0.8485	-0.4270	-0.3261	0.8386	-0.4185
23	0.1098	-0.6245	0.6693	0.9671	0.0363	0.4924	-0.4906	-0.6710	-0.2756	-0.4780	-0.5017	0.7117	1.4450	0.3381	-0.6302
24	-0.3262	0.9372	0.5229	-0.5790	0.8799	-0.9475	0.1873	0.3726	-0.5757	-0.0953	-0.2402	-0.5244	-0.9849	0.3367	-0.2850
25	-0.1922	0.7599	0.2407	-0.1361	0.7790	-0.9395	0.9462	-1.0303	-0.6706	0.4732	0.0768	-0.8743	-0.5633	-0.1477	0.2256
26	0.6752	-0.6072	-0.7413	-0.3618	0.0876	0.3070	0.4483	-0.2065	0.1616	-0.1138	-0.8122	0.6923	0.5202	1.7412	-0.2334
27	-0.5009	-0.7062	0.6193	-0.3612	-0.8909	-0.3943	-0.9085	-0.2684	0.6900	-0.1807	0.6809	0.4317	0.0111	-0.1222	0.5256
28	-1.9067	-0.3682	-0.3093	0.6313	-0.3408	-0.2721	-0.6553	3.2061	-0.4913	0.2066	-0.2936	0.7282	1.7477	0.1468	-0.4058
29	0.8570	-0.7847	0.6338	0.5753	0.4699	-0.9255	0.7026	1.1792	-0.0228	0.3216	0.4952	0.5771	-1.0988	0.0302	-0.5296
30	0.4185	-0.0293	-0.4681	-0.3711	0.7047	-0.6700	0.6797	-1.1190	-0.2077	-0.2999	0.3552	0.4739	-0.4818	-0.2014	-0.4575
31	-0.0224	-0.4323	0.2126	0.8201	0.1515	-0.8122	-0.6582	1.7894	0.2828	0.4736	0.6416	0.3246	0.1895	0.5499	-0.1593
32	0.8294	-0.2494	0.7588	0.1538	0.4198	-0.5857	0.0954	-0.3896	0.0791	0.3227	0.2257	-0.8409	0.4346	-0.4576	1.0434
33	0.0131	-0.5511	-0.2623	0.6765	-0.1012	-0.8857	-0.6339	-0.4714	0.7600	0.4112	0.8683	0.7065	0.9636	0.4013	0.3167
34	0.8536	0.2526	0.4705	-0.3500	0.1428	-1.0351	-0.1728	0.3305	1.1591	0.0994	-0.1394	-0.2557	0.6519	0.6511	0.8518
35	-0.4199	1.0449	0.1646	0.4321	0.7886	0.1917	-0.9133	0.6484	-0.4870	0.0208	-1.2715	-0.0938	0.2165	0.8928	0.2953
36	0.7504	-0.7848	-0.1876	0.8274	0.4180	0.4428	-0.3298	0.6437	-0.5963	0.7459	0.8908	0.1640	-0.7540	0.6931	-0.7824
37	-0.4234	0.7283	-0.8655	-1.0248	-0.8678	-0.4332	-0.4728	0.4813	-0.2478	-0.6773	0.0267	0.7002	1.6218	-0.5182	-0.6646
38	-0.1341	0.0090	-0.6788	0.9129	0.6990	0.1785	0.4464	-0.5599	0.4173	0.0506	-0.3749	0.9519	-0.5218	0.4279	0.2762
39	-0.2906	0.9748	0.7177	-0.0684	0.5251	0.1197	-0.4256	-0.3806	-0.9256	-0.0936	0.9897	-0.1443	0.8813	-0.4116	1.1890
40	0.8934	0.0929	0.4132	0.4256	-0.8531	0.0720	-0.1529	-0.3394	-0.5198	0.3399	0.5203	-0.7946	-0.2822	-0.5165	0.5218
41	0.7905	-0.2049	0.4716	-0.7489	-0.0722	0.7322	-0.0292	0.5354	-0.6362	0.6771	0.8609	0.3677	0.1919	-0.1584	-0.4539
42	0.3424	0.5231	0.1699	-0.2506	-0.9800	-0.6319	0.6461	-0.0535	0.6458	-0.4162	-0.0317	-0.4925	0.9718	-0.3362	-0.7398
43	0.4300	0.8635	0.0399	-0.7465	-0.0042	0.1897	0.4438	-1.2078	0.2763	0.9229	0.3084	-0.6428	-1.6820	-2.2023	0.7307
44	1.3169	-0.7822	0.6952	-0.7382	-0.0615	-0.8632	0.4963	1.0424	-0.2951	-0.2926	-0.0668	0.6532	0.8412	1.0327	0.0271
45	-0.0488	-0.4230	-0.3486	0.1400	-0.0692	0.7731	0.1678	-0.2599	-0.7775	0.2126	-0.1227	-0.5145	0.2311	0.1442	0.3242
46	0.4149	0.8550	0.8459	0.8402	-0.8181	-0.3212	-1.0572	-0.4873	-0.6332	-0.2565	0.6527	0.9046	-0.2754	-0.5887	0.7841
47	0.3721	-0.4946	0.6287	0.6917	0.3153	0.7051	1.2418	-1.0776	-0.1958	-0.0259	0.5665	0.5322	-0.3957	-0.1663	0.5822
48	-0.8237	-1.0638	-0.4722	-0.6796	-0.7943	0.2179	0.9923	-0.2150	-0.5681	1.5917	1.0280	-0.6955	-0.1447	0.6970	-0.5702
49	-0.9797	-0.8891	0.6380	1.1905	0.8908	-0.2338	0.6544	0.7415	0.5403	0.3146	-0.0925	1.0893	0.1697	0.9793	0.5908
50	0.2262	-1.0038	0.4097	-0.3896	-0.5557	-0.2401	-0.4543	-0.4144	-0.5085	-0.5720	-1.2596	0.4166	0.6632	-0.5756	-0.6528

	61	62	63	64	65	66	67	68	69	70	71	72	73	74	75
51	-1.0250	-0.5651	0.5186	0.0734	0.1689	0.1084	-0.3230	0.3982	0.3248	0.2288	0.9420	-0.7517	0.1194	0.4163	-0.6512
52	-0.9284	-0.3868	-0.0836	-1.1293	0.6376	-0.5139	0.4615	-0.6323	-0.9928	0.8276	0.9744	0.0964	-0.5549	0.4911	-0.7372
53	-0.5895	-0.0102	0.1473	-0.4817	-1.0461	0.5513	0.5497	0.5527	0.6531	-0.9498	0.3341	0.8390	0.0625	-1.0382	0.3198
54	0.2623	-0.5714	0.7043	-0.9750	0.3732	0.4474	0.3797	0.5224	0.7673	-0.1155	0.5409	-0.8092	-0.1548	0.2579	-0.1342
55	-0.6606	-0.0947	-0.7939	-0.2191	0.3476	1.0528	0.4794	0.0718	0.8414	-0.8246	0.4072	0.3167	-0.1690	0.1665	0.9193
56	0.7486	-1.0272	0.8190	-1.0014	0.3800	0.2554	0.2506	-0.1132	-0.1221	0.5651	0.7254	-1.0468	1.0888	-0.3945	-0.2637
57	0.4741	0.2008	-0.0433	-0.2476	-0.1692	-0.1871	1.0208	-0.1956	-0.3832	0.7917	0.5550	0.0114	-0.3380	-0.8195	0.9861
58	0.4687	0.9206	0.6075	0.4254	-0.4573	-0.2748	0.5393	0.2298	0.4326	-0.1304	0.8033	-0.7818	0.2298	-0.0362	-1.8407
59	0.6311	-0.8411	0.2922	-1.2087	-0.5726	-0.2381	-0.0989	1.0163	-0.5401	-0.6894	-0.6794	-0.2085	-0.7951	-1.3612	-1.0880
60	0.3612	0.5000	-0.8020	-0.5911	0.8329	-0.3006	0.9081	-0.3182	-0.4012	-1.3292	-0.7860	0.4508	0.8256	0.1568	0.4279
61	-0.7752	-0.6121	-1.0835	-0.4962	-0.0220	-0.2747	-0.9090	0.4848	0.1414	0.3191	-0.9328	-0.7783	0.7006	0.0715	0.4413
62	-0.6111	-0.3655	0.2024	-0.8737	-0.6619	0.9901	0.3556	-0.9730	1.2890	1.2331	-0.8836	-0.6480	0.9886	-0.3147	0.2376
63	-0.3973	-0.7894	-0.3075	-0.8064	-0.0307	-0.3184	-1.1092	-0.9097	0.9058	-0.9787	0.7188	-0.3497	0.5408	-0.9709	0.4644
64	0.7121	-0.3746	-0.5855	0.6870	0.3454	0.5621	-0.5126	0.8488	0.1279	-0.0804	-0.4762	0.3452	0.1345	-0.8551	0.6822
65	0.4081	-0.8677	-0.1385	-0.0373	-0.9922	-0.7188	-1.1423	0.3906	-1.0047	-0.4711	-0.2402	-0.1144	0.8466	0.3792	-0.5563
66	-0.6291	-0.4639	-0.9756	-0.5495	-0.7854	1.0272	-0.6885	-0.2500	0.9668	0.4682	0.3526	0.2137	-0.1155	-0.0592	0.4295
67	0.4459	-1.0200	-0.3780	0.2690	-0.2659	-0.2154	0.1335	0.8005	-0.6041	0.9152	0.2595	0.1316	0.0880	0.7204	0.9048
68	-0.8766	-0.0604	0.6137	-0.0490	1.0948	-0.3981	0.0056	-0.6178	0.2889	-0.4308	0.8972	0.0347	0.7851	0.5286	0.1993
69	0.6169	-0.4575	0.4058	-0.7755	-0.3386	-0.2743	-0.3356	-1.0195	0.0864	-0.2148	-0.6921	-0.1957	-0.0473	0.9114	-0.1868
70	-0.4833	0.2952	-0.6942	0.1893	0.8777	-0.1229	0.8605	0.6556	0.1913	0.6742	0.8406	0.8250	0.3345	-0.5666	0.0931
71	0.1937	-1.0259	-0.6360	0.8035	0.6979	-0.3032	0.1644	-0.7153	-1.1296	0.1913	0.9793	0.4735	0.9675	-0.6357	-0.6229
72	-0.1275	-0.3934	-0.1374	0.7057	-0.3772	0.1853	0.4157	0.0737	-0.0241	-0.1885	0.3753	-0.0990	-0.8542	-0.2963	0.1427
73	2.2733	-0.4639	-0.6242	0.0442	0.3585	0.5343	-0.9337	0.2645	-0.0964	1.3858	0.3124	-0.1290	-2.5948	0.1558	1.6318
74	1.8152	0.6818	-0.9228	-0.2675	0.3033	0.1856	-0.0221	-0.5306	-0.7550	-0.5010	-0.6429	0.1115	0.5747	0.8954	0.0036
75	-0.2415	-0.8964	-0.9580	0.9700	0.0155	0.2298	-0.2814	0.5769	0.6171	-0.7920	-0.5623	-0.4103	-0.4334	-0.5428	0.8022
76	0.9245	0.1191	-0.4734	-0.1048	-1.0193	0.7555	0.3839	0.5689	-0.8226	0.3084	-0.9716	-0.2893	-0.8101	-0.6330	-0.2297
77	-1.0335	0.9058	1.0998	-0.6191	0.7989	0.7269	0.1654	0.1083	-0.7719	0.7070	-1.2171	0.3038	-1.5981	-0.4699	1.0345
78	-0.4302	-0.6621	-0.6852	-0.4275	0.7341	0.2424	0.8121	0.4110	0.8498	0.9737	-0.6063	0.5523	0.7890	-0.2489	-0.1703
79	0.2732	0.4126	0.8686	0.7738	0.6210	0.7552	0.2911	0.4473	-0.2853	-1.1666	-0.8782	0.2200	-0.0330	1.0743	-0.0185
80	0.3432	-0.4987	-0.5675	-0.2307	0.3506	0.1891	0.8654	-1.1407	0.3111	-0.8999	-0.8413	0.6464	-0.9466	0.0148	-0.0646
81	-1.1457	0.6421	0.8319	-0.8886	-0.5608	0.0880	0.1656	-0.6948	0.7414	-0.5484	0.1807	-0.6406	-1.1542	-1.1240	-0.4132
82	0.2371	-0.2207	0.2970	0.1547	-1.2110	0.1753	-0.7905	0.1593	-0.9970	0.5716	0.7517	0.5534	-0.1576	0.2792	0.6066
83	-0.6035	-1.0820	-0.5165	0.7635	-0.3349	-0.6973	0.9402	-0.2561	-0.7423	0.6681	0.8006	0.8281	0.3089	0.4501	0.8750
84	0.3743	0.6886	-0.5019	-0.7491	0.8619	-0.0379	-0.9435	-0.7484	-0.8248	0.6722	-0.0455	0.4899	-0.0381	-0.6252	-0.5356
85	1.0383	0.7542	-0.0348	0.0180	-1.1056	0.5650	-0.6012	-1.1712	-0.0621	0.1420	0.0308	-0.5030	0.5047	0.0139	0.7308
86	0.2070	0.2385	-0.5684	-0.3406	0.2481	0.1472	-0.4706	-1.0146	-0.8606	-1.0427	-0.8593	-0.7507	-0.4091	0.6037	0.4529
87	0.2249	-0.4704	-0.9189	0.2189	0.8792	-0.3537	0.8205	0.4151	-0.7027	-0.0840	-0.9476	0.8893	0.7879	-0.4979	0.1627
88	-0.0157	-0.5902	-0.5734	-0.3865	-0.7345	-0.6745	-0.2628	0.5684	-0.4873	-0.4311	-0.9937	-0.0321	-0.2906	-0.5045	0.8144
89	0.1072	0.6134	-0.7476	-0.9613	-0.2081	-0.8932	-0.2543	-0.8343	-0.6516	0.0274	0.4529	-0.3507	-0.2934	1.3724	-0.7015
90	0.9392	-0.5049	0.8718	-0.1785	0.1896	-0.2486	0.0443	0.4383	-0.8993	0.6591	-0.9158	-0.8231	0.8463	0.7357	0.1960
91	0.1111	-0.9204	0.6404	-0.0703	-0.0761	-0.8706	1.0608	0.3299	0.6776	0.8448	0.8897	-0.9776	-0.4066	0.2103	0.8172
92	-1.5445	-0.5911	0.8369	-0.9003	-0.1098	-0.6174	0.4274	-0.3716	1.2041	0.6522	-0.8061	-0.2011	-1.0414	-1.0859	-0.1802
93	-3.5270	-0.2400	0.1437	0.4456	-0.8971	0.5730	-0.3505	1.0184	1.0395	-1.2390	0.5943	-0.4829	1.0254	-2.1667	0.5605
94	0.1152	-0.1222	0.4180	-1.0163	0.4757	0.1515	-0.1502	0.2952	0.0991	-0.9976	0.6150	-0.9157	0.0598	-0.4214	0.1498
95	-0.9742	-0.4729	0.0315	-0.6326	0.5693	-0.7020	0.1088	-0.6833	-0.5441	0.1867	-0.6605	-0.5693	0.1462	-0.5930	-0.0631
96	0.5422	0.1091	0.4611	-0.6892	-0.3734	0.5802	0.2301	2.3994	-0.0979	-1.9709	-0.9501	-0.4205	-0.7831	0.3601	-0.1078

