



Paulo Roberto Meneses de Paiva

**Modelos Neuroevolucionários com Echo State Networks
Aplicados à Identificação de Sistemas**

PUC-Rio - Certificação Digital Nº 1612970/CA

Dissertação de Mestrado

Dissertação apresentada como requisito parcial para
obtenção do grau de Mestre pelo Programa de Pós-
Graduação em Engenharia Elétrica da PUC-Rio.

Orientadora: Marley Maria Bernardes Rebuzzi Vellasco
Co-Orientador: José Franco Machado do Amaral

Rio de Janeiro
Setembro de 2018



Paulo Roberto Meneses de Paiva

**Modelos Neuroevolucionários com Echo State Networks
Aplicados à Identificação de Sistemas**

Dissertação apresentada como requisito parcial para obtenção do grau de Mestre pelo Programa de Pós-Graduação em Engenharia Elétrica da PUC-Rio. Aprovada pela Comissão Examinadora abaixo assinada.

Prof.^a Marley Maria Bernardes Rebuszi Vellasco

Orientadora
Departamento de Engenharia Elétrica – PUC-Rio

Prof. José Franco Machado do Amaral

Co-Orientador
UERJ

Prof. Pedro Henrique Gouvêa Coelho

UERJ

Prof.^a Karla Tereza Figueiredo Leite

UERJ

Prof. Márcio da Silveira Carvalho

Coordenador Setorial do Centro
Técnico Científico – PUC-Rio

Rio de Janeiro, 12 de setembro de 2018

Todos os direitos reservados. É proibida a reprodução total ou parcial do trabalho sem autorização da universidade, do autor e do orientador.

Paulo Roberto Meneses de Paiva

Graduou-se em Engenharia Elétrica com ênfase em Sistemas Eletrônicos pela Universidade do Estado do Rio de Janeiro (UERJ) em 2016.

Ficha Catalográfica

Paiva, Paulo Roberto Meneses de.

Modelos neuroevolucionários com Echo State Networks aplicados à identificação de sistemas / Paulo Roberto Meneses de Paiva; orientadora: Marley Maria Bernardes Rebuzzi Vellasco; co-orientador: José Franco Machado do Amaral. – 2018.

278 f. ; il. color.; 30 cm

Dissertação (mestrado) – Pontifícia Universidade Católica do Rio de Janeiro, Departamento de Engenharia Elétrica, 2018.

Inclui bibliografia

1. Engenharia Elétrica – Teses. 2. Echo State Networks. 3. Algoritmos evolutivos. 4. Identificação de sistemas. 5. Algoritmos evolutivos com inspiração quântica. 6. Estratégias evolucionárias. I. Vellasco, Marley Maria Bernardes Rebuzzi. II. Amaral, José Franco Machado do III. Pontifícia Universidade Católica do Rio de Janeiro. Departamento de Engenharia Elétrica. IV. Título.

CDD: 621.3

Agradecimentos

Aos meus orientadores, professora Marley Vellasco e professor José Franco, por todo o apoio, compreensão e incentivo durante todo o período do mestrado, que foram fundamentais para que eu pudesse prosseguir com os estudos, e pelo acompanhamento e parceria para a realização deste trabalho.

Aos meus familiares, em especial aos meus pais – Maria e José -, pela educação, carinho e apoio incondicional, e aos meus irmãos – Ângela, Cláudio, José e Maria – por compreenderem meus momentos de ausência dedicados ao estudo e por sempre me apoiarem.

À UERJ, em especial, aos professores do Departamento de Engenharia Eletrônica e de Telecomunicações desta instituição, pela sólida formação acadêmica durante o período no qual cursei minha graduação.

Ao CNPq, à FAPERJ e à PUC-Rio, pelos auxílios concedidos, sem os quais este trabalho não poderia ser realizado.

Aos professores do Departamento de Engenharia Elétrica da PUC-Rio, pelo ensino de qualidade e apoio.

Aos membros da banca examinadora, por aceitarem avaliar e sugerir melhorias ao trabalho.

Resumo

Paiva, Paulo Roberto Meneses de; Vellasco, Marley Maria Bernardes Rebuzzi, Amaral, José Franco Machado do. **Modelos neuroevolucionários com Echo State Networks aplicados à Identificação de Sistemas**. Rio de Janeiro, 2018. 278p. Dissertação de Mestrado - Departamento de Engenharia Elétrica, Pontifícia Universidade Católica do Rio de Janeiro.

Através das técnicas utilizadas em Identificação de Sistemas é possível obter um modelo matemático para um sistema dinâmico somente a partir de dados medidos de suas entradas e saídas. Por possuírem comportamento naturalmente dinâmico e um procedimento de treinamento simples e rápido, o uso de redes neurais do tipo *Echo State Networks* (ESNs) é vantajoso nesta área. Entretanto, as ESNs possuem hiperparâmetros que devem ser ajustados para que obtenham um bom desempenho em uma dada tarefa, além do fato de que a inicialização aleatória de pesos da camada interna destas redes (reservatório) nem sempre ser a ideal em termos de desempenho. Por teoricamente conseguirem obter boas soluções com poucas avaliações, o AEIQ-R (Algoritmo Evolutivo com Inspiração Quântica e Representação Real) e a estratégia evolucionária com adaptação da matriz de covariâncias (CMA-ES) representam alternativas de algoritmos evolutivos que permitem lidar de maneira eficiente com a otimização de hiperparâmetros e/ou pesos desta rede. Sendo assim, este trabalho propõe um modelo neuroevolucionário que define automaticamente uma ESN para aplicações de Identificação de Sistemas. O modelo inicialmente foca na otimização dos hiperparâmetros da ESN utilizando o AEIQ-R ou o CMA-ES, e, num segundo momento, seleciona o reservatório mais adequado para esta rede, o que pode ser feito através de uma segunda otimização focada no ajuste de alguns pesos do reservatório ou por uma escolha simples baseando-se em redes com reservatórios aleatórios. O método proposto foi aplicado a 9 problemas *benchmark* da área de Identificação de Sistemas, apresentando bons resultados quando comparados com modelos tradicionais.

Palavras-chave

Echo State Networks; Algoritmos evolutivos; Identificação de sistemas; Algoritmos evolutivos com inspiração quântica; Estratégias evolucionárias.

Abstract

Paiva, Paulo Roberto Meneses de; Vellasco, Marley Maria Bernardes Rebuzzi (Advisor), Amaral, José Franco Machado do (Co-advisor). **Neuro-evolutionary models with Echo State Networks applied to System Identification**. Rio de Janeiro, 2018. 278p. Dissertação de Mestrado - Departamento de Engenharia Elétrica, Pontifícia Universidade Católica do Rio de Janeiro.

Through System Identification techniques is possible to obtain a mathematical model for a dynamic system from its input/output data. Due to their intrinsic dynamic behavior and simple and fast training procedure, the use of Echo State Networks, which are a kind of neural networks, for System Identification is advantageous. However, ESNs have global parameters that should be tuned in order to improve their performance in a determined task. Besides, a random reservoir may not be ideal in terms of performance. Due to their theoretical ability of obtaining good solutions with few evaluations, the Real Coded Quantum-Inspired Evolutionary Algorithm (QIEA-R) and the Covariance Matrix Adaptation Evolution Strategy (CMA-ES) represent efficient alternatives of evolutionary algorithms for optimizing ESN global parameters and/or weights. Thus, this work proposes a neuro-evolutionary method that automatically defines an ESN for System Identification problems. The method initially focuses in finding the best ESN global parameters by using the QIEA-R or the CMA-ES, then, in a second moment, in selecting its best reservoir, which can be done by a second optimization focused on some reservoir weights or by doing a simple choice based on networks with random reservoirs. The method was applied to 9 benchmark problems in System Identification, showing good results when compared to traditional methods.

Keywords

Echo State Networks; Evolutionary algorithms; System identification; Quantum-inspired evolutionary algorithms; Evolution strategies.

Sumário

1	Introdução	32
1.1.	Motivação	36
1.2.	Objetivos	38
1.3.	Descrição da dissertação	39
1.4.	Organização do Trabalho	40
2	Identificação de Sistemas Não Lineares	42
2.1.	Introdução	42
2.2.	Definições sobre sistemas e modelos	44
2.3.	Procedimento típico de Identificação	45
2.4.	Representações não lineares	49
2.4.1.	Modelos NARX, NARMAX e seus derivados	49
2.4.2.	Representação por Espaço de Estados	51
2.4.3.	Modelos estruturados em blocos	52
2.4.4.	Séries de Volterra	54
2.4.5.	Redes Neurais Artificiais	54
2.4.6.	Modelos de dinâmica interna	59
3	Echo State Networks	60
3.1.	Introdução	60
3.2.	Arquitetura	61
3.3.	Funcionamento básico	62
3.4.	Equações básicas	64
3.4.1.	Atualização dos estados do reservatório	64
3.4.2.	Atualização das saídas da rede	65
3.5.	Treinamento	65
3.6.	Parâmetros que afetam o desempenho das ESNs	67
3.7.	Alguns métodos para avaliação do reservatório	68

3.7.1. Separação	69
3.7.2. Expoente de Lyapunov	70
3.7.3. <i>Separation Ratio Graph</i> (Gibbons, 2010)	72
3.7.4. Separation Ratio Graph (Niño, 2016)	73
3.8. ESNs implementando um modelo NARX	75
3.9. Alguns trabalhos anteriores	76
3.9.1. ESNs para identificação de sistemas	76
3.9.2. Otimização de ESNs por algoritmos evolutivos e/ou outras técnicas bioinspiradas	79
4 Algoritmos Evolutivos com Inspiração Quântica e Estratégias Evolucionárias	82
4.1. Introdução	82
4.2.3. Algoritmo Evolutivo com Inspiração Quântica e Representação Real – AEIQ-R	88
4.3. Estratégias Evolucionárias	94
4.3.1. Introdução	94
4.3.2. Estratégia evolucionária com adaptação da matriz de covariâncias – CMA-ES	96
4.3.2.1. Introdução	96
4.3.2.2. Descrição do algoritmo	98
5 Modelos neuroevolucionários baseados em ESNs otimizadas pelos algoritmos AEIQ-R e CMA-ES para Identificação de Sistemas	102
5.1. Introdução	102
5.2. Modelo Geral	103
5.2.1. Etapa 1 - Otimização de parâmetros gerais da ESN	104
5.2.2. Etapa 2 – Seleção do melhor reservatório	107
5.2.2.1. Random	107
5.2.2.2. SRG	108
5.2.2.3. Otimização do Reservatório	109
5.3. Resumo	113

6 Estudos de Caso	116
6.1. Introdução	116
6.2. Descrição dos problemas	116
6.2.1. Estudo de caso 1 - <i>Trocador de calor a vapor líquido-saturado</i>	116
6.2.2. Estudo de caso 2 - <i>Secador de cabelo</i>	118
6.2.3. Estudo de caso 3 – <i>Densidade de fluxo de calor através de uma parede de duas camadas</i>	119
6.2.4. Estudo de caso 4 – <i>Braço robótico flexível</i>	121
6.2.5. Estudo de caso 5 – <i>Reator tanque de agitação contínua</i>	122
6.2.6. Estudo de caso 6 – <i>Tanques em cascata</i>	124
6.2.7. Estudo de caso 7 – <i>Gerador de Vapor</i>	126
6.2.8. Estudo de caso 8 – <i>Braço do leitor de CD</i>	129
6.2.9. Estudo de caso 9 – <i>Caixa de prata</i>	131
6.3. Configurações dos experimentos	132
6.3.1. Configurações gerais dos experimentos	132
6.3.2. Configurações das ESNs	133
6.3.3. Divisão e pré-processamentos das bases de dados	134
6.3.4. Configurações dos Algoritmos Evolutivos	136
6.4. Resultados e discussões	138
6.4.1. Estudo de caso 1 - <i>Trocador de calor a vapor líquido-saturado</i>	140
6.4.2. Estudo de caso 2 – <i>Secador de cabelo</i>	148
6.4.3. Estudo de caso 3 – <i>Densidade de fluxo de calor através de uma parede de duas camadas</i>	156
6.4.4. Estudo de caso 4 – <i>Braço robótico flexível</i>	163
6.4.5. Estudo de caso 5 – <i>Reator tanque de agitação contínua</i>	171
6.4.6. Estudo de caso 6 – <i>Tanques em cascata</i>	182
6.4.7. Estudo de caso 7 – <i>Gerador de Vapor</i>	192
6.4.8. Estudo de caso 8 – <i>Braço do leitor de CD</i>	208
6.4.9. Estudo de caso 9 – <i>Caixa de prata</i>	220
6.5. Análise geral dos resultados	230
6.5.1. Resultados gerais	230
6.5.2. Análise estatística	234

7 Conclusão	241
8 Referências bibliográficas	244
Apêndice A – Valores de RMSE, SRG e de parâmetros globais das soluções de cada submodelo por experimento	261

Lista de figuras

Figura 1 - Variáveis de um sistema: \mathbf{u} – sinais de entrada, \mathbf{v} – ruídos mensuráveis, \mathbf{w} – ruídos não mensuráveis, \mathbf{x} – variáveis de estado do sistema, \mathbf{y} – saídas. Fonte: Adaptado de Ljung (1987) e Keesman (2011).....	44
Figura 2. O problema geral de Identificação de Sistemas. Fonte: Adaptado de Chinarro (2014).....	46
Figura 3. Procedimento adotado em Identificação de Sistemas. Fonte: Adaptado de Tavakolpour-Saleh et al. (2015)	46
Figura 4. Configuração série-paralelo utilizada na previsão 1 passo à frente.....	48
Figura 5. Configuração paralela utilizada na simulação ou previsão livre.....	48
Figura 6. Estratégia de dinâmica externa, onde $\mathbf{q} - \mathbf{1}$ representa o operador de atraso unitário no tempo. Fonte: Adaptado de Nelles (2001).....	50
Figura 7. Modelo de Hammerstein	52
Figura 8. Modelo de Wiener	53
Figura 9. Arquitetura simplificada de uma MLP	56
Figura 10. Modelo de um neurônio de uma rede MLP	56
Figura 11. Funções logística e tangente hiperbólica	57
Figura 12 – Arquitetura típica de uma ESN	61
Figura 13 – Regiões de um gráfico de razão de separação Fonte: Adaptado de Niño (2016) e Gibbons (2010).....	73
Figura 14 – Representação gráfica de um círculo unitário com as amplitudes de probabilidade de se observarem os estados “0” e “1” para um q-bit. Fonte: Adaptado de Cruz (2007)	84
Figura 15 - Pseudocódigo do AEIQ-B. Fonte: Adaptado de Han & Kim (2000) e Pinho (2010)	86
Figura 16 – Representação gráfica da rotação de um q-bit usando o operador q -gate.....	88

Figura 17 – Pseudocódigo do AEIQ-R. Fonte: Adaptado de Cruz (2007) e Pinho (2010)	89
Figura 18 – Função pulso retangular ou distribuição uniforme de probabilidade, onde L_{ij} e U_{ij} representam os valores de domínio inferior e superior, respectivamente, de $p_{ij}(x)$	90
Figura 19 – Indivíduo de uma ES convencional	94
Figura 20 – Mudança dos contornos de uma distribuição normal bivariada com a mudança da matriz C. Fonte: Adaptado de Pessanha (2018) e Johnson & Wichern (2007).....	97
Figura 21 – Pseudocódigo simplificado do CMA-ES. Fonte: Adaptado de Hansen (2016)	98
Figura 22 – Modelo geral proposto.....	103
Figura 23 – Etapa de otimização de parâmetros gerais	104
Figura 24 – Cromossomo utilizado na etapa “Otimização de parâmetros gerais da ESN”	105
Figura 25 – Etapa de seleção do melhor reservatório	107
Figura 26 – Módulo de Otimização do Reservatório	110
Figura 27- Determinação do cromossomo utilizado na etapa 2	111
Figura 28 – Pseudocódigo do modelo geral proposto	114
Figura 29 – Estrutura de um trocador de calor. Fonte: Adaptado de Bittanti & Piroddi (1997)	117
Figura 30 – Variáveis do estudo de caso 1	
(a) Entrada u – vazão de entrada do líquido no trocador.	
(b) Saída y – temperatura de saída do líquido no trocador.....	118
Figura 31 – Vista frontal de parte do <i>Feedback’s process trainer</i> PT326. Fonte: Adaptado de Leva & Papadoulos (2013).....	118
Figura 32 - Variáveis do estudo de caso 2	
(a) Entrada u – tensão na resistência de aquecimento.	
(b) Saída y – temperatura medida pelo termopar.	119
Figura 33 – Parede com duas camadas (tijolos e isolamento). Fonte: Adaptado de Castellanos (2008).....	120

Figura 34 - Variáveis do estudo de caso 3	
(a) Entrada 1: u_1 – temperatura interna da parede.	
(b) Entrada 2: u_2 – temperatura externa da parede	
(c) Saída y – densidade de fluxo de calor através da parede.	121
Figura 35 – Braço robótico flexível.	
Fonte: Adaptado de Ferreira & Serra (2012).....	121
Figura 36 - Variáveis do estudo de caso 4	
(a) Entrada: u – torque de reação na estrutura em relação ao solo.	
(b) Saída: y – aceleração do braço robótico.	122
Figura 37 – Reator tanque continuamente agitado.	
Fonte: Adaptado de Kandroodi & Moshiri (2011)	123
Figura 38 - Variáveis do estudo de caso 5	
(a) Entrada: u – fluxo de refrigerante $qC(t)$.	
(b) Saída 1: y_1 – concentração da mistura final CA_t .	
(c) Saída 2: y_2 – temperatura da mistura final $T(t)$	124
Figura 39 – Sistema de controle de fluido com dois tanques em cascata. Fonte: Adaptado de Wigren (2013) e Birpoutsoukis et al. (2018).....	125
Figura 40 - Variáveis do estudo de caso 5	
(a) Entrada: u – tensão aplicada na bomba.	
(b) Saída 1: y_1 – nível de água no tanque superior.	
(c) Saída 2: y_2 – nível de água no tanque inferior	126
Figura 41 – Esquema básico do gerador de vapor.	
Fonte: Adaptado de Nozari & Banadaki (2014).....	127
Figura 42 - Variáveis de entrada do estudo de caso 7	
(a) Entrada 1: u_1 – vazão de combustível.	
(b) Entrada 2: u_2 – vazão de ar	
(c) Entrada 3: u_3 – nível de água de referência.	
(d) Entrada 4: u_4 – distúrbios definidos pelo nível de água	128

Figura 43 - Variáveis de saída do estudo de caso 7	
(a) Saída 1: y_1 – pressão de vapor.	
(b) Saída 2: y_2 – excesso de oxigênio.	
(c) Saída 3: y_3 – nível de água no domo do gerador de vapor.	
(d) Saída 4: y_4 – vazão de vapor.....	129
Figura 44 - Operação mecânica de um dispositivo leitor de CD.	
Fonte: Adaptado de Barry (2004).....	129
Figura 45 - Variáveis de entrada do estudo de caso 8	
(a) Entrada 1: u_1 – força aplicada no atuador 1.	
(b) Entrada 2: u_2 – força aplicada no atuador 2.....	130
Figura 46 - Variáveis de saída do estudo de caso 8	
(a) Saída 1: y_1 – precisão de rastreamento 1.	
(b) Saída 2: y_2 – precisão de rastreamento 2.....	131
Figura 47 - Sistema mecânico massa-mola-amortecedor simulado pelo “caixa de prata”	131
Figura 48 - Variáveis do estudo de caso 9	
(a) Entrada: u – força aplicada na massa.	
(b) Saída: y – deslocamento da massa.....	132
Figura 49 - Organização geral dos experimentos.....	139
Figura 50 – Curva de evolução da etapa 1 (média da aptidão do melhor indivíduo em cada geração nos 10 experimentos) para o estudo de caso 1.....	141
Figura 51 - Curva de evolução da etapa 2 (média da aptidão do melhor indivíduo em cada geração nos 10 experimentos) para o estudo de caso 1.....	142
Figura 52 – Simulação da variável de saída y do estudo de caso 1 utilizando a melhor solução do submodelo AEIQ-R/Random/ESN	143
Figura 53 - Simulação da variável de saída y do estudo de caso 1 utilizando a melhor solução do submodelo AEIQ-R/SRG/ESN	144
Figura 54 - Simulação da variável de saída y do estudo de caso 1 utilizando a melhor solução do submodelo AEIQ-R/AEIQ-R/ESN.....	144
Figura 55 - Simulação da variável de saída y do estudo de caso 1 utilizando a melhor solução do submodelo CMA-ES/Random/ESN.....	145

Figura 56 - Simulação da variável de saída y do estudo de caso 1 utilizando a melhor solução do submodelo CMA-ES/SRG/ESN.....	145
Figura 57 - Simulação da variável de saída y do estudo de caso 1 utilizando a melhor solução do submodelo CMA-ES/CMA-ES/ESN	146
Figura 58 – Gráficos de razão de separação encontrados para as melhores soluções de cada submodelo para o estudo de caso 1	146
Figura 59 – Curva de evolução da etapa 1 (média da aptidão do melhor indivíduo em cada geração nos 10 experimentos) para o estudo de caso 2.....	149
Figura 60 - Curva de evolução da etapa 2 (média da aptidão do melhor indivíduo em cada geração nos 10 experimentos) para o estudo de caso 2.....	149
Figura 61 – Simulação da variável de saída y do estudo de caso 2 utilizando a melhor solução do submodelo AEIQ-R/Random/ESN	151
Figura 62 - Simulação da variável de saída y do estudo de caso 2 utilizando a melhor solução do submodelo AEIQ-R/SRG/ESN	151
Figura 63 - Simulação da variável de saída y do estudo de caso 2 utilizando a melhor solução do submodelo AEIQ-R/AEIQ-R/ESN.....	152
Figura 64 - Simulação da variável de saída y do estudo de caso 2 utilizando a melhor solução do submodelo CMA-ES/Random/ESN.....	152
Figura 65 - Simulação da variável de saída y do estudo de caso 2 utilizando a melhor solução do submodelo CMA-ES/SRG/ESN.....	153
Figura 66 - Simulação da variável de saída y do estudo de caso 2 utilizando a melhor solução do submodelo CMA-ES/CMA-ES/ESN	153
Figura 67 – Gráficos de razão de separação encontrados para as melhores soluções de cada submodelo para o estudo de caso 2.....	154
Figura 68– Curva de evolução da etapa 1 (média da aptidão do melhor indivíduo em cada geração nos 10 experimentos) para o estudo de caso 3.....	157
Figura 69 - Curva de evolução da etapa 2 (média da aptidão do melhor indivíduo em cada geração nos 10 experimentos) para o estudo de caso 3.....	157

Figura 70 - Simulação da variável de saída y do estudo de caso 3 utilizando a melhor solução do submodelo AEIQ-R/Random/ESN	158
Figura 71 - Simulação da variável de saída y do estudo de caso 3 utilizando a melhor solução do submodelo AEIQ-R/SRG/ESN	159
Figura 72 - Simulação da variável de saída y do estudo de caso 3 utilizando a melhor solução do submodelo AEIQ-R/AEIQ-R/ESN.....	159
Figura 73 - Simulação da variável de saída y do estudo de caso 3 utilizando a melhor solução do submodelo CMA-ES/Random/ESN.....	160
Figura 74 - Simulação da variável de saída y do estudo de caso 3 utilizando a melhor solução do submodelo CMA-ES/SRG/ESN.....	160
Figura 75 - Simulação da variável de saída y do estudo de caso 3 utilizando a melhor solução do submodelo CMA-ES/CMA-ES/ESN	161
Figura 76 – Gráficos de razão de separação encontrados para as melhores soluções de cada submodelo para o estudo de caso 3.....	161
Figura 77 - Curva de evolução da etapa 1 (média da aptidão do melhor indivíduo em cada geração nos 10 experimentos) para o estudo de caso 4.....	164
Figura 78 - Curva de evolução da etapa 2 (média da aptidão do melhor indivíduo em cada geração nos 10 experimentos) para o estudo de caso 4.....	165
Figura 79 - Simulação da variável de saída y do estudo de caso 4 utilizando a melhor solução do submodelo AEIQ-R/Random/ESN	166
Figura 80 - Simulação da variável de saída y do estudo de caso 4 utilizando a melhor solução do submodelo AEIQ-R/SRG/ESN	166
Figura 81 - Simulação da variável de saída y do estudo de caso 4 utilizando a melhor solução do submodelo AEIQ-R/AEIQ-R/ESN.....	167
Figura 82 - Simulação da variável de saída y do estudo de caso 4 utilizando a melhor solução do submodelo CMA-ES/Random/ESN.....	167
Figura 83 - Simulação da variável de saída y do estudo de caso 4 utilizando a melhor solução do submodelo CMA-ES/SRG/ESN.....	168
Figura 84 - Simulação da variável de saída y do estudo de caso 4 utilizando a melhor solução do submodelo CMA-ES/CMA-ES/ESN	168

Figura 85 - Gráficos de razão de separação encontrados para as melhores soluções de cada submodelo para o estudo de caso 4	169
Figura 86 - Curva de evolução da etapa 1 (média da aptidão do melhor indivíduo em cada geração nos 10 experimentos) para o estudo de caso 5	172
Figura 87 - Curva de evolução da etapa 2 (média da aptidão do melhor indivíduo em cada geração nos 10 experimentos) para o estudo de caso 5.....	172
Figura 88 - Simulação da variável de saída y_1 do estudo de caso 5 utilizando a melhor solução do submodelo AEIQ-R/Random/ESN	173
Figura 89 - Simulação da variável de saída y_2 do estudo de caso 5 utilizando a melhor solução do submodelo AEIQ-R/Random/ESN	174
Figura 90 - Simulação da variável de saída y_1 do estudo de caso 5 utilizando a melhor solução do submodelo AEIQ-R/SRG/ESN	174
Figura 91 - Simulação da variável de saída y_2 do estudo de caso 5 utilizando a melhor solução do submodelo AEIQ-R/SRG/ESN	175
Figura 92 - Simulação da variável de saída y_1 do estudo de caso 5 utilizando a melhor solução do submodelo AEIQ-R/AEIQ-R/ESN.....	175
Figura 93 - Simulação da variável de saída y_2 do estudo de caso 5 utilizando a melhor solução do submodelo AEIQ-R/AEIQ-R/ESN.....	176
Figura 94 - Simulação da variável de saída y_1 do estudo de caso 5 utilizando a melhor solução do submodelo CMA-ES/Random/ESN.....	176
Figura 95 - Simulação da variável de saída y_2 do estudo de caso 5 utilizando a melhor solução do submodelo CMA-ES/Random/ESN.....	177
Figura 96 - Simulação da variável de saída y_1 do estudo de caso 5 utilizando a melhor solução do submodelo CMA-ES/SRG/ESN.....	177
Figura 97 - Simulação da variável de saída y_2 do estudo de caso 5 utilizando a melhor solução do submodelo CMA-ES/SRG/ESN.....	178
Figura 98 - Simulação da variável de saída y_1 do estudo de caso 5 utilizando a melhor solução do submodelo CMA-ES/CMA-ES/ESN	178
Figura 99 - Simulação da variável de saída y_2 do estudo de caso 5 utilizando a melhor solução do submodelo CMA-ES/CMA-ES/ESN	179

Figura 100 - Gráficos de razão de separação encontrados para as melhores soluções de cada submodelo para o estudo de caso 5	179
Figura 101 - Curva de evolução da etapa 1 (média da aptidão do melhor indivíduo em cada geração nos 10 experimentos) para o estudo de caso 6.....	183
Figura 102 - Curva de evolução da etapa 2 (média da aptidão do melhor indivíduo em cada geração nos 10 experimentos) para o estudo de caso 6.....	183
Figura 103 - Simulação da variável de saída y_1 do estudo de caso 6 utilizando a melhor solução do submodelo AEIQ-R/Random/ESN	184
Figura 104 - Simulação da variável de saída y_2 do estudo de caso 6 utilizando a melhor solução do submodelo AEIQ-R/Random/ESN	185
Figura 105 - Simulação da variável de saída y_1 do estudo de caso 6 utilizando a melhor solução do submodelo AEIQ-R/SRG/ESN	185
Figura 106 - Simulação da variável de saída y_2 do estudo de caso 6 utilizando a melhor solução do submodelo AEIQ-R/SRG/ESN	186
Figura 107 - Simulação da variável de saída y_1 do estudo de caso 6 utilizando a melhor solução do submodelo AEIQ-R/AEIQ-R/ESN.....	186
Figura 108 - Simulação da variável de saída y_2 do estudo de caso 6 utilizando a melhor solução do submodelo AEIQ-R/AEIQ-R/ESN.....	187
Figura 109 - Simulação da variável de saída y_1 do estudo de caso 6 utilizando a melhor solução do submodelo CMA-ES/Random/ESN.....	187
Figura 110 - Simulação da variável de saída y_2 do estudo de caso 6 utilizando a melhor solução do submodelo CMA-ES/Random/ESN.....	188
Figura 111 - Simulação da variável de saída y_1 do estudo de caso 6 utilizando a melhor solução do submodelo CMA-ES/SRG/ESN.....	188
Figura 112 - Simulação da variável de saída y_2 do estudo de caso 6 utilizando a melhor solução do submodelo CMA-ES/SRG/ESN.....	189
Figura 113 - Simulação da variável de saída y_1 do estudo de caso 6 utilizando a melhor solução do submodelo CMA-ES/CMA-ES/ESN	189
Figura 114 - Simulação da variável de saída y_2 do estudo de caso 6 utilizando a melhor solução do submodelo CMA-ES/CMA-ES/ESN	190

Figura 115 - Gráficos de razão de separação encontrados para as melhores soluções de cada submodelo para o estudo de caso 6	190
Figura 116 - Curva de evolução da etapa 1 (média da aptidão do melhor indivíduo em cada geração nos 10 experimentos) para o estudo de caso 7	193
Figura 117 - Curva de evolução da etapa 2 (média da aptidão do melhor indivíduo em cada geração nos 10 experimentos) para o estudo de caso 7	193
Figura 118 - Simulação da variável de saída y_1 do estudo de caso 7 utilizando a melhor solução do submodelo AEIQ-R/Random/ESN	195
Figura 119 - Simulação da variável de saída y_2 do estudo de caso 7 utilizando a melhor solução do submodelo AEIQ-R/Random/ESN	195
Figura 120 - Simulação da variável de saída y_3 do estudo de caso 7 utilizando a melhor solução do submodelo AEIQ-R/Random/ESN	196
Figura 121 - Simulação da variável de saída y_4 do estudo de caso 7 utilizando a melhor solução do submodelo AEIQ-R/Random/ESN	196
Figura 122 - Simulação da variável de saída y_1 do estudo de caso 7 utilizando a melhor solução do submodelo AEIQ-R/SRG/ESN	197
Figura 123 - Simulação da variável de saída y_2 do estudo de caso 7 utilizando a melhor solução do submodelo AEIQ-R/SRG/ESN	197
Figura 124 - Simulação da variável de saída y_3 do estudo de caso 7 utilizando a melhor solução do submodelo AEIQ-R/SRG/ESN	198
Figura 125 - Simulação da variável de saída y_4 do estudo de caso 7 utilizando a melhor solução do submodelo AEIQ-R/SRG/ESN	198
Figura 126 - Simulação da variável de saída y_1 do estudo de caso 7 utilizando a melhor solução do submodelo AEIQ-R/AEIQ-R/ESN.....	199
Figura 127 - Simulação da variável de saída y_2 do estudo de caso 7 utilizando a melhor solução do submodelo AEIQ-R/AEIQ-R/ESN.....	199
Figura 128 - Simulação da variável de saída y_3 do estudo de caso 7 utilizando a melhor solução do submodelo AEIQ-R/AEIQ-R/ESN.....	200
Figura 129 - Simulação da variável de saída y_4 do estudo de caso 7 utilizando a melhor solução do submodelo AEIQ-R/AEIQ-R/ESN.....	200

Figura 130 - Simulação da variável de saída y_1 do estudo de caso 7 utilizando a melhor solução do submodelo CMA-ES/Random/ESN.....	201
Figura 131 - Simulação da variável de saída y_2 do estudo de caso 7 utilizando a melhor solução do submodelo CMA-ES/Random/ESN.....	201
Figura 132 - Simulação da variável de saída y_3 do estudo de caso 7 utilizando a melhor solução do submodelo CMA-ES/Random/ESN.....	202
Figura 133 - Simulação da variável de saída y_4 do estudo de caso 7 utilizando a melhor solução do submodelo CMA-ES/Random/ESN.....	202
Figura 134 - Simulação da variável de saída y_1 do estudo de caso 7 utilizando a melhor solução do submodelo CMA-ES/SRG/ESN.....	203
Figura 135 - Simulação da variável de saída y_2 do estudo de caso 7 utilizando a melhor solução do submodelo CMA-ES/SRG/ESN.....	203
Figura 136 - Simulação da variável de saída y_3 do estudo de caso 7 utilizando a melhor solução do submodelo CMA-ES/SRG/ESN.....	204
Figura 137 - Simulação da variável de saída y_4 do estudo de caso 7 utilizando a melhor solução do submodelo CMA-ES/SRG/ESN.....	204
Figura 138 - Simulação da variável de saída y_1 do estudo de caso 7 utilizando a melhor solução do submodelo CMA-ES/CMA-ES/ESN	205
Figura 139 - Simulação da variável de saída y_2 do estudo de caso 7 utilizando a melhor solução do submodelo CMA-ES/CMA-ES/ESN	205
Figura 140 - Simulação da variável de saída y_3 do estudo de caso 7 utilizando a melhor solução do submodelo CMA-ES/CMA-ES/ESN	206
Figura 141 - Simulação da variável de saída y_4 do estudo de caso 7 utilizando a melhor solução do submodelo CMA-ES/CMA-ES/ESN	206
Figura 142 - Gráficos de razão de separação encontrados para as melhores soluções de cada submodelo para o estudo de caso 7	207
Figura 143 - Curva de evolução da etapa 1 (média da aptidão do melhor indivíduo em cada geração nos 10 experimentos) para o estudo de caso 8	209
Figura 144 - Curva de evolução da etapa 2 (média da aptidão do melhor indivíduo em cada geração nos 10 experimentos) para o estudo de caso 8	210

Figura 145 - Simulação da variável de saída y_1 do estudo de caso 8 utilizando a melhor solução do submodelo AEIQ-R/Random/ESN	211
Figura 146 - Simulação da variável de saída y_2 do estudo de caso 8 utilizando a melhor solução do submodelo AEIQ-R/Random/ESN	211
Figura 147 - Simulação da variável de saída y_1 do estudo de caso 8 utilizando a melhor solução do submodelo AEIQ-R/SRG/ESN	212
Figura 148 - Simulação da variável de saída y_2 do estudo de caso 8 utilizando a melhor solução do submodelo AEIQ-R/SRG/ESN	212
Figura 149 - Simulação da variável de saída y_1 do estudo de caso 8 utilizando a melhor solução do submodelo AEIQ-R/AEIQ-R/ESN.....	213
Figura 150 - Simulação da variável de saída y_2 do estudo de caso 8 utilizando a melhor solução do submodelo AEIQ-R/AEIQ-R/ESN.....	213
Figura 151 - Simulação da variável de saída y_1 do estudo de caso 8 utilizando a melhor solução do submodelo CMA-ES/Random/ESN.....	214
Figura 152 - Simulação da variável de saída y_2 do estudo de caso 8 utilizando a melhor solução do submodelo CMA-ES/Random/ESN.....	214
Figura 153 - Simulação da variável de saída y_1 do estudo de caso 8 utilizando a melhor solução do submodelo CMA-ES/SRG/ESN.....	215
Figura 154 - Simulação da variável de saída y_2 do estudo de caso 8 utilizando a melhor solução do submodelo CMA-ES/SRG/ESN.....	215
Figura 155 - Simulação da variável de saída y_1 do estudo de caso 8 utilizando a melhor solução do submodelo CMA-ES/CMA-ES/ESN	216
Figura 156 - Simulação da variável de saída y_2 do estudo de caso 8 utilizando a melhor solução do submodelo CMA-ES/CMA-ES/ESN	216
Figura 157 - Gráficos de razão de separação encontrados para as melhores soluções de cada submodelo para o estudo de caso 8.....	217
Figura 158 - Curva de evolução da etapa 1 (média da aptidão do melhor indivíduo em cada geração nos 10 experimentos) para o estudo de caso 9.....	221
Figura 159 - Curva de evolução da etapa 2 (média da aptidão do melhor indivíduo em cada geração nos 10 experimentos) para o estudo de caso 9.....	221