	61	62	63	64	65	66	67	68	69	70	71	72	73	74	75
97	1.0529	0.1228	-0.3147	0.6944	0.9657	-0.8549	-0.2153	1.1747	-0.1503	-0.5934	-0.7907	0.8014	1.5970	1.3046	-0.1299
98	-0.9277	-0.1052	-0.8434	-0.5225	0.4849	-1.1205	0.7611	1.1624	0.1477	0.0727	-0.3398	-0.2430	0.7955	0.2280	0.0472
99	-0.3831	0.7284	-0.5160	-0.8206	0.7400	1.0940	0.8018	-0.8395	0.9414	0.3752	-0.0440	-0.2440	0.7927	0.1016	-0.8527
100	0.4268	0.4311	0.8513	0.4899	0.2937	-0.0323	-0.3999	0.0044	-0.0151	0.5978	0.7201	-0.0233	0.8672	-0.9419	1.4771

Tabela D8. Valores dos pesos das conexões entre a primeira camada oculta (120 neurônios) e a segunda camada oculta (100 neurônios) da rede *Magol11*. Coluna 76 a 90. matriz de pesos LW1.

	76	77	78	79	80	81	82	83	84	85	86	87	88	89	90
1	-0.6421	0.2376	-0.6523	-0.2616	0.9844	-0.4176	-0.0716	-0.0556	-0.8872	0.1066	-0.6743	-0.6681	-0.5735	0.7836	-0.7026
2	0.3005	0.7256	-0.1879	0.1887	0.9155	0.4146	-0.2846	0.6049	-0.7751	-0.3315	0.8693	-0.0871	-0.5032	0.4191	-0.9871
3	-0.3575	-0.3966	0.0363	0.7099	-0.3308	0.5129	-0.2228	-0.9911	-1.0274	-0.1256	0.0778	0.3847	0.3634	-0.2445	0.7092
4	1.2020	-0.4614	-0.1116	1.6238	0.4595	-0.3406	-0.5708	-0.2015	0.9323	-0.6872	-0.4369	-0.6789	0.7885	1.1111	-0.0548
5	-0.5185	-0.2384	-0.3167	0.1730	0.2656	0.0987	0.4467	0.8811	0.1510	0.8708	-0.8101	-0.2599	-0.1503	0.5572	-1.1064
6	-0.3812	0.5081	0.3939	0.5354	0.2808	0.4493	0.0515	0.0049	-0.5318	-0.5863	0.4431	-0.5378	0.2927	0.0667	-0.2508
7	-0.0480	0.3812	-0.0880	-0.3257	-0.8147	0.1654	0.1894	-0.6870	0.0307	-0.8869	-0.5570	0.2771	0.3878	0.8003	0.5306
8	-0.2265	0.3332	0.5593	0.8043	0.2451	-0.0728	-1.1740	0.3150	-0.6596	-1.0074	-0.3951	0.6041	0.5445	0.2202	-0.8265
9	0.0012	0.9989	0.5700	0.7644	-1.0496	0.4259	0.4118	-0.7011	-0.1848	0.6351	0.8819	0.8186	0.0379	0.5771	0.3978
10	-0.1006	0.2403	-0.8570	0.7948	0.3301	0.1388	0.8005	-0.8525	-0.0140	-0.1450	0.9151	0.0101	0.5078	0.4356	0.5885
11	0.7303	-0.4292	-0.3420	-1.0402	0.6573	-0.6074	0.6978	-0.2076	-0.5153	-0.2008	0.6916	-0.0712	0.6760	1.0433	0.8233
12	-0.6851	-0.9443	1.1625	0.3171	0.4742	0.5527	-0.4655	0.2262	1.1016	0.1027	0.9711	-0.9565	0.1752	-2.2667	0.5187
13	0.4807	0.0838	-0.3535	-0.3377	0.6471	0.2695	0.9259	-0.4631	-0.3961	0.8465	1.0441	-0.0405	-0.4034	-0.2469	-0.4779
14	0.0236	0.7910	-1.4883	0.1076	-0.3022	0.2666	0.8412	0.3134	-0.3446	0.7090	-0.8770	0.4452	-0.4016	0.4069	-0.2480
15	1.2403	0.5350	-0.2952	-1.0748	-0.2641	0.1416	0.5304	-0.4503	-1.0675	0.1271	0.0234	0.4606	0.2814	-0.7883	0.6417
16	-0.2631	0.3760	-0.9441	0.0606	0.0111	0.5085	-0.6440	-0.3311	1.0288	-0.0159	0.4568	0.3552	0.9885	0.1856	-0.0462
17	0.2550	-0.4745	-0.5001	-0.3629	0.0166	-0.5431	0.7503	-0.6027	0.9616	0.3527	-0.1191	0.2744	0.9228	0.0954	0.2095
18	-0.4814	0.0939	0.1790	-0.0868	-0.0353	-0.5724	0.4475	-0.2475	0.4882	0.3520	-0.8691	-0.0022	0.6281	0.3525	-0.6756
19	-3.6547	0.4846	-1.2444	2.0089	2.7369	-0.5320	-0.2198	-0.0457	1.1814	-0.1468	0.3920	-0.5975	0.1106	0.4923	1.1347
20	0.6156	0.4261	-0.1959	0.3124	0.3374	0.7543	0.7386	-0.5170	-0.1434	-0.6973	-0.8434	-0.2885	0.7329	-0.0691	-0.6726
21	0.1157	-1.1014	1.2830	-0.9380	1.4605	-0.8954	0.7030	0.5980	0.5793	-0.2024	-0.2698	-1.3472	-0.3524	0.6164	-0.1498
22	-0.1170	-0.8556	0.5758	0.2411	0.9335	1.0574	-0.1245	0.5306	0.1703	0.1444	-0.4715	-0.8001	0.2855	0.9153	0.2803
23	-0.8297	-0.1046	0.0596	0.2942	1.1419	-0.7432	-0.7987	0.3408	-0.3222	0.6434	0.9930	-0.1951	-0.6670	-0.3557	-0.7574
24	0.2198	-0.6565	0.4868	0.1798	0.9203	-0.4280	-0.4605	0.9474	-0.4389	-0.1350	-0.7680	0.4489	0.0393	-1.1070	0.7790
25	0.7456	-0.0197	0.9636	-0.1056	-1.0334	0.8642	0.4471	-0.3959	0.1852	0.8327	-0.5888	0.6720	-1.0486	-0.6854	-0.7928
26	-0.4131	0.1591	-1.3682	0.1511	-0.1163	0.3958	0.6759	0.1328	-0.1198	-0.7028	0.6926	-0.0128	0.7621	-0.6091	-0.4178
27	-0.9721	0.7070	-0.0706	0.1925	-0.2326	0.5325	-0.0935	-1.4096	0.0198	-0.6438	0.1936	0.5203	-0.9921	0.6938	-1.0180
28	1.5497	1.0058	1.0752	0.9907	-1.2063	-0.7648	-0.6633	-0.5978	-0.7002	-0.1193	0.5672	0.3289	-0.5234	0.8276	-0.9083
29	0.8681	0.0448	-0.8476	-0.0356	0.0913	-0.1036	-0.5444	-0.0197	0.9739	-0.7661	0.7104	0.7793	-0.2490	-0.6960	-0.0857
30	-0.6924	-0.4239	0.4561	-0.2661	-0.0173	1.0939	-0.1537	0.4225	0.7940	-0.1816	-0.5108	0.4454	0.7205	0.4378	-0.1386
31	-0.0032	0.4770	-1.3316	-0.0523	1.3622	0.9851	0.6514	1.9089	0.1935	-0.8108	-0.1786	0.5529	1.0494	-1.1395	0.8799
32	0.6083	-0.5159	0.1201	0.5051	0.3477	0.8710	0.6065	0.2682	0.4275	-0.4989	0.2198	0.7750	-0.1572	0.9204	-0.9271
33	-0.2198	1.6012	-1.2303	0.3453	0.7120	-0.5360	-0.1354	-0.6143	-0.7691	-0.6140	-0.0082	1.7900	-0.4806	-0.5665	-1.0258
34	0.9120	-0.0471	0.4377	0.1430	0.7514	0.8870	-0.3377	-0.1673	0.3856	0.5231	-0.7845	-2.9349	0.3242	-0.9752	0.1167

	76	77	78	79	80	81	82	83	84	85	86	87	88	89	90
35	-0.1396	0.6445	0.2460	-0.5671	-0.7363	0.0680	-0.8113	-0.7577	-0.1935	-0.0822	0.0920	-0.3793	0.9911	0.2820	0.9797
36	0.0269	-0.4320	-1.0332	0.0863	0.2991	0.4513	0.3575	-0.1883	0.5330	0.6493	0.1147	-0.3500	0.2263	-0.1050	-0.1680
37	0.9453	1.1611	-0.4903	0.7662	-1.3145	0.2329	-0.5996	-0.5533	-0.7310	-0.1650	-0.4160	0.2105	-0.1197	-0.3182	0.4687
38	0.7182	0.7421	-0.9964	-0.1661	-0.0113	-0.0902	0.7599	1.0536	0.7282	0.2231	-0.4019	-0.4117	-0.6459	0.2697	-0.0652
39	-0.8553	0.6497	0.6097	-0.1239	0.4003	0.3436	0.8939	0.3650	-0.9568	-0.4182	-0.9193	0.5156	0.5849	0.5633	-0.3523
40	0.1395	-0.6416	0.0520	-1.1351	-0.1231	-0.7664	0.4251	0.7658	0.0860	-0.8469	0.1172	0.5610	-0.4414	0.6196	0.3507
41	0.8678	0.2582	-1.5316	0.8020	-1.3464	-0.1563	0.4855	-0.0286	-0.2273	0.0906	-0.5088	-0.4798	-0.6763	0.0568	-0.5668
42	-0.7489	0.2463	0.8826	0.7884	0.1502	0.2290	0.2430	-0.1162	0.3050	-0.3759	0.2248	-0.3378	-0.5323	-0.3806	1.3761
43	-1.1848	0.4542	-1.0154	1.1687	0.5603	-0.4381	0.7029	0.1024	-0.1990	0.5185	1.1447	1.4190	0.8878	-0.2185	-0.1422
44	0.8257	0.2304	2.4975	-0.9597	0.1666	0.5520	0.4307	0.3498	-1.0083	0.6289	0.0172	-0.5036	0.7923	0.0956	0.1196
45	0.4280	-0.0131	0.8509	-0.0605	0.1767	0.5220	-0.3590	0.4976	-0.3738	-0.3190	-0.6612	0.0739	0.1497	-0.9587	0.4740
46	-0.6186	-0.4608	0.0525	0.3207	0.1000	0.5114	-0.4792	0.8650	-0.0088	0.0321	0.8627	-0.7571	0.2095	-0.7955	0.4214
47	-1.3869	-0.6612	0.5959	-0.4904	-0.2490	-0.6215	-0.7543	-0.0084	0.4414	-0.4740	-0.7216	1.1561	0.7388	-0.3240	0.1961
48	0.0408	-1.1251	-0.6425	0.7537	-0.5306	0.0395	0.8758	0.2323	-0.2271	-0.5142	-0.9324	-0.6081	-0.6450	1.2009	-0.5601
49	0.1107	0.7689	0.2417	0.9577	0.6132	-0.4173	0.9846	0.4118	-0.2335	0.7485	0.7793	0.3547	0.0158	0.5639	0.3104
50	0.2560	-0.2969	-0.1092	0.4518	0.9958	-0.8561	0.3264	0.4144	-0.6374	-0.8014	-0.2295	-0.9654	0.5236	-0.2034	-0.4982
51	-0.4585	-0.3312	-0.2717	0.5642	0.2086	-0.3605	0.7023	0.9255	0.6787	0.6773	0.1257	0.6849	-0.9278	0.4268	0.5370
52	-0.3294	0.2197	0.5980	0.8419	-0.3821	0.0553	0.2734	-0.2760	0.6717	-0.3724	0.0478	-0.8719	-0.8440	0.6998	0.7294
53	-0.6959	-0.2707	0.4660	-0.7285	0.3028	-0.9837	-0.2846	0.7000	-0.2808	-0.4798	0.6431	0.6829	-0.2246	-0.9395	0.6240
54	-0.6697	0.2947	0.4177	-0.4028	-0.0807	-0.2156	-0.2456	-0.1137	0.2287	-0.1121	-0.8596	0.0885	-0.8904	0.8191	-0.0315
55	-0.0816	-0.8258	-0.2332	-0.0557	0.1166	-0.9952	1.0032	-0.4541	0.1770	0.6696	-0.8384	-0.8858	0.8909	-0.3149	-0.4478
56	-1.3177	1.1362	0.0502	1.2971	0.3126	-0.5040	0.0375	-0.5988	-0.2165	0.2309	0.2420	0.4099	-0.6890	0.3898	0.1449
57	0.4052	-0.6543	-0.9674	0.5275	-0.9102	0.0734	-0.9663	-1.0066	-0.1192	-0.7588	-0.6451	1.0538	0.7688	0.2841	-0.4166
58	-0.2615	0.8508	-0.5287	-0.1246	0.4766	0.5148	-0.5743	-0.2601	-0.0948	0.0257	-0.0806	-0.7280	-0.9003	0.1280	-0.1586
59	0.0985	0.7048	0.7627	0.7395	0.8871	-0.6514	-0.4253	0.3280	-0.0275	0.3635	-0.3813	1.1243	-0.5572	0.7962	0.7429
60	1.6392	0.9891	-1.3941	-1.1381	-0.7152	-0.0845	-0.3185	0.6584	-0.6534	-0.9117	-0.3745	-0.2528	0.8614	0.2284	-0.7797
61	0.6313	-0.4333	-0.8416	-0.2841	-0.9852	-0.3179	1.1429	-0.0305	-0.3545	0.7703	0.7751	-0.5715	0.1082	0.3589	-0.7977
62	0.3149	0.7054	-0.1842	-0.9240	0.3599	-0.6701	-0.6442	-0.0186	0.5038	0.9320	-0.2238	1.1651	-0.0890	-0.4971	-0.0514
63	0.1542	0.6223	-0.2559	0.7282	0.9578	-0.2718	0.1176	0.0247	0.4809	-0.3548	-0.4013	-0.0718	0.1837	-0.1375	0.5031
64	0.1342	-1.0482	-0.7446	-0.7572	-0.1262	-0.1031	0.5615	-0.7095	0.6239	-0.7915	-0.1407	-0.6214	0.6084	-0.7840	0.2788
65	0.0706	0.3203	0.3936	0.0314	-0.2434	1.1195	0.7822	0.0547	0.0881	-0.2913	-0.5466	-2.2761	-0.8299	0.1017	0.4235
66	-0.6868	-0.4824	0.0945	0.7384	0.2812	0.6067	1.2006	-0.8679	0.6877	-0.5986	0.6090	-1.1681	0.2253	0.7171	-0.1666
67	-0.1802	0.5951	1.0482	-0.1452	-0.9038	0.5593	-1.1018	-0.8302	0.8854	0.0130	-0.9071	0.5670	0.5655	0.2430	-0.4999
68	-1.0949	0.5477	0.4209	-0.2087	0.1831	1.0639	0.2239	-0.2322	0.2694	-0.3234	-0.2419	-0.2556	-1.2777	0.1354	0.0904
69	-0.1860	0.6693	-0.0393	-1.0788	-0.5529	-0.6443	-0.4166	-0.2526	0.9198	0.1559	-0.1830	-0.4682	0.8364	-0.7667	0.2913
70	-0.2546	0.1049	-0.5358	-0.2344	-0.5194	0.3730	-0.2593	-0.2694	0.4981	-0.0007	0.8927	-0.6767	-0.7709	0.6702	-1.0075
71	0.4545	-0.4162	-0.8587	-0.0139	0.1835	-0.4251	0.0129	-0.2502	0.4450	-0.9473	-0.0485	0.9204	0.5349	0.7760	-0.5641
72	0.5469	-0.3858	-0.3656	-0.6070	-0.4204	0.3613	0.6570	-0.6216	0.7343	0.6313	0.6375	0.5525	1.0746	0.1788	-0.5900
73	-2.8742	0.8021	1.1616	-1.6527	-1.7103	2.0099	-0.1894	0.5918	-1.2715	0.1160	-1.0874	-0.1920	-1.0758	1.5230	-1.4811
74	1.0230	1.0593	0.4945	-0.1492	-0.7611	-0.2950	-0.9175	0.7468	-0.9157	0.0369	-0.7229	-1.6893	0.2410	0.8606	0.6123
75	-0.2142	0.8231	-0.6786	0.7837	0.9786	0.7174	0.2388	0.6106	0.4512	0.7469	-0.0821	-0.2726	-0.3230	-0.5108	1.0667
76	-0.7305	-0.3089	-0.2773	-0.6312	0.5736	-0.2616	0.3300	-0.3483	-0.6300	-0.1555	0.5303	0.2093	-0.5939	-0.7003	-0.7986
77	-1.3940	0.0241	1.0237	-0.5468	0.3811	-0.4897	-0.3805	0.0277	0.4666	-0.5121	0.5184	0.0751	-0.3325	-0.4296	-0.2603
78	0.6554	0.8961	-1.0534	-0.2986	-1.1671	-0.3056	0.4595	-0.8860	-0.9116	-0.8398	0.8069	-0.4340	-0.1772	0.6862	0.4803
79	1.1476	0.6122	-0.1517	1.0679	0.4496	-0.4615	-0.3540	0.4537	-0.7068	-0.3209	-0.0612	0.8290	-0.2680	-1.0014	0.3114
80	0.5501	-0.4743	0.7621	-0.8940	-0.8163	-0.4522	-0.7587	0.9693	0.9298	-0.7205	0.4711	0.0337	0.4158	-0.1574	0.6716