Figura 160 - Simulação da variável de saída y do estudo de caso 9 utilizando a melhor solução do submodelo AEIQ-R/Random/ESN	222
Figura 161 - Simulação da variável de saída y do estudo de caso 9 utilizando a melhor solução do submodelo AEIQ-R/SRG/ESN	223
Figura 162 - Simulação da variável de saída y do estudo de caso 9 utilizando a melhor solução do submodelo AEIQ-R/AEIQ-R/ESN.....	223
Figura 163 - Simulação da variável de saída y do estudo de caso 9 utilizando a melhor solução do submodelo CMA-ES/Random/ESN.....	224
Figura 164 - Simulação da variável de saída y do estudo de caso 9 utilizando a melhor solução do submodelo CMA-ES/SRG/ESN.....	224
Figura 165 - Simulação da variável de saída y do estudo de caso 9 utilizando a melhor solução do submodelo CMA-ES/CMA-ES/ESN	225
Figura 166 - Simulação da variável de saída y do estudo de caso 9 utilizando a melhor solução do submodelo AEIQ-R/Random/ESN (zoom em algumas amostras)	225
Figura 167 - Simulação da variável de saída y do estudo de caso 9 utilizando a melhor solução do submodelo AEIQ-R/SRG/ESN (zoom em algumas amostras)	226
Figura 168 - Simulação da variável de saída y do estudo de caso 9 utilizando a melhor solução do submodelo AEIQ-R/AEIQ-R/ESN (zoom em algumas amostras)	226
Figura 169 - Simulação da variável de saída y do estudo de caso 9 utilizando a melhor solução do submodelo CMA-ES/Random/ESN (zoom em algumas amostras)	227
Figura 170 - Simulação da variável de saída y do estudo de caso 9 utilizando a melhor solução do submodelo CMA-ES/SRG/ESN (zoom em algumas amostras)	227
Figura 171 - Simulação da variável de saída y do estudo de caso 9 utilizando a melhor solução do submodelo CMA-ES/CMA-ES/ESN (zoom em algumas amostras)	228
Figura 172 - Gráficos de razão de separação encontrados para as melhores soluções de cada submodelo para o estudo de caso 9	228

Lista de tabelas

Tabela 1- Parâmetros do AEIQ-R	93
Tabela 2 – Alguns valores de constantes e parâmetros do CMA-ES.....	100
Tabela 3 – Submodelos resultantes do modelo geral testados	133
Tabela 4 – Configurações das ESNs	134
Tabela 5 – Informações das bases de dados utilizadas nos estudos de casos.....	134
Tabela 6 - Divisões das bases de dados valores de T0 para cada estudo de caso	135
Tabela 7 – Intervalo de variação das variáveis otimizadas pelo AEIQ-R e pelo CMA-ES.....	136
Tabela 8 – Parâmetros de configuração do AEIQ-R	137
Tabela 9 - Parâmetros de configuração do CMA-ES	138
Tabela 10 – Resultados obtidos de RMSE pelos diferentes submodelos para o <i>benchmark Trocador de calor a vapor líquido-saturado</i> nos 10 experimentos realizados	141
Tabela 11 – Aptidão dos AEs na última geração (média de 10 experimentos) para o estudo de caso 1	142
Tabela 12 – Parâmetros globais da melhor solução encontrada por cada submodelo para o estudo de caso 2	143
Tabela 13 – Comparação de resultados com outros métodos para o <i>benchmark Liquid-Saturated Heat Exchanger</i>	147
Tabela 14 – Resultados obtidos de RMSE pelos diferentes submodelos para o <i>benchmark Secador de cabelo</i> nos 10 experimentos realizados	148
Tabela 15 – Aptidão dos AEs na última geração (média de 10 experimentos) para o estudo de caso 2	150
Tabela 16 – Parâmetros globais da melhor solução encontrada por cada submodelo para o estudo de caso 2	150
Tabela 17 – Comparação de resultados com outros métodos para o <i>benchmark Secador de cabelo</i>	155

Tabela 18 – Resultados obtidos de RMSE pelos diferentes submodelos para o <i>benchmark Densidade de fluxo de calor através de uma parede de duas camadas</i> nos 10 experimentos realizados.....	156
Tabela 19 – Aptidão dos AEs na última geração (média de 10 experimentos) para o estudo de caso 3	158
Tabela 20 – Parâmetros globais da melhor solução encontrada por cada submodelo para o estudo de caso 3	158
Tabela 21 – Resultados para o experimento extra para o estudo de caso 3 com 100 gerações na etapa 2	162
Tabela 22 – Comparação de resultados com outros métodos para o <i>benchmark Densidade de fluxo de calor através de uma parede de duas camadas</i>	162
Tabela 23 – Resultados obtidos de RMSE pelos diferentes submodelos para o <i>benchmark Braço robótico flexível</i> nos 10 experimentos realizados...	164
Tabela 24 - Aptidão dos AEs na última geração (média de 10 experimentos) para o estudo de caso 4	165
Tabela 25 - Parâmetros globais da melhor solução encontrada por cada submodelo para o estudo de caso 4	165
Tabela 26 - Comparação de resultados com outros métodos para o <i>benchmark Flexible Arm</i>	170
Tabela 27– Resultados obtidos de RMSE pelos diferentes submodelos para o <i>benchmark Reator tanque de agitação contínua</i> nos 10 experimentos realizados	171
Tabela 28 - Aptidão dos AEs na última geração (média de 10 experimentos) para o estudo de caso 5	172
Tabela 29 - Parâmetros globais da melhor solução encontrada por cada submodelo para o estudo de caso 5	173
Tabela 30 - Comparação 1 de resultados com outros métodos para o <i>benchmark Reator tanque de agitação contínua</i>	181
Tabela 31 - Comparação 2 de resultados com outros métodos para o <i>benchmark Reator tanque de agitação contínua</i>	181
Tabela 32– Resultados obtidos de RMSE pelos diferentes submodelos para o <i>benchmark Tanques em cascata</i> nos 10 experimentos realizados.....	182

Tabela 33 - Aptidão dos AEs na última geração (média de 10 experimentos) para o estudo de caso 6	183
Tabela 34 - Parâmetros globais da melhor solução encontrada por cada submodelo para o estudo de caso 6	184
Tabela 35 - Comparação de resultados com outros métodos para o <i>benchmark Tanques em cascata</i>	191
Tabela 36 – Resultados obtidos de RMSE pelos diferentes submodelos para o <i>benchmark Gerador de Vapor</i> nos 10 experimentos realizados	192
Tabela 37 - Aptidão dos AEs na última geração (média de 10 experimentos) para o estudo de caso 7	194
Tabela 38 - Parâmetros globais da melhor solução encontrada por cada submodelo para o estudo de caso 7	194
Tabela 39 - Comparação de resultados com outros métodos para o <i>benchmark Gerador de Vapor</i>	208
Tabela 40 – Resultados obtidos de RMSE pelos diferentes submodelos para o <i>benchmark Braço do leitor de CD</i> nos 10 experimentos realizados	209
Tabela 41 - Aptidão dos AEs na última geração (média de 10 experimentos) para o estudo de caso 8	210
Tabela 42 - Parâmetros globais da melhor solução encontrada por cada submodelo para o estudo de caso 8.....	210
Tabela 43 - Comparação de resultados com outros métodos para o <i>benchmark Braço do leitor de CD</i>	218
Tabela 44 – Resultados para o estudo de caso 8 – experimento extra.....	219
Tabela 45 - Comparação de resultados com os métodos de (Dudul; Ghatol, 2003) para o estudo de caso 8 – experimento extra	219
Tabela 46 – Resultados obtidos de RMSE pelos diferentes submodelos para o <i>benchmark Caixa de prata</i> nos 10 experimentos realizados.....	220
Tabela 47 - Aptidão dos AEs na última geração (média de 10 experimentos) para o estudo de caso 9	222
Tabela 48 - Parâmetros globais da melhor solução encontrada por cada submodelo para o estudo de caso 9	222
Tabela 49 - Comparação 1 de resultados com outros métodos para o <i>benchmark Caixa de prata</i>	229

Tabela 50 - Comparação 2 de resultados com outros métodos para o <i>benchmark Caixa de prata</i>	230
Tabela 51 – Resultados de RMSE no teste (média de 10 experimentos) para a simulação do conjunto de teste utilizando as diferentes variações do método geral nos 9 estudos de caso.....	231
Tabela 52 – Resultados por experimentos métodos AEIQ-R/Random/ESN e AEIQ-R/SRG/ESN para o estudo de caso <i>Braço do leitor de CD</i>	233
Tabela 53 – Teste de Wilcoxon Signed Ranks aplicado aos submodelos derivados do método geral	235
Tabela 54 - Resultados para o teste de Friedman para comparação entre os submodelos	236
Tabela 55 – Resultados de aplicação dos testes <i>post-hoc</i>	237
Tabela 56 – Resumo dos resultados de <i>fit</i> de teste para os estudos de caso 1, 2, 3 e 8.....	238
Tabela 57 – Resultados do teste de Wilcoxon Signed Ranks para os modelos da tabela 56	238
Tabela 58 – Resultados do teste de Friedman para os modelos da tabela 56	240
Tabela 59 – Melhores parâmetros globais encontrados pelo AEIQ-R em cada experimento no estudo de caso 1	261
Tabela 60 – Melhores parâmetros globais encontrados pelo CMA-ES em cada experimento no estudo de caso 1	262
Tabela 61 – Resultados de RMSE da solução final de cada submodelo por experimento no estudo de caso 1 (parte 1).....	262
Tabela 62 – Resultados de RMSE da solução final de cada submodelo por experimento no estudo de caso 1 (parte 2).....	262
Tabela 63 – Resultados de SRG dos submodelos em cada experimento no estudo de caso 1	263
Tabela 64 – Melhores parâmetros globais encontrados pelo AEIQ-R em cada experimento no estudo de caso 2.....	263
Tabela 65 – Melhores parâmetros globais encontrados pelo CMA-ES em cada experimento no estudo de caso 2.....	263
Tabela 66 – Resultados de RMSE da solução final de cada submodelo por experimento no estudo de caso 2 (parte 1).....	264

Tabela 67 – Resultados de RMSE da solução final de cada submodelo por experimento no estudo de caso 2 (parte 2).....	264
Tabela 68 – Resultados de SRG dos submodelos em cada experimento no estudo de caso 2.....	264
Tabela 69 – Melhores parâmetros globais encontrados pelo AEIQ-R em cada experimento no estudo de caso 3.....	265
Tabela 70 – Melhores parâmetros globais encontrados pelo CMA-ES em cada experimento no estudo de caso 3.....	265
Tabela 71 – Resultados de RMSE da solução final de cada submodelo por experimento no estudo de caso 3 (parte 1).....	266
Tabela 72 – Resultados de RMSE da solução final de cada submodelo por experimento no estudo de caso 3 (parte 2).....	266
Tabela 73 – Resultados de SRG dos submodelos em cada experimento no estudo de caso 3.....	266
Tabela 74 – Melhores parâmetros globais encontrados pelo AEIQ-R em cada experimento no estudo de caso 4.....	267
Tabela 75 – Melhores parâmetros globais encontrados pelo CMA-ES em cada experimento no estudo de caso 4.....	267
Tabela 76 – Resultados de RMSE da solução final de cada submodelo por experimento no estudo de caso 4 (parte 1).....	268
Tabela 77 – Resultados de RMSE da solução final de cada submodelo por experimento no estudo de caso 4 (parte 2).....	268
Tabela 78 – Resultados de SRG dos submodelos em cada experimento no estudo de caso 4.....	268
Tabela 79 – Melhores parâmetros globais encontrados pelo AEIQ-R em cada experimento no estudo de caso 5.....	269
Tabela 80 – Melhores parâmetros globais encontrados pelo CMA-ES em cada experimento no estudo de caso 5.....	269
Tabela 81 – Resultados de RMSE da solução final de cada submodelo por experimento no estudo de caso 5 (parte 1).....	270
Tabela 82 – Resultados de RMSE da solução final de cada submodelo por experimento no estudo de caso 5 (parte 2).....	270
Tabela 83 – Resultados de SRG dos submodelos em cada experimento no estudo de caso 5.....	270

Tabela 84 – Melhores parâmetros globais encontrados pelo AEIQ-R em cada experimento no estudo de caso 6.....	271
Tabela 85 – Melhores parâmetros globais encontrados pelo CMA-ES em cada experimento no estudo de caso 6.....	271
Tabela 86 – Resultados de RMSE da solução final de cada submodelo por experimento no estudo de caso 6 (parte 1).....	272
Tabela 87 – Resultados de RMSE da solução final de cada submodelo por experimento no estudo de caso 6 (parte 2).....	272
Tabela 88 – Resultados de SRG dos submodelos em cada experimento no estudo de caso 6.....	272
Tabela 89 – Melhores parâmetros globais encontrados pelo AEIQ-R em cada experimento no estudo de caso 7.....	273
Tabela 90 – Melhores parâmetros globais encontrados pelo CMA-ES em cada experimento no estudo de caso 7.....	273
Tabela 91 – Resultados de RMSE da solução final de cada submodelo por experimento no estudo de caso 7 (parte 1).....	274
Tabela 92 – Resultados de RMSE da solução final de cada submodelo por experimento no estudo de caso 7 (parte 2).....	274
Tabela 93 – Resultados de SRG dos submodelos em cada experimento no estudo de caso 7.....	274
Tabela 94 – Melhores parâmetros globais encontrados pelo AEIQ-R em cada experimento no estudo de caso 8.....	275
Tabela 95 – Melhores parâmetros globais encontrados pelo CMA-ES em cada experimento no estudo de caso 8.....	275
Tabela 96 – Resultados de RMSE da solução final de cada submodelo por experimento no estudo de caso 8 (parte 1).....	276
Tabela 97 – Resultados de RMSE da solução final de cada submodelo por experimento no estudo de caso 8 (parte 2).....	276
Tabela 98 – Resultados de SRG dos submodelos em cada experimento no estudo de caso 8.....	276
Tabela 99 – Melhores parâmetros globais encontrados pelo AEIQ-R em cada experimento no estudo de caso 9.....	277
Tabela 100 – Melhores parâmetros globais encontrados pelo CMA-ES em cada experimento no estudo de caso 9.....	277

Tabela 101 – Resultados de RMSE da solução final de cada submodelo por experimento no estudo de caso 9 (parte 1).....	278
Tabela 102 – Resultados de RMSE da solução final de cada submodelo por experimento no estudo de caso 9 (parte 2).....	278
Tabela 103 – Resultados de SRG dos submodelos em cada experimento no estudo de caso 9	278

Lista de acrônimos

AE: Algoritmo Evolutivo

AED: Algoritmos de Estimção de Distribuição de Probabilidade

AEIQ: Algoritmo Evolutivo com Inspiração Quântica

AEIQ-B: Algoritmo Evolutivo com Inspiração Quântica e Representação Binária

AEIQ-BR: Algoritmo Evolutivo com Inspiração Quântica e Representação Binário-Real

AEIQ-R: Algoritmo Evolutivo com Inspiração Quântica e Representação Real

ARMA: *AutoRegressive Moving Average model*

ARMAX: *AutoRegressive Moving Average eXogenous model*

ARX: *AutoRegressive eXogenous model*

BPTT: *Backpropagation Through Time*

CMA-ES: *Covariance Matrix Adapatation Evolution Strategy*

CR: Computação de Reservatório

DaISy: *Database for the Identification of Systems*

ES: *Evolution Strategy*

ESNs: *Echo State Networks*

GA: *Genetic Algorithm*

KNN: *K Nearest Neighbors*

MIMO: *Multiple Input Multiple Output*

MISO: *Multiple Input Single Output*

MLP: *Multilayer Perceptron*

NARMAX: *Nonlinear AutoRegressive Moving Average eXogenous model*

NARX: *Nonlinear AutoRegressive eXogenous model*

NFIR: *Nonlinear Finite Impulse Response model*

NOE: *Nonlinear Output Error model*

PSO: *Particle Swarm Optimization*

RBF: *Radial Basis Function*

RLS: *Recursive Least Squares*

RMSE: *Root Mean Squared Error*

RNA: *Redes Neurais Artificiais*

RNAF: *Redes Neurais Artificiais Feedforward*

RNAR: *Redes Neurais Artificiais Recorrentes*

RTRL: *Real Time Recurrent Learning*

SIMO: *Single Input Multiple Output*

SISO: *Single Input Single Output*

SR: *Separation Ratio*

SRG: *Separation Ratio Graph*

SVM: *Support Vector Machine*

WTSG: *Water Tube Steam Generator*

1

Introdução

Modelos matemáticos representam uma ferramenta de grande importância em diversos campos da ciência e da engenharia, pois através deles é possível simular o funcionamento de processos e fenômenos do mundo real, permitindo-se adquirir um maior entendimento sobre os mesmos, analisar e prever o comportamento de suas variáveis, além de serem fundamentais em tarefas de controle, projeto e otimização de sistemas (Aguirre, 2015; Billings, 2013; Nelles, 2001). Soma-se a isto o fato de os experimentos envolvendo a simulação de modelos serem normalmente mais rápidos, seguros e econômicos, pois não utilizam recursos do sistema original, e permitem avaliar configurações não existentes do mesmo e realizar testes que poderiam ser perigosos no mundo real (Billings, 2013). Desta forma, a obtenção de um modelo mais adequado à representação do sistema em estudo é de grande interesse, o que normalmente é conseguido por meio de técnicas que podem ser agrupadas em dois grandes grupos: modelagem pela física e identificação de sistemas (Aguirre et al., 1998).

Na modelagem pela física (ou caixa branca ou fenomenológica), é utilizado todo o conhecimento disponível sobre o sistema (Aguirre et al., 1998), partindo-se de leis físicas e equações matemáticas que governam o seu funcionamento (Aguirre, 2015). Entretanto, em muitas situações práticas, tais equações não estão disponíveis ou são difíceis de serem obtidas e terem seus parâmetros estimados em um tempo razoável, devido à complexidade do sistema a ser modelado. Neste contexto, a identificação de sistemas ou modelagem caixa preta se apresenta como uma técnica alternativa de modelagem.

Identificação de sistemas é uma área de modelagem matemática (Aguirre, 2015) em que são estudados métodos e algoritmos voltados à obtenção de modelos matemáticos de sistemas dinâmicos a partir de dados medidos de suas entradas e saídas (Aguirre, 2015; Billings, 2013; Ljung, 1987), pressupondo pouco ou nenhum conhecimento prévio sobre os mesmos. Neste sentido, quando a modelagem caixa branca é inviável de ser aplicada, as técnicas utilizadas em identificação de sistemas se tornam, normalmente, opções mais práticas e rápidas de serem utilizadas (Billings, 2013), pois isentam o usuário de conhecer

a fundo o sistema a ser modelado, bem como as relações matemáticas que descrevem o seu funcionamento.

Com a disponibilidade cada vez maior de computadores e, conseqüentemente, uma maior capacidade e facilidade para se monitorar processos e coletar dados sobre a dinâmica dos mesmos (Aguirre, 2015; Martins, 2016), o procedimento utilizado em identificação de sistemas tem ganhado destaque e se tornado necessário e desejável em praticamente todas as áreas do conhecimento humano (Aguirre et al., 1998). Outra tendência recente é a substituição de modelos lineares para sistemas por seus correspondentes não lineares (Aguirre et al., 1998), como as redes neurais artificiais, devido a estes terem uma capacidade maior para reproduzir certos fenômenos e comportamentos dinâmicos mais complexos (Aguirre, 2015; Aguirre & Mendes, 1996) do que aqueles.

Redes neurais artificiais (RNAs) são sistemas computacionais de processamento altamente paralelo e distribuído (Haykin, 2009), cujo princípio de funcionamento é inspirado no cérebro de seres vivos (Nelles, 2001; Silva et al., 2010), sendo capazes de aprender por meio de exemplos, armazenar e utilizar o conhecimento adquirido (Braga et al., 2000), além de atuarem como aproximadores universais para diversas classes de funções não lineares (Silva et al., 2010; Braga et al., 2000). Estas características conferem às RNAs a habilidade de assimilar complexas relações não lineares presentes em dados de entrada e saída e de simular o mecanismo que os produziu (Chen & Billings, 1992), tornando-as, portanto, boas candidatas para representar sistemas não lineares em aplicações de identificação de sistemas, seja na forma de suas arquiteturas *feedforward* (Narendra & Parthasarathy, 1990; Rego et al., 2014) ou recorrentes (Wang & Chen, 2006; Coban, 2013).

Redes neurais artificiais *feedforward* (RNAF), como as *multilayer perceptron* (MLP) e as *radial basis function* (RBF), são normalmente as arquiteturas de RNA mais utilizadas em identificação de sistemas (Schrauwen et al., 2007). Porém, RNAF são estáticas, devendo a informação temporal do sistema a ser modelado ser inserida de maneira externa através de atrasos temporais nas variáveis que compõem sua entrada. Este procedimento converte um problema temporal em espacial, alimentando-se iterativamente a rede por meio de janelas temporais, o que, além de ser uma forma não natural de tratar informações temporais, pode requerer um número grande de variáveis de entrada, especialmente em problemas de identificação de sistemas de alta ordem e/ou multivariáveis, o que resulta, ainda que linearmente no caso de RNA, em um aumento do

número de parâmetros do modelo (Nelles, 2001; Schrauwen et al., 2007) e na necessidade de seleção de quais variáveis regressoras são realmente relevantes para a aplicação.

Diferentemente das redes *feedforward*, redes neurais artificiais recorrentes (RNAR) possuem ligações de realimentação em sua estrutura (Haykin, 1999), o que faz com que as mesmas exibam uma memória interna, conseguindo armazenar e acessar o histórico de suas entradas e saídas. Desta forma, RNAR são naturalmente dinâmicas e conseguem representar de maneira direta, sem a necessidade de atrasos temporais em suas entradas, sistemas dinâmicos, nos quais suas respostas atuais dependem de seus estados anteriores. Sendo assim, RNAR se tornam ferramentas apropriadas à utilização não só em identificação de sistemas, mas também em outros problemas de natureza temporal como previsão de séries temporais e controle de processos (Silva et al., 2010). Por outro lado, RNAR são mais difíceis de treinar do que RNAF, tendo seus esquemas clássicos de treinamento, baseados na descida do gradiente em uma função de erro, como o *Backpropagation Through Time* (BPTT) (Werbos, 1990) e o *Real Time Recurrent Learning* (RTRL) (Williams; Zipser, 1989), apresentado problemas como convergência lenta, soluções subótimas e altíssimo custo computacional (Jaeger, 2010; Duan & Wang, 2016; Ozturk et al., 2007), o que limita bastante o uso de redes deste tipo. Muito dos problemas citados foram superados com a proposição, em 2001, por Herbert Jaeger, das *Echo State Networks*.

As *Echo State Networks* (ESNs) (Jaeger, 2010) são RNAR que possuem um processo de treinamento simplificado e de baixo custo computacional, sendo uma implementação simples e eficiente do paradigma conhecido como Computação de Reservatório (*Reservoir Computing, em inglês*), que também abrange métodos como *Liquid State Machines* (Maass et al., 2002), *Backpropagation Decorrelation* (Steil, 2004) e *Evolino* (Schmidhuber et al., 2007), os quais compartilham algumas similaridades com as ESNs. A simplicidade no funcionamento e no treinamento deste tipo de rede está relacionada à forma diferenciada como são tratadas a sua camada escondida, denominada reservatório dinâmico, e a sua camada de saída. O reservatório é uma arquitetura de rede recorrente e não linear composta por um grande número de neurônios inter e/ou auto conectados de maneira esparsa e aleatória, cuja matriz \mathbf{W} , que representa os pesos das conexões no reservatório, permanece fixa após a inicialização. Ainda, caso a matriz \mathbf{W} satisfaça determinadas condições (Jaeger, 2010), as conexões recorrentes no reservatório criam “ecos” dos sinais apresentados ao mesmo (Skowronski & Harris, 2007a), fazendo com que as saídas de seus neurônios (estados) atuem como uma função do histórico de

entradas e/ou saídas, conferindo às ESNs uma memória dinâmica. Diferentemente do reservatório, que faz uma expansão temporal e não linear dos sinais a ele apresentados, a camada de saída é responsável por realizar um mapeamento estático (e normalmente linear) dos estados produzidos pelo reservatório a fim de gerar os sinais de saída da rede. Como os pesos internos do reservatório, bem como os de conexões para o mesmo, permanecem fixos após a inicialização, somente os pesos de conexões à camada de saída são ajustados durante o aprendizado, o que é uma tarefa de regressão linear e com solução única (Boccatto et al., 2011; Xi et al., 2005), conferindo simplicidade e rapidez ao treinamento das ESNs. Desta forma, as ESNs se apresentam como uma alternativa bastante viável e atraente de se aplicar RNAR, o que já tem sido explorado por diversos autores, com relatos de bom desempenho, em aplicações como: previsão de séries temporais (Liu et al., 2015; Chitsazan et al., 2017; Jaeger & Haas, 2004; Babinec & Pospíchal, 2007a, 2007b; Ferreira & Ludermir, 2009, 2010, 2011; Sergio & Ludermir, 2012; Ferreira et al., 2013; Bianchi et al., 2015; Chouikhi et al., 2015, 2017), restauração de imagens (Duan & Wang, 2016), segmentação de imagens (Meftah et al., 2016; Jose & Mythili, 2008), reconhecimento de sinais de voz (Skowronski; Harris, 2007a; Verstraeten et al.; 2006; Skowronski & Harris, 2007b), controle não linear (Salmen & Plöger, 2005; Plöger et al., 2004; Wu et al., 2012; Xu et al., 2005; Park et al., 2017), equalização de canais (Boccatto et al., 2011; Jaeger & Haas, 2004), aprendizado por reforço (Jiang et al., 2008; Chatzidimitriou & Mitkas, 2010, 2013; Koprinkova-Hristova et al., 2010), agrupamento de dados (Koprinkova-Hristova & Angelova, 2013) e identificação de sistemas (Ishu et al., 2004; Xi et al., 2005; Deng & Zhang, 2006; Squartini et al., 2007; Zhou et al., 2010; Bian & Mou, 2011; Song et al., 2011; Kim & Billard, 2012; Li et al., 2012; Pan & Wang, 2012; Xing et al., 2012; Han & Wang, 2014; Lin et al., 2014; Lun et al., 2014; Cao et al., 2015; Antonelo et al., 2015; Lin & Zhang, 2016; Magerl, Stockreiter et al., 2016; Magerl, Ceperic et al.; 2015; Moon et al., 2016; Martins et al., 2015).

Algoritmos evolutivos com inspiração quântica (AEIQs) são fundamentados em conceitos da física quântica, como o de superposição de estados (Cruz, 2007), sendo considerados pertencentes à classe de algoritmos de estimação de distribuição (AED) (Defoin-Platel et al., 2009). Em comparação aos algoritmos genéticos tradicionais, os AEIQs, em geral, convergem de maneira mais eficiente, encontrando boas soluções em menos tempo, mesmo com populações com poucos indivíduos, o que é vantajoso para problemas de otimização em que uma avaliação é custosa computacionalmente (Cruz, 2007). Uma representação binária e bem sucedida para tais algoritmos, é o AEIQ-B

(Algoritmo Evolutivo com Inspiração Quântica e representação Binária), proposto por Han e Kim (2000), o qual obteve êxito em problemas de otimização combinatorial (Han & Kim, 2000, 2002) e detecção de faces (Jang et al., 2004). Posteriormente, uma versão de AEIQ mais adequada à solução de problemas numéricos, foi proposta por Cruz (2007) e denominada de AEIQ-R (Algoritmo Evolutivo com Inspiração Quântica e representação Real), superando o desempenho de diversos algoritmos convencionais em problemas de otimização de funções numéricas e no treinamento de uma arquitetura de rede neural recorrente tanto em casos supervisionados quanto não supervisionados (Cruz, 2007).

A Estratégia Evolutiva com Adaptação da Matriz de Covariâncias (CMA-ES - *Covariance Matrix Adaptation Evolution Strategy*) é um algoritmo evolutivo do tipo AED, assim como os AEIQs, que foi proposto inicialmente por Hansen & Ostermeier (1996). Este algoritmo tem seu princípio de funcionamento calcado na amostragem de soluções a partir de uma distribuição normal multivariada, cujos parâmetros (média, passo e matriz de covariâncias) vão sendo adaptados iterativamente ao longo do processo evolutivo (Strickler, 2017). Com este mecanismo de adaptação, o CMA-ES consegue superar alguns dos problemas enfrentados por algoritmos evolutivos, sendo capazes de lidar bem com a otimização de funções altamente não separáveis (Hansen, 2016), evitar situações de convergência prematura (Hansen, 2016), conseguir trabalhar bem com populações com poucos indivíduos (Hansen, 2016), o que reduz o tempo necessário para convergência, além de ser um algoritmo com pouquíssimos parâmetros a serem configurados pelo usuário, o que facilita em muito seu uso. Com estas vantagens, o CMA-ES é um algoritmo evolutivo altamente competitivo, sendo considerado o otimizador estado da arte para funções mono-objetivas contínuas (Hansen et al., 2010) e aplicado com sucesso em diversos problemas reais (Hansen, 2009, 2016).

1.1. Motivação

Apesar das vantagens citadas e dos bons resultados em diversas aplicações, as ESNs são sensíveis a determinados parâmetros necessários à sua geração (parâmetros gerais ou globais ou hiperparâmetros), os quais devem ser ajustados de forma a extrair o melhor desempenho das mesmas em uma determinada tarefa, como o tamanho e a conectividade do seu reservatório e o raio espectral associado à sua matriz \mathbf{W} , além de outros como

fatores de escala de suas matrizes de entrada e de *feedback* e constantes de neurônios do tipo *leaky integrator* (Jaeger, 2010). O processo de ajuste manual destes parâmetros pode ser difícil e/ou trabalhoso, dependendo de experimentação, conhecimento prévio sobre os mesmos, intuição, sorte e experiência de uso (Ishu et al., 2004). Uma estratégia comum para solucionar este problema é associar às ESNs uma técnica de otimização, como os algoritmos evolutivos, tornando automática a sintonia de seus parâmetros gerais, assim como melhorando o desempenho das mesmas na tarefa pretendida (Ferreira & Ludermir, 2009; Sergio & Ludermir, 2012; Martins et al., 2015; Niño, 2016; Martins, 2016).

Uma outra questão a ser levantada sobre o desempenho das ESNs é que embora um reservatório esparso e aleatório conduza a uma estratégia simples de treinamento, é de se esperar que uma rede com os mesmos hiperparâmetros, mas cujo reservatório seja idealmente ajustado para uma determinada tarefa possa apresentar um desempenho superior (Ferreira, 2011; Boccato et al., 2011). Em vista disto, com o objetivo de melhorar esta estratégia não totalmente satisfatória de inicialização de pesos, diversos esquemas para seleção/adaptação da camada interna das ESNs têm sido foco de pesquisa na literatura, dentre eles, o uso de regras de aprendizado não supervisionadas inspiradas na plasticidade sináptica biológica (Williams & Zipser, 1989), como Anti-Hebb (Babinec & Pospíchal, 2007a), Anti – Oja (Jaeger & Haas, 2004; Yusoff et al., 2016) e Bienenstock-Cooper-Munro (Yusoff et al., 2016), para treinamento dos pesos do reservatório, tal como o ajuste de ganhos e *bias* de neurônios internos por meio de regras baseadas no gradiente descendente e inspiradas na plasticidade intrínseca de neurônios biológicos (Jose & Mythili, 2008; Steil, 2007; Verstraeten et al., 2007), além da proposição de diferentes métricas de qualidade que permitem avaliar o reservatório em uma dada tarefa antes de efetivamente aplicá-lo para tal, como o coeficiente de Lyapunov (Gibbons, 2010; Verstraeten et al., 2007; Schrauwen et al., 2008; Legenstein & Maass, 2007), coeficientes e gráficos de razão de separação em classificação (Gibbons, 2010) e previsão (Niño, 2016), entre outras. Como já pesquisado por outros autores (Ferreira, 2011; Ishu et al., 2004; Ferreira & Ludermir, 2010, 2011; Chatzidimitriou & Mitkas, 2013; Chouikhi et al., 2017), algoritmos evolutivos (AEs) e outras técnicas de otimização bioinspiradas, além de atuarem como otimizadores de hiperparâmetros para ESNs, podem otimizar a camada interna das mesmas, atuando diretamente no ajuste dos pesos das conexões do reservatório.

A otimização de parâmetros gerais, ainda que resulte em um espaço de busca menor, devido à natureza aleatória do reservatório, pode gerar ruídos na função de aptidão

(Ferreira & Ludermir, 2011), causando imprecisões na determinação dos hiperparâmetros e dificultando a convergência de algoritmos evolutivos. Tal dificuldade pode ser contornada através da obtenção do desempenho médio de várias ESNs para uma mesma configuração de hiperparâmetros (Martins, 2016), porém, esta estratégia aumenta o tempo requerido para uma avaliação devido à inicialização e o treinamento de várias redes. Por outro lado, otimizar diretamente os pesos do reservatório pode levar a um custo computacional significativo, devido ao grande número de parâmetros a serem otimizados (N^2 , sendo N o número de neurônios internos) (Ferreira & Ludermir, 2011). Portanto, nas duas situações de otimização, é importante que o algoritmo evolutivo utilizado atinja um bom desempenho com um menor número de avaliações possível, de forma a poupar tempo. Neste sentido, por teoricamente fornecerem boas soluções com poucas avaliações, o Algoritmo Evolutivo com Inspiração Quântica e Representação Real (Cruz, 2007) e a Estratégia Evolutiva com Adaptação da Matriz de Covariâncias (Hansen & Ostermeier, 1996) surgem como alternativas potencialmente promissoras de AEs a serem aplicadas.

1.2. Objetivos

Do exposto no subitem acima, o objetivo principal desta dissertação é implementar um modelo neuroevolutivo composto por *Echo State Networks* otimizadas por um algoritmo evolutivo para aplicações de identificação de sistemas. No modelo proposto, o processo de ajuste das *Echo State Networks* terá duas etapas, sendo a etapa 1 focada na otimização dos parâmetros globais necessários à geração das ESNs por um algoritmo evolutivo e a etapa 2 centrada na determinação do reservatório mais adequado para os hiperparâmetros definidos na etapa anterior. Como alternativas de algoritmos evolutivos na primeira etapa serão utilizados o AEIQ-R e o CMA-ES. Para a etapa 2, serão avaliadas três opções para a seleção do reservatório, sendo a primeira uma escolha simples baseada em ESNs com reservatórios aleatórios, a segunda baseada no coeficiente SRG de Niño (2016) e uma terceira conduzida através de uma nova otimização focada no ajuste dos pesos do reservatório utilizando o algoritmo evolutivo da etapa anterior.

O modelo neuroevolutivo proposto permitirá encontrar automaticamente a ESN mais adequada, em termos de parâmetros globais e de pesos do reservatório, para problemas de identificação de sistemas dinâmicos utilizando técnicas evolutivas (AEIQ-R ou CMA-ES) capazes de buscar em um tempo computacional adequado por uma

representação final que simule com boa precisão o sistema em estudo. Do ponto de vista do usuário, será abstraído do mesmo o conhecimento a respeito dos hiperparâmetros das ESNs, exigindo apenas a configuração de alguns parâmetros do AEIQ-R e praticamente nenhum do CMA-ES. As diferentes alternativas de composição do modelo serão avaliadas em bases de dados *benchmark* em identificação de sistemas disponíveis publicamente.

1.3. Descrição da dissertação

Esta dissertação foi elaborada seguindo-se as seguintes etapas:

- **Revisão bibliográfica** – Pesquisou-se na literatura sobre trabalhos que envolviam a utilização de *Echo State Networks* em Identificação de Sistemas e sobre a otimização destas redes por algoritmos evolutivos e/ou técnicas similares.

- **Embasamento teórico** – Nesta etapa, inicialmente foi realizado um estudo sobre as principais representações não lineares utilizadas em identificação de sistemas, bem como sobre o uso de redes neurais artificiais nesta área. Posteriormente, estudou-se sobre *Echo State Networks*, abordando o princípio de funcionamento, arquitetura, treinamento e parâmetros que afetam o desempenho destas redes, além de um método para avaliação e seleção do reservatório que utiliza o coeficiente SRG de Niño (2016). Finalmente, foram estudados os Algoritmos Evolutivos com Inspiração Quântica, abordando as representações binária, real e binária-real, e as estratégias evolutivas, com foco na Estratégia Evolutiva com Adaptação da Matriz de Covariâncias.

- **Concepção do modelo** – O modelo foi implementado no ambiente do MATLAB (MATLAB 2016b), utilizando como base o *toolbox* para ESNs de Jaeger (2009), o algoritmo do AEIQ-R de Cruz (2007), a versão em MATLAB do CMA-ES (Hansen, 2012) e o código fonte do coeficiente SRG (Niño, 2016). Como exposto anteriormente, o modelo foi elaborado com o intuito de tornar automática a configuração das ESNs em problemas de identificação de sistemas, focando-se no ajuste de seus parâmetros gerais numa primeira etapa e na seleção do reservatório mais adequado para estes hiperparâmetros numa segunda fase, além de se utilizar algoritmos evolutivos que fornecem boas soluções com poucas avaliações para reduzir o tempo computacional da aplicação.