	76	77	78	79	80	81	82	83	84	85	86	87	88	89	90
81	-0.7646	0.4990	0.3727	0.0560	0.0997	0.8508	0.4832	-0.0469	0.8305	-0.2629	-0.0479	0.5392	0.3491	0.6496	0.2117
82	0.4004	0.0784	0.0711	0.7781	0.5605	0.8405	0.8534	-0.5491	0.4860	0.5415	0.3441	-0.9151	-0.1878	-0.6851	0.5710
83	0.4290	-0.2879	0.4569	0.5535	0.5383	-1.0879	-1.1691	-0.6423	0.0411	0.7132	-1.1254	-0.1954	-0.0611	-0.6388	-0.2172
84	-0.6908	-0.9648	-0.1346	-0.7178	-0.0254	0.9401	0.8283	-0.5979	-0.2264	0.2569	1.1105	0.9811	-0.8679	-0.9814	0.2852
85	-0.2761	-0.8650	-1.3777	-0.8572	-0.1491	0.2412	-0.3363	-0.7975	0.4627	0.2289	1.1712	-0.5430	-0.3701	0.2799	0.3238
86	-0.9142	0.7630	-0.4034	-0.9513	0.7304	-0.7043	0.2451	0.6369	-0.9693	0.8255	0.3405	-1.3044	0.3464	0.8583	0.2314
87	0.3012	-1.1607	-0.6777	-0.3893	0.2185	0.1785	-0.6054	0.1194	0.5191	0.8214	0.7921	-1.2620	-0.0401	0.8983	0.9375
88	0.9278	0.8010	-0.2101	0.5336	1.2116	-0.0407	-0.4161	0.5579	0.4427	-0.3648	0.5833	-0.9738	-0.1138	-0.5013	0.8196
89	-0.4680	0.9629	-0.4103	0.2532	2.0597	-0.4104	0.6928	-1.5459	0.7793	0.3549	-0.5904	-0.7910	0.3580	-0.8929	-0.6150
90	0.2659	-0.9500	-0.8215	-0.3477	0.2140	0.4155	-0.2897	0.1833	-0.3401	0.7689	1.1553	0.7974	0.7628	0.5260	0.7553
91	-0.2985	-0.4959	-0.3450	0.2677	-0.3680	0.7467	-0.0922	-0.2245	0.7782	-0.2995	0.0302	-0.3096	0.9153	-0.6845	0.2844
92	-0.3204	-0.2975	0.4935	-0.3289	0.8458	0.8573	-0.4310	-0.3962	0.2807	0.3609	0.8450	0.0270	0.3330	0.8527	-1.0524
93	-1.0825	-0.0398	-1.1736	1.3971	1.8559	-0.3209	0.0191	0.9373	0.7262	-0.0821	-0.2157	0.0950	-0.5960	0.3554	-0.4704
94	0.1116	0.7376	0.2892	-0.4238	0.2168	0.6877	1.1120	0.8767	-0.2004	-0.5139	-0.1421	0.6521	0.7441	0.2200	-0.3328
95	0.8059	-0.3434	0.4083	-0.3340	-0.6708	-0.2890	-0.4091	0.2723	0.6113	-0.5683	0.1609	-0.9184	-0.1777	-0.8659	-0.0869
96	0.8784	-0.3124	-1.1853	-0.7416	-0.2031	0.7806	1.0483	2.7605	1.2315	0.9161	-0.5420	-0.3299	-0.2125	1.7601	-0.7153
97	-0.5951	0.2451	0.6924	0.7091	0.3136	-0.8855	0.8943	1.0330	0.2944	-0.0665	-0.7099	-1.0259	-0.7756	-0.7485	0.5865
98	-0.4931	0.7142	-0.9122	0.1593	0.3581	0.5696	-1.0823	-0.0231	-1.1260	-0.5641	0.2273	-0.6695	-0.8192	-0.9244	0.3393
99	0.0127	-1.1065	0.6479	-0.8557	0.9657	0.5363	0.0179	0.4614	0.2887	-0.6332	0.1326	-0.3283	0.1666	-0.8777	-0.3213
100	-0.5023	-1.9944	0.7334	-0.5966	-0.5444	-0.5279	-0.0676	-0.6334	0.3972	0.5663	0.0695	1.1731	-0.6159	0.9283	0.1234

Tabela D9. Valores dos pesos das conexões entre a primeira camada oculta (120 neurônios) e a segunda camada oculta (100 neurônios) da rede *Magola11*. Coluna 91 a 105. Matriz de pesos LW1.

	91	92	93	94	95	96	97	98	99	100	101	102	103	104	105
1	0.4557	-0.1476	-0.1278	1.0515	0.1859	0.3594	0.3610	-0.7672	-0.0907	-0.4680	-0.1893	0.9797	-0.6815	0.6503	0.1921
2	-0.8707	-0.6457	0.6325	0.6875	-0.0658	-0.2775	0.5071	0.5831	0.6988	0.5795	-0.4759	-0.3282	0.6513	-0.4121	-0.0676
3	0.7800	-0.6574	-0.1162	0.0601	-0.0663	-0.2928	0.3845	0.3146	-0.2678	-1.1319	-0.9094	0.7577	-0.5226	-0.1260	0.3888
4	0.4892	0.5455	-0.1298	-1.0703	0.2674	0.7071	-0.2089	-0.2839	-0.8482	-0.0880	1.1011	0.6472	-0.9664	0.8241	0.1525
5	0.3636	-0.1561	-0.9062	-0.3557	0.6161	-0.0326	0.9358	-0.7541	0.7498	0.2543	-0.1982	0.7025	0.9735	0.8997	0.5604
6	0.5543	-0.7736	-0.1817	-0.4927	0.3612	-0.6884	-0.4986	1.1496	0.3891	-1.0032	-0.4018	-0.3507	0.5225	0.3439	0.0884
7	0.1208	-0.4987	-0.5846	0.0474	-0.0811	0.3191	-0.4255	-0.6787	-0.3599	0.1191	-0.4382	0.7460	0.1832	0.2265	0.7253
8	-0.7756	0.4996	0.6100	1.0657	-0.4622	0.7456	0.0820	0.3626	0.0056	-0.0428	0.5644	1.2107	0.3896	-0.6823	-0.3094
9	0.0829	-0.8812	0.6491	-0.3904	0.2544	-0.8576	-0.4527	0.0594	0.7717	-0.1231	-0.5336	0.6376	0.3437	-0.3276	0.1836
10	0.2990	-1.0437	0.9575	-0.4003	-0.7568	-0.2127	0.1478	0.3571	0.2676	-0.7339	0.5777	0.0977	-1.0796	-0.5922	0.7497
11	0.2208	0.6276	0.9292	-1.2299	-0.1526	-1.2547	0.4029	-1.2208	-0.5680	-0.2709	0.6379	0.1402	-0.3764	-0.6675	0.0809
12	-2.3430	0.4879	-1.0206	-0.9026	-1.4554	-1.1357	-0.0997	1.1034	-0.5823	-0.1055	-0.7312	-0.1311	0.3895	1.9379	0.9175
13	0.0561	0.5018	-0.4527	-0.6009	-0.8363	0.4738	0.7462	-0.6927	-0.0288	-0.7099	-0.1810	0.1127	-0.8964	-0.7467	0.8681
14	-0.0360	-0.2928	0.3604	-0.7485	-0.7641	-0.7063	0.3936	-0.3483	0.5710	0.1646	-1.0571	-0.5836	-0.2416	1.1349	0.2331
15	-0.0409	0.7558	-0.1792	-0.2569	-0.7911	-0.2496	1.1746	-1.1380	-0.8302	-0.5893	-0.4720	-0.2148	-0.7575	-0.5243	0.3050
16	-0.5415	-0.0764	0.2491	0.1199	-0.0568	0.2390	-0.7597	0.1534	-0.1642	0.3434	-0.0959	0.4285	0.7569	-0.3216	-0.8814
17	0.3392	0.3928	-0.7210	-0.6934	0.7746	-0.3211	-0.4511	0.2224	-0.7655	0.4732	0.3685	-0.9008	0.5805	-0.8309	-0.4997
18	-0.0902	0.3717	0.0366	0.1490	-0.8381	0.9438	0.5401	0.7589	1.2438	-0.5157	-1.3696	0.3623	-0.5130	0.3370	0.4528