• **Estudos de casos** – Com a elaboração do modelo no ambiente do MATLAB, para a avaliação dos diferentes abordagens do modelo proposto foram selecionados problemas *benchmarks* em identificação de sistemas de variadas características, cujas bases de dados estão disponíveis publicamente no repositório DaISy – *Database for the Identification of Systems* (De Moor, 2018) e no artigo “Three free data sets for development and benchmarking in nonlinear system identification” (Wigren, 2013). Os resultados deste trabalho foram comparados com autores que utilizaram as mesmas bases de dados, além das abordagens propostas serem avaliadas por meio de testes estatísticos.

1.4.

Organização do Trabalho

O restante deste trabalho está organizado da seguinte maneira:

• **Capítulo 2** – São apresentados alguns conceitos em identificações de sistemas e as principais representações não lineares, inclusive redes neurais artificiais, utilizadas nesta área.

• **Capítulo 3** – Aborda as *Echo State Networks*, cobrindo-se esquema básico de funcionamento, arquitetura, treinamento e principais parâmetros de configuração. Também abordam-se alguns métodos para avaliação do reservatório e discorre-se sobre trabalhos da literatura que utilizaram *Echo State Networks* em aplicações de identificação de sistemas, além de outros que otimizaram ESNs por meio de algoritmos evolutivos.

• **Capítulo 4** – Apresenta os Algoritmos Evolutivos com Inspiração Quântica nas suas representações binária e real, algumas estratégias evolutivas clássicas e a Estratégia Evolutiva com Adaptação da Matriz de Covariâncias, passando-se pelo pseudocódigo e pela descrição das principais etapas de cada algoritmo.

• **Capítulo 5** – É apresentado o modelo neuroevolutivo proposto, iniciando-se pela descrição geral do mesmo e posteriormente detalhando-se cada uma de suas duas etapas com os algoritmos utilizados, os parâmetros globais otimizados e o esquema para seleção do reservatório.

• **Capítulo 6** – São descritos os problemas utilizados como estudos de caso e apresentados os resultados obtidos com as abordagens propostas neste trabalho. Em cada um dos problemas são comparados os resultados obtidos neste trabalho com os de métodos que utilizaram os mesmos conjuntos de teste nas bases de dados. Testes estatísticos são aplicados para comparar as abordagens propostas.

- **Capítulo 7** – São apresentadas as conclusões do trabalho e as sugestões para pesquisas futuras.
- **Capítulo 8** – Apresenta as referências bibliográficas utilizadas no trabalho.
- **Apêndice A** – Apresenta os principais resultados dos submodelos em todos os experimentos realizados.

2 Identificação de Sistemas Não Lineares

2.1. Introdução

Um modelo (matemático) de um sistema é um análogo matemático do mesmo (Aguirre, 2015) que consegue representar, de maneira aproximada, características essenciais de seu comportamento (Chinarro, 2014). Através de um modelo é possível simular o funcionamento de um sistema e estimar como o mesmo se comportaria em determinadas condições operacionais, sem a necessidade de realização de testes experimentais com o processo real, o que é, em geral, um procedimento mais econômico e seguro (Coelho & Coelho, 2004). Entretanto, a adequação dos resultados obtidos durante a simulação depende da obtenção de um modelo que consiga representar adequadamente as características desejadas do sistema alvo para um determinado fim (Coelho & Coelho, 2004). Entre as abordagens utilizadas para a modelagem de sistemas, destacam-se:

- Modelagem caixa branca ou física ou analítica – onde um modelo é totalmente obtido por meio do equacionamento das relações matemáticas entre as variáveis do sistema. Requer conhecimento profundo sobre o sistema, bem como das leis físicas, químicas, biológicas, etc que descrevem os fenômenos nele existentes. Pode ser difícil e/ou demorada de ser aplicada em sistemas complexos, tendo-se que, em muitos casos, serem consideradas simplificações e/ou aproximações sobre o funcionamento dos mesmos, porém a estrutura do modelo resultante tem relação direta com a do sistema real.

- Modelagem caixa preta ou empírica – na qual o modelo de um sistema é completamente derivado dos dados de suas variáveis de entrada e saída. Requer pouco ou nenhum conhecimento sobre o sistema, sendo normalmente uma alternativa mais simples e rápida à abordagem caixa branca, porém o modelo obtido por este mecanismo geralmente não reflete a estrutura do sistema real, sendo mais difícil sua interpretação física.

- Modelagem caixa cinza – representa um meio termo entre as duas modelagens acima citadas, onde ambas as abordagens estão presentes em algum nível no processo de

obtenção de um modelo matemático. Desta forma, existem diferentes tipos de se realizar uma modelagem caixa cinza, como, por exemplo, a incorporação de algum conhecimento prévio sobre o sistema que ajude na determinação da estrutura de um modelo caixa preta, tornando-o mais preciso e/ou fisicamente mais significativo, ou a estimação dos parâmetros de um modelo derivado por relações físico-matemáticas através de dados de entrada e saída.

A segunda categoria de modelagem supracitada (caixa preta) é normalmente referida como Identificação de Sistemas (Aguirre, 2015). A Identificação de Sistemas tem como objetivo obter modelos matemáticos de sistemas dinâmicos a partir de dados observados de suas entradas e saídas (Ljung, 1987), requerendo pouco ou nenhum conhecimento sobre os mesmos. As técnicas utilizadas nesta área baseiam-se na informação de entrada e saída de um sistema dinâmico para estimar modelos que visam explicar, pelo menos em parte e de forma aproximada, a relação de causa e efeito presente nos dados (Aguirre, 2015), abstraindo-se do que ocorre internamente no sistema. Devido aos significativos avanços tecnológicos em diferentes áreas, como engenharia elétrica, ciência da computação e medicina, o interesse em Identificação de Sistemas é crescente (Coelho & Coelho, 2004).

Muitos sistemas reais são não lineares. Ainda nestes casos, embora a utilização de modelos lineares seja mais simples e de obtenção mais fácil, tais modelos conservam sua acurácia somente nas proximidades de determinado ponto de operação do sistema (Moreira, 2008). Por outro lado, modelos não lineares conseguem representar um conjunto maior e mais complexo de comportamentos dinâmicos (Aguirre, 2015; Aguirre & Mendes, 1996) e em uma faixa maior de operação do sistema (Moreira, 2008) em comparação a modelos lineares.

Desta forma, este capítulo tem como objetivo fazer uma rápida introdução à Identificação de Sistemas, revisando alguns conceitos sobre modelos e sistemas e descrevendo as principais etapas do processo de identificação, além de apresentar brevemente os principais modelos não lineares utilizados nesta área.

2.2. Definições sobre sistemas e modelos

Um sistema pode ser definido como uma entidade em que variáveis de diferentes tipos interagem entre si e produzem sinais observáveis (saídas) (Ljung, 1987; Keesman, 2011). Um sistema pode receber estímulos externos provenientes de sinais de entrada, que são manipuláveis e mensuráveis, ou de ruídos ou distúrbios, os quais não são manipuláveis e podem ser mensuráveis ou não (Ljung, 1987; Keesman, 2011) (Figura 1). As variáveis internas de um sistema são denominadas estados e resumem os efeitos de entradas e ruídos anteriores sobre o mesmo (Keesman, 2011).

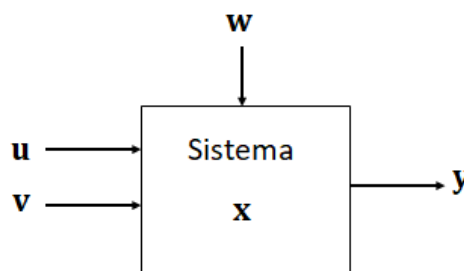


Figura 1 - Variáveis de um sistema: u – sinais de entrada, v – ruídos mensuráveis, w – ruídos não mensuráveis, x – variáveis de estado do sistema, y – saídas. Fonte: Adaptado de Ljung (1987) e Keesman (2011)

Um sistema dinâmico é um sistema que possui memória, em outras palavras, suas respostas atuais dependem também de valores de estímulos anteriores (Ljung, 1987). Neste caso, as relações entre as variáveis do sistema são modeladas por meio de equações diferenciais ou de diferenças. Por outro lado, sistemas estáticos não possuem memória, sendo um caso especial dos sistemas dinâmicos onde suas saídas em um instante de tempo qualquer são dependentes somente dos valores das entradas naquele instante de tempo. Assim sendo, as relações entre as variáveis de um sistema estático são descritas por meio de equações algébricas.

Sistemas lineares possuem o mesmo comportamento independente do ponto de operação (Martins, 2016). Matematicamente, sistemas lineares satisfazem o princípio da superposição. Por outro lado, sistemas não lineares apresentam diferentes comportamentos em diferentes pontos de operação (Martins, 2016), que é o caso da maioria dos sistemas reais, e não satisfazem o princípio da superposição.

Um sistema é considerado invariante no tempo se o seu comportamento não muda com o tempo (Aguirre, 2015) e variante, caso contrário. Em outras palavras, pode-se dizer

que em um sistema invariante no tempo, a lei que determina como as variáveis evoluem no tempo, conhecida como dinâmica do sistema, permanece inalterada (Aguirre, 2015).

Como anteriormente citado, um modelo matemático é uma representação simplificada de um sistema, na forma de uma equação matemática, que reproduz aproximadamente algumas de suas características. Assim como os sistemas, os modelos também podem ser lineares ou não lineares, dinâmicos ou estáticos. Os modelos ainda podem ser classificados em: discretos ou contínuos no tempo, monovariáveis ou multivariáveis, estocásticos ou determinísticos, paramétricos ou não paramétricos.

Modelos contínuos no tempo descrevem as suas variáveis continuamente ao longo do tempo, enquanto que modelos discretos relacionam suas variáveis apenas em instantes específicos de tempo, nos quais os sinais são medidos ou amostrados com uma certa periodicidade.

Modelos monovariáveis são os que possuem somente uma entrada e uma saída, sendo também conhecidos como modelos SISO (*Single Input Single Output*). Modelos multivariáveis possuem mais de uma entrada e/ou mais de uma saída, que é o caso dos modelos SIMO (*Single Input Multiple Output*), MISO (*Multiple Input Single Output*) ou MIMO (*Multiple Input Multiple Output*).

Modelos determinísticos não possuem incerteza em seus parâmetros ou variáveis, enquanto que modelos estocásticos incluem algum tipo de incerteza em seus parâmetros ou variáveis para lidar com processos aleatórios presentes em sistemas reais (Aguirre, 2015). Nesta situação, as saídas de modelos estocásticos são variáveis do tipo aleatórias.

Modelos paramétricos, na maioria das vezes, referem-se a modelos que possuem coeficientes ou números que os caracterizam, enquanto que modelos não paramétricos são representações gráficas, como a resposta ao impulso ou a resposta em frequência de um sistema (Aguirre, 2015).

2.3. Procedimento típico de Identificação

Supondo-se que estejam disponíveis dados de sinais de entrada ($\mathbf{u}(k)$) e de saída ($\mathbf{y}(k)$) de um sistema desconhecido, de uma maneira simplificada, as técnicas utilizadas em Identificação de Sistemas objetivam encontrar um modelo que ao ser excitado por $\mathbf{u}(k)$ retornem sinais de saída ($\hat{\mathbf{y}}(k)$) mais próximos de $\mathbf{y}(k)$. Para isto, parâmetros do

modelo são ajustados segundo algum critério de desempenho estabelecido, como, por exemplo, a minimização de uma função de erro ($e(k)$) entre as saídas do modelo ($\hat{y}(k)$) e as saídas desejadas ($y(k)$) (Figura 2).

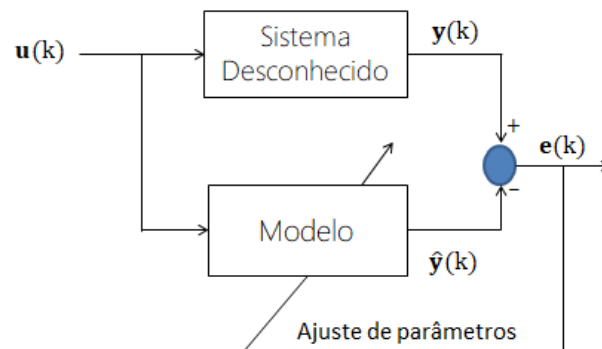


Figura 2. O problema geral de Identificação de Sistemas. Fonte: Adaptado de Chinarro (2014)

De forma a atingir tal objetivo, normalmente o procedimento empregado em Identificação de Sistemas consiste de 4 etapas principais: experimentação e coleta de dados; seleção do modelo e de sua estrutura; estimação de parâmetros e validação do modelo (Figura 3).

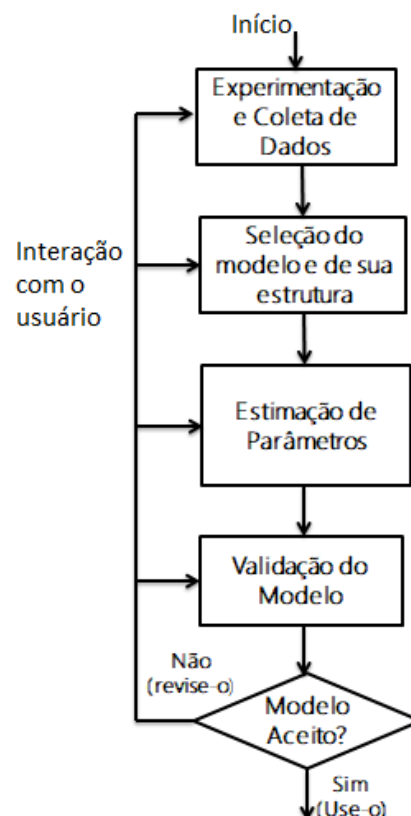


Figura 3. Procedimento adotado em Identificação de Sistemas. Fonte: Adaptado de Tavakolpour-Saleh et al. (2015)

Na etapa de experimentação e coleta de dados são gerados os dados necessários ao processo de identificação. Quando possível, testes dinâmicos devem ser aplicados ao

sistema de forma que o comportamento dinâmico de interesse esteja presente nos dados. Neste caso, devem ser definidos adequadamente o período de amostragem e os tipos de sinais utilizados como entrada. No caso de identificação de sistemas lineares é mais importante o conteúdo de frequências do espectro dos sinais de entrada, enquanto que no caso não linear, a forma dos sinais de excitação também é importante (Aguirre, 2015). Quando não é possível por algum motivo realizar tais testes em uma planta real, dados de operação normal do sistema podem ser utilizados (Narendra & Parthasarathy, 1990).

A segunda etapa da Figura 3 envolve a escolha do modelo (ou conjunto de modelos), dentre os variados tipos possíveis (lineares ou não lineares, discretos ou contínuos no tempo, paramétricos ou não paramétricos, etc), para representar o sistema. No caso de representações lineares como o ARX (*AutoRegressive with eXogenous inputs*) e o ARMAX (*AutoRegressive Moving Average with eXogenous inputs*), descritas em (Aguirre, 2015), a estrutura do modelo é completamente definida pela escolha do conjunto de termos regressores a serem utilizados como entrada (Petrovic et al., 2000). No caso de representações não lineares como NARX (*Nonlinear AutoRegressive with eXogenous inputs*) e o NARMAX (*Nonlinear AutoRegressive Moving Average with eXogenous inputs*), descritos posteriormente, além do conjunto de termos regressores, deve-se definir também o tipo de mapeamento não linear (funções polinomiais, racionais, redes neurais, etc) a ser utilizado no modelo (Petrovic et al., 2000).

Uma vez que a estrutura do modelo escolhido para representar o sistema é determinada, a equação matemática que o representa possui coeficientes desconhecidos (parâmetros), os quais devem ser determinados para a sua completa definição. A estimação de tais parâmetros corresponde a resolver um problema de otimização, no qual um algoritmo de estimação irá determinar os parâmetros mais adequados ao modelo de forma a minimizar uma função custo baseada em um critério de desempenho previamente escolhido. Se o modelo for linear nos parâmetros, o método dos mínimos quadrados pode ser aplicado, caso contrário métodos baseados na descida do gradiente ou algoritmos evolutivos podem ser utilizados.

A última fase do procedimento mostrado na Figura 3 (validação do modelo) visa verificar se o conjunto de modelos obtidos incorporam as características desejadas do sistema original, bem como se há um candidato significativamente melhor que os demais (Aguirre, 2015). Para tal, o procedimento mais usual é o de comparar o desempenho dos modelos finais em um conjunto de dados de validação (ou teste) que é diferente dos dados utilizados para estimá-los (dados de identificação ou treinamento). Dependendo da

aplicação, o modelo resultante pode ser utilizado para previsão de 1 passo à frente (configuração série-paralelo) ou para simulação ou previsão livre (configuração paralela) (Figuras 4 e 5). No caso da previsão de 1 passo à frente, o modelo dispõe dos valores de entrada e saída do sistema real até o passo anterior para fazer a previsão (Figura 4). No caso da simulação ou previsão livre, o modelo utiliza como entrada somente os valores anteriores dos sinais de entrada do sistema e/ou valores anteriores de suas próprias previsões (Figura 5).

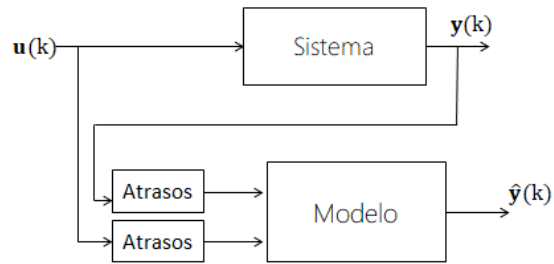


Figura 4. Configuração série-paralelo utilizada na previsão 1 passo à frente

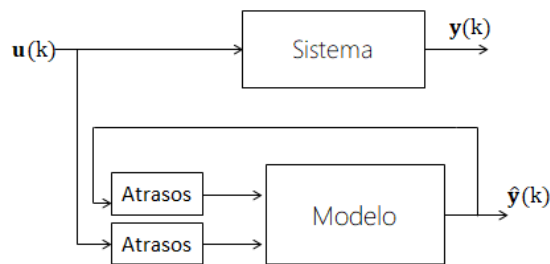


Figura 5. Configuração paralela utilizada na simulação ou previsão livre

Segundo Aguirre (2015), os indicadores de previsão de 1 passo à frente não são confiáveis para validar um modelo, pois em alguns casos, dependendo do período de amostragem, pode-se ter pouca diferença entre uma amostra e outra e até mesmo um modelo “ingênuo” pode obter um bom desempenho na previsão. Neste caso, para validação, é preferível uma métrica baseada na previsão livre ou simulação, pois representa uma situação mais realística, onde as saídas do sistema real não estão disponíveis e ainda assim, o modelo resultante deve reproduzir aproximadamente o comportamento do sistema.

Caso o modelo final não atenda as expectativas na validação, uma das etapas anteriores deve ser revisada.

2.4. Representações não lineares

Nesta seção serão revistos alguns modelos não lineares utilizados para representar sistemas. Para simplificar a notação das equações, será assumido que os modelos são do tipo SISO, porém tais equações podem ser naturalmente estendidas para o caso multivariável. O foco se dará na descrição de modelos discretos no tempo.

2.4.1. Modelos NARX, NARMAX e seus derivados

Os modelos descritos nesta seção são também conhecidos como modelos do tipo entrada/saída e conseguem representar uma quantidade enorme de sistemas não lineares (Nelles, 2001). De uma maneira geral, tais modelos podem ser derivados da seguinte equação:

$$\hat{y}(k) = f(\boldsymbol{\varphi}(k)) \quad (1)$$

onde k é um passo de tempo discreto, \hat{y} é a saída do modelo, $f(\cdot)$ é uma função não linear e $\boldsymbol{\varphi}(k)$ é o vetor de regressores.

O vetor de regressores $\boldsymbol{\varphi}(k)$ pode incluir valores anteriores de sinais de entrada, das saídas do sistema ou do modelo e de erros de previsão. Se $\boldsymbol{\varphi}(k)$ contém valores anteriores dos sinais de entrada e saída do sistema, o modelo resultante é denominado NARX (*Nonlinear AutoRegressive with eXogenous inputs*), cuja equação é a seguinte:

$$\hat{y}(k) = f(u(k-1), \dots, u(k-n_u), y(k-1), \dots, y(k-n_y)) \quad (2)$$

onde $u(k)$ e $y(k)$ são, respectivamente, os sinais de entrada e saída do sistema no passo de tempo k , enquanto que, n_u e n_y representam, respectivamente, os maiores atrasos na entrada e na saída.

Se em adição a valores passados de $u(k)$ e $y(k)$, valores anteriores de sinais de ruído ($e(k)$) são considerados em $\boldsymbol{\varphi}(k)$, o modelo resultante é denominado NARMAX (*Nonlinear AutoRegressive Moving Average with eXogenous inputs*), que é representado por:

$$\hat{y}(k) = f(u(k-1), \dots, u(k-n_u), y(k-1), \dots, y(k-n_y), e(k-1), \dots, e(k-n_e)) \quad (3)$$

onde $e(k)$ é o termo de ruído no passo k e n_e é maior atraso para $e(k)$.

Se em (2), o modelo é estimado em uma configuração paralela, onde ao invés das saídas do sistema, o vetor $\boldsymbol{\varphi}(k)$ utiliza valores anteriores das saídas do próprio modelo, o modelo NARX se torna o modelo NOE (*Nonlinear Output Error*), sendo descrito por:

$$\hat{y}(k) = f(u(k-1), \dots, u(k-n_u), \hat{y}(k-1), \dots, \hat{y}(k-n_y)) \quad (4)$$

Se o vetor $\boldsymbol{\varphi}(k)$ contém somente informações anteriores sobre o sinal de entrada $u(k)$, o modelo é denominado NFIR (*Nonlinear Finite Impulse Response*), cuja descrição é dada por:

$$\hat{y}(k) = f(u(k-1), \dots, u(k-n_u)) \quad (5)$$

A forma mais comum utilizada para implementar os modelos acima citados é denominada de “estratégia de dinâmica externa” (Nelles, 2001). Neste caso, a função $f(\cdot)$ é implementada por um modelo não linear estático, cujas entradas são versões atrasadas dos sinais $u(k)$ e/ou $y(k)$ e/ou $e(k)$ (Figura 6).

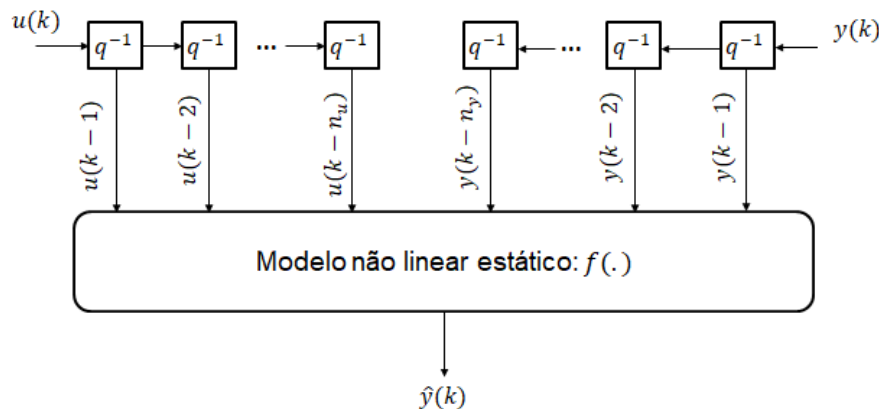


Figura 6. Estratégia de dinâmica externa, onde q^{-1} representa o operador de atraso unitário no tempo. Fonte: Adaptado de Nelles (2001)

Questões relativas à como implementar a função $f(\cdot)$, que valores atribuir a n_u , n_y e n_e e quais termos regressores utilizar em $\boldsymbol{\varphi}(k)$ fazem parte da determinação da estrutura destes modelos. Alternativas de modelos estáticos para a implementação de $f(\cdot)$ incluem funções polinomiais, funções racionais, redes neurais *feedforward*, modelos baseados na lógica *fuzzy* (Nelles, 2001).

Se uma função polinomial com grau de não linearidade $l \in \mathbb{N}$ (F^l) é utilizada para implementar o modelo NARMAX, o modelo resultante NARMAX polinomial é expresso por:

$$\hat{y}(k) = F^l[u(k-1), \dots, u(k-n_u), y(k-1), \dots, y(k-n_y), e(k-1), \dots, e(k-n_e)] \quad (6)$$

que na forma expandida corresponde a:

$$\begin{aligned} \hat{y}(k) = & \theta_0 + \sum_{i_1=1}^{n_T} \theta_{i_1} z_{i_1}(k) + \sum_{i_1=1}^{n_T} \sum_{i_2=1}^{n_T} \theta_{i_1 i_2} z_{i_1}(k) z_{i_2}(k) + \dots \\ & \dots + \sum_{i_1=1}^{n_T} \dots \sum_{i_l=i_1-1}^{n_T} \theta_{i_1 \dots i_l} z_{i_1}(k) \dots z_{i_l}(k) \end{aligned} \quad (7)$$

onde $\theta_{i_1 \dots i_l}$ são parâmetros do modelo, $n_T = n_u + n_y + n_e$, $z_1(k) = y(k-1), \dots, z_{n_y}(k) = y(k-n_y)$, $z_{n_y+1} = u(k-1), \dots, z_{n_y+n_u} = u(k-n_u)$, $z_{n_y+n_u+1} = e(k-1), \dots, z_{n_y+n_u+n_e} = e(k-n_e)$.

Um modelo NARMAX racional, por sua vez, pode ser expresso como uma razão entre duas funções polinomiais $A^{l_1}[\cdot]$ e $B^{l_2}[\cdot]$ de graus l_1 e l_2 , respectivamente:

$$\hat{y}(k) = \frac{A^{l_1}[y(k-1), \dots, y(k-n_y), u(k-1), \dots, u(k-n_u), e(k-1), \dots, e(k-n_e)]}{B^{l_2}[y(k-1), \dots, y(k-n_y), u(k-1), \dots, u(k-n_u), e(k-1), \dots, e(k-n_e)]} \quad (8)$$

Questões envolvendo a estimação de parâmetros dos modelos NARMAX polinomiais e racionais são discutidas por Aguirre (2015).

2.4.2. Representação por Espaço de Estados

Uma representação mais geral para sistemas dinâmicos tanto lineares quanto não lineares é através de modelos de espaço de estados, que são expressos por:

$$\begin{cases} x_1(k) = h_1(x_1(k-1), x_2(k-1), \dots, x_m(k-1), u(k-1)) \\ x_2(k) = h_2(x_1(k-1), x_2(k-1), \dots, x_m(k-1), u(k-1)) \\ \vdots \\ x_m(k) = h_m(x_1(k-1), x_2(k-1), \dots, x_m(k-1), u(k-1)) \\ \hat{y}(k) = g(x_1(k), x_2(k), \dots, x_m(k), u(k)) \end{cases} \quad (9)$$

onde $x_1(k), x_2(k), \dots, x_m(k)$ representam os estados do sistema, $h_1(\cdot), h_2(\cdot), \dots, h_m(\cdot)$, são funções (lineares ou não lineares) que representam a atualização dos estados e $g(\cdot)$ é uma função (linear ou não linear) que representa o mapeamento de saída.

Diferentemente dos modelos entrada/saída onde somente $f(\cdot)$ é aproximada, para a completa representação de um sistema dinâmico por um modelo de espaço de estados, $m + 1$ funções ($h_1(\cdot), h_2(\cdot), \dots, h_m(\cdot)$ e $g(\cdot)$) devem ser aproximadas, o que depende da disponibilidade de dados de todos os estados do sistema, além da informação entrada/saída, o que em situações reais é difícil de ocorrer, especialmente em casos de sistemas não lineares (Billings, 2013; Nelles, 2001).

2.4.3.

Modelos estruturados em blocos

Os modelos estruturados em blocos representam modelos que podem ser representados por uma interconexão de modelos não lineares estáticos com modelos lineares dinâmicos. Dois exemplos bem conhecidos de modelos deste tipo são os de Hammerstein e de Wiener.

No modelo de Hammerstein, a não linearidade estática acontece antes do modelo dinâmico linear, como mostrado na Figura 7.

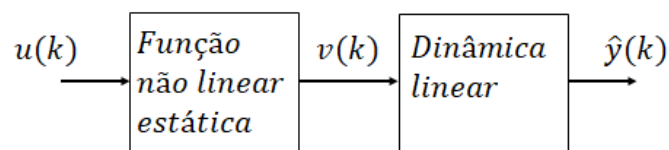


Figura 7. Modelo de Hammerstein

Caso seja utilizado o modelo ARX (*autoregressive model with exogenous inputs*), que é um caso especial do NARX onde $f(\cdot)$ é linear, para a parte linear da figura 7, o modelo de Hammerstein é expresso como:

$$v(k) = f(u(k)) \quad (10)$$

$$\hat{y}(k) = b_1 v(k-1) + \dots + b_{n_v} v(k-n_v) - a_1 y(k-1) + \dots + a_{n_y} y(k-n_y) \quad (11)$$

onde $u(k)$ e $y(k)$ são os sinais de entrada e saída do sistema, respectivamente, $\hat{y}(k)$ é a saída do modelo, $v(k)$ é um sinal intermediário não mensurável, $f(\cdot)$ é uma não linearidade estática, $a_1, \dots, a_{n_u}, b_1, \dots, b_{n_v}$ são parâmetros do modelo linear e n_v e n_y são os maiores atrasos em $v(k)$ e $y(k)$, respectivamente.

No modelo de Wiener, a não linearidade estática acontece depois do modelo dinâmico linear, como mostrado na Figura 8.

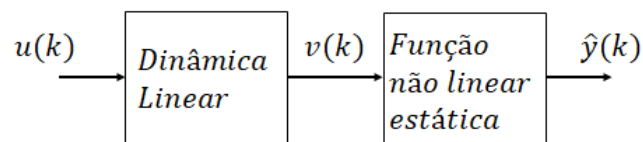


Figura 8. Modelo de Wiener

Com o modelo ARX para a parte linear da figura 8, o modelo de Wiener é representado por:

$$v(k) = a_1 u(k-1) + \dots + a_{n_u} u(k-n_u) + b_1 v(k-1) + \dots + b_{n_v} v(k-n_v) \quad (12)$$

$$\hat{y} = f(v(k)) \quad (13)$$

onde $a_1, \dots, a_{n_u}, b_1, \dots, b_{n_v}$ são parâmetros do modelo linear, n_u e n_v são os maiores atrasos em $u(k)$ e $v(k)$, respectivamente.

Uma versão mais generalizada dos modelos citados, denominada Hammerstein-Wiener (Wills et al., 2013), envolve um modelo linear entre dois blocos de não linearidades estáticas diferentes.

A separação entre blocos lineares e não lineares destes tipos de modelo facilita o tratamento dos mesmos por técnicas de controle não-linear (Aguirre, 2015). Embora a não linearidade de $f(\cdot)$ possa ser implementada por qualquer mapeamento não linear, no caso de modelos Hammerstein, é mais comum o uso de polinômios (Nelles, 2001). Alternativas para a estimação de parâmetros dos modelos Wiener e Hammerstein são encontradas nos trabalhos de Liu et al. (2006), Emara-Shabaik et al. (2001) e Wills et al. (2013).

2.4.4. Séries de Volterra

As séries de Volterra representam um modelo não linear clássico para Identificação de Sistemas (Nelles, 2001), que pode ser expresso como:

$$\hat{y}(k) = \theta_0 + \sum_{l=1}^L H_l(k) \quad (14)$$

com:

$$H_l(k) = \sum_{m_1=1}^{n_u} \dots \sum_{m_l=1}^{n_u} \theta_l(m_1, \dots, m_l) u(k - m_1) \dots u(k - m_l) \quad (15)$$

onde $\theta_0, \theta_1(m_1, \dots, m_l), \dots, \theta_l(m_1, \dots, m_l)$ são parâmetros do modelo, denominados núcleos de Volterra, l é a ordem da série e n_u é o maior atraso na entrada $u(k)$.

Para o caso de ordem 3, a série de Volterra se torna:

$$\begin{aligned} \hat{y}(k) = & \theta_0 + \sum_{m_1=1}^{n_u} \theta_1(m_1) u(k - m_1) + \\ & + \sum_{m_1=1}^{n_u} \sum_{m_2=1}^{n_u} \theta_2(m_1, m_2) u(k - m_1) u(k - m_2) \\ & + \sum_{m_1=1}^{n_u} \sum_{m_2=1}^{n_u} \sum_{m_3=1}^{n_u} \theta_3(m_1, m_2, m_3) u(k - m_1) u(k - m_2) u(k - m_3) \end{aligned} \quad (16)$$

Como mostrado por (14) e (15), a série de Volterra é um modelo do tipo NFIR polinomial, que não utiliza recursão do sinal de saída. Neste caso, o número de parâmetros necessários para explicar a saída somente com valores prévios da entrada é extremamente grande, o que dificulta a utilização prática deste modelo em Identificação de Sistemas, sendo o mesmo aplicável somente em problemas de baixa complexidade (Aguirre, 2015; Nelles, 2001).

2.4.5. Redes Neurais Artificiais

Redes neurais artificiais (RNAs) são modelos da área de Inteligência Computacional cujo princípio de funcionamento é inspirado no cérebro de seres vivos (Silva et al., 2010). Elas são compostas por unidades de processamento simples (neurônios) que são interligados e organizados em camadas. Elas possuem características como aprendizagem por meio de dados de entrada e saída, generalização do conhecimento

aprendido e capacidade de aproximar universalmente funções não lineares (Haykin, 1999) as quais fazem das mesmas boas candidatas para a representação de sistemas não lineares.

RNAs podem ser do tipo *feedforward*, quando não há ligações de recorrência entre seus neurônios, ou recorrentes, quando existe alguma ligação de realimentação entre eles. Redes *feedforward* são estáticas enquanto redes recorrentes são sistemas dinâmicos. Apesar de serem dinâmicas, os métodos tradicionais de treinamento para redes recorrentes, como o BPTT (*Backpropagation Through Time*) (Werbos, 1990) e o RTRL (*Real Time Recurrent Learning*) (Williams & Zipser, 1989), apresentam problemas como convergência lenta e altíssimo custo computacional (Jaeger, 2010; Duan & Wang, 2016; Ozturk et al., 2007), limitando o uso destas redes. Diante disto, a maioria dos trabalhos em Identificação de Sistemas se concentra em redes *feedforward* (Nelles, 2001) utilizando a estratégia de dinâmica externa, apesar de o panorama ter mudado um pouco recentemente com as redes recorrentes da área de Computação de Reservatório, como as ESNs (descritas no capítulo 3).

Assim sendo, nesta seção será descrita simplificadamente uma arquitetura de RNA *feedforward*, denominada *multilayer perceptron* (MLP), que talvez seja a mais utilizada na literatura, deixando um exemplo de rede recorrente para o próximo capítulo, onde serão abordadas as ESNs, que são o foco desta dissertação.

Redes neurais do tipo *multilayer perceptron* (MLP) são caracterizadas por possuírem pelo menos uma camada intermediária (ou escondida) entre as camadas de entrada e saída. Uma arquitetura simplificada para este tipo de rede com uma camada escondida e uma saída é mostrada na Figura 9:

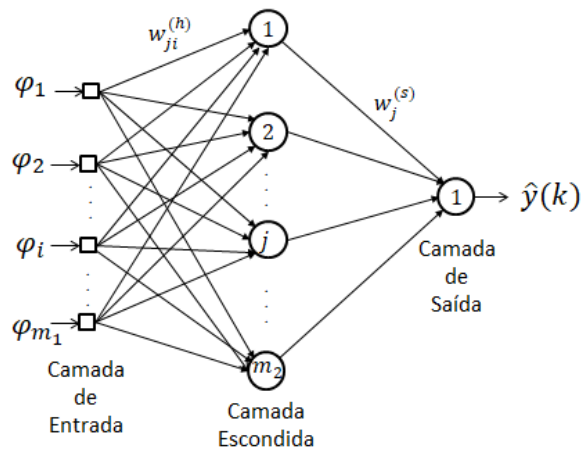


Figura 9. Arquitetura simplificada de uma MLP

Pela figura 9, pode-se ver que a MLP utilizada como exemplo possui m_1 sinais de entrada, que estão representados por $\varphi_1, \dots, \varphi_{m_1}$, uma camada escondida com m_2 neurônios e uma saída $\hat{y}(k)$. O peso de uma conexão entre um sinal de entrada i e um neurônio j da camada escondida é representado por w_{ji} , enquanto w_j representa o peso de uma conexão de um neurônio j da camada intermediária com o único neurônio de saída. A composição interna de um neurônio da MLP, tomando como exemplo um neurônio j da camada escondida é mostrada na figura 10:

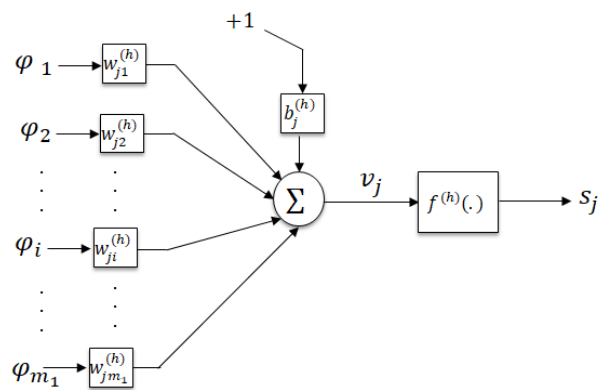


Figura 10. Modelo de um neurônio de uma rede MLP

Através do modelo de processador mostrado na Figura 10, pode-se obter a saída de um neurônio da camada intermediária como:

$$s_j = f_j^{(h)} \left(\sum_{i=1}^{m_1} w_{ji}^{(h)} \varphi_i + b_j^{(h)} \right) \quad (17)$$

onde $f_j^{(h)}$ é a função de ativação de um neurônio j da camada escondida, w_{ji}^h é o peso de conexão de uma entrada φ_i com um neurônio intermediário j e $b_j^{(h)}$ é o *bias* do neurônio j da camada intermediária.

A expressão da saída da rede da figura 9 pode ser expressa como:

$$\hat{y}(k) = f^{(s)}(\sum_{j=1}^{m_2} w_j^{(s)} s_j + b^{(s)}) \quad (18)$$

onde $f^{(s)}$ é a função de ativação do neurônio da camada de saída, $w_j^{(s)}$ é o peso de conexão de um neurônio j da camada interna com o neurônio de saída, s_j é a saída de um neurônio j da camada escondida e $b^{(s)}$ é o *bias* do neurônio de saída.

Substituindo (17) em (18), a saída da rede pode ser expressa como:

$$\hat{y}(k) = f^{(s)}(\sum_{j=1}^{m_2} w_j^{(s)} f_j^{(h)}(\sum_{i=1}^{m_1} w_{ji}^{(h)} \varphi_i + b_j^{(h)})) + b^{(s)} \quad (19)$$

Para atender restrições relativas ao algoritmo de treinamento normalmente empregado nas MLPs, o *backpropagation*, $f_j^{(h)}$ normalmente é uma função de ativação não linear e totalmente diferenciável, como a logística ou a tangente hiperbólica (Figura 11):

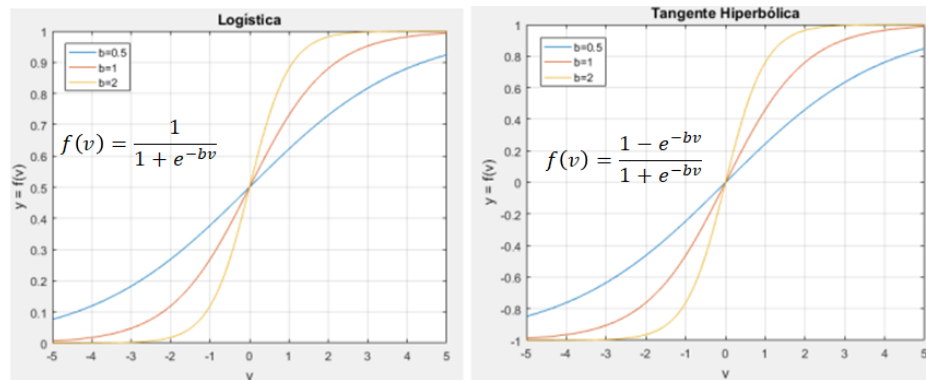


Figura 11. Funções logística e tangente hiperbólica

Em aplicações de identificação de sistemas, é comum o uso da função de ativação linear na camada de saída ($f^{(s)}(v) = v$) com $b^{(s)} = 0$ (Aguirre, 2015), assim (19) é simplificada a:

$$\hat{y}(k) = \sum_{j=1}^{m_2} w_j^{(s)} f_j^{(h)}(\sum_{i=1}^{m_1} w_{ji}^{(h)} \varphi_i + b_j^{(h)}) \quad (20)$$

Por (20), pode-se ver que a MLP faz um mapeamento não linear e estático de seus sinais de entrada $\boldsymbol{\varphi} = [\varphi_1, \dots, \varphi_{m_1}]$. Ainda, pode-se demonstrar (Kolmogorov, 1957; Cybenko, 1989) que a MLP pode aproximar qualquer função contínua através da superposição de diversas funções sigmoidais (como a logística e a tangente hiperbólica), o que justifica o uso destes modelos para aproximação de funções e identificação de sistemas (Silva et al., 2010). Entretanto, pelo fato de a MLP realizar somente mapeamentos estáticos, para a identificação de sistemas dinâmicos, deve-se fornecer para a rede valores atrasados dos sinais de entrada e saída do sistema a ser modelado. Neste caso, a MLP é utilizada como na Figura 6, implementando a função não linear $f(\cdot)$ em uma estratégia de dinâmica externa. Por exemplo, se suas entradas forem $\boldsymbol{\varphi} = [u(k-1), \dots, u(k-n_u), y(k-1), \dots, y(k-n_y)]$, a MLP implementa o modelo NARX.

A estimação de parâmetros (ou treinamento) de redes MLP corresponde à determinação dos pesos e *bias* de seus neurônios, o que normalmente é realizado utilizando o algoritmo *backpropagation*. Neste algoritmo, há a aplicação sucessiva de duas fases denominadas propagação direta (*forward*) e propagação reversa (*backward*).

A fase de propagação direta visa somente obter as respostas da rede frente às suas entradas. Para tal, são mantidos fixos todos os valores de pesos e *bias* da rede e são apresentados à mesma os sinais de entrada e seus efeitos são propagados camada a camada até que sejam obtidas as respostas da MLP na camada de saída.

Na fase de propagação reversa, as saídas produzidas pela rede são comparadas com as respostas desejadas e é gerado um sinal de erro, que é propagado para as camadas anteriores no sentido contrário ao fluxo de informação da rede, isto é da camada de saída até a primeira camada escondida. Desta forma, somente o erro dos neurônios da camada de saída são calculados em função da resposta desejada, sendo o erro de um neurônio de uma camada intermediária estimado em função dos erros dos neurônios da camada imediatamente seguinte conectados a ele, ponderados pelos pesos das conexões entre eles (Braga et al., 2000). Em função dos erros obtidos em cada neurônio, os pesos e *bias* são ajustados, utilizando o método do gradiente descendente (Silva et al., 2010; Haykin, 2009; Braga et al., 2000), de forma a deixar as respostas da rede próximas às desejadas.

Uma descrição detalhada do *backpropagation* é realizada por Silva et al. (2010), Haykin (2009) e Braga et al. (2000). Deve-se ressaltar que quando a MLP é treinada em

uma configuração paralela, implementando, por exemplo, o modelo NOE, onde $\boldsymbol{\varphi} = [u(k-1), \dots, u(k-n_u), \hat{y}(k-1), \dots, \hat{y}(k-n_y)]$, a rede se comporta como recorrente e algoritmos apropriados para o treinamento de redes recorrentes devem ser empregados.

Com relação à estrutura do modelo, além da questão de determinar quais termos regressores utilizar como entrada, que é comum aos modelos de dinâmica externa, quando as MLPs são utilizadas para identificação de sistemas, deve-se determinar também o número de camadas escondidas a serem utilizadas na rede, bem como a quantidade de neurônios nela(s) presente(s) e suas respectivas funções de ativação, o que normalmente depende de um processo de tentativa e erro.

2.4.6. Modelos de dinâmica interna

Uma outra abordagem para a identificação de sistemas dinâmicos não lineares é através da estratégia de dinâmica interna. Modelos com dinâmica interna possuem memória interna e podem ser expressos por equações em espaço de estados como (Nelles, 2001):

$$\hat{\mathbf{x}}(k) = h(\hat{\mathbf{x}}(k-1), u(k-1)) \quad (21)$$

$$\hat{y}(k) = g(\hat{\mathbf{x}}(k)) \quad (22)$$

onde $\hat{\mathbf{x}}(k)$ é um vetor de estados internos do modelo, $h(\cdot)$ e $g(\cdot)$ são funções não lineares.

Os estados internos do modelo $\hat{\mathbf{x}}(k)$ não possuem ligação com os estados do sistema, sendo normalmente em número muito maior do que eles (Nelles, 2001). Em contraste aos modelos de dinâmica externa, modelos de dinâmica interna não exigem a utilização de valores passados das entradas e saídas do sistema, o que reduz consideravelmente o número de variáveis de entrada do modelo (Nelles, 2001). Um exemplo de modelos de dinâmica interna são as redes neurais recorrentes (Nelles, 2001), uma vez que a existência de ligações de realimentação em sua estrutura cria um modelo de espaço de estados (Becerra et al., 2002) devido à influência do estado atual de um neurônio nos estados de outros neurônios em um passo de tempo seguinte.

3 Echo State Networks

3.1. Introdução

A área de Computação de Reservatório (CR) introduziu uma nova maneira de se entender, treinar e utilizar redes neurais artificiais recorrentes (Lukosevicius et al., 2012), que foi iniciada com a proposição das *Echo State Networks* (Jaeger, 2010) e das *Liquid State Machines* (Maass et al., 2002) no começo da década passada e posteriormente ganhou extensões como o *Backpropagation Decorrelation* (Steil, 2004) e o Evolino (Schmidhuber et al., 2007). Modelos CR tratam de forma diferenciada as suas partes recorrentes e as suas camadas de saídas, sendo alternativas aos métodos de treinamento de RNAR baseados na descida do gradiente, como o BPTT (Werbos, 1990) e o RTRL (Williams & Zipser, 1989), que adaptam indistintamente todos os pesos de uma RNAR, além de apresentarem problemas de convergência lenta e subótima e altíssimo custo computacional. Desta forma, devido ao alto custo computacional, o uso do BPTT e do RTRL é conveniente somente em aplicações em que uma RNAR com poucos neurônios na camada escondida é suficiente para se atingir bons resultados (Jaeger, 2013).

As ESNs, uma possível (e talvez a mais simples e mais utilizada) realização de CR, foram propostas por Jaeger em 2001 (Jaeger, 2010) e possuem como principal vantagem o seu processo de treinamento, que é tipicamente simples e computacionalmente eficiente, uma vez que adapta apenas os pesos de uma camada estática de saída de neurônios, sendo normalmente desempenhado por uma regressão linear, e mantém fixos os pesos internos e direcionados à parte dinâmica destes modelos, que é denominada reservatório. Desta forma, as ESNs preservam a capacidade de processamento naturalmente dinâmico das redes neurais recorrentes, o que é importante para tarefas de natureza temporal, como em identificação de sistemas dinâmicos, porém se utilizam de uma estratégia de treinamento de baixo custo computacional, o que permite superar muitos dos problemas apresentados com os algoritmos clássicos citados (BPTT e o RTRL).

Cabe ressaltar que outras alternativas de métodos para o treinamento de RNAR, que normalmente convergem mais rapidamente que os métodos clássicos BPTT e RTRL (Williams, 1992; Trebatický, 2005), são os algoritmos derivados do clássico filtro de Kalman (Haykin, 2009), como o filtro de Kalman estendido (Haykin, 2009; Williams, 1992; Trebatický, 2005), o filtro de Kalman estendido e desacoplado (Haykin, 2009), o filtro de Kalman “*Unscented*” (Trebatický, 2005) e o filtro de Kalman “*npr*” (Trebatický, 2005).

Com isto, este capítulo tem como objetivo abordar os principais conceitos relacionados às redes neurais recorrentes do tipo *Echo State Networks*, passando pela descrição do esquema básico de funcionamento, arquitetura, treinamento e principais parâmetros de configuração destas redes. Também são abordados alguns métodos para avaliação do reservatório e discorre-se sobre trabalhos da literatura que utilizaram *Echo State Networks* em aplicações de identificação de sistemas, além de outros que otimizaram ESNs por meio de algoritmos evolutivos.

3.2. Arquitetura

Uma *Echo State Network* é composta por 3 camadas principais: entrada, reservatório dinâmico e saída, sendo uma de suas arquiteturas típicas a mostrada na Figura 12:

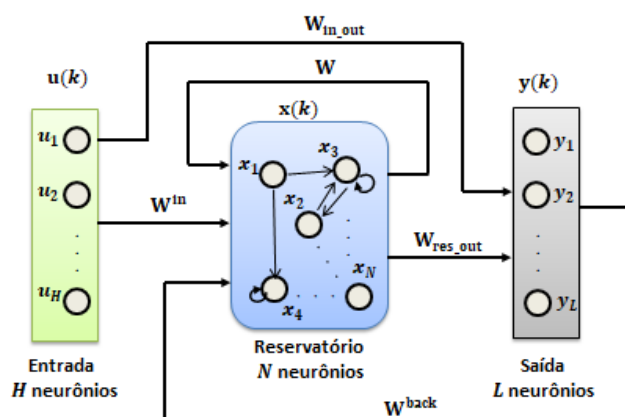


Figura 12 – Arquitetura típica de uma ESN

Como mostrado pela Figura 12, a camada de entrada de uma ESN recebe H sinais de entrada, os quais são representados pelo vetor $\mathbf{u}(k) = [u_1(k) \ u_2(k) \ \dots \ u_H(k)]^T$, o reservatório contém N neurônios, cujos estados de ativação são representados por $\mathbf{x}(k) = [x_1(k) \ x_2(k) \ \dots \ x_N(k)]^T$ e os L sinais de saída da rede são representados pelo vetor $\mathbf{y}(k) = [y_1(k) \ y_2(k) \ \dots \ y_L(k)]^T$. Os pesos de conexões da camada de entrada com o reservatório são representados por \mathbf{W}^{in} , de tamanho $N \times H$, pesos de conexões de realimentação da camada de saída para o reservatório são representadas pela matriz \mathbf{W}^{back} , que possui dimensões $N \times L$, enquanto que pesos de conexões entre os neurônios do reservatório são representados pela matriz \mathbf{W} , de tamanho $N \times N$, que inclusive admite conexões de um neurônio com ele mesmo. As matrizes $\mathbf{W}_{\text{in_out}}$ e $\mathbf{W}_{\text{res_out}}$, de dimensões $L \times H$ e $L \times N$, respectivamente, representam os pesos de conexões da camada de entrada e do reservatório para a camada de saída. Estas conexões podem ser agrupadas em uma única matriz de pesos de saída, denominada \mathbf{W}^{out} e de dimensões $L \times (H + N)$. As matrizes \mathbf{W} , \mathbf{W}^{in} e \mathbf{W}^{back} permanecem fixas após serem inicializadas aleatoriamente, enquanto que \mathbf{W}^{out} é a única matriz que é ajustada durante o treinamento. Em algumas configurações de ESNs, também é possível encontrar conexões de realimentação da camada de saída para a camada de saída, o que não foi explorado neste trabalho, pois em testes iniciais foi observado um comportamento instável da rede quando estas conexões eram utilizadas.

3.3. Funcionamento básico

O reservatório é o elemento principal de uma ESN. Ele possui um grande número de neurônios com conexões recorrentes, esparsas e inicializadas aleatoriamente. Ao se fornecerem ao reservatório sinais provenientes da entrada da rede e realimentados de sua camada de saída ponderados por \mathbf{W}^{in} e \mathbf{W}^{back} , respectivamente, ele atua como uma função temporal não linear, expandindo o histórico de tais sinais ($\mathbf{u}(k), \mathbf{u}(k - 1), \dots, \mathbf{y}(k - 2), \mathbf{y}(k - 1)$) em uma representação de espaço de estados não linear, que é expressa através dos estados de ativação de seus neurônios internos $\mathbf{x}(k)$ (Lukosevicius; Jaeger, 2009).

Com o objetivo de enriquecer a representação dada por $\mathbf{x}(k)$ e favorecer a ocorrência de comportamentos dinâmicos diversificados dentro do reservatório, o número de neurônios nesta camada deve ser grande e a sua matriz de pesos \mathbf{W} deve ser esparsa em algum nível (Jaeger, 2010). Desta forma, o número de neurônios do reservatório N e a conectividade c_W desta matriz são hiperparâmetros que devem ser definidos durante a geração de uma ESN. Outro importante hiperparâmetro referente ao reservatório é o raio espectral de sua matriz \mathbf{W} ($\rho(\mathbf{W})$), que corresponde ao maior autovalor em valor absoluto entre os autovalores de \mathbf{W} . Tal hiperparâmetro é importante para questões de estabilidade da rede e está relacionado a um conceito chave para o funcionamento de uma ESN, que é a propriedade de *echo state*.

De uma maneira intuitiva, a propriedade de *echo state* estabelece que após a ESN ter sido executada por um longo período de tempo, os estados de ativação dos neurônios do reservatório em um passo de tempo k ($\mathbf{x}(k)$) dependerão somente do histórico das entradas e saídas anteriores da rede (Jaeger, 2013). A existência de tal efeito depende de propriedades algébricas da matriz \mathbf{W} . Como abordado por Jaeger (2010, 2013), uma condição prática que em muitas situações resulta em uma ESN que satisfaz a propriedade de *echo state* é fazer com que o raio espectral de \mathbf{W} não exceda uma unidade ($\rho(\mathbf{W}) < 1$). Tal condição significa que todos os autovalores da matriz \mathbf{W} estão contidos dentro de um círculo unitário no plano complexo, o que embora não garanta, é uma condição necessária para que a rede tenha um comportamento estável (Straub et al., 2012). Maiores detalhes sobre esta propriedade podem ser encontrados em (Jaeger, 2010, 2013).

A camada de saída das ESNs é estática e normalmente linear e combina os estados produzidos pelo reservatório $\mathbf{x}(k)$ (e entradas $\mathbf{u}(k)$) conectadas diretamente à mesma) de forma a gerar as saídas da rede $\mathbf{y}(k)$.

3.4. Equações básicas

3.4.1. Atualização dos estados do reservatório

Os estados de ativação do reservatório dependem das entradas da rede, das realimentações provenientes da camada de saída e dos estados do reservatório no passo de tempo anterior. Desta forma, considerando neurônios sigmoidais estáticos no reservatório, a atualização de seus estados é dada por:

$$\mathbf{x}(k) = \mathbf{f}^{\text{res}}(\mathbf{W}^{\text{in}}\mathbf{u}(k) + \mathbf{W}\mathbf{x}(k-1) + \mathbf{W}^{\text{back}}\mathbf{y}(k-1)) \quad (23)$$

onde $\mathbf{f}^{\text{res}}(\cdot) = [f_1^{\text{res}}(\cdot), \dots, f_N^{\text{res}}(\cdot)]$ representa o vetor de funções de ativação dos neurônios do reservatório, que normalmente são do tipo tangente hiperbólica (*tanh*).

Uma outra frequente opção de neurônios para o reservatório são os do tipo *leaky integrator* (Jaeger, 2010). Diferentemente dos neurônios sigmoidais simples, neurônios *leaky integrator* são individualmente dinâmicos (Jaeger et al., 2007) e conseguem reter localmente parte do seu estado de ativação anterior, cuja influência em seu estado de ativação atual vai decaindo ao longo do tempo na ausência de excitação (Martins, 2016).

Quando neurônios *leaky integrator* são utilizados no reservatório, uma das possíveis representações para a equação de atualização dos estados da rede é a seguinte (Lukosevicius & Jaeger, 2009; Lukosevicius, 2012):

$$\mathbf{x}(k) = (1 - \alpha) \mathbf{x}(k-1) + \alpha \mathbf{f}^{\text{res}}(\mathbf{W}^{\text{in}}\mathbf{u}(k) + \mathbf{W}\mathbf{x}(k-1) + \mathbf{W}^{\text{back}}\mathbf{y}(k-1)) \quad (24)$$

onde $\alpha \in (0,1]$ é a taxa de decaimento, mais conhecida como *leakage rate*, que foi assumida a mesma para todos os neurônios do reservatório.

O valor de α controla a dinâmica de atualização da rede (Lukosevicius & Jaeger, 2009) e pode ser ajustado de tal forma a acomodar a rede às características temporais de uma dada tarefa (Jaeger et al., 2007). Quanto mais próximo de 1, menos cada neurônio retém informação do seu estado anterior e no limite, $\alpha = 1$, a ESN se comporta como se tivesse neurônios sigmoidais estáticos em seu reservatório e a equação (24) se reduz à (23). Como consequência, uma ESN com neurônios *leaky integrator* com um valor de α adequado tem um desempenho ao menos similar a uma ESN com neurônios sigmoidais

simples em seu reservatório (Lukosevicius & Jaeger, 2009). Por estes motivos, neurônios do tipo *leaky integrator*, serão os utilizados no modelo proposto nesta dissertação, o qual é descrito no capítulo 5 deste documento.

3.4.2. Atualização das saídas da rede

As saídas da rede são influenciadas pelos estados do reservatório e pelas entradas que são conectadas diretamente à camada de saída. Com isso, as saídas da rede são obtidas conforme:

$$\mathbf{y}(k) = \mathbf{f}^{\text{out}}(\mathbf{W}^{\text{out}}[\mathbf{u}(k)^\top \mathbf{x}(k)^\top]^\top) \quad (25)$$

onde $\mathbf{f}^{\text{out}}(\cdot) = [f_1^{\text{out}}(\cdot), \dots, f_N^{\text{out}}(\cdot)]$ representa o vetor de funções dos neurônios de saída, que são normalmente funções não lineares, $[\mathbf{u}(k)^\top \mathbf{x}(k)^\top]^\top$ representa a concatenação dos vetores $\mathbf{u}(k)$ e $\mathbf{x}(k)$, enquanto que \top é o operador de transposição de matrizes.

3.5. Treinamento

Uma visão geral de diferentes métodos para o treinamento da camada de saída das ESNs pode ser encontrado nos trabalhos de Lukosevicius & Jaeger (2009), Lukosevicius (2012); Lovlid (2013) e Goudarzi & Stefanovic (2014). Nesta seção, será descrito o treinamento *off-line* de uma ESN desempenhado por meio da pseudoinversa, que será o utilizado nesta dissertação por possuir menor custo computacional (Niño, 2016) e ser mais simples para aplicações de modelagem de sistemas (Martins, 2016).

O objetivo do treinamento é determinar os pesos de saída \mathbf{W}^{out} de forma a minimizar o erro médio quadrático (MSE – *Mean Squared Error*) entre o vetor de saídas desejadas $\mathbf{y}_d(k)$ e o vetor de saídas da rede $\mathbf{y}(k)$. Com este objetivo, o treinamento por meio da pseudoinversa determina \mathbf{W}^{out} seguindo-se os seguintes passos:

1 – Gera-se as matrizes \mathbf{W}^{in} , \mathbf{W}^{back} e \mathbf{W} conforme:

$$\mathbf{W}^{\text{in}} = s_{in} \mathbf{W}_0^{\text{in}} \quad (26)$$

$$\mathbf{W}^{\text{back}} = s_{\text{back}} \mathbf{W}_0^{\text{back}} \quad (27)$$

$$\mathbf{W} = \rho_W \frac{\mathbf{W}_0}{\rho(\mathbf{W}_0)} \quad (28)$$

onde s_{in} , s_{back} são fatores de escala de entrada e de realimentação, respectivamente e \mathbf{W}_0^{in} , $\mathbf{W}_0^{\text{back}}$ são matrizes de tamanhos $N \times H$, $N \times L$, respectivamente, que são inicializadas aleatoriamente. A matriz \mathbf{W}_0 é uma matriz esparsa de tamanho $N \times N$ e conectividade c_W , que também é inicializada aleatoriamente, $\rho(\mathbf{W}_0)$ é o raio espectral (autovalor de maior valor absoluto) de \mathbf{W}_0 e ρ_W é um fator multiplicativo que resultará no raio espectral da matriz \mathbf{W} . A divisão $\mathbf{W}_0/\rho(\mathbf{W}_0)$ resulta numa matriz de raio espectral unitário, que ao ser multiplicada por ρ_W gera a matriz \mathbf{W} com raio espectral. De forma a gerar uma ESN com a propriedade de *echo state*, faz-se $\rho_W < 1$.

Uma vez que \mathbf{W}^{in} , \mathbf{W}^{back} e \mathbf{W} são geradas, elas permanecem fixas e prossegue-se ao próximo passo do treinamento.

2 – Para cada padrão entrada/saída de treinamento, calculam-se os estados do reservatório $\mathbf{x}(k)$ utilizando (23) ou (24), fazendo-se $\mathbf{y}(k) = \mathbf{y}_d(k)$ (saída desejada) e usando-se $\mathbf{x}(0) = \mathbf{0}$ e $\mathbf{y}(0) = \mathbf{0}$ (ou quaisquer outros valores arbitrários) como condições iniciais.

3 – Para cada passo de tempo k maior ou igual a um tempo inicial T_0 (*washout time*), coletam-se os sinais concatenados $[\mathbf{u}(k)^\top \mathbf{x}(k)^\top]$ e as saídas desejadas $\mathbf{y}_d(k)$ como novas linhas para matrizes \mathbf{M} e \mathbf{D} , que são criadas, respectivamente, como:

$$\mathbf{M} = \begin{bmatrix} \mathbf{u}(T_0) & \mathbf{x}(T_0) \\ \vdots & \vdots \\ \mathbf{u}(T) & \mathbf{x}(T) \end{bmatrix} \quad (29)$$

$$\mathbf{D} = \begin{bmatrix} \mathbf{y}_d(T_0) \\ \vdots \\ \mathbf{y}_d(T) \end{bmatrix} \quad (30)$$

onde T é o tamanho da base de dados de treinamento.

O período de *washout* T_0 é utilizado para que o transiente inicial devido à inicialização aleatória dos estados do reservatório não seja considerado durante o cálculo de \mathbf{W}^{out} .

Caso se utilize uma função de ativação não linear na camada de saída, deve-se armazenar em \mathbf{D} para cada $k \geq T_0$ os valores $\mathbf{f}^{\text{out}-1}(\mathbf{y}_d(k))$ de forma a cancelar a não linearidade.

4 – Finalmente, no último passo do treinamento, os pesos de saída ótimos são determinados como:

$$\mathbf{W}^{\text{out}} = (\mathbf{M}^\dagger \mathbf{D})^\top \quad (31)$$

onde \dagger e \top são os operadores de pseudoinversa e de transposição de matrizes, respectivamente.

3.6. Parâmetros que afetam o desempenho das ESNs

Embora as ESNs apresentem um treinamento simples e de baixo custo computacional, é necessária a correta definição de seus hiperparâmetros para que a rede apresente um bom desempenho em uma dada tarefa. Baseando-se em Jaeger (2013) e Lukosevicius (2012), os principais parâmetros globais que afetam o desempenho das ESNs são os seguintes:

- **Tamanho do reservatório (N)** – controla o número de neurônios do reservatório e deve ser escolhido levando-se em consideração a complexidade da tarefa e o tamanho da base de dados de treinamento (T). Como observado por (Jaeger, 2013), é aconselhável não exceder os limites de $T/10$ (mínimo) e $T/2$ (máximo) para valores de N como uma precaução contra o problema de *overfitting*. Quanto mais difícil for a tarefa, maior o valor de N deve ser (Jaeger, 2013), porém maior será o custo computacional associado com o aumento de seu valor (Lukosevicius, 2012).

- **Conectividade ou densidade de conexões (c_W)** – define a relação entre o número de conexões não nulas no reservatório e o total de pesos em \mathbf{W} . Definindo-se este

parâmetro, define-se também o nível de esparsidade $(1 - c_W)$ no reservatório requerido para garantir uma dinâmica variada para a rede.

- **Raio espectral** (ρ_W) – é o parâmetro mostrado em (28). É utilizado para escalar a matriz de pesos do reservatório, que foi inicializada aleatoriamente e normalizada por seu raio espectral ($\mathbf{W}_0/\rho(\mathbf{W}_0)$). Ele representa o raio espectral resultante para a matriz \mathbf{W} e deve satisfazer $\rho_W < 1$ para gerar uma ESN com a propriedade de *echo state* na maioria dos casos. Quanto mais próximo a 1 este parâmetro for, mais adequada à problemas com dinâmica lenta a ESN se torna (Jaeger, 2013), o que pressupõe que o usuário deva conhecer previamente o comportamento dinâmico da tarefa.

- **Fatores de escala das matrizes** \mathbf{W}^{in} (s_{in}) e \mathbf{W}^{back} (s_{back}) – são os parâmetros apresentados em (26) e (27), respectivamente. Eles são fatores multiplicativos para as matrizes \mathbf{W}^{in} e \mathbf{W}^{back} e controlam quanto maior ou menor os valores absolutos destas matrizes podem ser. Valores absolutos altos para os pesos de \mathbf{W}^{in} e \mathbf{W}^{back} conduzem os neurônios do reservatório a uma região de atuação mais não linear e valores absolutos pequenos levam a uma operação mais linear dos mesmos, considerando um reservatório com função de ativação *tanh*.

- **Leakage rate** (α) – é o parâmetro presente em (24). Pode ser definido como a velocidade com que o reservatório atualiza sua dinâmica no tempo discreto (Lukosevicius, 2012) quando neurônios do tipo *leaky integrator* são utilizados.

3.7. Alguns métodos para avaliação do reservatório

Esta seção tem como objetivo descrever o método de *separation ratio graph* (SRG) de Niño (2016), que é uma extensão do método homônimo inicialmente proposto por Gibbons (2010) no contexto de classificação de padrões. Como o método de Gibbons (2010) é derivado das métricas de separação (Norton & Ventura, 2009, 2010; Goodman & Ventura, 2006) e do expoente de Lyapunov (Verstraeten et al., 2007; Schrauwen et al., 2008; Legenstein & Maass, 2007), tais técnicas também são abordadas brevemente nesta seção.

Os métodos citados são alguns dos exemplos de métricas que foram desenvolvidas para avaliar a qualidade de um reservatório que foi criado aleatoriamente antes de efetivamente utilizá-lo na tarefa pretendida e sem a necessidade de realizar o treinamento de uma ESN (ou de outro método CR em questão).

3.7.1. Separação

Separação é uma métrica que permite medir a distância média de separação entre os estados produzidos pelo reservatório em resposta a diferentes classes de padrões de entrada (Gibbons, 2010). Ela foi proposta por Goodman & Ventura (2006) e expandida por Norton & Ventura (2010).

Assumindo-se que padrões de entrada e estados correspondentes do reservatório estejam divididos em classes discretas, a ideia por trás de se utilizar esta métrica é baseada no raciocínio de que, com o objetivo de se facilitar a classificação:

- Estados do reservatório associados à mesma classe devem ter distância de separação pequena e;
- Estados do reservatório associados a classes diferentes devem ter distância de separação grande.

Formalmente, dados um número Q de estados na mesma classe x_1, x_2, \dots, x_Q , define-se a média do centro de massa (μ) da classe O_j como:

$$\mu(O_j(k)) = \frac{\sum_{i=1}^P x_i(k)}{Q} \quad (32)$$

Uma vez definido o centro de massa para cada classe, definem-se a distância inter classes ($C_d(k)$) e a variância intra classe ($C_v(k)$) como:

$$C_d(k) = \sum_{m_1=1}^M \sum_{m_2=1}^M \frac{\|\mu(O_{m_1}(k)) - \mu(O_{m_2}(k))\|}{M^2} \quad (33)$$

$$C_v(k) = \frac{1}{M} \sum_{m=1}^M \frac{\sum_{i=1}^P \|\mu(O_m(k)) - x_i(k)\|}{Q} \quad (34)$$

onde M é o número total de classes e $\|\cdot\|$ é a norma euclidiana.

Finalmente, tem-se a separação ($sep_x(k)$) como:

$$sep_x(k) = \frac{C_d(k)}{C_v(k) + 1} \quad (35)$$

É mostrado por Goodman & Ventura (2006) que o aumento da separação tem correlação com uma maior precisão no mapeamento da camada de saída quando os reservatórios são gerados aleatoriamente (Gibbons, 2010). Com isto, a separação pode ser utilizada como uma medida de avaliação da qualidade do reservatório antes do treinamento, dando-se preferência a reservatórios com maior separação entre os estados (Gibbons, 2010).

A limitação associada ao uso de sep_x está relacionada ao fato de que os padrões de entradas devem estar associadas a classes discretas distintas. Quando esta situação não é possível, pode-se estender o conceito de separação para problemas que não possuam classes previamente definidas assumindo-se que dois padrões de entrada significativamente diferentes ($\mathbf{u}_i(k)$ e $\mathbf{u}_j(k)$) devem gerar estados correspondentes ($\mathbf{x}_i(k)$ e $\mathbf{x}_j(k)$, respectivamente) que também são significativamente diferentes (Gibbons, 2010). Em outras palavras, é desejada a seguinte relação:

$$\frac{\|\mathbf{x}_i(k) - \mathbf{x}_j(k)\|}{\|\mathbf{u}_i(k) - \mathbf{u}_j(k)\|} \geq 1 \quad (36)$$

3.7.2. Expoente de Lyapunov

Expoentes de Lyapunov correspondem à taxa exponencial média de divergência ou convergência de duas órbitas infinitesimalmente próximas no espaço de fase de um sistema dinâmico (Wolf et al., 1985). Eles atuam como uma maneira de detectar e mensurar o comportamento caótico de sistemas dinâmicos.

Uma das maneiras de estimar os expoentes de Lyapunov foi o método desenvolvido por Rosenstein et al. (1993), onde o expoente de Lyapunov é numericamente estimado como a taxa média da distância de separação entre os estados de um sistema que são vizinhos mais próximos:

$$\lambda(k) = \gamma \sum_{m=1}^M \ln\left(\frac{d_j(k)}{d_j(0)}\right) \quad (37)$$

onde $\lambda(k)$ é o expoente de Lyapunov no passo k , γ é uma constante, $d_j(k) = \|\mathbf{x}_i(k) - \mathbf{x}_j(k)\|$ é a distância de separação entre os estados \mathbf{x}_i e \mathbf{x}_j do sistema, que são vizinhos mais próximos, e $d_j(0)$ é o estado inicial do sistema.

A extensão do método de Rosenstein et al. (1993) para estimar o coeficiente de Lyapunov para redes da área de RC foi feita por Gibbons (2010) ao substituir em (37) $d_j(0)$ pela distância de separação entre os padrões de entrada mais próximos $\mathbf{u}_i(k)$ e $\mathbf{u}_j(k)$ que são responsáveis por gerar, respectivamente, os estados \mathbf{x}_i e \mathbf{x}_j no reservatório. Desta forma, o expoente de Lyapunov para redes com reservatório é estimado como:

$$\lambda(k) = \gamma \sum_{m=1}^N \ln\left(\frac{\|\mathbf{x}_i(k) - \mathbf{x}_j(k)\|}{\|\mathbf{u}_i(k) - \mathbf{u}_j(k)\|}\right) \quad (38)$$

Se $\lambda(k) > 0$, o comportamento da rede é caótico e pequenas mudanças nos padrões de entrada resultam em grandes mudanças nos estados gerados pelo reservatório (Gibbons, 2010). Por outro lado, se $\lambda(k) < 0$, o comportamento da rede é estável e pequenas mudanças na entrada convergem em estados no reservatório que tendem a um atrator (Gibbons, 2010).

É mostrado (Legenstein & Maass, 2007; Bertschinger & Natschläger, 2004) que o desempenho do reservatório é maximizado próximo ao “limiar do caos”, entre o comportamento caótico e o estável, onde $\lambda(k) \approx 0$. Neste caso, o argumento da função $\ln(\cdot)$ em (38) tende a 1, desta forma o comportamento ótimo para distâncias de separação pequenas ocorre quando:

$$\frac{\|\mathbf{x}_i(k) - \mathbf{x}_j(k)\|}{\|\mathbf{u}_i(k) - \mathbf{u}_j(k)\|} \cong 1 \quad (39)$$

3.7.3. Separation Ratio Graph (Gibbons, 2010)

Do exposto anteriormente, pela métrica de separação deseja-se que $\frac{\|\mathbf{x}_i(k) - \mathbf{x}_l(k)\|}{\|\mathbf{u}_i(k) - \mathbf{u}_l(k)\|} \geq 1$, assumindo que a distância de separação entre $\mathbf{u}_i(k)$ e $\mathbf{u}_l(k)$ é grande. Por outro lado, pelo expoente de Lyapunov, para uma operação do reservatório perto do limiar do caos, espera-se que $\frac{\|\mathbf{x}_j(k) - \mathbf{x}_j(k)\|}{\|\mathbf{u}_j(k) - \mathbf{u}_j(k)\|} \cong 1$, assumindo-se uma distância de separação pequena entre $\mathbf{u}_j(k)$ e $\mathbf{u}_j(k)$. Com o objetivo de unificar as duas métricas (separação e expoente de Lyapunov), Gibbons (2010) definiu a razão de separação (*separation ratio* - SR) como a relação entre a distância de separação entre os estados de um reservatório e a distância de separação entre as entradas que os originaram:

$$SR = \frac{\|\mathbf{x}_i(k) - \mathbf{x}_j(k)\|}{\|\mathbf{u}_i(k) - \mathbf{u}_j(k)\|} \quad (40)$$

Quando $\mathbf{u}_i(k)$ e $\mathbf{u}_j(k)$ são muito diferentes, pela métrica de separação, espera-se que $SR \geq 1$. Por outro lado, pelo expoente de Lyapunov, espera-se que $SR \cong 1$. Com isso, um valor de $SR \cong 1$ permite atender a ambas situações, sendo o valor alvo em tal métrica.