	91	92	93	94	95	96	97	98	99	100	101	102	103	104	105
19	0.7428	0.2342	-0.3890	0.2297	0.5926	0.3642	0.3915	0.3567	-0.8417	-0.0952	1.3889	-0.7786	2.0496	1.7690	0.6650
20	0.3182	0.7715	0.3421	0.1294	0.1441	0.2539	0.4729	0.5166	0.6472	-0.5425	0.2278	-0.4442	-0.0273	0.6054	-0.9184
21	-0.5294	0.1505	-0.8517	0.7006	-0.0718	-0.6085	-0.8925	0.2830	0.7835	0.0010	-1.0084	-0.9704	-0.0979	0.5306	-0.8222
22	-0.9197	-0.8527	-0.4884	-0.2472	-0.4459	0.6855	-0.2570	-0.0955	-0.1578	-0.6524	0.1581	-0.5795	-0.3551	0.9005	-0.1586
23	0.6158	0.2746	0.9378	-0.1033	-0.6618	0.5660	-0.9810	-0.9810	-0.7489	0.5962	-0.4409	-0.6506	-1.3363	-0.1124	-0.1081
24	-0.5076	0.4369	0.2677	-0.1047	-0.4501	-0.1867	-0.7885	0.8978	-0.2799	0.1483	-0.2932	-0.2097	-0.7423	0.3677	-0.6387
25	0.7268	-0.1396	-0.9438	0.3893	-0.8298	0.7760	1.2177	-0.8675	-0.6063	-0.3599	-0.0766	-0.1977	0.8417	0.5069	-0.5033
26	0.0673	-0.0221	0.9544	-1.0734	0.7208	0.7828	0.8468	0.3781	0.5516	-0.4582	0.9039	-0.3362	-1.0049	0.1835	0.8910
27	1.0926	-0.2660	-0.7980	0.2121	0.2877	-0.2251	-0.6754	0.1939	0.5797	0.6089	0.3382	0.8647	1.0155	-0.2075	-0.7750
28	0.7751	-0.4466	0.3940	0.6959	0.9103	-1.6346	-0.2036	0.5481	-0.7063	-0.9607	-1.5677	-0.7689	-0.0685	0.0526	0.0558
29	0.7959	0.9561	0.8331	0.8896	-0.5439	0.3544	0.0137	0.2366	-0.4930	-1.0128	-0.5688	-0.0805	0.2488	0.4502	0.4347
30	-0.7573	0.0840	-0.8214	-0.2746	0.1718	0.1577	0.5108	0.2537	-0.2893	-0.3521	-0.8574	0.2082	1.1774	-0.6767	0.7021
31	0.1759	-1.2313	0.8650	0.7581	-0.1898	-0.7687	-0.3276	-0.4373	-0.8138	0.2410	-0.8676	0.6120	3.2287	-1.9029	1.0239
32	-0.7137	0.0914	-0.3619	0.2029	-0.6074	-0.2146	-0.3992	0.6983	0.6998	0.1767	-0.1860	0.5116	-0.4632	0.7077	-0.6761
33	0.4681	0.3298	1.0714	0.4989	0.5297	-0.7001	0.4874	-0.7549	0.3297	-0.1841	0.9796	1.0962	1.0289	-0.6423	1.2777
34	-0.4612	0.8972	-0.9385	-0.7510	1.2461	-0.5386	-0.8428	-0.4285	-0.6081	-0.2014	0.2169	0.0016	0.9486	-0.2474	0.1077
35	-0.2683	-1.1608	-0.9445	0.0282	-0.9279	0.1534	0.9448	-0.1987	0.3020	-0.9527	-0.0304	0.3421	-0.2397	-0.3150	-0.3079
36	-0.0589	0.2158	0.9480	-0.7717	-0.1488	-0.3505	-0.3280	0.5425	0.5877	0.0131	-0.1220	0.4552	0.7644	-0.3274	-0.1602
37	-0.2061	-0.4785	-0.4726	-0.2268	-0.7772	-1.0719	-0.0956	-0.7964	0.5323	-0.7229	0.0825	-0.1750	1.4109	0.4690	0.5256
38	-0.9050	-0.8239	0.7093	-1.0534	0.2308	-0.8894	-0.7951	0.2811	-0.4689	0.2628	-0.7295	0.3070	0.1389	0.7196	0.1347
39	0.8089	0.5530	-0.8761	-0.0829	-0.3437	-0.4792	-0.3130	0.7343	0.0896	-0.3983	-0.3811	0.6033	0.0061	-0.1679	0.2494
40	0.2774	-0.3071	0.4890	0.4301	0.6940	-0.6353	-0.0079	-0.2617	-0.5432	0.9212	0.1826	-0.5743	0.6777	-0.2555	0.0931
41	-0.2904	-0.2617	0.2446	0.9420	0.0032	-0.1223	1.1031	0.2245	0.5018	-0.9996	1.0186	0.7299	-0.1075	-0.0242	-0.1574
42	-0.1562	0.6302	0.7382	0.6803	0.6509	0.4142	-0.5206	-0.1224	-0.1245	0.0991	0.4574	-0.0307	0.5800	-0.6086	-1.2181
43	0.6905	-0.0522	-0.5922	-0.8781	-1.0242	2.0924	-0.1845	1.0730	0.7035	-0.3494	-0.7042	0.8697	1.7975	1.3420	1.4706
44	-0.3080	-0.2060	-0.0317	-0.4672	0.5122	-1.1818	-0.3265	-0.6391	-0.1383	-0.9188	0.7747	0.5431	-0.4434	-0.2360	-1.4389
45	0.7206	1.1014	0.2748	0.5413	0.2704	-0.3647	-0.6370	-0.1015	-0.7450	0.3941	0.2260	-0.7547	0.6908	0.6884	-0.3974
46	-0.1313	-0.6272	0.4131	-0.6541	0.3557	-0.6772	-0.4752	0.8892	0.1920	0.9900	0.7735	0.0127	0.1040	0.2232	0.0735
47	1.0784	0.1693	-0.1040	0.6474	-0.0963	-1.3996	0.0750	0.1057	-0.6084	0.4703	-1.1234	-0.4428	1.2104	0.1391	-0.8191
48	-0.9744	-0.1845	0.6617	-0.3284	-1.0391	-0.6402	-0.5353	0.6646	0.2815	0.7443	0.5774	0.7377	-0.4316	1.1079	-0.9819
49	-0.4196	-0.6975	-0.8289	-0.9193	0.8641	-0.5970	-1.0063	0.2403	-0.7396	-0.8058	0.8918	0.0151	-0.6737	-0.6935	-0.0381
50	1.4135	-0.5929	-0.5135	0.3738	-0.9812	0.4687	-1.1316	0.7043	-0.2586	-0.3666	-0.2179	-0.8032	-0.2795	-0.0293	-0.7898
51	0.9537	0.2374	0.3755	0.6911	-1.0958	1.5464	0.8712	-0.3630	-0.5998	0.4906	-1.8473	-0.5683	-0.9971	1.2576	-0.2488
52	-0.8582	-0.6028	-0.3498	1.0233	0.7919	0.8335	0.6764	-0.3604	-0.7859	0.2645	-0.7580	0.7315	0.6754	-0.4288	0.5873
53	0.9003	-0.1568	-0.2867	0.3867	0.8156	-0.2445	-0.7625	0.4203	0.2345	0.6212	-0.0207	-0.9393	-0.5043	-0.3336	0.6808
54	-0.4471	-0.3251	0.0792	-0.1045	-0.5910	-0.7725	0.2384	0.7075	-0.3032	-1.0069	-0.6867	-0.1641	0.1817	0.7440	1.0287
55	-0.4197	-0.9097	0.0086	-0.6299	-0.3553	-0.5331	-0.9039	0.7637	0.1444	-0.4859	-0.6376	0.1185	0.5181	0.0104	-0.1956
56	0.4911	0.1542	0.2925	0.8862	1.0759	-0.9396	0.5676	-0.2327	-0.1219	-1.0887	1.5799	0.1444	0.1394	0.8629	-0.0139
57	0.4076	0.1525	-0.4557	0.1165	0.1595	0.5082	-0.6044	-0.0166	0.7693	0.5619	0.8270	-0.4332	0.4470	-0.1783	-0.3394
58	-0.9658	-0.8042	-0.4877	-0.5438	-0.1114	-1.0384	0.9204	-0.0091	-0.0577	-0.7500	0.3920	-1.3165	0.2309	0.0521	0.7101
59	0.5028	-0.7862	-0.9127	-0.6424	-0.9119	0.0561	0.6614	-0.0525	-0.8209	-0.7291	-0.0759	0.5350	0.1441	-0.9254	-1.0554
60	0.1817	0.7231	-1.3926	0.4745	-0.0422	-1.7670	1.4745	0.1071	0.1132	-0.4424	-0.1456	-0.7736	-1.2250	-0.1670	-0.1431
61	-0.7699	-0.1607	1.1885	0.8053	0.7121	-0.0867	-1.0231	-0.0888	0.2652	0.6015	0.2412	0.9134	-0.3080	-0.4595	0.8120
62	-1.1341	-0.7054	-1.3146	0.3110	-0.8158	-0.9873	0.1880	-0.4084	-0.3405	-1.2043	-0.6997	0.0347	-1.4719	-0.9059	-0.1690
63	-0.5406	-0.0422	-0.8179	0.6572	0.7375	0.1906	0.6534	0.7151	0.6986	0.4882	-0.8960	-0.0976	0.5188	0.0440	-1.0582
64	0.1023	0.2538	0.6472	-0.6583	0.6169	0.9738	-0.4948	-0.6530	0.2892	-0.0273	0.8249	-0.5479	-0.6206	0.4258	0.2032

	91	92	93	94	95	96	97	98	99	100	101	102	103	104	105
65	-0.4447	0.3139	-0.1910	0.4935	0.2443	-0.7103	-0.9845	0.6079	-0.2897	-0.5957	0.5993	0.6279	0.3108	-0.8779	1.0026
66	-0.4439	0.4448	0.5123	0.8549	0.8672	0.1323	0.4770	0.2222	0.0660	1.2061	0.3126	0.7071	0.1177	-0.4632	-0.0963
67	-0.2692	0.9643	0.3890	0.1681	0.1786	-0.5952	0.1586	0.2527	-0.1114	-0.5581	0.3306	0.0114	-0.7787	0.8693	-0.3994
68	-0.2134	-0.1216	-0.0048	-0.3504	0.4430	-0.3009	0.7533	0.6547	1.0205	0.6139	0.3507	-0.0429	-0.3300	0.7262	0.3618
69	0.3138	0.5246	-0.1373	0.8203	-0.4607	-0.1457	-0.7956	-0.9574	0.9559	0.5973	-0.7659	0.3056	0.0944	-0.2343	-0.2567
70	-0.9372	0.3054	-0.0054	-0.1771	-0.9187	0.8092	-0.5522	0.9787	-0.9383	0.2751	-0.2695	-0.2766	0.3799	0.1572	0.3808
71	-0.4341	0.1576	0.3172	1.0026	-0.3768	-0.1748	1.0246	-0.5754	-0.3256	-0.7776	-0.0162	0.9500	0.7412	0.0947	-1.0493
72	-0.4658	0.0777	-0.3579	-0.9845	0.1872	0.1313	-0.2097	0.4077	0.6249	-0.2342	-0.0692	-0.9641	0.5995	0.5147	-0.9633
73	0.2042	-0.6462	-0.8867	0.8256	-2.9632	3.0224	0.0851	0.7746	1.1949	0.6554	-0.1741	0.4715	5.0304	2.5132	-2.1143
74	-0.1688	0.3342	0.9343	-0.8173	0.9661	-0.2581	1.0242	-0.9208	-0.1973	-0.1073	0.7841	-0.0194	0.6003	-0.9846	1.1808
75	-0.6829	0.4786	0.0060	0.2951	0.9750	-0.1779	-0.5236	0.1186	0.3540	-0.3609	-0.1776	-0.4976	0.3941	-0.1628	-0.0355
76	-0.4877	0.5616	0.1657	-0.7644	0.5303	-0.4964	-0.2315	0.3673	0.4665	-0.1089	0.9752	-0.8676	0.7516	0.5025	0.5063
77	0.3269	-1.2038	-0.1485	-0.7663	-1.4606	0.5090	-0.7023	-1.0239	-0.6571	-1.0295	-0.0427	-0.3188	0.7032	1.6294	-0.0942
78	0.3424	0.2415	0.2869	0.6553	0.4036	0.1537	0.5827	-0.5825	0.2239	0.0187	1.3359	0.6240	-0.1597	-0.3819	0.7379
79	-0.1362	-0.4524	-0.3272	0.5002	1.1855	0.1118	0.3162	-0.4259	-0.6328	0.2442	-0.2165	0.1668	-1.6203	-0.6146	0.1064
80	0.6756	-0.0201	0.4921	0.3028	-0.6191	0.3164	-0.1334	-0.0866	-1.1113	0.5855	-0.9677	-0.7068	0.4718	-0.0659	-0.7967
81	-0.3050	0.0361	0.9694	-0.5414	-1.4014	0.1640	-0.0218	-0.3234	-0.6549	0.6271	-1.2837	-0.3917	0.2523	1.1954	-0.1850
82	0.7381	0.2548	0.3544	-0.8154	0.7358	-0.6018	-0.3915	0.0451	-0.0480	0.2694	-0.6955	0.8459	0.0576	-0.4218	0.4822
83	-0.2432	-0.5719	-0.9644	-0.9363	0.3765	-0.7146	-0.5494	1.6424	0.5744	-0.9554	0.0383	-0.7003	0.6236	0.3548	0.9846
84	0.6658	-0.7407	-0.9049	0.0867	-0.8593	-0.8325	-0.6189	-0.9629	-0.3271	0.8417	0.1728	0.1857	-0.5389	0.0111	0.8277
85	0.6391	-0.9321	0.5506	-0.3320	-0.2758	0.1929	0.8807	-0.5421	-0.7851	1.1565	-0.6261	0.6760	-1.2688	-0.7261	-0.4606
86	-0.1469	-0.1189	-0.4151	0.0525	0.4321	-0.8265	0.5536	0.0482	-0.6191	-0.6782	0.2768	-1.1252	0.0592	-0.9502	0.0310
87	0.5445	0.5343	-0.1929	0.2385	-0.9795	0.0952	0.2591	0.7717	0.4681	-1.0719	-1.5229	-0.1410	-0.5552	0.9880	0.1641
88	-0.2363	-0.0959	-0.9190	-0.2066	-0.8363	0.3885	-0.2710	-0.6287	0.0666	1.0933	0.1986	-0.5801	-0.9907	-0.7268	0.2698
89	0.5289	0.7204	-0.6206	-0.7590	-0.3182	-1.6332	0.2935	0.6617	0.4961	-0.1718	0.4904	0.3786	-0.5546	1.1229	1.2113
90	-0.2716	-1.0399	0.5806	-1.1602	-0.8256	0.0426	-0.5899	-0.1919	-0.8886	0.6163	0.7620	-0.8946	0.2942	0.6735	0.3318
91	0.8727	-0.1435	-0.1099	1.0064	-0.6550	0.5334	-0.8323	-0.8288	0.1461	-0.3034	1.0501	-0.1089	-0.6675	0.0754	0.8466
92	-0.0251	-0.2720	-0.1673	-0.8522	0.0336	0.6780	-0.4871	-0.7316	0.7771	0.1893	-1.1435	0.5931	0.3170	0.9070	-0.0415
93	-0.9861	-0.8033	-0.7361	0.7080	-1.2612	0.6024	-0.1635	0.1543	-0.3338	-0.8578	-1.5217	0.1326	1.5964	0.0239	0.1156
94	-0.4708	0.9572	0.5785	0.9120	0.1756	0.8435	-0.6989	-0.6946	-0.7210	-0.4316	1.1580	-0.1642	0.8638	0.5013	-0.7792
95	-0.8186	-0.6282	-0.9329	0.1400	-0.0814	-0.7139	0.8415	-0.5580	0.1322	-0.6907	0.1364	0.3381	0.8314	-0.4817	-0.8132
96	-1.1286	-0.2884	0.3030	-0.1206	0.7451	-0.3107	-1.2777	0.5034	0.4588	-0.4217	0.4853	-0.7229	0.9023	0.1700	-0.0566
97	0.1902	-0.3129	0.5615	-0.7808	1.3696	-0.6542	-0.3651	-0.1198	-1.0187	0.2766	0.1356	-0.3219	-1.3572	-1.1523	0.0495
98	0.8067	0.3806	0.0157	-0.6666	-0.4484	0.0992	-0.6637	0.5053	1.2994	-1.0533	0.0079	-0.5641	-0.6624	-0.2909	0.6739
99	-0.4969	-0.8554	-0.8158	0.4979	-0.4831	0.7530	-0.5932	-0.3164	0.1142	0.8348	0.1779	-0.1650	-0.2059	0.3209	0.3228
100	-0.1265	-0.8038	0.7445	0.6622	0.8211	0.5954	1.1271	0.2306	0.6875	0.5131	-0.0779	-0.4305	0.5282	0.0918	0.9364