Em adição à razão de separação, Gibbons (2010) introduziu a ideia dos gráficos de razão de separação (*separation ratio graphs* - SRGs). Os SRGs permitem representar de forma gráfica a razão de separação de um reservatório e interpretar qualitativamente o seu comportamento. Para isto, dado um conjunto de P padrões de entrada $\{\mathbf{u}_1(k), \mathbf{u}_2(k), \dots, \mathbf{u}_P(k)\}$ e os correspondentes estados gerados por estes padrões $\{\mathbf{x}_1(k), \mathbf{x}_2(k), \dots, \mathbf{x}_P(k)\}$, para cada $i, j \in [1, p]$, a distância de separação entre os padrões de entrada ($\|\mathbf{u}_i(k) - \mathbf{u}_j(k)\|$) é plotada contra a distância de seus estados correspondentes ($\|\mathbf{x}_i(k) - \mathbf{x}_j(k)\|$), gerando um gráfico como o da figura 13:

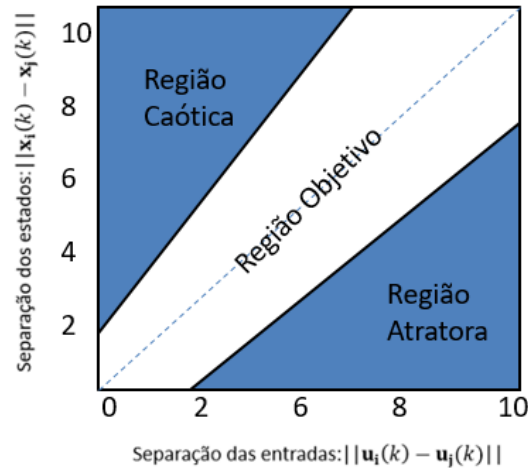


Figura 13 – Regiões de um gráfico de razão de separação. Fonte: Adaptado de Niño (2016) e Gibbons (2010)

A região objetivo na figura 13 ($SR \cong 1$) ocorre quando $||\mathbf{x}_i(k) - \mathbf{x}_j(k)|| \cong ||\mathbf{u}_i(k) - \mathbf{u}_j(k)||$. Utilizando-se uma regressão linear para ajustar uma reta aos dados, tal região objetivo é alcançada quando a reta resultante se aproxima da reta pontilhada na figura 13, isto é, seu coeficiente angular é próximo de 1 ($a \cong 1$) e seu coeficiente linear é aproximadamente nulo ($b \cong 0$). A região objetivo é também conhecida como “limiar do caos”, se situando entre as zonas caótica e atratora na figura 13. Estas últimas representam situações indesejadas onde entradas com distância de separação grande geram estados com separação pequena (atratora) e entradas similares geram estados muito diferentes (caótica).

3.7.4. Separation Ratio Graph (Niño, 2016)

O método SRG proposto por Gibbons (2010) foi aplicado a tarefas de classificação, sendo uma extensão do mesmo para tarefas de previsão *multi-step* de séries temporais realizada por Niño (2016). No trabalho de Niño (2016), o SRG é utilizado como um critério para selecionar o reservatório mais adequado dentre um conjunto de n ESNs inicializadas aleatoriamente, cujos parâmetros de configuração são otimizados por um algoritmo genético em uma etapa prévia. Como observado por Niño (2016), tal método permite avaliar se os estados gerados pelo reservatório são interpretados de forma correta, isto é, estímulos similares (entradas mais próximas) geram respostas (estados) similares. Para isto, a metodologia utilizada por (Niño, 2016) pode ser dividida em 4 passos:

- 1 – Agrupamento em pares dos padrões de entrada;
- 2 – Geração dos estados do reservatório para cada par de padrões agrupados no passo 1;
- 3 – Cálculo do coeficiente SRG;
- 4 – Seleção do reservatório mais adequado.

Na fase 1, para cada um dos padrões de entrada do conjunto de treinamento, é buscado o seu vizinho mais próximo utilizando o algoritmo *knn* (*k nearest neighbors*) com $k = 1$ com a distância euclidiana ($d_{i,j}(\mathbf{u}_i, \mathbf{u}_j)$) usada para quantificar a separação entre dois padrões de entrada i e j :

$$d_{i,j}(\mathbf{u}_i, \mathbf{u}_j) = d(\mathbf{u}_j, \mathbf{u}_i) = \sqrt{(\mathbf{u}_i - \mathbf{u}_j)^2} \quad (41)$$

Após o agrupamento em pares dos padrões, em uma segunda etapa, cada padrão (\mathbf{u}_j) e seu vizinho mais próximo (\mathbf{u}_j) são apresentados ao reservatório, sendo obtidos os seus estados correspondentes (\mathbf{x}_j e \mathbf{x}_j). Como observado por Niño (2016), durante esta fase, deve-se cumprir com a condição de que todos os estados sejam gerados com um padrão \mathbf{u} partindo do mesmo lugar, o que em outras palavras, significa não considerar o estado imediatamente anterior como histórico para gerar o estado atual.

Em uma terceira etapa, com os p padrões agrupados com o seu vizinho mais próximo e seus correspondentes estados coletados, Niño (2016) definiu o *SRG* como um coeficiente que é calculado como:

$$SRG = \sum_{j=1}^p ||d(\mathbf{u}_j) - d(\mathbf{x}_j)|| \quad (42)$$

onde $d(u_j) = ||\mathbf{u}_j(k) - \mathbf{u}_j(k)||$ e $d(x_j) = ||\mathbf{x}_j(k) - \mathbf{x}_j(k)||$ representam, respectivamente, a distância de separação entre o padrão de entrada \mathbf{u}_j e seu vizinho mais próximo \mathbf{u}_j e a distância de separação de seus respectivos estados \mathbf{x}_j e \mathbf{x}_j .

Em uma última etapa em que se deseja avaliar a qualidade de um reservatório gerado, de maneira similar ao *SR* (Gibbons, 2010), o comportamento desejado, próximo ao limiar do caos, para o coeficiente de (42) é quando $SRG \cong 0$. Desta forma, ao se avaliar

um conjunto de reservatórios, escolhe-se como melhor aquele com o coeficiente *SRG* mais baixo. A visualização gráfica das distâncias de separação dos padrões de entrada contra a distância de seus respectivos estados continua válida no procedimento adotado por Niño (2016).

3.8. ESNs implementando um modelo NARX

Como mostrado por (23) ou (24), as conexões recorrentes no reservatório criam nas ESNs um modelo de espaço de estados, com realimentação da saída, cujas variáveis de estado são as saídas de seus neurônios internos. Desta forma, as ESNs são modelos de dinâmica interna, com capacidade de representar sistemas dinâmicos sem a necessidade de se prover explicitamente e iterativamente sinais atrasados no tempo como entrada para a rede, como seria no caso de se utilizarem redes neurais *feedforward* em uma estratégia de dinâmica externa.

De outro ponto de vista, tal representação em espaço de estados das ESNs pode ser convertida em uma representação do tipo NARX. Para isto, assumindo uma ESN com camada de saída linear, com realimentação da saída para o reservatório e com a propriedade de *echo state*, após um certo período de tempo, para cada estado interno do reservatório x_i existe uma função de eco $e_i(\cdot)$ (Jaeger, 2010, 2013), que faz um mapeamento não linear das entradas e saídas anteriores. Com isto, o estado de ativação de um neurônio i qualquer do reservatório um passo de tempo k é função do histórico de entradas e saídas anteriores:

$$x_i(k) = e_i(\mathbf{u}(k), \mathbf{u}(k-1), \dots, \mathbf{y}(k-2), \mathbf{y}(k-1)) \quad (43)$$

A saída da rede, por sua vez, é uma combinação linear dos estados gerados pelo reservatório:

$$\mathbf{y}(k) = \sum w_i^{out} x_i(k) = \sum w_i^{out} e_i(\mathbf{u}(k), \mathbf{u}(k-1), \dots, \mathbf{y}(k-2), \mathbf{y}(k-1)) \quad (44)$$

O termo $w_i^{out} e_i(\cdot)$ pode ser visto como uma função não linear equivalente $\mathbf{G}(\cdot)$, então (44) se torna:

$$\mathbf{y}(k) = \mathbf{G}(\mathbf{u}(k), \mathbf{u}(k-1), \dots, \mathbf{y}(k-2), \mathbf{y}(k-1)) \quad (45)$$

Por (45), pode ser visto que as ESNs implementam o modelo NARX de maneira direta (Martins, 2016; Jaeger, 2013), isto é, sem a necessidade de utilizar sinais atrasados no tempo como entradas adicionais para a rede.

3.9. Alguns trabalhos anteriores

3.9.1. ESNs para identificação de sistemas

No artigo “Identification of Motion” de Ishu et al. (2004) uma ESN é utilizada para modelar o movimento de um robô autônomo subaquático (“twin-burger”). Este artigo é também um dos primeiros a utilizar algoritmos evolutivos para otimizar ESNs. Na abordagem de Ishu et al. (2004), a otimização das ESNs é dividida em dois estágios. Inicialmente, são otimizados 3 parâmetros globais (tamanho do reservatório, raio espectral e a densidade de conexões no reservatório) por uma estratégia evolutiva. Em uma segunda etapa, os pesos das matrizes \mathbf{W}^{in} , \mathbf{W} e \mathbf{W}^{back} são, em um caso, totalmente otimizados e, em outra situação, parcialmente ajustados (mantendo a topologia da rede fixa). A abordagem que otimizou totalmente os pesos da rede na segunda etapa obteve resultados ligeiramente melhores do que a que otimizou parcialmente os pesos, porém as duas últimas foram melhores do que atuar somente na otimização dos 3 hiperparâmetros citados.

No trabalho de Xi et al. (2005), uma ESN é utilizada para identificar e simular um sistema caótico não linear, denominado Circuito de Chua. É dado enfoque especial à utilização das ESNs como um modelo de espaços de estados não linear, sendo mostrado que a ESN utilizada para modelar o sistema atua como uma realização não linear e de fase não mínima do Circuito de Chua no espaço de estados, além de a mesma conseguir prever de maneira satisfatória as saídas do sistema durante o teste. Também é observado que o estado inicial da rede exerce influência em seu desempenho, podendo ser livremente escolhido durante o treinamento, porém adequadamente determinado durante a fase de teste (Xi et al., 2005).

Em um trabalho (Li et al., 2012) envolvendo identificação de sistemas para uma aplicação de controle, uma ESN compõe, juntamente com uma *support vector machine* (SVM), uma estratégia para controlar um sistema caótico desconhecido. A ESN identifica a planta e atua como previsor de sua próxima saída, o que é utilizado para determinar a ação de controle no passo seguinte, enquanto que a SVM identifica o modelo inverso da planta, atuando como um cancelador de ruído. De acordo com os resultados de Li et al. (2012), a simulação da estratégia de controle atingiu bons resultados, apresentando estabilidade e praticamente reduzindo completamente os efeitos adversos de ruído.

No trabalho de Bian & Mou (2011), uma ESN é utilizada para identificar o sistema de movimentação de um veículo submarino não tripulado. Foi considerado que o veículo poderia realizar 6 tipos de movimentos desacoplados, utilizando-se uma ESN para modelar cada um deles. Neste trabalho, um algoritmo genético foi utilizado para otimizar 4 hiperparâmetros da rede: o raio espectral, o fator de escala dos pesos de entrada, a densidade de conexões e o tamanho do reservatório. De acordo com Bian & Mou (2011), a rede atingiu uma boa capacidade de aproximação com um rápido treinamento.

Um algoritmo de mínimos quadrados recursivo (RLS - *recursive least squares*) é utilizado para treinar de forma *online* uma ESN por Song et al. (2011). Tal estratégia é aplicada na identificação de sistemas não lineares. Foi verificado neste trabalho que a estratégia RLS – ESN apresentou desempenho superior em relação a um modelo linear ARMA (*autoregressive moving average*) na identificação tanto de um sistema não linear estático quanto de um sistema dinâmico não linear.

Em trabalho de Moon et al. (2016), um sistema MIMO (“*water wall system*”) componente de uma planta termoelétrica com 6 entradas e 3 saídas é identificado utilizando 3 modelos diferentes: um linear, uma rede *feedforward* com treinamento por *backpropagation* e uma ESN. Na comparação dos 3 modelos, foi verificada a superioridade dos modelos com redes neurais sobre o modelo linear, porém, em comparação à rede *feedforward*, a ESN mostrou um desempenho ligeiramente melhor em termos de precisão e muito superior em relação ao tempo de treinamento.

No artigo de Xing et al. (2012), a ESN é utilizada tanto para obter o modelo de um dispositivo denominado músculo pneumático, que apresenta características fortemente não lineares e de histerese, como para compor uma estratégia de controle para o mesmo. Durante a modelagem do sistema, a ESN é treinada de forma *online* utilizando

o algoritmo RLS e tem 3 de seus parâmetros globais (raio espectral, fator de escala dos pesos de entrada e conectividade do reservatório) otimizados pelo algoritmo PSO (*Particle Swarm Optimization*). Para controlar o sistema, uma outra ESN faz a identificação inversa da planta e juntamente com um controlador PID gera as ações de controle. Pelos resultados obtidos por Xing et al. (2012), foi verificado que a ESN conseguiu aproximar com boa precisão as características não lineares do músculo pneumático e a estratégia ESN-PID demonstrou ser auto adaptativa à dinâmica não linear do sistema e aos distúrbios externos.

Em uma aplicação recente de Lin & Zhang (2016), uma estratégia que emprega uma ESN treinada por um filtro de Kalman (KF – *Kalman Filter*) é usada para modelar um processo de moagem de cimento bruto com moinhos verticais. Em tal trabalho, a abordagem ESN-KF obteve desempenho superior tanto em termos de precisão na previsão do conjunto de teste quanto em termos de tempo de treinamento em comparação à uma rede *feedforward* com treinamento por *backpropagation*.

Em outro recente trabalho (Magerl et al., 2015), uma ESN é utilizada para modelar a resposta temporal de 3 tipos de circuitos integrados altamente não-lineares: driver cc-cc, misturador de RF e de um circuito de interface de conexão de redes locais. Foi verificado por Magerl et al. (2015), que a ESN conseguiu obter desempenho comparável à uma RNAR de referência com o mesmo número de neurônios, porém com um tempo de treinamento extremamente inferior. Também é observado neste trabalho, que o hiperparâmetro mais sensível da ESN em dois dos três estudos de caso foi a *leakage rate*.

Na dissertação de mestrado de Martins (2016) é proposto um método que define automaticamente uma ESN para aplicações de identificação de sistemas. O método proposto aplica o algoritmo genético (GA - *genetic algorithm*, em inglês) ou o PSO (*Particle Swarm Optimization*) para otimizar os parâmetros globais de uma ESN, que são o tamanho do reservatório, o raio espectral, constantes de ajuste das matrizes de entrada e de realimentação, escala do sinal de realimentação e constante de tempo do neurônio *leaky integrator*. Como para uma mesma configuração de parâmetros globais podem ser gerados reservatórios com desempenhos completamente diferentes, na etapa de avaliação de uma configuração é realizada uma média de um conjunto de redes com os mesmos hiperparâmetros. Após esta etapa inicial, em uma segunda etapa, é escolhido o reservatório mais adequado para a ESN como aquele que apresentou melhor desempenho

na simulação do conjunto de treinamento dentre um subconjunto de redes inicializadas com os mesmos parâmetros globais determinados anteriormente. O método proposto foi aplicado com sucesso na identificação em malha fechada do sistema de nível de um gerador de vapor de uma usina nuclear.

Outros exemplos de trabalhos que utilizaram ESNs para identificação de sistemas são os de Deng & Zhang (2006), Squartini et al. (2007), Zhou et al. (2010), Kim & Billard (2012), Pan & Wang (2012), Han & Wang (2014), Lin et al. (2014), Lun et al. (2014), Cao et al. (2015), Martins et al. (2015).

3.9.2. Otimização de ESNs por algoritmos evolutivos e/ou outras técnicas bioinspiradas

Além dos trabalhos de Ishu et al. (2004), Bian & Mou (2011), Xing et al. (2012) e Martins (2016), citados anteriormente, outros autores empregaram alguma técnica de otimização bioinspirada para ajustar automaticamente os parâmetros globais e/ou os pesos do reservatório de uma ESN. Um deles é o trabalho de Ferreira & Ludermir (2011), em que é proposto um método baseado em um algoritmo genético que automaticamente define uma ESN buscando simultaneamente pelos melhores parâmetros globais, pesos e topologia da rede e sem reescalar \mathbf{W} por um valor de raio espectral. São otimizados simultaneamente os valores de pesos de \mathbf{W} , \mathbf{W}^{in} , \mathbf{W}^{back} e matriz extras de *bias* \mathbf{W}^{bias} , além de parâmetros globais como o tamanho e o tipo de função de ativação do reservatório, o tipo de treinamento (regressão *ridge* ou pseudoinversa), *leakage rate*, nível de ruído introduzido nos estados do reservatório durante o treinamento, além de variáveis binárias definindo ou não a existência de conexões entre as diferentes camadas da ESN. Como proposto pelo método, o valor de raio espectral não é ajustado, com a justificativa de que reescalar \mathbf{W} incorre em um alto custo computacional, além de que ainda é possível obter a propriedade de *echo state* mesmo com $\rho_W > 1$. Como tal método otimiza simultaneamente parâmetros globais e pesos, neste trabalho foi desenvolvido um operador de *crossover* para cruzamento de reservatórios com tamanhos diferentes. O método RCDESIGN foi comparado com dois outros métodos denominados “*Classical Searching*”, que otimiza somente parâmetros globais (tamanho do reservatório, raio espectral e conectividade) e “*Topology and Radius Searching*” por parâmetros globais

(tamanho do reservatório, raio espectral e conectividade) e de topologia (existência ou não de conexões entre as camadas de entrada, bias, reservatório e de saída) da rede no problema de previsão de 3 séries temporais de velocidade do vento. Dos resultados obtidos no trabalho, verificou-se que a abordagem RCDESIGN apresentou os melhores resultados de previsão, além de um menor custo computacional. Viu-se também que não foi necessário limitar o raio espectral em um valor menor do que 1 para se gerar bons reservatórios.

Seguindo na mesma ideia de não reescalar o raio espectral de \mathbf{W} , Basterrech et al. (2014) propuseram uma estratégia em que somente um subconjunto de pesos do reservatório são otimizados através do PSO (*Particle Swarm Optimization*), mantendo-se fixos os parâmetros globais da rede, previamente definidos empiricamente. Nos problemas considerados por Basterrech et al. (2014), tal estratégia gerou melhores resultados em comparação a uma ESN canônica, com reservatório totalmente aleatório.

Uma extensão do trabalho de Basterrech et al. (2014) é realizada por Chouikhi et al., (2015, 2017). Nestes trabalhos, o PSO otimiza não só um subconjunto de pesos de \mathbf{W} , mas também alguns pesos de \mathbf{W}^{in} e \mathbf{W}^{back} . Parâmetros globais são ajustados por tentativa e erro antes da otimização e não há reescala para um valor de raio espectral no reservatório. A metodologia ESN-PSO apresentou desempenho 100 vezes superior em comparação a uma ESN convencional, com reservatório aleatório, no problema de previsão da série temporal Mackey-Glass (Chouikhi et al., 2015), além de ter diversos métodos em problemas *benchmark* e reais de previsão de séries temporais (Chouikhi et al., 2017).

O trabalho de Yuenyong (2015) visa realizar uma estratégia simples e eficiente computacionalmente para a otimização de ESNs, além de resolver o problema de ruído gerado na função de aptidão devido à inicialização aleatória do reservatório quando se busca por parâmetros globais da rede. Para isto, no método de Yuenyong (2015), geram-se aleatoriamente uma única vez as matrizes \mathbf{W}_0^{in} e \mathbf{W}_0 , salvando-as em memória juntamente com o raio espectral de \mathbf{W}_0 . A partir daí, o método consiste em otimizar usando o CMA-ES os parâmetros globais s_{in} e ρ_w , que serão utilizados para gerar, respectivamente, as matrizes \mathbf{W}^{in} e \mathbf{W} escalando-se as matrizes *templates* \mathbf{W}_0^{in} e \mathbf{W}_0 previamente salvas. Tal método é eficiente computacionalmente e torna determinística a otimização de uma ESN, mas, para a utilização de tal abordagem, outros parâmetros da

rede, como o número de neurônios do reservatório e a densidade de conexões no mesmo devem ser previamente conhecidos.

Dentro da otimização de parâmetros gerais da rede, pode-se destacar os trabalhos de Ferreira & Ludermir (2009) e Sergio & Ludermir (2012), onde, respectivamente, o GA e o PSO são utilizados para otimizar o tamanho do reservatório, o raio espectral, o tipo de função no reservatório e a existência ou ausência de conexões da camada de entrada com a saída e da saída com a saída em aplicações de previsão de séries temporais. Também em uma tarefa para previsão de séries temporais, em sua dissertação, Rueda (2014) desenvolveu uma ferramenta, que utiliza o GA ou o PSO para otimizar parâmetros globais da rede (tamanho do reservatório, percentual de conectividade no reservatório, raio espectral e presença ou não de realimentação da saída) e o tamanho da janela utilizada na previsão.

Outro trabalho que merece destaque é o de Niño (2016). Como já citado na seção 3.7.4, este trabalho foca na otimização de uma ESN para a tarefa de previsão de séries temporais. Para isto, inicialmente busca-se os parâmetros globais da rede (número de neurônios no reservatório, percentual de conexões em \mathbf{W} , raio espectral, fator de escala de \mathbf{W}^{in} e fator de escala de \mathbf{W}^{back}) e a janela de previsão utilizando um algoritmo genético. Na segunda etapa, busca-se o reservatório mais adequado para uma ESN utilizando método SRG descrito em 3.7.4.

Uma série de outros trabalhos utilizaram algoritmos evolutivos e/ou outras técnicas bioinspiradas para otimizar parâmetros globais e/ou pesos das ESNs, como Ferreira (2011), Duan & Wang (2016), Ferreira & Ludermir (2010, 2011), Chatzidimitriou & Mitkas (2010, 2013), Martins et al. (2015) e Roeschies & Igel (2010) e muitos outros não citados, cujas descrições serão omitidas nesta dissertação.

4

Algoritmos Evolutivos com Inspiração Quântica e Estratégias Evolucionárias

4.1.

Introdução

Algoritmos evolutivos (AEs) são modelos computacionais estocásticos inspirados em processos naturais de evolução e que funcionam como ferramentas alternativas aos métodos tradicionais de busca e otimização na solução de diversos problemas (Linden, 2012). Tais algoritmos possuem um mecanismo de busca que vai evoluindo uma população de indivíduos (possíveis soluções) ao longo de gerações por meio de operadores bioinspirados, como cruzamento, seleção e mutação e com base na aptidão de cada indivíduo, o que é medido através de uma função de avaliação, que é específica para cada problema. Por lidarem ao mesmo tempo com múltiplos indivíduos, tais algoritmos apresentam um paralelismo intrínseco que os tornam naturalmente implementáveis em computadores paralelos e mais adequados em relação a métodos tradicionais de otimização na solução de problemas complexos e/ou custosos computacionalmente, além de apresentarem boa capacidade de busca global, atingindo bons resultados e sendo flexíveis e adaptáveis à diversas aplicações (Moore & Narayanan, 1995; Back et al., 1997). Exemplos de AEs clássicos são os algoritmos genéticos (Holland, 1975), a programação evolucionária (Fogel et al., 1966) e as estratégias evolucionárias (Rechenberg, 1965, 1973; Schwefel, 1965, 1975).

Como este trabalho visa o desenvolvimento de um modelo que permita definir automaticamente uma ESN, optou-se por empregar algoritmos evolutivos para a otimização dos hiperparâmetros e/ou pesos da rede, o que já foi abordado por outros autores, como visto na seção 3.9. Além disso, objetivando reduzir o custo computacional associado ao processo de busca, optou-se pela utilização de algoritmos evolutivos que normalmente fornecem boas soluções com poucas avaliações, como o AEIQ-R (Cruz, 2007) e o CMA-ES (Hansen & Ostermeier, 1996). Com isto, as seções a seguir abordam brevemente os conceitos principais necessários ao entendimento destes dois algoritmos.

4.2.

Algoritmos Evolutivos com Inspiração Quântica

4.2.1.

Introdução

Um conceito chave em mecânica quântica é o de superposição de estados. Por esta propriedade, é possível que uma partícula atômica esteja simultaneamente em vários estados (Pinho, 2010). Genericamente, assumindo que uma partícula possa estar em n_e estados possíveis, a localização quântica de uma partícula pode ser descrita por um vetor quântico de estados $|\Psi\rangle$ (Narayanan & Moore, 1996):

$$|\Psi\rangle = \alpha_1|A_1\rangle + \alpha_2|A_2\rangle + \dots + \alpha_{n_e}|A_{n_e}\rangle \quad (46)$$

onde $\alpha_1, \alpha_2, \dots, \alpha_{n_e}$ são números complexos que representam os pesos de cada possível estado A_1, A_2, \dots, A_{n_e} .

Desta forma, por (46) é possível afirmar que uma partícula está ao mesmo tempo em n_e estados, entretanto, o ato de observar o estado quântico, o colapsa a um único estado (Narayanan & Moore, 1996).

A computação quântica é um termo cunhado para descrever processos computacionais que se baseiam na superposição de estados e em outros fenômenos específicos da mecânica quântica e que são capazes de reduzir esforço e complexidade computacional na resolução de determinados problemas (Cruz, 2007). Tal redução de complexidade parte da ideia de que o espaço de busca abrangido por um sistema quântico cresce exponencialmente de acordo com o seu tamanho (Rieffel & Polak, 2000). Esta afirmação é embasada pelo fato de que, diferentemente de em um computador clássico em que a menor unidade de informação é o *bit*, que pode assumir o estado “0” ou “1”, em um computador quântico a menor unidade de representação é o *bit* quântico ou simplesmente *q-bit*. Um bit quântico pode assumir o estado “0”, “1” ou uma superposição dos mesmos, o que utilizando a notação de *Dirac* é representado por:

$$|\Psi\rangle = \alpha|0\rangle + \beta|1\rangle \quad (47)$$

onde $|\Psi\rangle$ é o estado de um q -bit, α e β são números complexos tal que $|\alpha|^2$ e $|\beta|^2$ representam as probabilidades de se observarem os estados $|0\rangle$ e $|1\rangle$, respectivamente. Ainda, α e β satisfazem:

$$|\alpha|^2 + |\beta|^2 = 1 \quad (48)$$

o que é graficamente representado pela figura 14 (assumindo α e β reais):

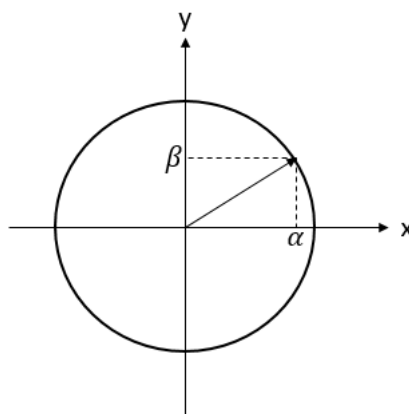


Figura 14 – Representação gráfica de um círculo unitário com as amplitudes de probabilidade de se observarem os estados “0” e “1” para um q -bit. Fonte: Adaptado de Cruz (2007)

É interessante de se observar que com m q -bits, pode-se representar 2^m estados simultaneamente, o que dá a ideia do “paralelismo exponencial”. Desta forma, algoritmos quânticos podem considerar simultaneamente uma região muito maior do espaço de busca de um dado problema, mesmo que certos estados não venham a ser utilizados, o que os tornam, em teoria, exponencialmente mais eficientes do que seus análogos clássicos (Cruz, 2007; Rieffel & Polak, 2000).

Embora uma eficiência computacional maior seja anunciada pela computação quântica, de acordo com Cruz (2007), dois problemas atualmente impedem que a mesma se torne uma ferramenta útil: (i) a dificuldade em se construir computadores quânticos e (ii) a dificuldade de se implementarem algoritmos para aproveitarem a capacidade de processamento destes dispositivos. De forma a contornar tais dificuldades, o conceito de “computação com inspiração quântica” foi apresentado por Moore & Narayanan (1995).

A ideia da “computação com inspiração quântica” é criar algoritmos clássicos (e portanto, capazes de serem executados em computadores clássicos) que tirem proveito dos paradigmas da mecânica quântica, de forma a melhorar o desempenho dos mesmos

na resolução de problemas (Cruz, 2007). No âmbito da computação evolucionária, seguindo esta filosofia, foi criado por Han & Kim (2000), uma versão de algoritmo evolutivo com inspiração quântica e representação binária (AEIQ-B), que posteriormente ganhou extensões como o AEIQ-R (algoritmo evolutivo com inspiração quântica e representação real) de Cruz (2007) e o AEIQ-BR (algoritmo evolutivo com inspiração quântica e representação binário-real) de Pinho (2010). Tais algoritmos possuem como principais características (Pinho, 2010; Botelho, 2014):

- Capacidade de otimização global de um problema;
- Representação probabilística do espaço de busca e;
- Necessidade de poucos indivíduos na população para atingirem bons resultados, o que reduz em muito o número de avaliações necessárias em um processo de otimização e torna os AEIQs mais eficientes em termos de convergência em relação aos algoritmos genéticos convencionais.

Dito isto, as subseções seguintes dedicam-se a descrever de forma sucinta o AEIQ-B e o AEIQ-R.

4.2.2.

Algoritmo Evolutivo com Inspiração Quântica e Representação Binária – AEIQ-B

O algoritmo evolutivo com inspiração quântica e representação binária (AEIQ-B) aqui descrito foi proposto por Han & Kim (2000, 2002), sendo inicialmente aplicado em problemas de otimização combinatória. A figura 15 mostra o pseudocódigo do AEIQ-B:

<p>AEIQ-B</p> <p>Início</p> <p>1: $t \leftarrow 0$</p> <p>2: Inicializa a população quântica $Q(t)$ com representação em q-bits</p> <p>3: Gera a população clássica $P(t)$ com representação binária, observando $Q(t)$</p> <p>4: Avalia $P(t)$</p> <p>5: $B(t) \leftarrow P(t)$, onde $B(t)$ é a população dos melhores indivíduos clássicos</p> <p>6: Enquanto (condição de parada) faça</p> <p>7: $t \leftarrow t+1$</p> <p>8: Gera $P(t)$ observando $Q(t-1)$</p> <p>9: Avalia $P(t)$</p> <p>10: $B(t) \leftarrow$ melhores indivíduos de $[P(t) \cup B(t-1)]$</p> <p>11: $Q(t) \leftarrow$ atualização ($Q(t-1)$)</p> <p>12: Fim Enquanto</p> <p>Fim</p>
--

Figura 15 - Pseudocódigo do AEIQ-B. Fonte: Adaptado de Han & Kim (2000) e Pinho (2010)

No AEIQ-B, um gene quântico g simula o funcionamento de um q -bit, assumindo probabilisticamente uma superposição dos estados “0” e “1” e sendo representado por um par de números complexos α e β :

$$g = \begin{pmatrix} \alpha \\ \beta \end{pmatrix}, |\alpha|^2 + |\beta|^2 = 1 \quad (49)$$

O ato de se observar ou instanciar um gene quântico o converte (colapsa) em um valor inteiro (neste caso, 0 ou 1). O valor de $|\alpha|^2$ indica a probabilidade de se obter o valor 0 ao se observar o gene quântico, enquanto que $|\beta|^2$ informa a probabilidade de se obter 1 neste processo. Baseando-se nesta ideia, no AEIQ-B, coexistem duas populações, a população quântica $Q(t)$ e a população clássica $P(t)$.

A população quântica num instante de tempo t ($Q(t)$) é composta por N_Q indivíduos quânticos:

$$Q(t) = [q_1^t \ q_2^t \ q_3^t \ \dots \ q_{N_Q}^t] \quad (50)$$

onde q é um indivíduo quântico formado por m q -bits:

$$q = \left[\begin{pmatrix} \alpha_1 \\ \beta_1 \end{pmatrix} \begin{pmatrix} \alpha_2 \\ \beta_2 \end{pmatrix} \begin{pmatrix} \alpha_3 \\ \beta_3 \end{pmatrix} \dots \begin{pmatrix} \alpha_m \\ \beta_m \end{pmatrix} \right] \quad (51)$$

No momento de inicialização da população quântica $Q(t)$ (passo 2 da figura 15), todos os indivíduos quânticos são gerados de forma que inicialmente exista a mesma probabilidade de se gerarem os estados 0 e 1 quando os q -bits forem observados, isto é, $\alpha_j = \beta_j = \frac{1}{\sqrt{2}}, \forall j = 1, 2, \dots, m$.

A população clássica $P(t)$ é gerada a partir da observação da população quântica $Q(t)$ (passos 3 e 8 na figura 15), o que converte cada indivíduo quântico em uma *string* binária, que é obtida através do colapso de cada q -bit em um valor inteiro (0 ou 1). Cada gene clássico de um indivíduo de $P(t)$ pode ser obtido de seu correspondente gene quântico em $Q(t)$ utilizando-se a seguinte heurística (Pinho, 2010):

1 – Gera-se um número aleatório $r \in [0,1]$;

2 – Se r estiver entre 0 e $|\alpha|^2$, atribui-se 0 ao gene clássico, caso contrário, atribui-se 1 ao mesmo.

Após a geração e a avaliação de $P(t)$ (passos 4 e 9 da figura 14), uma segunda população clássica $B(t)$, que é inicialmente equivalente à $P(t)$ (passo 5 na figura 15), é utilizada para manter os melhores indivíduos clássicos, armazenando as melhores soluções de $P(t) \cup B(t - 1)$ (passo 10 da figura 15).

No último passo do *loop* de funcionamento do AEIQ-B, a população quântica é atualizada modificando α e β com o objetivo de aumentar a probabilidade de se observarem melhores indivíduos. Isto é feito através do operador q -gate (Han & Kim, 2000), que “rotaciona” α e β na direção de valores que potencialmente podem gerar indivíduos de melhor aptidão (α' e β'):

$$\begin{pmatrix} \alpha' \\ \beta' \end{pmatrix} = \begin{pmatrix} \cos(\theta) & -\text{sen}(\theta) \\ \text{sen}(\theta) & \cos(\theta) \end{pmatrix} \begin{pmatrix} \alpha \\ \beta \end{pmatrix} \quad (52)$$

onde θ é o ângulo de rotação (figura 16).

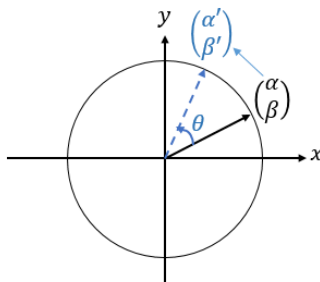


Figura 16 – Representação gráfica da rotação de um q-bit usando o operador q -gate.

Maiores detalhes sobre o funcionamento do AEIQ-B podem ser consultados nos trabalhos de Han & Kim (2000, 2002).

4.2.3. Algoritmo Evolutivo com Inspiração Quântica e Representação Real – AEIQ-R

O algoritmo evolutivo com inspiração quântica e representação real (AEIQ-R), descrito nesta seção, é uma versão de AEIQ mais apropriada para a resolução de problemas numéricos que foi proposta por Cruz (2007), sendo uma extensão do AEIQ-B de Han & Kim (2000). O AEIQ-R foi aplicado com sucesso por Cruz (2007) em problemas *benchmark* de otimização numérica e no treinamento de redes neurais recorrentes para tarefas de previsão de séries temporais e aprendizado por reforço. A figura 17 mostra o pseudocódigo do algoritmo:


```

AEIQ-R
Início
1:   $t \leftarrow 1$ 
2:  Inicializa população quântica  $Q(t)$  com  $N_Q$  indivíduos com  $G$  genes cada
3:  Enquanto ( $t \leq Gen$ ) faça
4:    Gera população clássica  $P(t)$  com representação real observando  $Q(t)$ 
5:    Se ( $t=1$ ) então
6:       $B(t) \leftarrow P(t)$ , onde  $B(t)$  é a população de melhores indivíduos clássicos
7:    Senão
8:       $P(t) \leftarrow$  recombinação clássica entre  $P(t)$  e  $B(t - 1)$ 
9:      Avalia  $P(t)$ 
10:      $B(t) \leftarrow$  melhores indivíduos de  $[P(t) \cup B(t - 1)]$ 
11:    Fim Se
12:     $Q(t + 1) \leftarrow$  atualização ( $Q(t)$ )
13:     $t \leftarrow t + 1$ 
14:  Fim Enquanto
Fim

```

Figura 17 – Pseudocódigo do AEIQ-R. Fonte: Adaptado de Cruz (2007) e Pinho (2010)

De forma análoga ao AEIQ-B, no AEIQ-R coexistem duas populações de indivíduos: a quântica $Q(t)$ e a clássica $P(t)$. Entretanto, diferentemente do AEIQ-B, o AEIQ-R é fundamentado na abordagem de funções de onda (Cruz, 2007), onde os genes quânticos deste algoritmo simulam a superposição de estados contínuos e não discretos (como o “0” e o “1” de um q -bit) os quais são utilizados para obter números reais na representação de seus indivíduos. Para representar genes quânticos reais, que são contínuos em seu domínio, Cruz (2007) fez uso de funções densidade de probabilidade (fdps):

$$g_{ij}(x) = p_{ij}(x) \quad (53)$$

onde $g_{ij}(x)$ representa o gene quântico j de um indivíduo quântico i e $p_{ij}(x)$ é uma fdp que satisfaz:

$$\int_{-\infty}^{+\infty} p_{ij}(x) dx = 1 \quad (54)$$

Com isso, a função $p_{ij}(x)$ representa a densidade de probabilidade de se observar um determinado valor para um gene quântico quando a superposição do mesmo for colapsada (Cruz, 2007). Dentre as diferentes fdps possíveis (normal, exponencial, *weibull*, etc) para representar $p_{ij}(x)$, por simplicidade, Cruz (2007) escolheu uma função

de pulso retangular ou distribuição uniforme (figura 18), desenvolvendo todos os operadores de seu algoritmo baseando-se nesta distribuição.