Tabela D10. Valores dos pesos das conexões entre a primeira camada oculta (120 neurônios) e a segunda camada oculta (100 neurônios) da rede *Magola11*. Coluna 106 a 120. Matriz de pesos LW1.

	106	107	108	109	110	111	112	113	114	115	116	117	118	119	120
1	-0.9366	-0.9546	0.4058	0.8315	-0.4885	0.1729	0.6960	-0.7635	-0.1208	-1.1967	-0.3127	0.6855	-0.8584	0.9272	0.0385
2	-0.8643	0.1498	-0.0353	0.9456	0.7133	0.6322	-0.7773	0.5607	0.1888	-0.4512	-0.5852	-0.4369	0.9849	-0.8837	0.0356

	106	107	108	109	110	111	112	113	114	115	116	117	118	119	120
3	-0.5788	0.7370	-0.0589	0.3640	0.8546	-0.3144	0.7787	0.8148	0.8064	-0.0590	0.6421	-0.2694	0.4815	0.8591	-0.6553
4	0.1143	-0.7832	0.1228	1.0506	-0.4088	-0.1355	0.6081	-0.0987	-0.7732	-0.8850	-0.7769	0.1422	-0.1602	-2.5721	-0.8939
5	-0.5815	0.1428	-0.8067	-0.1891	-0.5258	-0.1071	-0.7995	-0.6100	-0.0961	-0.2232	0.8564	-0.6664	0.8709	0.7126	0.3691
6	-0.1766	0.8601	-0.4632	-0.0547	-0.3909	-0.3078	-0.4022	-0.4107	-0.0065	-0.5126	-0.7771	0.3676	0.7428	0.6998	0.0143
7	0.5794	0.2893	0.8032	-0.7716	-0.1098	-0.0054	-0.9618	-0.8589	0.5404	0.3057	1.0714	0.3201	-0.2041	0.7229	0.4240
8	0.0185	-0.4292	0.3904	-0.1215	-0.2382	0.1839	0.5755	-0.8557	0.8562	-0.1760	0.0226	0.7284	-0.0838	-0.5047	0.3853
9	-0.0395	-0.3991	0.6987	0.6130	0.5800	-0.1088	-0.4938	-0.2087	0.4381	0.5346	0.1265	-0.2608	0.1144	-0.8843	-1.1208
10	1.8165	-0.0746	0.1329	0.9195	0.1087	-0.1332	-0.7628	-0.7478	-1.4805	1.1198	0.6518	-0.1122	-0.1967	0.7491	0.7033
11	0.2473	0.9166	-0.6729	0.0179	0.4649	0.3685	-0.7958	-0.8023	-0.2285	-0.3072	-0.1090	-1.3025	0.6347	-0.6325	-0.5849
12	0.0043	0.8629	0.7113	0.4596	-1.7928	-0.7077	0.0671	0.6565	-0.9763	-1.3855	-0.5860	0.2828	0.4673	0.7781	1.0223
13	-0.0454	-0.3233	1.0620	1.0650	-0.0653	-0.9881	-0.2199	-0.3656	-0.9383	0.7507	0.1179	-0.7393	-0.7363	-0.4488	0.4211
14	0.3621	-1.2493	0.3384	-0.9637	0.4078	-0.8633	0.0230	0.5369	1.6108	-0.0054	0.2993	-1.2194	-0.3065	1.4973	0.5492
15	0.0662	1.0459	0.3302	-0.0687	0.9612	0.5616	-0.0883	-0.2723	0.2799	0.7349	0.7978	-0.4374	0.0148	0.3960	-1.1457
16	-0.2328	-1.0738	0.4443	-0.3737	-0.0313	0.1318	1.0957	0.6179	0.1176	-0.6646	-1.3884	0.2246	-0.4072	0.4928	0.6033
17	-0.5952	-0.6644	0.4419	0.5345	0.6585	-0.3396	-0.2672	0.1277	0.4378	-0.3244	-0.2051	-0.8440	-0.1387	-0.0441	-0.3901
18	-0.2430	0.7270	-0.3121	0.4229	-0.5928	-0.4709	-0.5232	-0.5988	-0.2286	0.6827	0.0251	0.4156	-0.3538	-0.3296	0.2089
19	0.2400	0.6911	-0.5690	0.9881	0.4078	-0.2908	-1.5350	0.3374	1.0164	-0.3385	-1.1740	2.0776	-0.5399	-1.1153	2.1027
20	0.7645	-0.5433	-0.1328	-0.6096	0.3352	0.6221	-0.5769	-0.3389	-0.8068	-0.4588	0.3671	0.6007	0.1027	0.1951	0.4291
21	0.3778	-0.7205	-0.4278	0.2319	0.9134	0.8121	0.2364	-0.0548	-0.7062	-0.9594	-0.3574	-0.9322	-0.4114	-0.5746	-0.2319
22	0.2628	0.6249	-0.9129	0.0452	-0.5492	-0.7396	-0.8208	0.6557	-0.8155	-0.5173	0.4088	-0.5630	0.9984	-0.4595	-0.5204
23	0.5566	-0.2242	0.7378	-0.2611	-0.6250	-0.0015	-0.2636	0.7266	0.0491	1.3330	-0.1135	0.1823	0.4708	0.8727	0.2893
24	0.6886	-0.8911	-0.4155	-0.0238	-0.5443	0.8704	-0.2821	0.2452	-0.2606	-0.0897	-0.4894	0.8593	-0.0043	0.6444	0.3154
25	0.0894	-0.6890	0.2215	-1.0146	0.1131	-0.2226	-1.1031	0.3649	0.5617	0.6491	-0.8624	-0.8413	0.9154	0.9533	-0.5328
26	0.7357	-0.8925	-0.5622	0.4187	-1.1882	0.3511	-1.0481	0.4869	-0.5941	-0.4510	-0.8449	0.8848	-0.6836	0.6006	0.3891
27	-0.0373	0.1065	-0.7691	0.0628	-0.5729	0.8742	-0.6307	0.0148	-0.6707	-0.4167	-0.5950	0.4577	1.1408	0.1238	0.7516
28	-1.4592	-1.8202	-0.4834	-0.3965	-0.9254	0.6103	-1.1230	0.7719	0.3846	0.1175	1.5027	2.4236	0.1389	-0.1708	-1.9071
29	-0.7198	-0.4823	-0.4488	0.6583	-0.3673	0.0346	-0.2889	0.4609	-0.9205	1.1003	-0.6852	0.7796	0.0909	-0.1471	0.1392
30	-0.9287	1.2286	-0.5376	0.6660	0.9662	0.2779	-0.1231	0.3163	1.3995	-0.1360	-0.0586	-0.3703	-0.1020	-0.2075	0.6785
31	0.5971	-0.0338	-0.1290	0.9338	1.1143	0.9154	0.1008	-0.1274	-0.8234	0.2222	-0.8956	-0.0779	0.9945	-0.0664	2.2871
32	-0.3670	-0.1235	0.2474	0.3014	-0.7418	0.0354	-0.9804	-0.7093	0.3553	0.2578	-0.1972	-0.1410	0.2456	0.2602	0.6458
33	-0.7122	0.1551	0.0944	0.0531	-0.7030	-0.9191	-0.2268	-0.0479	0.3676	0.6786	-0.3339	0.2115	0.0628	0.5114	-1.1198
34	1.0116	-0.0089	-0.1781	0.4282	1.3435	0.8051	0.1634	-0.3370	-0.7823	-0.4776	0.2496	-1.3214	1.0030	-1.6604	1.3166
35	0.9471	0.3926	0.0392	0.8351	0.6962	0.5641	0.2900	0.8952	-0.6419	-0.1781	0.3119	0.4369	0.4483	-0.2988	-0.7197
36	-0.6200	0.9263	-0.7175	0.8759	0.2922	-0.3743	-0.5726	-0.1439	0.8671	0.1391	-0.1212	0.2172	0.6303	0.1006	0.1705
37	-1.2567	-0.3051	0.6632	-0.3971	0.0162	-1.1201	0.1217	0.5618	1.1315	0.7252	-0.7885	0.8005	-0.1931	1.0202	0.2179
38	0.5888	-0.4962	-0.5715	0.2578	0.2453	0.0935	0.7124	0.7370	-0.2520	-0.7118	-0.4636	0.2095	-0.5178	-0.9563	-0.2747
39	0.3159	0.2101	-0.8442	0.3278	0.8231	-0.0783	-0.1599	-0.7793	0.2548	-0.6999	-0.2312	-0.7917	0.4848	1.0559	-0.6374
40	0.4181	0.6892	0.6363	1.0996	0.1611	0.5051	-0.0984	0.1709	-0.1188	-1.1329	-0.5692	0.7516	-0.6471	0.6519	0.6896
41	0.8465	-0.7754	0.5312	0.8751	-0.3266	-0.4459	-0.6818	0.8280	-0.9562	-0.5165	-0.3708	0.8606	-0.5493	0.5727	-1.2459
42	0.7400	-1.1035	-0.1580	0.8226	0.8533	-1.1063	0.6659	-0.8998	0.7738	0.6442	0.5305	0.4389	-0.6783	0.0488	0.6820
43	-0.0121	-0.3493	-0.1361	0.4114	0.0607	-0.7710	0.6800	-0.6323	-0.3922	1.1466	-0.6082	1.7836	-0.8593	0.3389	0.4240
44	-1.7894	-0.9673	0.5005	-0.9463	-0.1786	-0.5033	0.7562	0.2900	0.9950	-0.4149	-0.3931	-0.9490	-0.0302	-1.5473	0.7496
45	0.4156	0.8853	-0.2162	0.0212	-0.0009	0.8191	-0.6689	0.6420	-0.0413	-0.8969	0.4698	0.7804	-0.4265	-0.2055	-0.5045
46	-0.8245	0.7888	-0.1169	0.6522	0.7597	-0.3790	-0.7155	0.9793	0.7319	-0.0730	-0.9811	0.5822	-0.4910	-0.1353	0.4185
47	-1.0572	-0.4398	-0.4300	-1.0088	0.0482	0.5742	0.0311	-1.0119	-0.8035	-0.6404	1.6353	-0.5255	0.4409	0.2749	-0.1984
48	0.7287	0.6142	-1.1283	0.0761	0.3792	0.2735	-0.4041	-0.6674	0.5203	0.6811	-0.6262	-0.7352	-0.0541	-0.3040	-0.3951

	106	107	108	109	110	111	112	113	114	115	116	117	118	119	120
49	-0.3707	-0.5922	0.3813	0.2900	-0.6202	-0.8327	0.9820	-0.6726	0.4072	-0.0621	-0.0746	0.4726	-0.3542	0.4449	-0.3193
50	-0.3790	-1.0312	0.4766	-0.3539	-0.4366	-0.5628	0.9994	-0.0186	0.7642	0.4681	-1.1257	-0.1487	-0.3171	-0.3916	0.0066
51	0.8807	0.1433	-0.7416	-0.1930	-0.2966	0.2492	0.2769	-0.9240	0.0846	0.6032	-0.6502	0.1870	-0.2670	0.4795	0.0200
52	0.3269	0.2445	-0.9527	0.4058	0.6743	-0.1815	-0.4004	-0.0821	1.2051	0.4428	-0.5911	0.1236	0.7393	0.6795	0.8551
53	0.7811	-0.7785	-0.2497	0.2625	-0.3783	0.1912	0.5814	0.4496	0.3213	-0.2971	-0.6156	0.1782	-0.9638	-0.6821	-0.4563
54	0.8417	0.2885	-0.4622	-0.5915	0.0164	0.3008	0.4956	-0.4497	1.0735	0.4931	0.9264	-0.8557	-0.0998	-0.8065	0.6569
55	0.4411	0.0592	0.8693	-0.1611	0.1624	-0.2101	0.5383	0.8582	-0.4706	0.5197	0.3785	0.9473	0.0075	-0.5738	-0.2296
56	-0.6577	0.7398	0.6091	-0.7796	0.8721	0.1498	-0.2025	0.8598	0.5960	-0.2829	0.1590	0.4072	0.6898	-0.6651	-0.9930
57	-0.3307	0.7650	-1.2341	-0.0829	0.7164	-1.1302	0.8795	-0.6846	0.7288	0.0549	-0.4067	-0.3707	0.8569	-0.6819	0.6344
58	-0.5552	-0.8182	-0.0794	0.3933	0.4513	0.0039	0.6988	0.6522	-0.1299	0.6781	0.1514	-0.5597	0.7204	-0.7310	-0.4130
59	0.7806	0.9463	-0.3713	-0.1770	-0.7808	-0.1286	1.0431	0.5284	0.2115	0.1145	0.1671	0.9728	0.6289	-0.2691	1.8670
60	0.2507	-0.6918	-0.7046	-0.7084	-0.1621	0.4170	-0.5006	0.0343	-1.0806	1.3474	0.0985	1.2130	-0.8110	0.2839	-0.8457
61	0.4517	-0.7362	-0.8480	0.8288	-0.1880	0.0002	1.1548	0.1500	-0.0334	-0.0634	0.3362	0.8689	0.7478	-0.5393	0.3715
62	-0.3366	-2.7066	0.9818	0.4246	-0.2471	0.6292	0.3456	-0.3497	-1.5864	-0.3468	-0.5755	1.0627	-0.0333	0.1751	-0.1104
63	0.9351	0.5378	0.0736	0.5877	-0.8581	-0.3408	0.5523	0.8897	0.4662	0.5497	0.1791	0.1794	0.0015	-0.4781	0.8197
64	0.1200	-0.4297	-0.3374	-0.3974	-0.6067	-0.0237	-0.0795	0.0865	0.0888	0.8173	-0.8258	0.1480	-0.7379	0.2301	0.0045
65	0.1781	0.9494	-0.5361	-0.1835	0.8346	0.7909	1.0074	-1.3971	0.2159	-0.7308	0.8527	-0.8495	0.1137	-0.9466	0.9897
66	-0.8547	0.6767	0.5234	1.1852	0.6910	0.5190	0.1676	-0.5552	-0.9060	-0.1025	-0.7218	0.3930	-0.1595	-0.1822	0.8315
67	-0.1670	0.7350	-0.2389	0.6281	-0.5258	0.6674	-0.7870	-0.4367	0.6517	0.2115	-0.0748	-0.8467	0.0075	-0.2039	-0.1867
68	-0.3311	-0.2229	-1.1178	-0.4739	-0.7220	-0.7890	-0.9780	-1.3894	0.2940	0.7611	0.5110	0.9253	0.3656	-0.1800	-0.6909
69	0.0824	0.0793	0.3750	-0.2307	0.9425	0.5183	0.6948	-0.8963	0.8136	0.0489	0.1584	0.6479	0.6353	0.5905	-0.7921
70	-0.3056	-0.3207	0.0487	-0.8441	0.7844	-0.1322	-0.8284	-0.7582	0.1838	0.0761	-0.9344	-0.1164	0.3735	-0.4393	-0.7622
71	-0.1936	0.0121	-0.0724	-0.1452	-0.4898	-0.8256	0.1306	-0.7800	0.9072	-0.5669	0.9043	-0.8519	-0.5360	0.2803	0.6909
72	-0.0745	-0.5367	0.2504	-0.6228	-0.6931	0.4797	0.2500	0.2046	-0.0509	0.8460	0.4256	0.8608	0.1025	-0.4993	0.2148
73	0.2754	-0.8671	0.8449	-0.5336	-1.8042	-0.7827	0.6209	-0.9150	0.2020	-0.6200	0.2067	-1.3675	-0.3832	4.7874	-0.2847
74	0.3310	0.0520	0.1284	-0.4120	-0.1410	1.0534	-0.3324	-0.4696	-0.8156	-0.6594	-0.7234	-0.6183	-0.7601	0.3575	0.5773
75	0.5363	-0.9833	1.0161	0.3798	-0.4802	0.9325	-0.0609	0.0722	0.0891	-0.8702	0.8291	0.2345	-0.1312	0.9514	0.7143
76	-0.4344	-0.8719	0.2699	0.4486	0.5343	-0.1116	0.4435	0.2799	-0.7111	0.0759	0.3583	0.6271	0.2834	-0.2111	-0.5756
77	0.6494	0.8634	0.6007	-0.7334	0.0470	-0.2198	-0.2851	0.6258	0.9446	0.3912	0.0184	-0.4558	0.5920	0.4667	-0.3631
78	-0.9415	-0.0896	0.8163	-0.6277	-0.7321	-0.7227	-0.1219	-0.8350	0.7985	0.0584	0.6862	0.5440	-0.8606	0.0476	-1.4106
79	-0.7301	1.1932	-0.0645	-0.4298	0.3730	-0.5495	0.5769	0.6426	0.1154	-1.0220	-0.2221	-1.8115	0.1583	-1.0937	0.2840
80	-0.1036	-0.5720	0.9869	0.0153	0.2494	0.2490	-0.4374	-0.2784	0.1894	-0.6724	-0.4981	-0.7944	0.1868	-0.5048	-1.0331
81	-0.8076	1.1962	-0.2009	-0.4921	-0.6490	-0.5927	-1.2036	0.2317	0.3098	1.0075	-0.4893	0.2529	-0.8096	0.6730	-0.8068
82	0.0120	0.6216	1.1600	0.8182	0.5167	-0.7255	-0.2042	0.6166	0.0575	-0.9081	-0.4719	0.7338	-1.0302	-0.7406	0.6063
83	-0.3913	-0.3906	0.6420	-0.8233	-0.5794	0.4007	0.2587	-0.7354	0.1912	0.7502	0.7756	-0.4856	0.0594	0.8573	-0.0503
84	-0.0252	-0.2676	0.2417	0.5831	-0.5504	-0.2113	0.0767	-0.2437	0.9291	-1.0051	0.4495	0.0379	-1.0292	-0.5716	-0.3189
85	-0.3065	-1.3956	0.0632	0.2309	-1.3233	0.1359	0.7515	-0.0246	-0.8949	0.5769	0.7200	-0.2136	0.2290	-1.0820	-1.0527
86	0.4065	0.9380	-0.5363	0.3552	0.3671	0.6870	0.3370	-0.0518	-0.8191	0.4889	0.3292	-0.3520	-0.3195	-0.3641	0.1113
87	-0.1362	0.2048	0.3778	0.3735	0.6328	0.1673	1.0841	-0.7975	-0.9277	-0.1212	0.1237	-0.1853	-0.4523	-0.4838	0.2757
88	0.9669	-0.1704	0.8413	0.6828	0.2818	1.0533	-0.4921	0.1870	0.1420	-0.2294	-0.9271	-0.1865	-1.0305	-0.7769	-0.7121
89	0.6391	0.0104	0.8083	0.7076	-0.5588	1.0570	-0.1390	0.3408	-0.9091	-0.4857	0.8344	0.0005	0.2727	-0.9653	-0.7566
90	0.4480	-0.0079	0.7205	0.4619	-0.8257	0.1703	0.5845	-0.1892	-0.5821	-0.6607	-0.8553	0.4531	-0.4789	-0.1582	-0.7874
91	-0.6832	0.1588	0.0066	-0.7951	-0.5618	0.3317	-0.2702	-0.8745	0.2250	0.3065	-0.8098	0.2778	0.6903	-0.0256	-0.6787
92	-0.6649	0.7035	-0.5080	-0.7908	-1.5695	0.7098	0.2959	-0.7125	0.1292	-1.4402	-0.8940	1.2789	0.8952	-0.8385	-0.6804
93	-0.7093	0.2558	-0.2867	0.1914	-0.8871	-0.2968	0.1195	-0.4651	-0.1459	0.6196	-0.7614	2.9462	0.4127	0.8305	0.3523
94	0.8830	0.9816	0.7855	0.2561	0.9010	0.1338	-0.3508	0.0625	0.6844	-0.4499	-1.0412	-0.4041	0.2516	1.0520	0.3490