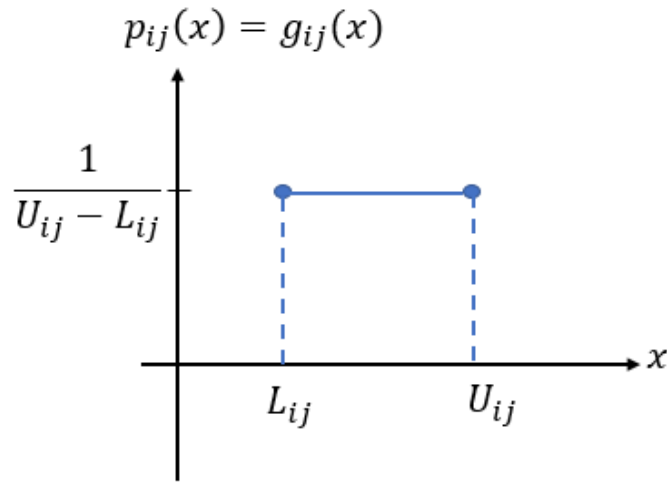


Figura 18 – Função pulso retangular ou distribuição uniforme de probabilidade, onde L_{ij} e U_{ij} representam os valores de domínio inferior e superior, respectivamente, de $p_{ij}(x)$

Utilizando-se como referência a largura σ_{ij} ($\sigma_{ij} = U_{ij} - L_{ij}$) e o centro do pulso μ_{ij} ($\mu_{ij} = \frac{U_{ij} + L_{ij}}{2}$), a função $p_{ij}(x)$ pode ser representada por:

$$p_{ij}(x) = \begin{cases} \frac{1}{\sigma_{ij}}, & \text{se } \left(\mu_{ij} - \frac{\sigma_{ij}}{2} \leq x \leq \mu_{ij} + \frac{\sigma_{ij}}{2}\right) \\ 0, & \text{outro caso} \end{cases} \quad (55)$$

Desta forma, no AEIQ-R, a população quântica $Q(t)$, que representa a superposição dos possíveis estados que um indivíduo clássico de $P(t)$ pode assumir (Cruz, 2007), com N_Q indivíduos é representada por:

$$Q(t) = [q_1, q_2, \dots, q_{N_Q}] \quad (56)$$

onde q_i é indivíduo quântico i que pode ser representado por:

$$q_i = [g_{i1} = p_{i1} = \binom{\mu_{i1}}{\sigma_{i1}}, g_{i2} = p_{i2} = \binom{\mu_{i2}}{\sigma_{i2}}, \dots, g_{iG} = p_{iG} = \binom{\mu_{iG}}{\sigma_{iG}}] \quad (57)$$

e sendo G o número de genes quânticos de q_i .

No início do algoritmo (passo 2 da figura 17), N_Q indivíduos quânticos são gerados para formar a população quântica inicial do algoritmo. Segundo Cruz (2007), este

processo pode ocorrer de forma que os pulsos iniciais tenham largura $(U_{ij} - L_{ij})/N_Q$ com seus centros distribuídos uniformemente ao longo do domínio de todas as variáveis ou de maneira que σ_{ij} e μ_{ij} de todos os indivíduos quânticos sejam obtidos a partir dos limites inferiores e superiores das variáveis de domínio do problema, isto é, valores L_{ij} e U_{ij} de cada gene quântico de um indivíduo q_i coincidam, respectivamente, com os limites inferiores e superiores de cada variável do problema.

Uma vez que a população quântica $Q(t)$ inicial é obtida, o AEIQ-R inicia um *loop* por Gen gerações, onde no passo 4 a população clássica $P(t)$ é gerada. Neste processo, um gene clássico x_{ij} é obtido de um gene quântico g_{ij} , o que para o caso da distribuição uniforme da figura 18 consiste em obter x_{ij} usando –se (Cruz, 2007):

$$x_{ij} = r_{ij} \cdot \sigma_{ij} + \left(\mu_{ij} - \frac{\sigma_{ij}}{2} \right) \quad (58)$$

onde r_{ij} é um número gerado aleatoriamente no intervalo $[0, 1]$.

De (56), para cada indivíduo clássico de $P(t)$ são obtidos G genes clássicos a partir de G genes quânticos de $Q(t)$. Entretanto, de acordo com Cruz (2007), com o objetivo de manter um procedimento de amostragem adequado, o número de indivíduos de $P(t)$ (N_C) é sempre maior ou igual ao número de indivíduos de $Q(t)$ (N_Q) ($N_C \geq N_Q$). Uma maneira de se atingir tal objetivo é fazer com que o tamanho da população clássica seja um múltiplo do tamanho da população quântica, isto é, $N_C = kN_Q$, onde k é um número inteiro positivo.

Pela figura 17, antes da avaliação da população clássica, no passo 8 há uma operação de recombinação entre $P(t)$ e $B(t - 1)$, onde $B(t)$ é a população que armazena os melhores indivíduos clássicos. Esta etapa é realizada utilizando uma operação de *crossover* aritmético, que é descrita como:

$$x_{ij}^t = \begin{cases} x_{ij_B}^t + \lambda^t (x_{ij_P}^t - x_{ij_B}^t), & \text{se } r_{ij}^t \text{ estiver entre } 0 \text{ e } C_r \\ x_{ij_B}^t, & \text{caso contrário} \end{cases} \quad (59)$$

onde r_{ij} é um número aleatório entre 0 e 1 sorteado para cada gene, λ é um número aleatório entre 0 e 1, C_r é a taxa de *crossover*, x_{ij_B} é um gene clássico de um indivíduo

sorteado de $B(t - 1)$, x_{ij_P} é um gene clássico de um indivíduo sorteado de $P(t)$ e x_{ij_F} é o gene do indivíduo clássico resultante do cruzamento.

Antes de se finalizar o *loop* de funcionamento do AEIQ-R, no passo 12 da figura 17, há a atualização da população quântica $Q(t)$. Tal processo consiste em determinar os novos valores de μ_{ij} e σ_{ij} para a próxima geração. Este procedimento deve ser realizado de forma a mapear as soluções mais promissoras, movendo-se o centro dos pulsos na direção destas soluções, e reduzir o espaço de busca, diminuindo a largura dos pulsos (Cruz, 2007). No caso do ajuste do centro dos pulsos, os valores de μ_{ij} são movidos na direção dos melhores indivíduos de $B(t)$ seguindo-se:

$$\mu_{ij}^{t+1} = \mu_{ij}^t + \lambda^t (x_{ij}^t - \mu_{ij}^t) \quad (60)$$

onde λ é o mesmo número de (59) e x_{ij} é um indivíduo clássico de $B(t)$.

Entre as alternativas para a atualização de σ_{ij} , há a regra do 1/5 (Michalewicz, 1994) que foi a utilizada por Cruz (2007), um decaimento linear ou um decaimento exponencial da largura dos pulsos, que foi o utilizado por Pinho (2010) (e também será a utilizada neste trabalho) no algoritmo AEIQ-BR (Algoritmo Evolutivo com Inspiração Quântica e Representação Binário-Real). No caso da regra do 1/5, a modificação da largura dos pulsos ocorre de forma igual e homogênea para todos os indivíduos quânticos (Cruz, 2007), sendo a heurística utilizada para determinar o aumento ou a diminuição da largura dos pulsos a seguinte (Michalewicz, 1994):

- Se menos de 20% da população clássica criada na geração atual tiver uma avaliação melhor do que na geração anterior, a largura do pulso é reduzida;
- Se a melhora da avaliação da população clássica atual em relação à geração anterior for maior do que 20%, a largura do pulso é aumentada;
- Caso a taxa de melhora for exatamente 1/5, não é realizada nenhuma alteração.

Matematicamente, a regra do 1/5 pode ser expressa como (Cruz, 2007):

$$\sigma_{ij}^{t+1} = \begin{cases} \sigma_{ij}^t \cdot \delta, & \text{se } \varphi < 1/5 \\ \sigma_{ij}^t / \delta, & \text{se } \varphi > 1/5 \\ \sigma_{ij}^t, & \text{se } \varphi = 1/5 \end{cases} \quad (61)$$

onde δ é um número aleatório entre 0 e 1 e φ é a taxa de melhora dos indivíduos clássicos em relação à geração anterior.

No caso de se utilizar uma regra de decaimento exponencial para a atualização da largura dos pulsos, a atualização de σ_{ij} segue (Pinho, 2010):

$$\sigma_{ij}^{t+1} = \sigma_{ij}^t + \lambda^t (\max(x_{ij}^t) - \min(x_{ij}^t) - \sigma_{ij}^t) \quad (62)$$

onde λ é o mesmo de (57), $\max(x_{ij}^t)$ e $\min(x_{ij}^t)$ representam, respectivamente, os valores máximo e mínimo de um indivíduo clássico de $B(t)$.

A atualização de $Q(t)$ pode ser feita a cada geração ou em intervalo fixo e pré-definido de gerações (controlado pelo hiperparâmetro upG) e pode ser aplicada em todos os indivíduos quânticos ou sobre uma porcentagem dos mesmos (definida pelo metaparâmetro C_Q).

Uma vez definido o funcionamento básico do AEIQ-R, a tabela 1 mostra os parâmetros gerais do algoritmo que devem ser definidos pelo usuário:

Tabela 1- Parâmetros do AEIQ-R

Parâmetro	Descrição
N_C	Número de indivíduos clássicos
N_Q	Número de indivíduos quânticos
C_r	Taxa de <i>crossover</i> clássico
Gen	Número de gerações
upG	Intervalo de gerações para atualizar $Q(t)$
C_Q	Taxa de atualização de $Q(t)$

Uma explicação mais detalhada sobre o funcionamento do AEIQ-R pode ser encontrada no trabalho de Cruz (2007).

4.3. Estratégias Evolucionárias

4.3.1. Introdução

As estratégias evolucionárias (do inglês, *evolutionary strategies* - ES) foram desenvolvidas na Alemanha durante as décadas de 1960 e 1970 por Rechemberg (1965, 1973) e Schwefel (1965, 1975) para a resolução de problemas de otimização numérica e de forma independente e quase simultânea aos algoritmos genéticos (Nelles, 2001). Em uma ES, cada indivíduo é representado por um vetor v_i que contém tanto parâmetros relativos ao problema de otimização em questão ($x = x_1, \dots, x_n$) quanto parâmetros da estratégia, que são representados pelo vetor $\sigma = \sigma_1, \sigma_2, \dots, \sigma_n$, que pode incluir além de desvios padrões σ_i , que representam o passo das mutações, taxas de mutação e recombinação, entre outros (figura 19).

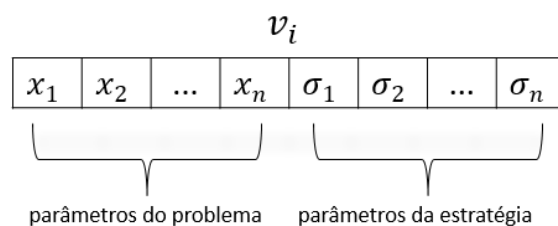


Figura 19 – Indivíduo de uma ES convencional

Com este tipo de representação, as estratégias evolucionárias possuem como principal vantagem o fato de que os parâmetros da própria estratégia são conjuntamente evoluídos com os parâmetros do problema, as caracterizando como algoritmos auto adaptáveis. Outra característica importante das ESs é o fato de que o operador predominantemente utilizado para modificar seus indivíduos a cada geração é o de mutação gaussiana. Neste caso, a operação de mutação é realizada em duas fases, sendo inicialmente perturbados aleatoriamente cada σ_i do vetor de desvios padrões gerando-se um σ_i' correspondente e, posteriormente, adicionado a cada x_i do vetor de parâmetros do problema um valor amostrado aleatoriamente de uma distribuição normal de média 0 e desvio atualizado correspondente σ_i' , gerando-se o x_i' (Linden, 2012; Silva, A. F., 2014):

$$\begin{cases} \sigma_i' = \sigma_i + N(0,1) \\ x_i' = x_i + N(0, \sigma_i') \end{cases} \quad (63)$$

onde σ_i' e x_i' são, respectivamente, os valores para a próxima geração de σ_i e x_i de uma posição de um indivíduo e $N(0, \sigma)$ representa a distribuição normal, que é dada por:

$$N(0, \sigma, x) = \frac{e^{-\frac{1}{2}\left(\frac{x}{\sigma}\right)^2}}{\sigma\sqrt{2\pi}} \quad (64)$$

Quando a operação de *crossover* é utilizada nas ESs, ela pode envolver a recombinação de dois indivíduos aleatoriamente escolhidos ou a combinação linear de vários progenitores para se obter um filho.

O mecanismo de seleção da população da geração seguinte nas ESs é realizado após os operadores de variação (mutação e/ou *crossover*) com o objetivo de substituir a população atual pelos melhores indivíduos resultantes da reprodução. Neste caso, algumas estratégias evolucionárias clássicas podem ser destacadas:

- (1+1) – ES – representa a versão mais simples de ES, onde a população a cada geração é composta por um único indivíduo, que através da mutação gera um único filho. A seleção da população para a geração seguinte é realizada escolhendo-se o melhor indivíduo entre o pai e o filho na geração atual ou escolhendo-se aleatoriamente um dos mesmos (Rechenberg, 1973).

- $(\mu + \lambda)$ – ES – neste tipo de estratégia, um número de descendentes $\lambda \geq 1$ a cada geração é obtido a partir de um número de pais $\mu \geq 1$. Durante a seleção, a população para a geração seguinte é obtida escolhendo-se os μ melhores indivíduos dentre o total de $\mu + \lambda$ pais e filhos da geração atual.

- (μ, λ) – ES – onde durante a seleção um conjunto de λ filhos, que compõem a população seguinte, substituem seus μ progenitores, não importando quão bons os μ pais sejam em comparação aos seus λ descendentes. Neste caso, o número de descendentes é sempre maior do que o número de indivíduos utilizados como pais, isto é, $\lambda > \mu$.

A versão de ES entendida como o estado da arte para problemas de otimização numérica é a estratégia evolutiva com adaptação da matriz de covariâncias, mais

conhecida como CMA-ES (*covariance matrix adaptation evolution strategy*), cujo algoritmo será descrito na subseção seguinte.

4.3.2. Estratégia evolucionária com adaptação da matriz de covariâncias – CMA-ES

4.3.2.1. Introdução

O algoritmo CMA-ES foi proposto por Hansen & Ostermeier (1996) e é considerado o AE estado da arte para a otimização de parâmetros reais (Hansen et al., 2010; Hansen & Kern, 2004; Hansen & Ostermeier, 2001). Em relação aos parâmetros que normalmente são adaptados em uma ES convencional, no CMA-ES é também adaptada uma matriz de covariâncias \mathbf{C} , que descreve a dependência de cada par de variáveis (parâmetros do problema a ser otimizado) de uma distribuição normal multivariada (Strickler, 2017). Uma distribuição normal multivariada, por sua vez, pode ser descrita por:

$$N_p(\mathbf{m}, \mathbf{C}, \mathbf{x}) \sim \mathbf{m} + N_p(\mathbf{0}, \mathbf{C}, \mathbf{x}) = \frac{1}{2\pi^{p/2}|\mathbf{C}|^{1/2}} e^{-\frac{1}{2}(\mathbf{x}-\mathbf{m})^\top(\mathbf{C})^{-1}(\mathbf{x}-\mathbf{m})} \quad (65)$$

onde \mathbf{x} é um vetor com p variáveis, $\mathbf{m} \in \mathbb{R}^p$ é o centroide (ou vetor de médias) da distribuição e $\mathbf{C} \in \mathbb{R}^{p \times p}$ é uma matriz simétrica e positiva definida, denominada matriz de covariâncias, que é dada por:

$$\mathbf{C} = \begin{bmatrix} \sigma_{11} & \sigma_{12} & \cdots & \sigma_{p1} \\ \sigma_{21} & \sigma_{22} & \cdots & \sigma_{p2} \\ \vdots & \vdots & \ddots & \vdots \\ \sigma_{p1} & \sigma_{p2} & \cdots & \sigma_{pp} \end{bmatrix} \quad (66)$$

onde σ_{ij} é a covariância entre as variáveis i e j .

A matriz de covariâncias muda a forma da distribuição, o que é exemplificado nos gráficos de contorno da figura 20, que são obtidos igualando-se (65) a valores constantes para o caso bivariado (Johnson & Wichern, 2007).

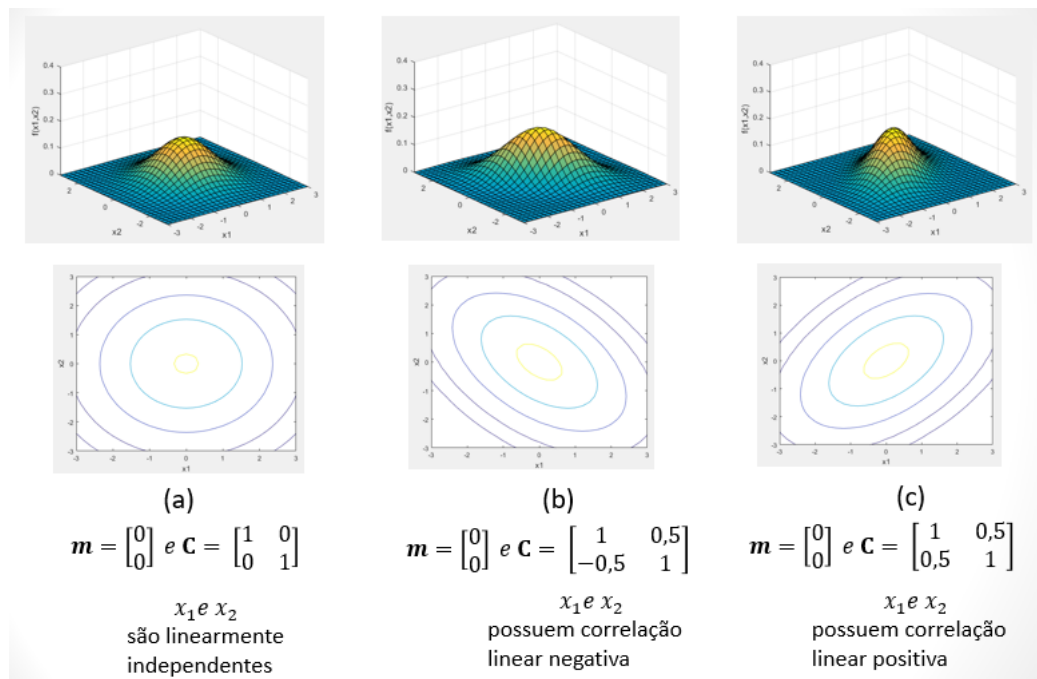


Figura 20 – Mudança dos contornos de uma distribuição normal bivariada com a mudança da matriz \mathbf{C} . Fonte: Adaptado de Pessanha (2018) e Johnson & Wichern (2007)

Como mostrado pela figura 20, quando a matriz \mathbf{C} é diagonal, as variáveis do problema são linearmente independentes, fazendo com que o contorno da distribuição seja uma circunferência centrada em $\mathbf{m} = [m_1, m_2]$. Em outros casos, os contornos da distribuição são elipses centradas em $\mathbf{m} = [m_1, m_2]$, cujos eixos estão nas direções dos autovetores de \mathbf{C} (Johnson & Wichern, 2007).

A adaptação da matriz de covariâncias, que é um dos pontos centrais do CMA-ES, tem como objetivo fazer com que as linhas de contorno da distribuição normal se aproximem das linhas de contorno da função de aptidão do problema, o que é análogo a aprender o inverso da matriz hessiana em métodos quase-Newton (Hansen, 2016). Outro ponto importante do CMA-ES é a adaptação, de forma independente da matriz \mathbf{C} , do tamanho do passo global do algoritmo (σ), o que é importante para evitar uma convergência prematura e ao mesmo tempo permitir uma convergência rápida do algoritmo para uma solução ótima (Hansen, 2016).

Uma descrição simplificada do algoritmo é dada a seguir.

4.3.2.2. Descrição do algoritmo

De uma maneira bem simplificada, omitindo muitos detalhes de funcionamento, a figura 21 mostra o pseudocódigo do CMA-ES:

CMA-ES	
1:	Entrada: centroide inicial \mathbf{m} , passo global inicial σ e tamanho da população λ
2:	Inicialize parâmetros do algoritmo (com a matriz de covariâncias $\mathbf{C} = \mathbf{I}$)
3:	Repita
4:	Mutação: gere λ indivíduos amostrando uma distribuição normal $N(\mathbf{m}, \sigma^2 \mathbf{C})$
5:	Avalie a população de λ indivíduos
6:	Selecione μ pais
7:	Atualize o centroide \mathbf{m} através da recombinação dos μ pais do passo 6
8:	Adapte a matriz de covariâncias \mathbf{C}
9:	Adapte o tamanho de passo global σ
10:	Até critério de parada ser atingido

Figura 21 – Pseudocódigo simplificado do CMA-ES. Fonte: Adaptado de Hansen (2016)

O passo 4 da figura 21 é a etapa principal do algoritmo que envolve a amostragem da população com λ indivíduos através de uma mutação envolvendo uma distribuição normal multivariada. Para isto, a cada geração, λ indivíduos independentes são gerados através da seguinte equação básica de amostragem (Hansen, 2016):

$$\mathbf{x}_i^{(g+1)} \sim \mathbf{m}^{(g)} + \sigma^{(g)} N_p(\mathbf{0}, \mathbf{C}^{(g)}), i = 1, 2, \dots, \lambda \quad (67)$$

onde $\mathbf{x}_i^{(g+1)}$ é o indivíduo (filho) i na geração $g + 1$, $\mathbf{m}^{(g)}$ é o centroide da distribuição na geração g , $\sigma^{(g)}$ é o tamanho de passo global do algoritmo na geração g , \mathbf{C} é a matriz de covariâncias, $\lambda \geq 2$ é o tamanho da população do algoritmo (ou número de filhos) e $N_p(\mathbf{0}, \mathbf{C}^{(g)})$ é uma distribuição normal de média 0 e matriz de covariância \mathbf{C} de tal forma que $\mathbf{m}^{(g)} + \sigma^{(g)} N_p(\mathbf{0}, \mathbf{C}^{(g)}) \sim N_p(\mathbf{m}, \sigma^2 \mathbf{C}^{(g)})$.

Depois de gerados, os λ indivíduos são avaliados (passo 5) e ordenados de acordo as suas respectivas aptidões. Feito isto, os μ ($\mu < \lambda$) melhores dentre os $x_1^{(g+1)}, x_2^{(g+1)}, \dots, x_\lambda^{(g+1)}$ são selecionados no passo 6 (seleção por truncamento).

Os μ indivíduos (pais) selecionados no passo 6 são utilizados no passo 7 para atualizar o centroide \mathbf{m} da distribuição através de uma operação de recombinação, que realiza uma média ponderada desses μ indivíduos (Hansen, 2016):

$$\mathbf{m}^{(g+1)} = \sum_{i=1}^{\mu} w_i \mathbf{x}_{i:\lambda}^{(g+1)} \quad (68)$$

$$\sum_{i=1}^{\mu} w_i = 1, w_1 \geq w_2 \geq \dots \geq w_{\mu} > 0 \quad (69)$$

onde $x_{i:\lambda}^{(g+1)}$ representa o i -ésimo melhor indivíduo da população e $w_{i=1\dots\mu} \in \mathbb{R}^+$ representa o peso de cada indivíduo na recombinação, com os indivíduos de melhor ranqueamento contribuindo com um peso maior na operação.

A adaptação da matriz de covariâncias \mathbf{C} e do passo global σ dependem dos caminhos de evolução \mathbf{p}_{σ} e \mathbf{p}_c , que são vetores que resumem o caminho percorrido sequencialmente pelo algoritmo durante a evolução através dos centroides mais recentes, sendo obtidos através de decaimentos exponenciais (Hansen, 2016):

$$\mathbf{p}_c^{(g+1)} = (1 - c_c)\mathbf{p}_c^{(g)} + \sqrt{c_c(2 - c_c)\mu_{eff}} \frac{\mathbf{m}^{(g+1)} - \mathbf{m}^{(g)}}{\sigma^{(g)}} \quad (70)$$

$$\mathbf{p}_{\sigma}^{(g+1)} = (1 - c_{\sigma})\mathbf{p}_{\sigma}^{(g)} + \sqrt{c_{\sigma}(2 - c_{\sigma})\mu_{eff}} \mathbf{C}^{(g)-\frac{1}{2}} \frac{\mathbf{m}^{(g+1)} - \mathbf{m}^{(g)}}{\sigma^{(g)}} \quad (71)$$

onde c_c , c_{σ} e μ_{eff} são constantes reais.

Desta forma, no passo 8, considerando o valor de \mathbf{p}_c e de forma similar ao cálculo do centroide em (68), a adaptação da matriz de covariâncias \mathbf{C} envolve um cálculo ponderado, onde é dado peso maior aos indivíduos com melhor aptidão (Hansen, 2016):

$$\begin{aligned} \mathbf{C}^{(g+1)} = & (1 - c_1 - c_{\mu} \sum_{i=1}^{\lambda} w_i) \mathbf{C}^{(g)} + c_1 \mathbf{p}_c^{(g+1)} \mathbf{p}_c^{(g+1)\top} + \\ & c_{\mu} \sum_{i=1}^{\lambda} w_i \left(\frac{\mathbf{x}_{1:\lambda}^{(g+1)} - \mathbf{m}^{(g)}}{\sigma^{(g)}} \right) \left(\frac{\mathbf{x}_{1:\lambda}^{(g+1)} - \mathbf{m}^{(g)}}{\sigma^{(g)}} \right)^{\top} \end{aligned} \quad (72)$$

onde c_1 e c_{μ} são constantes reais.

No passo 9, para a adaptação do passo global σ , o caminho de evolução \mathbf{p}_{σ} obtido em (71) é comparado ao seu valor esperado, que é expresso por $E(\|N(0, I)\|)$. Quando a norma de \mathbf{p}_{σ} é menor do que o seu valor esperado, significa que indivíduos mais próximos do centroide foram selecionados (como pais) e σ deve diminuir para concentrar as

mutações. Por outro lado, se σ é maior do que o valor esperado, então σ deve aumentar para expandir as mutações (Santos & Tinós, 2011):

$$\sigma^{(g+1)} = \sigma^{(g)} \exp\left(\frac{c_\sigma}{d_\sigma} \left(\frac{\|p_\sigma^{(g+1)}\|}{E(\|N(0,I)\|)} - 1\right)\right) \quad (73)$$

onde c_σ e d_σ são constantes reais e $E(\cdot)$ denota a função valor esperado.

Os passos 1 e 2 do algoritmo, que foram omitidos até então, referem-se à inicialização do algoritmo. No passo 1, deve-se fornecer ao algoritmo valores iniciais para o centroide ($\mathbf{m}^{(0)}$) e para o passo global do algoritmo ($\sigma^{(0)}$), que dependem do domínio do problema a ser resolvido. O tamanho da população pode ser livremente escolhido, uma vez que as constantes c_1 e c_μ previnem a degeneração do algoritmo até para pequenas populações (ex. $\lambda = 9$), que normalmente contribuem para a convergência em um tempo computacional menor (Hansen, 2016). Caso não se forneça um valor para λ , o algoritmo assume um valor padrão para o mesmo (tabela 2). Estes 3 parâmetros ($\mathbf{m}^{(0)}$, $\sigma^{(0)}$ e λ (opcional)), juntamente com um critério de parada, como o número máximo de gerações, são os normalmente fornecidos pelo usuário para o algoritmo. No passo 2, os demais parâmetros são inicializados com valores padrão, mostrados na tabela 2, e não necessitam da intervenção do usuário. A tabela 2 ainda mostra os valores padrão e iniciais de algumas constantes e parâmetros do CMA-ES, a relação completa pode ser encontrada no tutorial de Hansen (2016).

Tabela 2 – Alguns valores de constantes e parâmetros do CMA-ES

$\mathbf{p}_c^{(0)} = \mathbf{0}; \mathbf{p}_\sigma^{(0)} = \mathbf{0}; \mathbf{C}^{(0)} = \mathbf{I}$
$\lambda = 4 + \lfloor 3 \ln(p) \rfloor; \mu = \lfloor \lambda/2 \rfloor$
$\mu_{eff} = \frac{1}{\sum_{i=1}^{\mu} w_i^2}; c_\sigma = \frac{\mu_{eff}+2}{p+\mu_{eff}+5}$
$d_\sigma = 1 + 2 \max\left(0, \sqrt{\frac{\mu_{eff}-1}{p+1}} - 1\right) + c_\sigma$
$c_c = \frac{4 + \mu_{eff}/p}{p + 4 + 2\mu_{eff}/p}$
$c_1 = \frac{2}{(p+1,3)^2 + \mu_{eff}}$
$c_\mu = \min\left(1 - c_1, 2 \frac{\mu_{eff}-2 + 1/\mu_{eff}}{(p+2)^2 + \mu_{eff}}\right)$

Maiores detalhes sobre o funcionamento do CMA-ES podem ser encontrados nos trabalhos de Hansen & Ostermeier (1996, 2001), Hansen et al. (2010), Hansen (2016) e Hansen & Kern (2004).

5

Modelos neuroevolucionários baseados em ESNs otimizadas pelos algoritmos AEIQ-R e CMA-ES para Identificação de Sistemas

5.1.

Introdução

Conforme já abordado em capítulos anteriores, as redes neurais recorrentes são modelos de dinâmica interna, com capacidade de representar naturalmente sistemas dinâmicos, sem a necessidade de se realizarem atrasos temporais em seus sinais de entrada. As ESNs, RNARs da área de Computação de Reservatório, herdam tais características, porém com um processo de treinamento simplificado que envolve uma simples regressão linear. O uso das ESNs para identificação de sistemas traz vantagens como uma representação natural para sistemas dinâmicos e a consequente redução considerável do espaço de variáveis de entrada, além de um custo computacional baixo na etapa de ajuste de parâmetros devido ao treinamento rápido da rede. Entretanto, o ajuste manual dos parâmetros globais que afetam o desempenho das ESNs pode ser difícil e/ou demorado, uma vez que depende de testes experimentais e da experiência e conhecimento sobre tais parâmetros. Além disso, a inicialização de um reservatório totalmente aleatório pode não refletir a estratégia ideal para se extrair o melhor desempenho destas redes.

Levando em consideração as questões levantadas, neste capítulo será apresentado um modelo neuroevolucionário, resultante da união de ESNs com algoritmos evolutivos. O modelo proposto será composto por duas etapas, com a etapa 1 focando na otimização dos parâmetros gerais da rede e a segunda etapa se concentrando na seleção do reservatório mais adequado para uma ESN com os hiperparâmetros da etapa 1 fixos. Este modelo permitirá a definição de forma automática de uma ESN, tanto em termos de seus parâmetros globais, como em termos de pesos de seu reservatório.

Além do ajuste de parâmetros gerais da rede, a otimização dos pesos do reservatório por meio de AEs é uma questão já abordada por outros autores, como visto

na seção 3.9, apresentando bons resultados, apesar do espaço de busca ser consideravelmente maior neste caso. Por este motivo, um dos métodos da etapa de seleção do reservatório será a otimização dos pesos do mesmo por um AE. Como métodos alternativos para a seleção do reservatório, a segunda etapa disporá do SRG de Niño (2016), apresentado na seção 3.7.4, e de uma escolha simples baseada em reservatórios aleatórios.

O modelo utilizará como opções de algoritmos evolutivos, o AEIQ-R e o CMA-ES. O primeiro, pela capacidade de fornecer boas soluções com poucas avaliações, o que permitirá reduzir custo computacional em ambas as etapas do modelo proposto (na etapa 1, devido à avaliação de várias redes para um mesmo indivíduo e na segunda, devido ao maior espaço de busca). O segundo, além de ser capaz de operar bem com poucos indivíduos na população, é um algoritmo auto adaptativo, com pouquíssimos parâmetros a serem configurados pelo usuário, de uso consagrado na literatura e considerado estado da arte em termos de otimização numérica, inclusive com aplicações anteriores envolvendo a otimização de ESNs (Jiang et al., 2008; Yuenyong, 2015; Roeschies & Igel, 2010). Como aditivo, a utilização do AEIQ-R para a otimização de ESNs é inédita.

O modelo será detalhado nas subseções seguintes.

5.2. Modelo Geral

A figura 22 mostra a estrutura geral do modelo proposto:

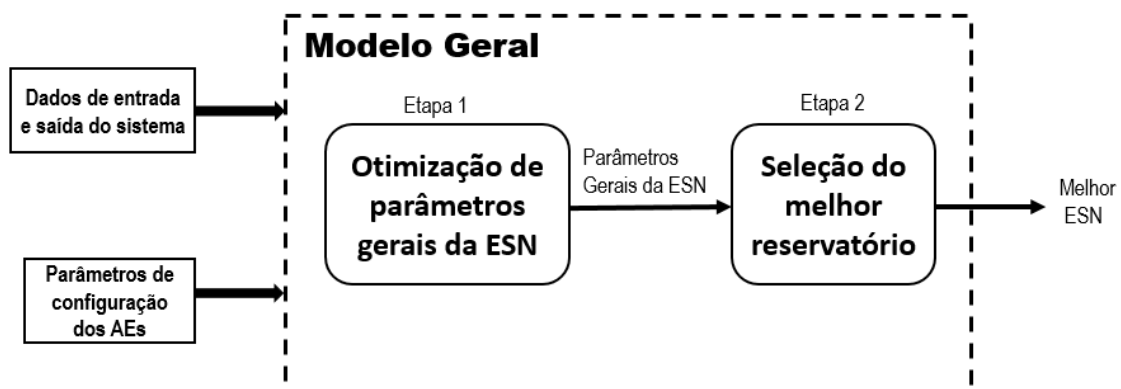


Figura 22 – Modelo geral proposto

Como mostrado pela figura 22, o modelo geral recebe como entradas do usuário a base de dados (dividida em treinamento, validação e teste), com as entradas e saídas do sistema dinâmico a ser identificado, e os parâmetros de configuração dos AEs presentes no método (AEIQ-R e CMA-ES) para cada etapa. A partir destas entradas, o modelo automaticamente define e retorna como saída uma ESN que simula o funcionamento do sistema alvo a partir das informações presentes nos dados, o que é feito determinando-se inicialmente seus parâmetros globais através do módulo “Otimização de parâmetros gerais da ESN” e selecionando-se o reservatório mais adequado para estes hiperparâmetros através do módulo “Seleção do melhor reservatório”. A ideia de dividir o processo de otimização das ESNs em duas fases distintas, uma focando nos parâmetros globais da rede e outra na seleção do reservatório, também é explorada nos trabalhos de Ishu et al. (2004), Martins (2016) e Niño (2016). As etapas do método proposto serão detalhadas a seguir.

5.2.1.

Etapa 1 - Otimização de parâmetros gerais da ESN

A ideia geral utilizada nesta etapa é mostrada na figura 23:

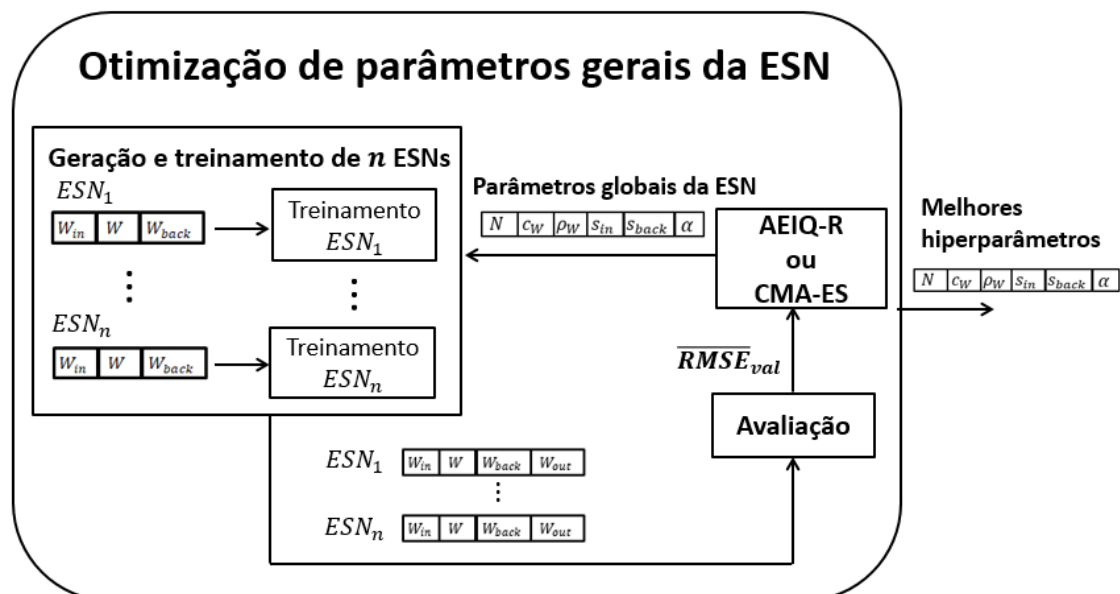


Figura 23 – Etapa de otimização de parâmetros gerais

Como mostrado pela figura 23, esta etapa tem como objetivo encontrar a melhor configuração de parâmetros gerais para uma ESN por meio de uma busca evolutiva utilizando o AEIQ-R ou o CMA-ES. Os parâmetros globais da ESN que são otimizados nesta etapa são:

- 1 - O número de neurônios no reservatório (N);
- 2 - A densidade de conexões no reservatório (c_W);
- 3 - O raio espectral da matriz \mathbf{W} (ρ_W);
- 4 - O fator de escala de \mathbf{W}^{in} (s_{in});
- 5 - O fator de escala de \mathbf{W}^{back} (s_{back}) e;
- 6 - A *leakage rate* (α).

Desta forma, o cromossomo utilizado nesta etapa, que representa um indivíduo para os AEs empregados, é o mostrado na figura 24:

N	c_W	ρ_W	s_{in}	s_{back}	α
1	2	3	4	5	6

Figura 24 – Cromossomo utilizado na etapa “Otimização de parâmetros gerais da ESN”

Como também mostrado pela figura 23, para cada conjunto de hiperparâmetros que são retornados por um cromossomo do AEIQ-R ou do CMA-ES, são inicializadas e treinadas n ESNs com estes hiperparâmetros. Após isto, cada uma dessas n ESNs realiza a simulação livre do conjunto de dados de validação do sistema e a avaliação de um indivíduo (f_{aval_1}) é calculada através da média dos erros de simulação dessas n redes. A métrica de erro escolhida foi a raiz quadrada do erro quadrático médio (RMSE – *Root Mean Squared Error*), então f_{aval_1} é expressa por:

$$f_{aval_1} = \overline{RMSE}_{val} = \frac{\sum_{j=1}^n RMSE_{val}(j)}{n} \quad (74)$$

onde \overline{RMSE}_{val} representa o RMSE médio no conjunto de validação de n redes, $RMSE_{val}(j)$ representa o RMSE no conjunto de validação de uma rede j , que é dado por:

$$RMSE_{val}(j) = \frac{1}{N_y} \sum_{i=1}^{N_y} \sqrt{\sum_{k=1}^{T_v} \frac{(\hat{y}_i(k) - y_i(k))^2}{T_v}} \quad (75)$$

onde N_y é o número de variáveis de saída, $\hat{y}_i(k)$ e $y_i(k)$ representam, respectivamente, o valor simulado pela rede j e o valor real da saída i do sistema no passo de tempo k e T_v é o tamanho da base de dados de validação.

A ideia de se utilizar o desempenho médio de n redes para compor a avaliação de um indivíduo nesta etapa tem como objetivo reduzir a influência de reservatórios aleatórios que podem introduzir ruídos em f_{aval_1} e levar a conclusões imprecisas sobre os parâmetros globais da ESN durante a evolução, devido à característica estocástica da rede e uma vez que os pesos da mesma não estão codificados no cromossomo. Tal estratégia de avaliação para um indivíduo foi também utilizada por Martins (2016) durante a otimização de hiperparâmetros da ESN por um algoritmo genético, entretanto, utilizando o desempenho médio no conjunto de treinamento para uma determinada configuração de parâmetros globais.

Cabe observar que no caso do CMA-ES, como o vetor de variáveis a serem otimizadas é composto por 6 hiperparâmetros da ESN, o vetor de médias \mathbf{m} e a matriz de covariâncias \mathbf{C} , parâmetros internos e também ajustados pela estratégia, serão dados por:

$$\mathbf{m} = [m_1 \ m_2 \ m_3 \ m_4 \ m_5 \ m_6] \quad (76)$$

$$\mathbf{C} = \begin{bmatrix} \sigma_{11} & \cdots & \sigma_{16} \\ \vdots & \ddots & \vdots \\ \sigma_{61} & \cdots & \sigma_{66} \end{bmatrix} \quad (77)$$

onde na equação 77, σ_{12} , por exemplo, é a covariância entre as variáveis N e c_w , seguindo a numeração adotada para cada variável no cromossomo (24).

Ao final do processo evolutivo, a etapa 1 retorna como saída os melhores parâmetros globais da ESN ($N, c_w, \rho_w, s_{in}, s_{back}, \alpha$), obtidos através da minimização de f_{aval_1} , os quais serão utilizados na etapa 2.

5.2.2. Etapa 2 – Seleção do melhor reservatório

Esta etapa irá determinar o melhor reservatório para a configuração de parâmetros globais definidos pela etapa 1 (figura 25).



Figura 25 – Etapa de seleção do melhor reservatório

Como mostrado pela figura 25, o usuário poderá optar dentre 3 diferentes métodos para a seleção do melhor reservatório: *random*, SRG e otimização do reservatório. Cada um destes módulos será descrito em seguida.

5.2.2.1. Random

Esta é a abordagem mais simples utilizada na etapa 2, cuja escolha da melhor rede se baseia na seleção de uma ESN cujo reservatório aleatório, dentre um conjunto de n ESNs com os mesmos parâmetros globais, resulta no menor erro de validação. Para isto, o método *random* se resume nos seguintes passos:

1 – Com os parâmetros globais da etapa 1 ($N, c_W, \rho_W, s_{in}, s_{back}, \alpha$) fixos, são inicializadas aleatoriamente n ESNs;

2 – As n ESNs do passo 1 são então treinadas e realizam a simulação livre do conjunto de validação, calculando-se para cada rede um valor de $RMSE_{val}$;

3 – A ESN cujo reservatório aleatório resultou no menor valor de $RMSE_{val}$ é escolhida como a melhor rede pelo método *random*, retornando-se como saída não só os pesos de sua correspondente matriz \mathbf{W} , mas também as suas respectivas matrizes \mathbf{W}^{in} , \mathbf{W}^{back} e \mathbf{W}^{out} (uma vez que a rede já foi treinada).

4 – Com a melhor rede do passo 3, pode-se realizar a simulação do conjunto de teste.

Processo similar de escolha do reservatório é realizado em (Martins, 2016), porém com uma escolha da melhor rede baseada no menor erro de teste.

5.2.2.2. SRG

Este módulo utiliza o coeficiente SRG de Niño (2016), apresentado na seção 3.7.4, para escolher o reservatório mais adequado para a ESN, o que pode ser resumido em 4 passos:

1 – Assim como no método *random*, n ESNs são inicializadas aleatoriamente com os mesmos parâmetros globais ($N, c_W, \rho_W, s_{in}, s_{back}, \alpha$) determinados pela etapa 1;

2 – Para cada uma das ESNs criadas no passo 1 é calculado o correspondente coeficiente SRG, o que não requer o treinamento destas redes;

3 – A ESN com o menor coeficiente SRG, que em teoria apresenta o reservatório com o comportamento mais adequado (próximo ao limiar do caos), é escolhida como a melhor pelo método, retornando-se como saída sua correspondente matriz \mathbf{W} , acompanhada das matrizes \mathbf{W}^{in} e \mathbf{W}^{back} .

4 – A melhor rede escolhida no passo 3 pelo método SRG pode ser então treinada e simular os conjuntos de validação e de teste.

Cabe ressaltar que para o cálculo do SRG, neste trabalho é feita uma modificação no método originalmente proposto por Niño (2016). Devido ao fato de que o vetor de realimentação de saída $\mathbf{y}(k-1)$, juntamente com o vetor $\mathbf{u}(k)$, funcionar como uma entrada do ponto de vista do reservatório, para o cálculo do coeficiente SRG, neste trabalho, é formado o pseudo padrão de entrada $\tilde{\mathbf{u}}(k) = [\mathbf{u}(k) \mathbf{y}(k-1)]$. A partir disto,

como no método original, cada pseudo padrão e seu vizinho mais próximo são apresentados ao reservatório e os seus correspondentes estados são gerados. Ainda, para o cálculo do SRG, neste trabalho, é considerado que cada pseudo padrão parte do mesmo estado inicial $\mathbf{x}(T_0)$, isto é, $\mathbf{x}(k - 1) = \mathbf{x}(T_0)$ na equação de geração de estados para todos os pseudo padrões de entrada. Após isto, o coeficiente SRG é calculado como:

$$SRG = \sum_{j=1}^p ||d(\tilde{\mathbf{u}}_j) - d(\mathbf{x}_j)|| \quad (78)$$

onde $d(\tilde{\mathbf{u}}_j)$ e $d(\mathbf{x}_j)$ representam, respectivamente, a distância euclidiana de separação entre o pseudo padrão de entrada $\tilde{\mathbf{u}}_j$ e seu vizinho mais próximo $\tilde{\mathbf{u}}_j$ e a distância de separação de seus respectivos estados \mathbf{x}_j e \mathbf{x}_j .

Como uma medida mais justa para a comparação das abordagens *random* e SRG, nos estudos de caso descritos no capítulo 6, as mesmas n ESNs inicializadas para utilização no método *random* serão as mesmas utilizadas pelo método SRG. Em outras palavras, o passo 1 dos métodos *random* e SRG será o mesmo.

5.2.2.3. Otimização do Reservatório

Este módulo determina o melhor reservatório para a configuração de parâmetros globais definidos na etapa 1 utilizando um novo processo de otimização por meio de um AE (AEIQ-R ou CMA-ES). Porém, neste caso, o AE atua diretamente na otimização dos pesos de \mathbf{W} para uma ESN com os parâmetros globais determinados na etapa 1 fixos. Com o objetivo de preservar a densidade de conexões do reservatório c_W determinada na etapa 1 e reduzir o espaço de busca, somente os pesos de \mathbf{W} com valores inicialmente diferentes de zero são otimizados pelo AE. As matrizes \mathbf{W}^{in} e \mathbf{W}^{back} , uma vez inicializadas, permanecem fixas durante todo o processo e, antes da etapa de avaliação, a matriz \mathbf{W} é rescalada de forma a manter o raio espectral ρ_W da etapa 1. Uma estratégia similar de otimização dos pesos do reservatório é vista em Ishu et al. (2004), porém com uma ESN com apenas 5 neurônios no reservatório. No trabalho de Basterrech et al. (2014), também há a otimização de alguns pesos do reservatório, porém sem a reescala

da matriz \mathbf{W} por um valor de ρ_W e sem a otimização de parâmetros gerais da rede em uma etapa prévia.

A figura 26 mostra a ideia geral utilizada neste método:

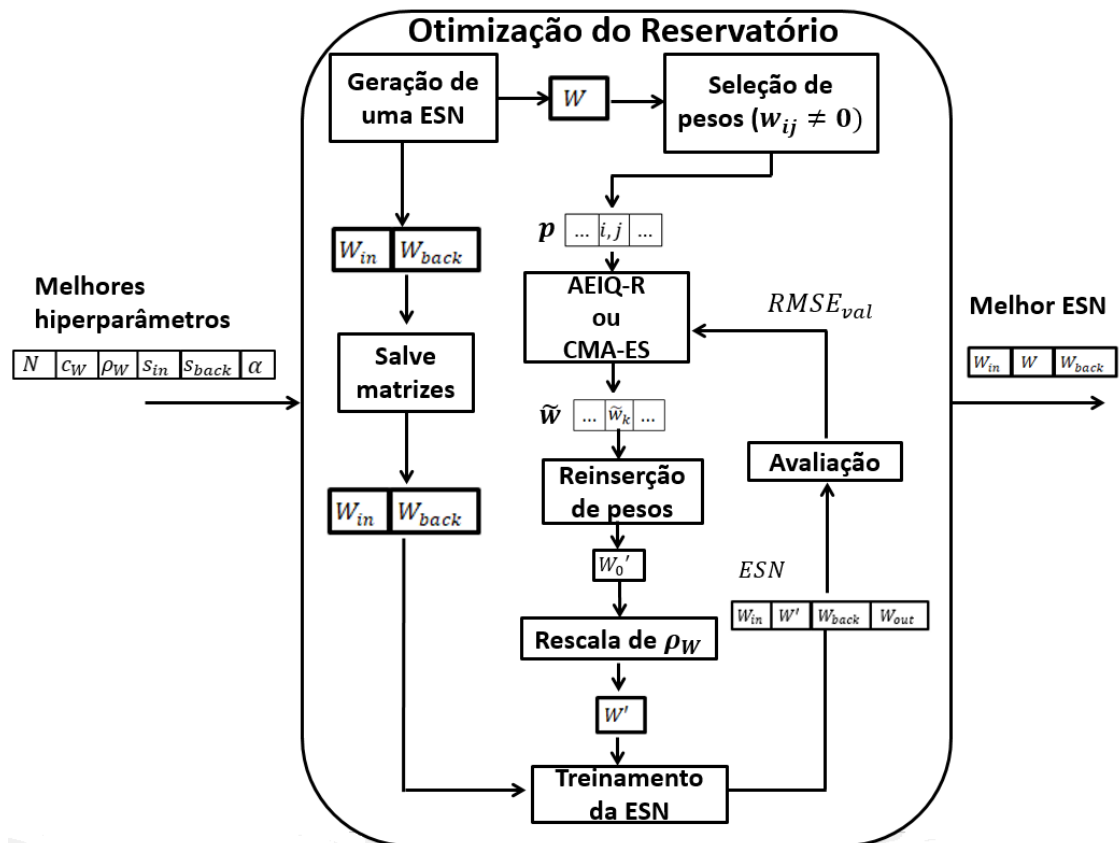


Figura 26 – Módulo de Otimização do Reservatório

Como mostrado pela figura 26, a partir dos hiperparâmetros ($N, c_W, \rho_W, s_{in}, s_{back}, \alpha$) definidos pela etapa 1, uma ESN é aleatoriamente inicializada, salvando-se as matrizes \mathbf{W}^{in} e \mathbf{W}^{back} para uso posterior e encaminhando-se a matriz \mathbf{W} para o bloco “Seleção de pesos”. Neste bloco, são determinadas as posições i, j de \mathbf{W} que correspondem a valores de pesos w_{ij} diferentes de zero, salvando-se tais posições em um vetor auxiliar \mathbf{p} (figura 27). A partir disto, é iniciado um processo evolutivo (utilizando o AEIQ-R ou o CMA-ES), cujo cromossomo $\tilde{\mathbf{w}}$ corresponde aos valores de pesos de \mathbf{W} nas posições determinadas por \mathbf{p} (figura 27).

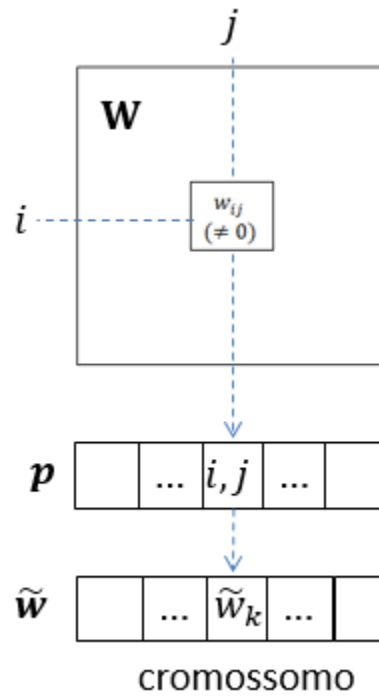


Figura 27- Determinação do cromossomo utilizado na etapa 2

Durante o processo evolutivo, antes da etapa de avaliação, cada vetor de pesos $\tilde{\mathbf{w}}$ é reinserido em \mathbf{W} nas posições dadas por \mathbf{p} , formando-se uma nova matriz \mathbf{W}_0' , que é reescalada para o raio espectral ρ_W determinado na etapa 1, obtendo-se \mathbf{W}' :

$$\mathbf{W}' = \frac{\mathbf{W}_0'}{\rho(\mathbf{W}_0')} \rho_W \quad (79)$$

Com \mathbf{W}' e as matrizes \mathbf{W}^{in} e \mathbf{W}^{back} , salvas anteriormente, a ESN é treinada e a avaliação de um indivíduo (f_{aval_2}) é dada pelo desempenho desta rede treinada durante a simulação livre do conjunto de validação:

$$f_{\text{aval}_2} = RMSE_{\text{val}} = \frac{1}{N_y} \sum_{i=1}^{N_y} \sqrt{\sum_{k=1}^{T_v} \frac{(\hat{y}_i(k) - y_i(k))^2}{T_v}} \quad (80)$$

onde N_y é o número de variáveis de saída, $\hat{y}_i(k)$ e $y_i(k)$ representam, respectivamente, o valor simulado pela rede e o valor real da saída i do sistema no passo de tempo k e T_v é o tamanho da base de dados de validação.

Após o processo evolutivo, a matriz \mathbf{W} final (com os valores de $\tilde{\mathbf{w}}$ otimizados) é retornada pelo módulo “Otimização do reservatório”, juntamente com as matrizes \mathbf{W}^{in} e \mathbf{W}^{back} , que foram salvas inicialmente.

Com esta metodologia, ainda que somente pesos inicialmente não nulos de \mathbf{W} sejam otimizados e \mathbf{W}^{in} e \mathbf{W}^{back} permaneçam fixas, o número de parâmetros a serem otimizados pelo AE pode ser muito grande, resultando em um espaço de busca muito grande, com no máximo N^2 variáveis (assumindo um reservatório totalmente conectado). Por outro lado, como os parâmetros globais da rede e suas matrizes \mathbf{W}^{in} e \mathbf{W}^{back} permanecem fixos e os pesos não nulos de \mathbf{W} estão codificados no cromossomo, a avaliação de um indivíduo é direta, reduzindo-se o tempo necessário para uma avaliação, uma vez que não é necessária a inicialização e o treinamento de n redes, como na etapa 1, onde o cromossomo codifica somente os hiperparâmetros da rede. Indiscutivelmente, os outros dois módulos (*random* e SRG), por serem mais simples, fornecerão soluções em um menor tempo computacional. Entretanto, como já abordado na seção 3.9, diferentes abordagens anteriores (Ishu et al., 2004; Ferreira & Ludermir, 2011; Chouikhi et al., 2015, 2017; Basterrech et al., 2014) já obtiveram bons resultados otimizando-se parcial ou totalmente os pesos de \mathbf{W} e/ou das matrizes \mathbf{W}^{in} e \mathbf{W}^{back} , o que influenciou a decisão de incluir neste trabalho uma estratégia voltada à otimização dos pesos de \mathbf{W} através do módulo “Otimização do reservatório” como um dos módulos na etapa 2 do modelo geral proposto.

Antes de se prosseguir para a próxima subseção, a sequência de passos abaixo resume a ideia empregada neste módulo “Otimização do reservatório”:

- 1 – Gere aleatoriamente uma ESN ($\mathbf{W}^{\text{in}}, \mathbf{W}, \mathbf{W}^{\text{back}}$) utilizando os hiperparâmetros anteriormente determinados na etapa 1 ($N, c_W, \rho_W, s_{in}, s_{back}, \alpha$);
- 2 – Salve as matrizes \mathbf{W}^{in} e \mathbf{W}^{back} do passo 1;
- 3 – Determine as posições (i, j) nas quais os pesos de \mathbf{W} são diferentes de zero e as salve em vetor auxiliar \mathbf{p} ;
- 4 – Inicialize uma busca evolutiva com o AEIQ-R ou com o CMA-ES, onde o cromossomo $\tilde{\mathbf{w}}$ corresponde aos pesos do reservatório nas posições determinadas por \mathbf{p} ;
- 5 – Durante a avaliação de um indivíduo ($\tilde{\mathbf{w}}$) utilize o seguinte procedimento:

5.1 – Reinsira em \mathbf{W} o vetor de pesos $\tilde{\mathbf{w}}$ nas posições dadas por \mathbf{p} , criando uma matriz \mathbf{W}_0' ;

5.2 – Reescale \mathbf{W}_0' para o raio espectral ρ_W determinada na fase 1, obtendo-se \mathbf{W}' ;

5.3 – Forme uma nova ESN com \mathbf{W}^{in} e \mathbf{W}^{back} do passo 2 e \mathbf{W}' do passo 5.2 e a treine;

5.4 - A avaliação de um indivíduo é o desempenho da ESN do passo 5.3 na simulação livre do conjunto de validação;

6 – Ao final do processo evolutivo, a matriz \mathbf{W} final é formada com os melhores valores de $\tilde{\mathbf{w}}$;

7 – O método retorna como saída a matriz \mathbf{W} do passo 6 e as matrizes \mathbf{W}^{in} e \mathbf{W}^{back} do passo 2, com as quais pode-se formar uma ESN final, que pode ser treinada e simular os conjuntos de validação e de teste.

Cabe observar que nesta fase, no caso do CMA-ES, o vetor de médias será um vetor \mathbf{m} de tamanho $1 \times n_p$ e a matriz de covariâncias \mathbf{C} terá dimensões $n_p \times n_p$, onde n_p será o número de pesos não nulos a serem otimizados que compõem o vetor $\tilde{\mathbf{w}}$.

5.3.

Resumo

Uma vez que cada uma das etapas (e sub etapas) tenham sido apresentadas, a figura 28 resume na forma de pseudocódigo o modelo geral proposto:

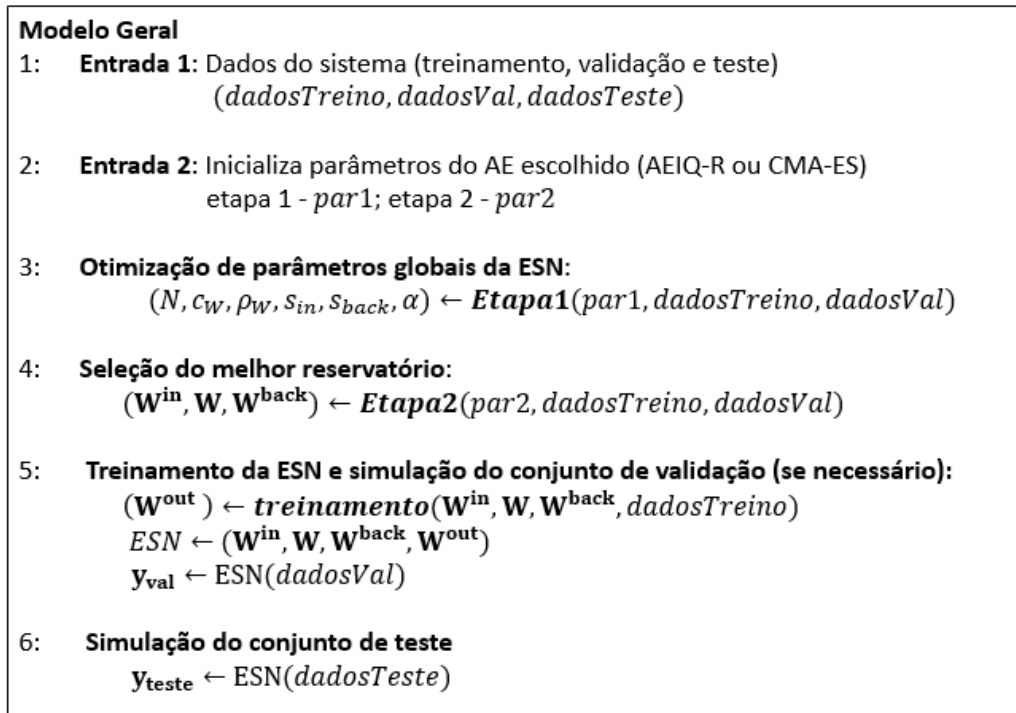


Figura 28 – Pseudocódigo do modelo geral proposto

Como mostrado pela figura 28, nos passos 1 e 2 da figura 28, o usuário é responsável por fornecer ao modelo, a base de dados do sistema dividida em treinamento, validação e teste, além de escolher e definir os parâmetros de configuração do algoritmo evolutivo escolhido (AEIQ-R ou CMA-ES) para as etapas 1 e 2.

No passo 3, os parâmetros globais da rede são otimizados utilizando a etapa 1 do método geral. A etapa 1 pode ser desempenhada pelo AEIQ-R ou pelo CMA-ES, utilizando como cromossomo os parâmetros $(N, c_W, \rho_W, S_{in}, S_{back}, \alpha)$ e (74) como função de aptidão, que retorna o desempenho médio no conjunto de validação de n ESNs para cada configuração de hiperparâmetros dada por um indivíduo.

No passo 4 da figura 28, é selecionado o reservatório mais adequado para os hiperparâmetros do passo 3, utilizando a etapa 2 do método, que retorna também as matrizes W^{in} e W^{back} . Na etapa 2, pode-se optar por utilizar métodos simples para a escolha do reservatório como o *random* e o SRG ou iniciar uma nova busca evolutiva utilizando o AEIQ-R ou o CMA-ES, através do método “Otimização do reservatório”. Nos módulos *random* e SRG, a melhor rede é obtida escolhendo-se dentre um conjunto de n ESNs com os mesmos hiperparâmetros da etapa 1, a rede cujo reservatório, inicializado aleatoriamente, resultar no menor $RMSE_{val}$ (caso do *random*) e menor

coeficiente SRG (caso do SRG), calculado conforme (78). No módulo “Otimização do reservatório” são otimizados pelo AEIQ-R ou pelo CMA-ES os pesos inicialmente não nulos de \mathbf{W} de uma ESN inicializada aleatoriamente com os hiperparâmetros definidos na etapa 1, mantendo-se \mathbf{W}^{in} e \mathbf{W}^{back} fixas, rescalando-se \mathbf{W} para o raio espectral ρ_W da etapa 1 e utilizando-se (75) como função de aptidão.

Após a etapa 2, a rede final, se necessário, é treinada e simula conjunto de validação (passo 5 da figura 28) e pode finalmente realizar a simulação livre do conjunto de dados de teste (passo 6 da figura 28).

6 Estudos de Caso

6.1. Introdução

Com o objetivo de verificar o desempenho do método proposto, foram selecionados 9 problemas *benchmark* da área de Identificação de Sistemas, cujas bases de dados estão publicamente disponíveis no repositório DaISy (*Database for the Identification of Systems*) (De Moor, 2018) e no artigo “Three free data sets for development and bechmarking in nonlinear system identification” (Wigren, 2013). As seções seguintes abordam a descrição de cada um dos problemas e de suas respectivas bases de dados, bem como a apresentação e a discussão dos resultados obtidos pelo método proposto em cada estudo de caso. Também é apresentada a configuração utilizada nos experimentos, bem como é realizada uma comparação do desempenho dos diferentes submodelos resultantes do modelo geral proposto. Testes estatísticos são aplicados nos resultados gerais obtidos para comparação das diferentes abordagens propostas e, quando possível, para comparação com os resultados obtidos por outros métodos de outros autores.

6.2. Descrição dos problemas

6.2.1. Estudo de caso 1 - *Trocador de calor a vapor líquido-saturado*

Este estudo de caso refere-se à identificação de um trocador de calor, no qual água, que flui por um tubo de cobre, é aquecida através da troca indireta de calor com um fluxo de vapor saturado e pressurizado (Bittanti & Piroddi, 1997).

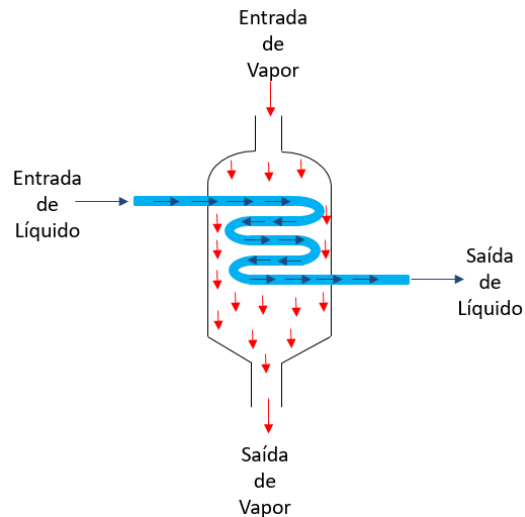


Figura 29 – Estrutura de um trocador de calor. Fonte: Adaptado de Bittanti & Piroddi (1997)

Tal processo é caracterizado por possuir comportamento não linear, de fase não-mínima e variante no tempo (Bittanti & Piroddi, 1997; Al-Dhaifallah et al., 2015; Soleimani et al., 2010). A base de dados utilizada neste trabalho pode ser acessada *online* (De Moor, 2018) e contém 4000 amostras (simuladas) da vazão do líquido na entrada do trocador (u - entrada) e a temperatura do líquido na saída do trocador (y - saída) que foram obtidas com um período de amostragem de 1,0 s e mantendo-se a temperatura do líquido e a do vapor aplicados na entrada do trocador constantes (Bittanti & Piroddi, 1997) (figura 30).

- Entrada:
 - u – vazão de entrada do líquido
- Saída:
 - y – temperatura de saída do líquido

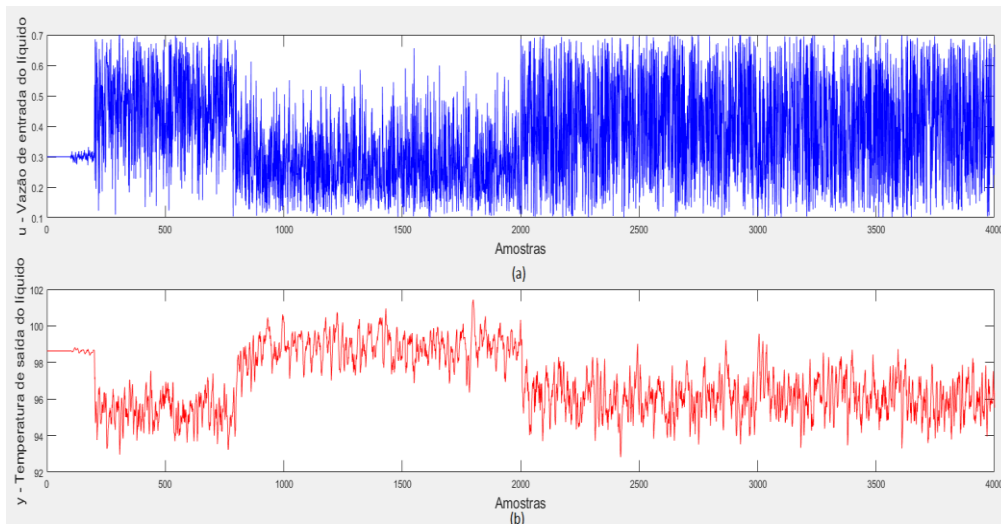


Figura 30 – Variáveis do estudo de caso 1 (a) Entrada u – vazão de entrada do líquido no trocador. (b) Saída y – temperatura de saída do líquido no trocador.

6.2.2. Estudo de caso 2 - Secador de cabelo

O problema abordado neste estudo de caso refere-se à modelagem de um dispositivo de laboratório (*Feedback's Process Trainer PT326*), que possui funcionamento análogo ao de um secador de cabelo (figura 31) (Ljung, 1987).

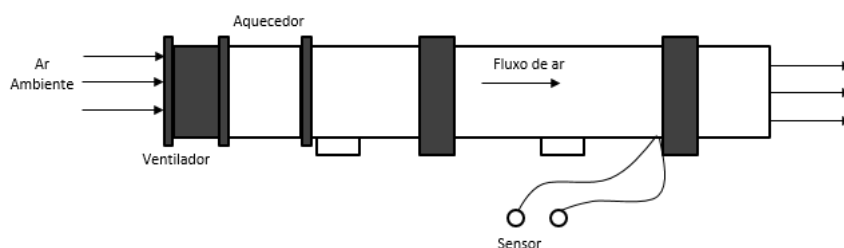


Figura 31 – Vista frontal de parte do *Feedback's process trainer* PT326. Fonte: Adaptado de Leva & Papadoulos (2013).

O processo envolvido no dispositivo da figura 31 envolve a sucção de ar ambiente através de um ventilador centrífugo para dentro de um tubo, onde o ar é aquecido em seu interior por meio de um aquecedor e retorna para a atmosfera. Um termopar mede a temperatura de saída do ar.

A base de dados utilizada neste estudo de caso, que pode ser obtida no repositório DaISy (De Moor, 2018), possui 1000 amostras, que foram obtidas através de medidas reais (Ljung, 1987) da tensão elétrica aplicada à resistência elétrica do aquecedor

(variável de entrada – u) e da temperatura do ar que sai para a atmosfera (variável de saída – y) com um período de amostragem de 0,08 s (figura 32):

- Entrada:

- u – tensão elétrica aplicada ao aquecedor

- Saída:

- y – temperatura do ar de saída

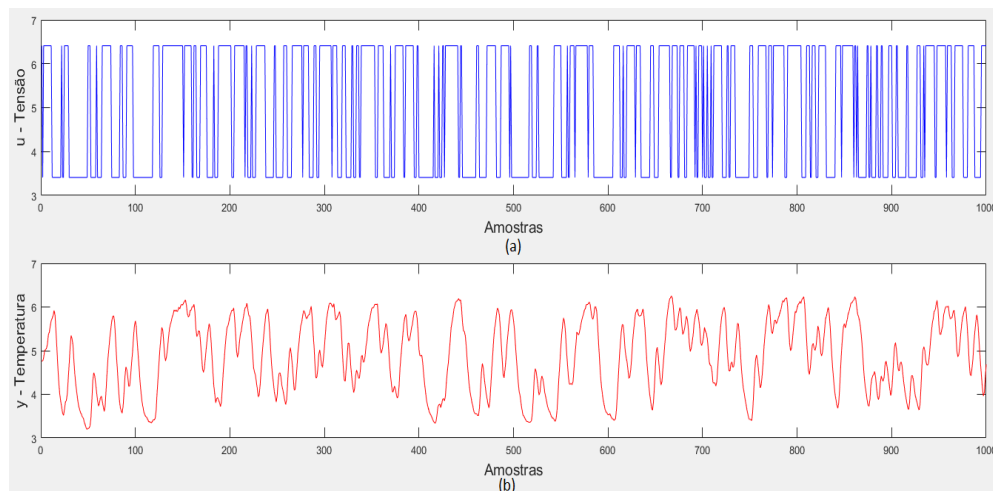


Figura 32 - Variáveis do estudo de caso 2 (a) Entrada u – tensão na resistência de aquecimento. (b) Saída y – temperatura medida pelo termopar.

6.2.3.

Estudo de caso 3 – Densidade de fluxo de calor através de uma parede de duas camadas

Este estudo de caso envolve a identificação do comportamento dinâmico do fluxo de calor através de uma parede de duas camadas (tijolos e isolamento) (Bloem, 1994) (figura 33).

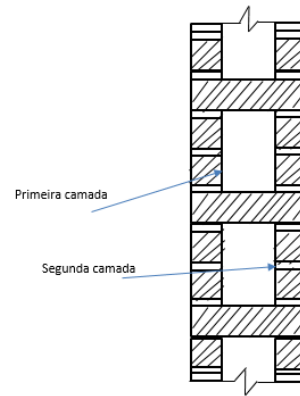


Figura 33 – Parede com duas camadas (tijolos e isolamento). Fonte: Adaptado de Castellanos (2008).