	106	107	108	109	110	111	112	113	114	115	116	117	118	119	120
95	-0.8073	0.4775	0.1948	-0.0571	-0.4243	0.7401	-0.4671	0.5162	-0.2842	-0.6807	-0.6099	-0.2349	0.5492	-0.7456	0.1304
96	0.8134	-0.3929	-0.6360	-0.2851	1.3475	0.2087	0.2746	-0.1598	0.2547	-0.5398	2.2182	0.7896	0.5500	0.7212	-0.6538
97	0.8617	0.2914	0.8260	-0.6956	-0.0227	0.1923	0.1442	-0.4594	-0.7604	0.5592	1.0413	-2.1567	-0.6674	-0.0226	1.1549
98	-0.4876	-0.4727	0.6643	0.4573	-0.1120	-0.2531	-0.1526	-0.6920	-0.4902	0.4469	0.4653	0.6167	0.0417	-0.5481	-0.9451
99	-0.2334	-0.5425	-0.6967	1.0205	-0.3655	-0.9112	-0.8589	0.0259	0.8283	0.5385	0.7726	-0.1364	-0.1761	0.7330	-0.1652
100	-0.5980	0.3595	-0.5824	0.2712	-0.4233	0.3992	-0.2636	0.7506	-0.6037	-0.4588	0.4232	0.0974	0.0976	-0.0855	-0.9515

Tabela D11. Valores dos pesos das conexões entre a segunda camada oculta (100 neurônios) e a camada de saída da rede *Magola11*. Matriz de pesos LW2

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
1	0.4831	0.7827	-0.7493	1.4641	-1.4004	-0.7075	0.9734	-0.9436	-0.5225	-0.0571	-1.0009	-0.9725	1.1068	0.0361	1.2820
2	-0.9206	0.9588	0.0063	1.1094	-0.2619	-0.0261	0.8018	-1.3257	-0.3195	-0.0644	0.0313	-0.0134	0.9426	0.0191	-0.0004
3	1.3577	-0.7306	0.5621	0.8621	-0.2110	0.0047	-0.7458	-1.1949	-0.4406	1.3358	0.0768	-0.0053	1.2798	1.9616	0.0088

	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30
1	1.3220	-0.6180	-1.7779	-0.0885	0.5719	-1.0113	1.1448	0.0008	0.4105	-0.9917	0.0092	1.0069	0.0202	1.5950	-0.7959
2	1.5400	-0.0956	-0.3271	-0.0029	0.2304	-1.2972	1.0172	-0.0068	-1.0834	-0.0605	0.0180	0.9495	0.0252	0.0692	-0.8687
3	0.9473	-0.1054	-0.3013	0.0012	0.6477	0.0664	0.7413	1.4219	-0.9236	-0.1830	1.4767	0.8430	2.8673	0.0048	-0.7693

	31	32	33	34	35	36	37	38	39	40	41	42	43	44	45
1	-1.7403	-0.5309	-1.0712	-0.0006	0.5224	-0.2481	-0.0008	0.0427	1.7456	-0.7255	0.5254	-1.9983	0.3780	0.0282	-0.0114
2	-0.0275	0.1825	-1.2079	0.0051	0.0085	0.3279	-0.0027	1.0915	1.9563	0.0372	0.5682	-0.0361	0.3784	0.0102	-0.6475
3	-0.0281	0.1606	-0.0088	-1.7138	0.0192	0.3015	0.4017	-0.6454	1.5710	0.0611	-1.9376	-0.0520	0.0152	1.9160	0.3526

	46	47	48	49	50	51	52	53	54	55	56	57	58	59	60
1	-0.5438	1.3907	1.6551	-0.8157	0.8954	-1.2351	-0.3009	0.4788	0.5643	-0.0660	-0.7199	-1.1227	1.6504	0.8120	1.0323
2	-0.4321	0.3504	1.0059	-1.0848	1.2291	-0.5854	-0.1550	0.2025	0.6493	-0.3963	-0.7945	-1.2795	1.8048	1.4976	0.9803
3	-0.5334	-0.0683	0.1108	-0.4616	1.1541	0.1321	-0.1406	0.1745	-0.0077	-0.4381	-0.0062	-1.0180	1.9174	1.2410	0.0892

	61	62	63	64	65	66	67	68	69	70	71	72	73	74	75
1	1.1805	-0.5549	0.5566	0.4753	0.5075	1.1897	0.5829	0.1637	-0.6977	1.0787	0.5060	-0.6946	-1.5540	0.8897	-0.4774
2	1.0712	-0.6326	0.2213	-0.5053	0.5209	0.6623	-0.1108	0.0257	-0.3490	0.0812	1.0129	0.1900	-0.3375	1.0993	-0.6258
3	1.0250	-0.5361	0.1596	-0.0025	0.4952	0.5281	0.5121	0.0272	-0.2700	0.0952	-0.1489	0.4103	-0.2808	-0.0884	-0.6889

	76	77	78	79	80	81	82	83	84	85	86	87	88	89	90
1	-0.1638	0.0845	0.5443	-1.1845	0.5285	0.1930	0.0170	-1.8635	-1.2641	1.2456	0.1186	1.7215	-1.2774	-1.0364	-0.5925
2	0.0888	-0.2817	0.7248	-1.6453	-0.0226	0.8462	0.9410	-0.0375	-0.0140	-0.2492	0.1265	1.8279	-1.0282	-1.0890	0.1749
3	-0.3192	-0.2540	1.1296	-0.1108	-0.0520	-0.0350	0.4658	-0.0162	-0.0348	0.1333	-1.0553	0.1343	-0.8382	-0.8788	0.0130

	91	92	93	94	95	96	97	98	99	100
1	0.6187	1.3069	-0.0157	1.3423	0.3614	2.3467	0.4373	-0.6305	0.9635	-2.5034
2	-0.0323	1.4323	-0.0200	1.4216	0.0316	0.0208	0.5091	-0.0522	-0.0444	-2.8625
3	0.0893	0.8300	-3.1325	1.7677	0.1166	0.0286	0.0194	-0.0458	0.1363	-0.0278

Tabela D12. Valores dos elementos do primeiro vetor de biases, b1, (120 elementos) da rede *Magola11*.

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
0.9868	-0.4674	-0.0459	-0.5392	0.9232	0.5020	0.6130	-0.8425	1.4347	-0.1915	0.1407	0.8334	0.4507	-0.0159	-0.8013
16	17	18	19	20	21	22	23	24	25	26	27	28	29	30
-0.2307	0.8283	0.7761	-0.0485	1.1734	-0.8041	0.2068	0.3194	-1.1995	-0.8593	-1.1947	-0.7685	0.3101	-0.7451	-0.3221
31	32	33	34	35	36	37	38	39	40	41	42	43	44	45
-1.9298	0.2883	-0.1100	1.1344	0.4932	-0.7283	1.1572	-0.5944	-0.6352	0.2601	1.2320	-0.6527	-0.2072	-0.4840	0.0917
46	47	48	49	50	51	52	53	54	55	56	57	58	59	60
0.1525	-0.3417	-0.1217	-0.7859	-0.7744	-1.3855	0.4689	-1.2584	0.4811	-0.2310	0.4347	-0.8056	0.8088	0.9174	0.4752
61	62	63	64	65	66	67	68	69	70	71	72	73	74	75
0.3491	1.2156	0.3633	0.0807	0.3779	0.6094	-0.2423	-1.2874	-0.3874	-0.4853	-0.3197	-0.2800	0.1598	-0.0222	-0.7360
76	77	78	79	80	81	82	83	84	85	86	87	88	89	90
-1.0512	0.4373	-0.1923	-1.4327	1.0115	0.1764	1.0520	0.6718	1.1062	-0.1841	1.1761	-0.0271	-0.3968	1.0694	0.7216
91	92	93	94	95	96	97	98	99	100	101	102	103	104	105
-1.3786	-1.1545	0.4759	-0.8318	-0.2784	0.2138	-0.4645	-0.5987	-1.0997	0.9074	0.6399	-0.6812	-1.6274	-0.2189	0.0491
106	107	108	109	110	111	112	113	114	115	116	117	118	119	120
-0.6237	0.9903	0.6789	-0.0646	0.5200	-0.6198	-0.5894	1.0708	0.4112	0.1355	1.1445	1.5269	-1.4775	-0.6375	-3.0380

Tabela D13. Valores dos elementos do segundo vetor de biases, b2, (100 elementos) da rede *Magola11*.

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
0.6128	0.3050	-0.8353	0.1054	0.6886	-0.0770	-0.2883	0.8219	0.4762	1.0765	-0.4008	0.0805	-0.7995	0.5173	0.4504
16	17	18	19	20	21	22	23	24	25	26	27	28	29	30
0.1607	-0.6168	0.4102	0.2731	-0.7524	-0.4486	0.3059	-0.8003	-0.7370	0.5475	0.0323	0.5683	0.5758	-0.6618	0.6623
31	32	33	34	35	36	37	38	39	40	41	42	43	44	45
-0.4447	0.2051	0.7299	-0.2683	-0.6638	-0.3423	0.2263	0.0322	-0.4330	0.1366	0.0125	-0.4543	0.4580	0.5517	-1.2538
46	47	48	49	50	51	52	53	54	55	56	57	58	59	60
0.8927	0.0975	-1.0068	-0.0865	-0.7656	-0.1401	0.5802	0.7161	0.8337	0.7137	-0.4888	-0.6153	0.9422	-0.2629	-1.0546
61	62	63	64	65	66	67	68	69	70	71	72	73	74	75
-0.8513	-0.7656	0.7412	0.9919	0.5757	-0.5587	-0.5996	0.5991	0.6564	-0.6916	-0.1264	-0.7604	-0.6867	-0.9033	0.8258
76	77	78	79	80	81	82	83	84	85	86	87	88	89	90
0.4139	-0.3950	0.3035	-0.4539	-0.3579	0.0307	-0.2124	-0.6570	0.7374	0.0981	1.0631	0.6931	0.8526	-0.5200	0.2567
91	92	93	94	95	96	97	98	99	100					
-0.4208	0.2634	-0.1652	-0.7440	-0.1950	0.3817	-0.5548	0.0687	0.1617	-0.2659					

Tabela D14. Valores dos elementos do terceiro vetor de biases, b3, (3 elementos) da rede *Magola11*.

1	2	3
-0.15660	-0.64170	-0.94516