Este sistema térmico possui 2 entradas, as quais são a temperatura interna (u_1) e a temperatura externa da parede (u_2), e 1 saída, que é a densidade de fluxo de calor através da parede (y). A base de dados possui 1680 amostras das variáveis citadas (período de amostragem não informado) e pode ser baixada do repositório DaISy (De Moor, 2018) (figura 34):

- Entradas:
 - u_1 – temperatura interna da parede
 - u_2 – temperatura externa da parede
- Saída:
 - y – fluxo de calor através da parede

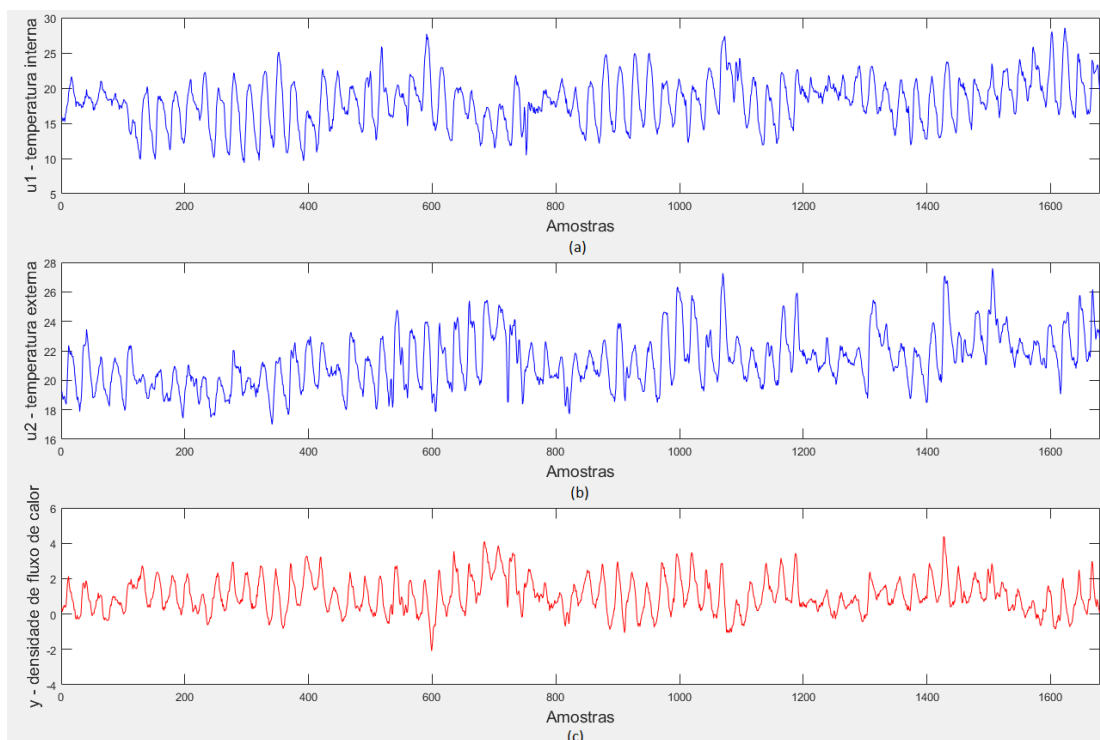


Figura 34 - Variáveis do estudo de caso 3 (a) Entrada 1: u_1 – temperatura interna da parede. (b) Entrada 2: u_2 – temperatura externa da parede (c) Saída y – densidade de fluxo de calor através da parede.

6.2.4. Estudo de caso 4 – Braço robótico flexível

Este estudo de caso envolve a identificação de um braço robótico flexível instalado em um motor elétrico (figura 35).

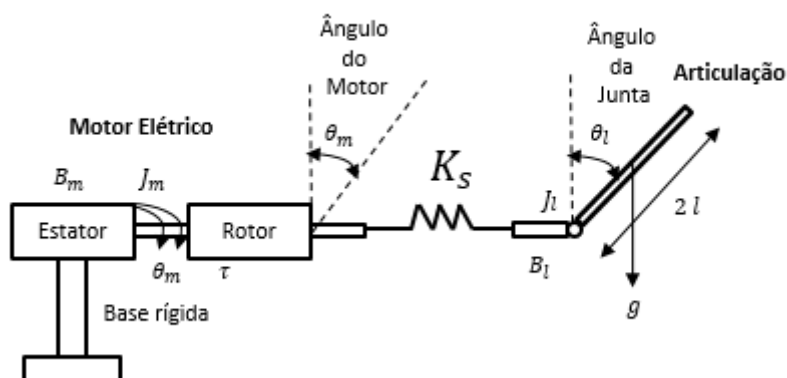


Figura 35 – Braço robótico flexível. Fonte: Adaptado de Ferreira & Serra (2012).

A base de dados utilizada, disponibilizada *online* pelo repositório DaISy (De Moor, 2018), contém 1024 amostras (com período de amostragem desconhecido) da aceleração do braço robótico (variável de saída - y) em resposta ao torque de reação na estrutura em relação ao solo (variável de entrada - u) (figura 36):

- Entrada:

- u – torque de reação na estrutura em relação ao solo

- Saída:

- y – aceleração do braço robótico

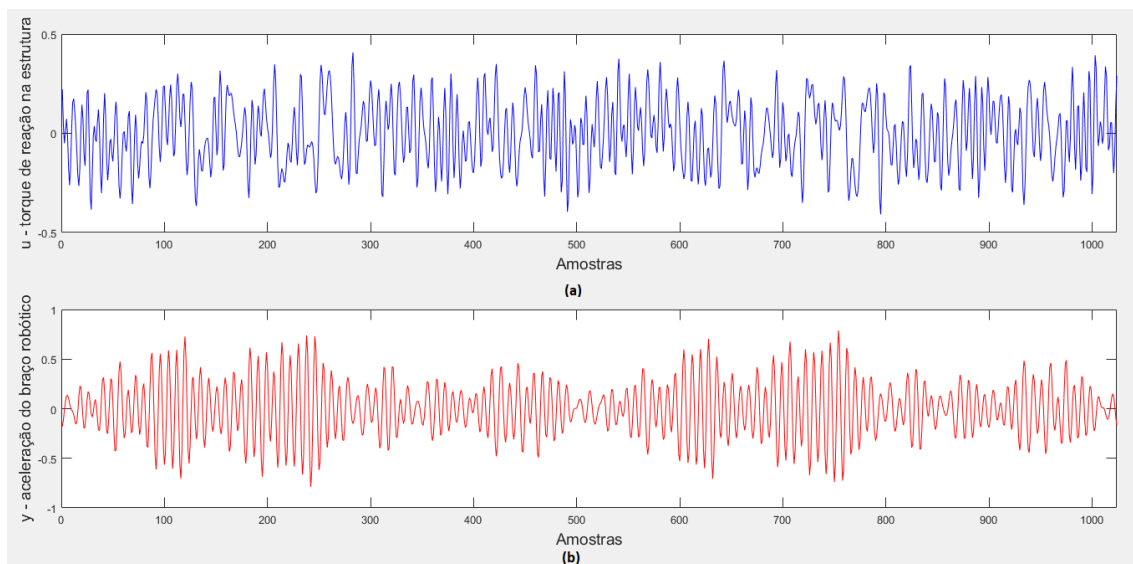


Figura 36 - Variáveis do estudo de caso 4 (a) Entrada: u – torque de reação na estrutura em relação ao solo. (b) Saída: y – aceleração do braço robótico.

6.2.5.

Estudo de caso 5 – *Reator tanque de agitação contínua*

Este estudo de caso envolve a modelagem de uma reação exotérmica em um reator tanque continuamente agitado (CSTR – *Reator tanque de agitação contínua*) e refrigerado por um único fluxo de refrigerante (figura 37).

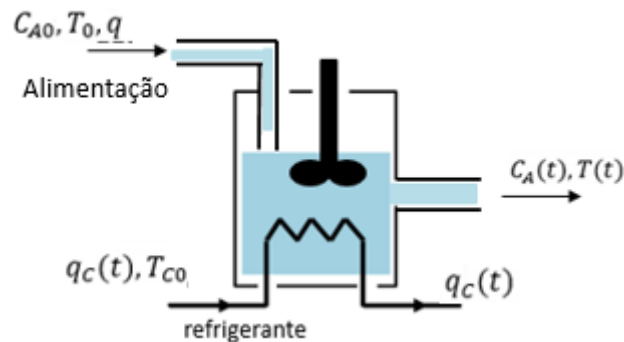


Figura 37 – Reator tanque continuamente agitado. Fonte: Adaptado de Kandroodi & Moshiri (2011)

Dentro do CSTR da figura 37, dois produtos químicos são misturados e através de uma reação química exotérmica (com liberação de calor) produzem um composto A de concentração $C_A(t)$ e com temperatura $T(t)$ (Lightbody & Irwin, 1997). Um fluxo $q_c(t)$ de um produto químico refrigerante permite variar a temperatura $T(t)$ e a concentração $C_A(t)$ da mistura final. Tal processo possui um comportamento dinâmico altamente não linear (Kandroodi & Moshiri, 2011) e pode ser expresso pelas seguintes equações diferenciais não lineares (Lightbody & Irwin, 1997; Pellegrinetti & Bentsman, 1996):

$$\dot{C}_A(t+d) = \frac{q}{v}(C_{A0} - C_A(t+d)) - k_0 C_A(t+d) e^{-\frac{E}{RT(t)}} \quad (81)$$

$$\begin{aligned} \dot{T}(t) = & \frac{q}{v}(T_0 - T(t)) + k_1 C_A(t+d) e^{-\frac{E}{RT(t)}} + \\ & + k_2 q_c(t) (1 - e^{-\frac{k_3}{q_c(t)}}) (T_{C0} - T(t)) \end{aligned} \quad (82)$$

onde C_{A0} é a concentração inicial da mistura, q é a vazão com que os produtos químicos são alimentados, T_0 é a temperatura inicial da mistura, T_{C0} é a temperatura do refrigerante, d é um atraso no tempo e k_0 , E , R , v , k_1 , k_2 e k_3 são constantes termodinâmicas e químicas.

A base de dados utilizada neste problema está disponível no repositório DaIisy (De Moor, 2018) e contém 7500 amostras (período de amostragem de 0,1 min) obtidas do modelo do CSTR, considerando como entrada para o sistema o fluxo de refrigerante $q_c(t)$ e como saídas a concentração ($C_A(t)$) e a temperatura $T(t)$ da mistura final (figura 38):

- Entrada:
 - u – fluxo de refrigerante $q_C(t)$
- Saídas:
 - y_1 – concentração da mistura final $C_A(t)$
 - y_2 – temperatura da mistura final $T(t)$

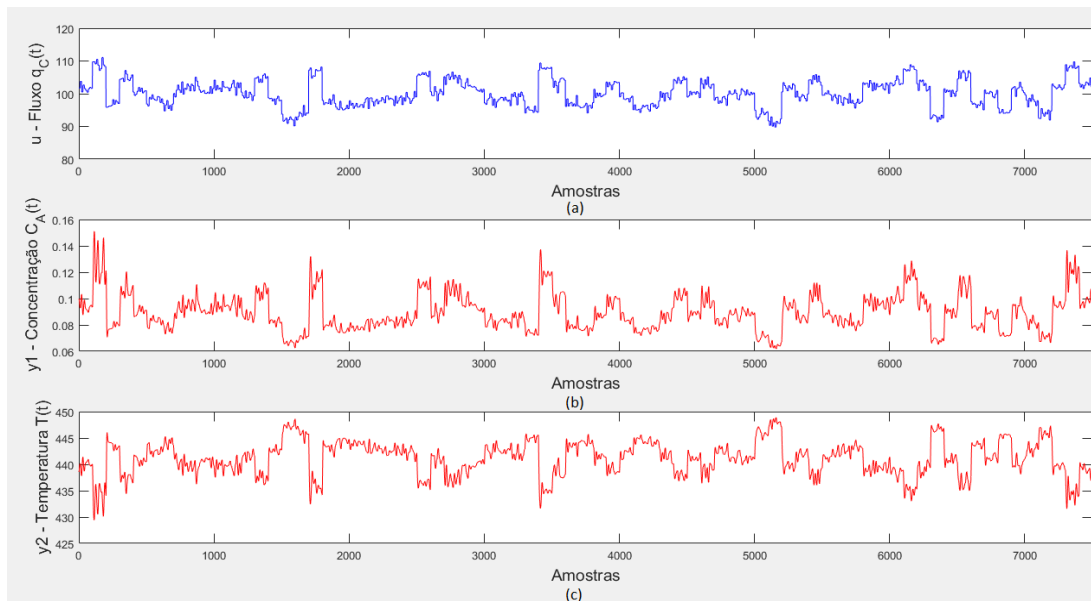


Figura 38 - Variáveis do estudo de caso 5 (a) Entrada: u – fluxo de refrigerante $q_C(t)$. (b) Saída 1: y_1 – concentração da mistura final $C_A(t)$. (c) Saída 2: y_2 – temperatura da mistura final $T(t)$

6.2.6. Estudo de caso 6 – *Tanques em cascata*

Este problema consiste na identificação de um sistema de controle de fluido composto por dois tanques ligados em cascata e alimentados por uma bomba (figura 39).

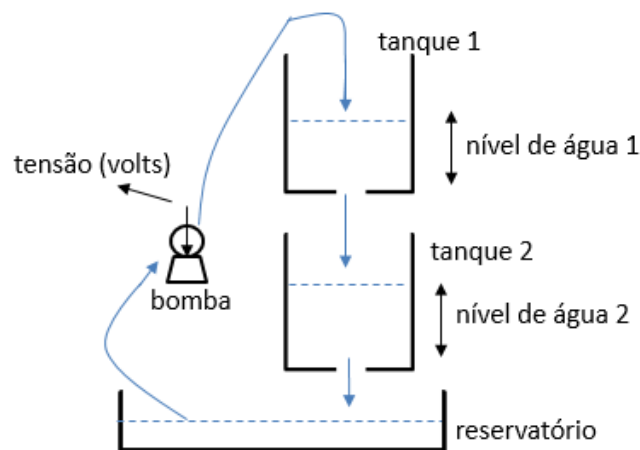


Figura 39 – Sistema de controle de fluido com dois tanques em cascata. Fonte: Adaptado de Wigren (2013) e Birpoutsoukis et al. (2018)

O sistema mostrado na figura 39 é altamente não linear (Wigren, 2013) e possui como entrada a tensão aplicada na bomba (u) e como saídas os níveis de água nos tanques superior (y_1) e inferior (y_2). A modelagem caixa branca para tal processo, após aplicação do princípio de Bernoulli e conservação de massa, no espaço de estados é dada por (Wigren, 2013):

$$\begin{pmatrix} \dot{x}_1(t) \\ \dot{x}_2(t) \end{pmatrix} = \begin{pmatrix} -k_1\sqrt{x_1(t)} \\ k_2\sqrt{x_1(t)} - k_3\sqrt{x_2(t)} \end{pmatrix} + \begin{pmatrix} k_4u(t) \\ 0 \end{pmatrix} + \begin{pmatrix} w_1(t) \\ w_2(t) \end{pmatrix} \quad (83)$$

$$\begin{pmatrix} y_1(t) \\ y_2(t) \end{pmatrix} = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} \begin{pmatrix} x_1(t) \\ x_2(t) \end{pmatrix} + \begin{pmatrix} e_1(t) \\ e_2(t) \end{pmatrix} \quad (84)$$

onde $x_1(t)$ e $x_2(t)$ são estados, $w_1(t)$ e $w_2(t)$ são distúrbios, $e_1(t)$ e $e_2(t)$ são erros de medida e k_1, k_2, k_3 e k_4 são parâmetros do sistema.

A base de dados utilizada neste estudo de caso contém 2500 amostras, com um período de amostragem de 5,0 s, e pode ser livremente acessada em (Wigren, 2013) (figura 40):

- Entrada:
 - u – tensão elétrica na bomba
- Saídas:

- y_1 – nível de água no tanque superior

- y_2 – nível de água no tanque inferior

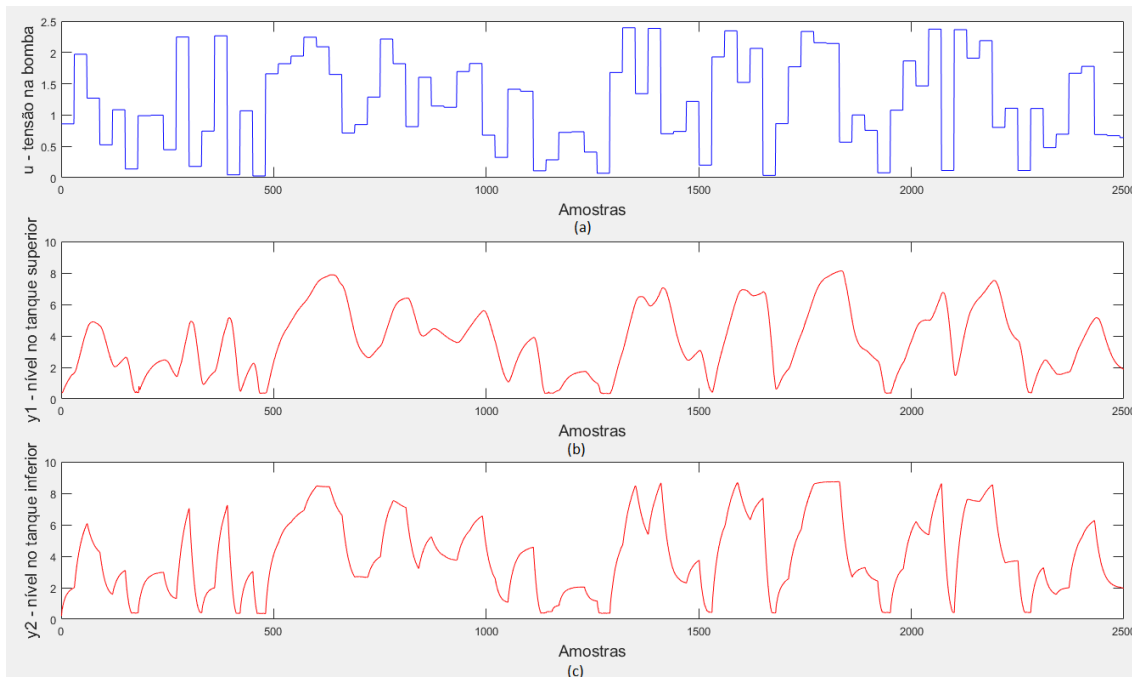


Figura 40 - Variáveis do estudo de caso 5 (a) Entrada: u – tensão aplicada na bomba. (b) Saída 1: y_1 – nível de água no tanque superior. (c) Saída 2: y_2 – nível de água no tanque inferior

6.2.7. Estudo de caso 7 – Gerador de Vapor

Este estudo de caso representa a identificação de um sistema MIMO cujos dados são obtidos de um modelo (Pellegrinetti & Bentsman, 1996) para um gerador de vapor da planta termelétrica de Abbott localizada em Champaign-Illinois, EUA (figura 41). Tal sistema possui um comportamento altamente não linear, instável, de resposta inversa, de fase não mínima com ruídos e atrasos no tempo devido a distúrbios de carga (Pellegrinetti & Bentsman, 1996). Tais efeitos são herdados no modelo utilizado para gerar os dados, uma vez que o mesmo consegue representar os comportamentos dinâmicos mais importantes presentes no gerador de vapor de Abbott (Pellegrinetti & Bentsman, 1996).

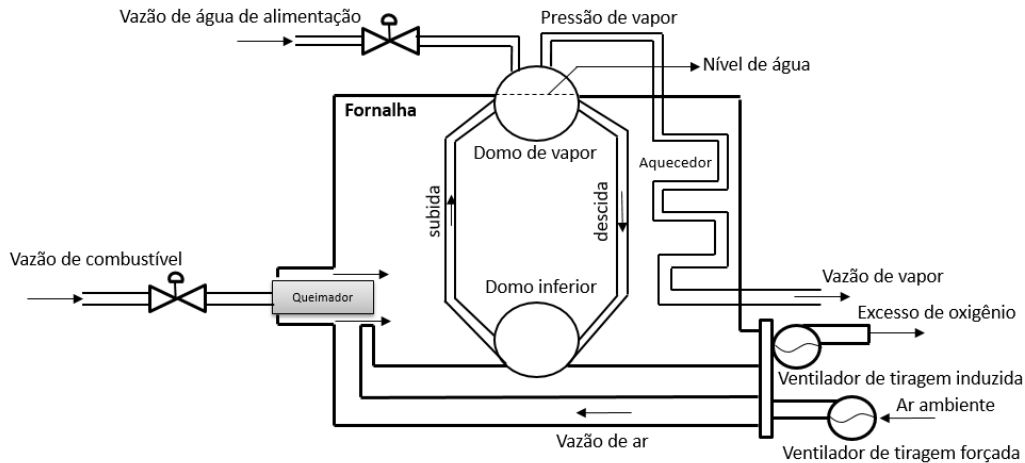


Figura 41 – Esquema básico do gerador de vapor. Fonte: Adaptado de Nozari & Banadaki (2014).

O gerador de vapor da figura 41 é do tipo WTSG (*Water Tube Steam Generator*), tendo como função principal produzir vapor e água aquecidos para utilização em processos industriais e na geração de energia elétrica (Nozari & Banadaki, 2014). Em seu interior existe uma fornalha onde ocorre queima de combustível (óleo ou gás), liberando gases quentes que transferem calor para os tubos de água via convecção. Bolhas de vapor se formam no tubo de água mais próximo ao queimador (tubo de subida) e sobem até o domo de vapor, onde ocorre a separação entre vapor e água (Nozari & Banadaki, 2014). O vapor que sai do domo inferior é substituído por água que flui em sentido contrário no tubo de descida, mantendo-se um ciclo natural de substituição (Nozari & Banadaki, 2014). Para compensar o vapor “perdido” no domo de vapor, é fornecido um fluxo contínuo de água de alimentação. Um ventilador de tiragem forçada fornece ar de combustão para o queimador, enquanto que um ventilador de tiragem induzida retira gases resultantes da combustão da fornalha. Antes de ser liberado para o ambiente, o vapor proveniente do domo superior do gerador de vapor passa por um aquecedor, onde é superaquecido.

A base de dados para este problema pode ser baixada de forma online no repositório DaISy (De Moor, 2018) e possui 9600 amostras, obtidas com um período de amostragem de 3 s. As entradas para este sistema são: a vazão de combustível (u_1), a vazão de ar (u_2), o nível de água de referência (u_3) e os distúrbios definidos pelo nível de água (u_4) (figura 42). As saídas para este sistema são: a pressão de vapor no domo

superior (y_1), o excesso de oxigênio nos gases de exaustão (y_2), o nível de água no domo de vapor (y_3) e a vazão de vapor produzido (y_4) (figura 43).

- Entradas:

- u_1 – vazão de combustível

- u_2 – vazão de ar

- u_3 – nível de água de referência

- u_4 – distúrbios definidos pelo nível de água

- Saídas:

- y_1 – pressão de vapor

- y_2 – excesso de oxigênio

- y_3 – nível de água no domo de vapor

- y_4 – vazão de vapor

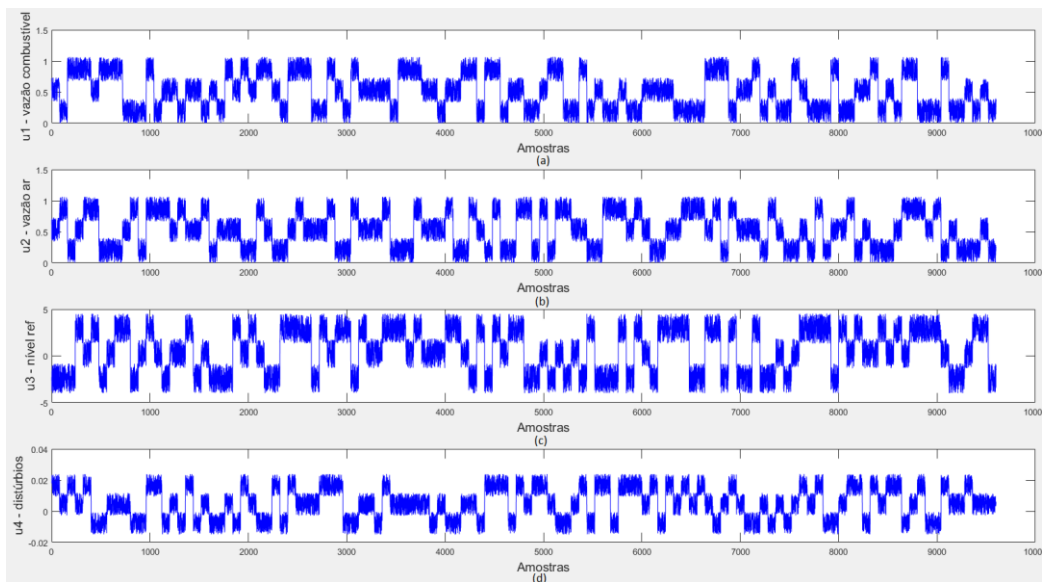


Figura 42 - Variáveis de entrada do estudo de caso 7 (a) Entrada 1: u_1 – vazão de combustível. (b) Entrada 2: u_2 – vazão de ar (c) Entrada 3: u_3 – nível de água de referência. (d) Entrada 4: u_4 – distúrbios definidos pelo nível de água

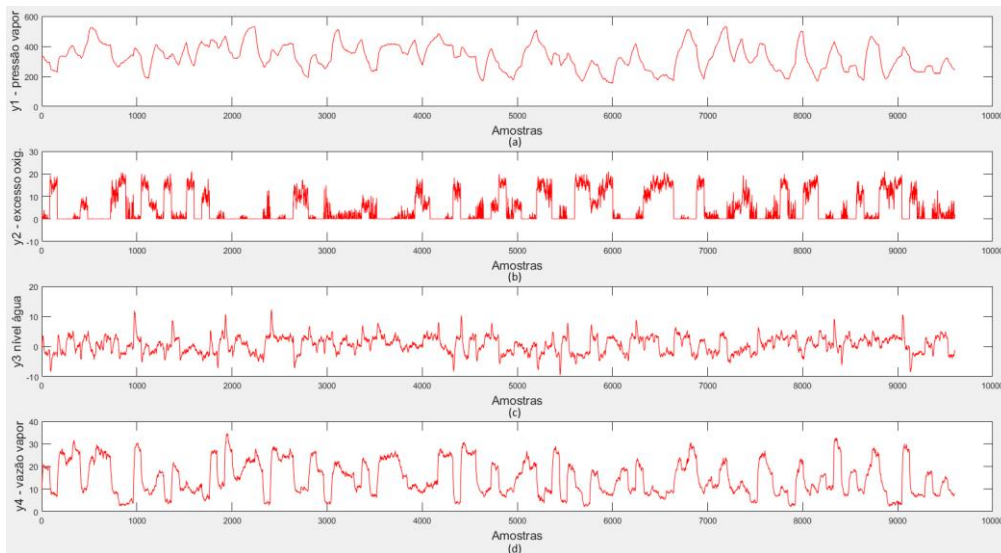


Figura 43 - Variáveis de saída do estudo de caso 7 (a) Saída 1: y_1 – pressão de vapor. (b) Saída 2: y_2 – excesso de oxigênio. (c) Saída 3: y_3 – nível de água no domo do gerador de vapor. (d) Saída 4: y_4 – vazão de vapor.

6.2.8. Estudo de caso 8 – Braço do leitor de CD

O benchmark “Braço do leitor de CD” trata da identificação da operação mecânica de um dispositivo leitor de CD (figura 44).

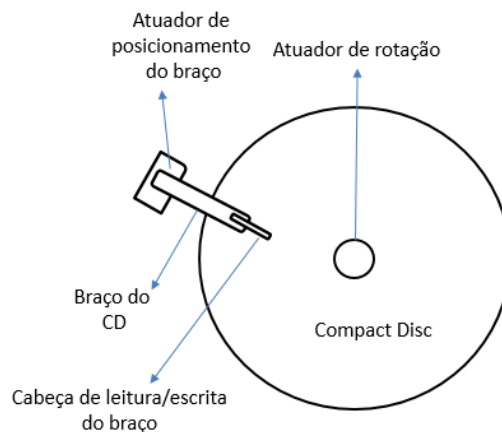


Figura 44 - Operação mecânica de um disposto leitor de CD. Fonte: Adaptado de Barry (2004)

O sistema mecânico da figura acima tem um comportamento dinâmico complexo e difícil de ser modelado por meio de equações físicas (Dudul & Ghatol, 2003), além de ser multivariável, apresentando como entradas (u_1, u_2) as forças aplicadas pelos atuadores (um para posicionamento do braço e outro para a rotação do CD) e suas saídas

(y_1, y_2) estão relacionadas com a precisão de rastreamento do braço em relação à superfície do CD. A base de dados para este problema possui 2048 amostras, com período de amostragem de 10 s, e foi obtida do repositório DaISy (De Moor, 2018) (figuras 45 e 46).

- Entradas:

- u_1, u_2 – forças aplicadas pelos atuadores

- Saídas:

- y_1, y_2 – precisão de rastreamento do braço

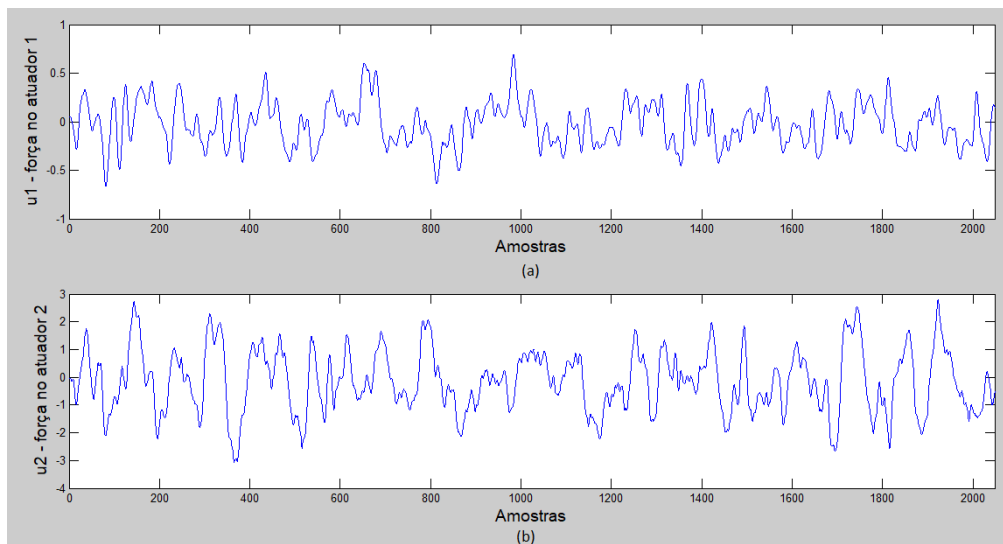


Figura 45 - Variáveis de entrada do estudo de caso 8 (a) Entrada 1: u_1 – força aplicada no atuador 1. (b) Entrada 2: u_2 – força aplicada no atuador 2

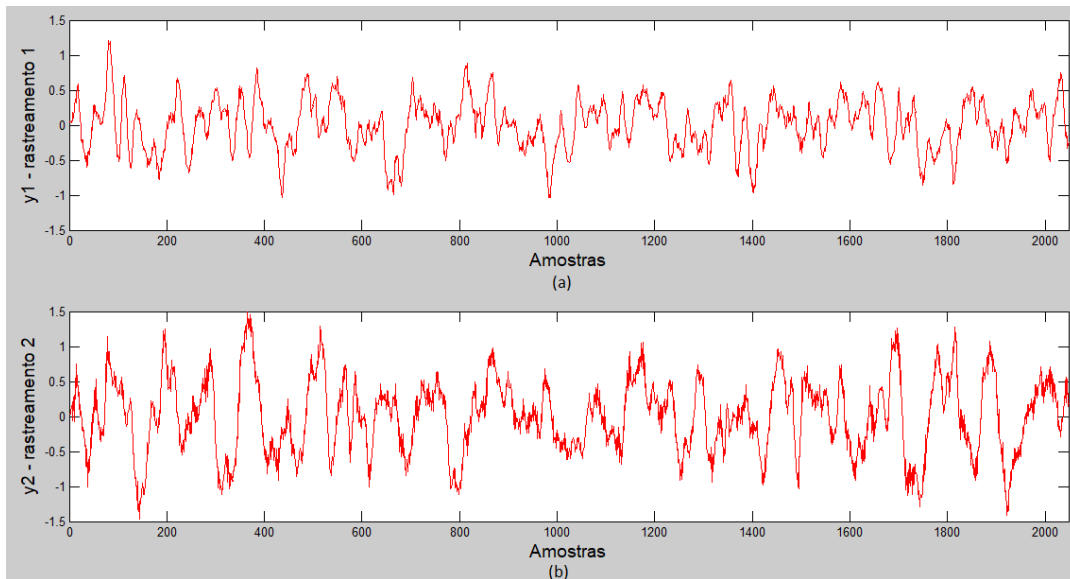


Figura 46 - Variáveis de saída do estudo de caso 8 (a) Saída 1: y_1 – precisão de rastreamento 1. (b) Saída 2: y_2 – precisão de rastreamento 2

6.2.9.

Estudo de caso 9 – Caixa de prata

Este estudo de caso refere-se à identificação de um circuito eletrônico de laboratório denominado “caixa de prata” que simula o funcionamento de um sistema mecânico de segunda ordem com uma m em movimento, um amortecedor viscoso d e uma mola não linear cuja constante elástica $k(y)$ é função do deslocamento y (figura 47).

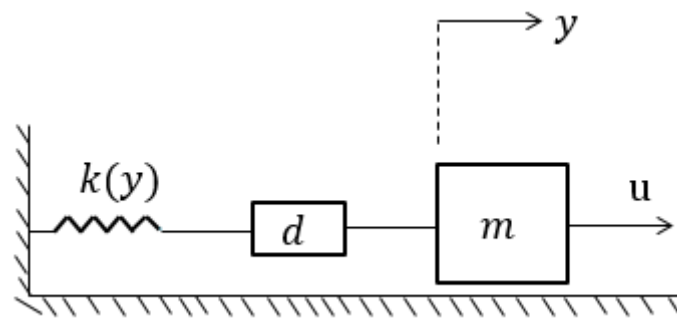


Figura 47 - Sistema mecânico massa-mola-amortecedor simulado pelo “caixa de prata”

O circuito “caixa de prata” é projetado para relacionar o deslocamento y (saída) à força aplicada u (entrada) seguindo a seguinte equação diferencial:

$$m\ddot{y} + d\dot{y} + k(y)y = u \quad (85)$$

com:

$$k(y) = a + by^2 \quad (86)$$

onde a e b são constantes.

A base de dados utilizada neste estudo de caso contém 131072 amostras (obtidas com um período de amostragem de 1/610,35 s e pode ser baixada livremente em (Wigren, 2013) (figura 48).

- Entrada:

- u – força aplicada na massa

- Saída:

- y – deslocamento da massa

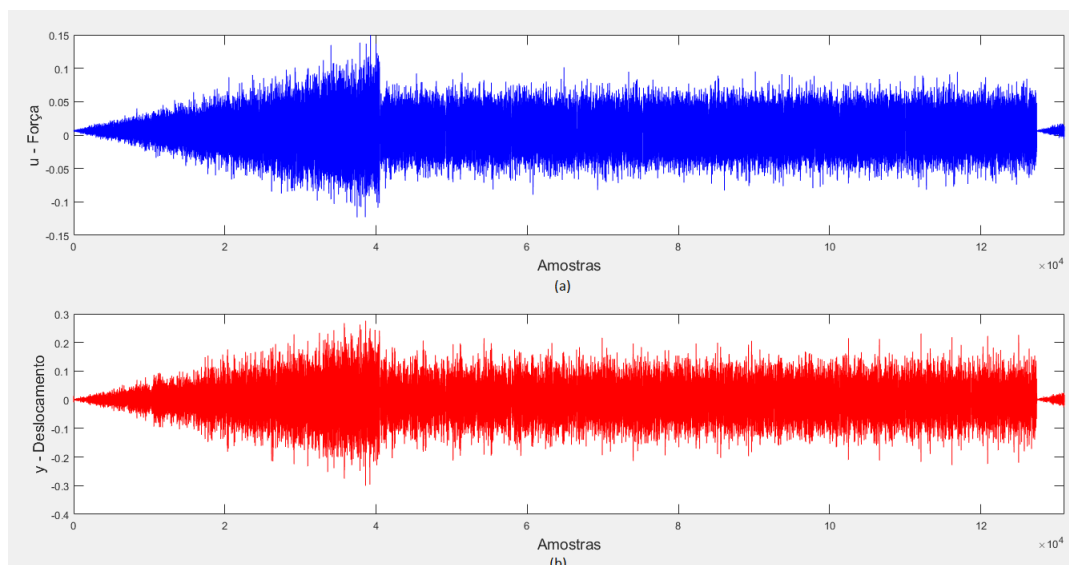


Figura 48 - Variáveis do estudo de caso 9 (a) Entrada: u – força aplicada na massa . (b) Saída: y – deslocamento da massa.

6.3. Configurações dos experimentos

6.3.1. Configurações gerais dos experimentos

Os experimentos para verificar o desempenho do modelo proposto nos problemas *benchmark* de Identificação de Sistemas foram executados no MATLAB (MATLAB

2016b) utilizando-se como base o *toolbox* para ESNs de Jaeger (2009) e os códigos fontes do AEIQ-R de Cruz (2007), do CMA-ES de Hansen (2012) e do coeficiente SRG de Niño (2016). Com o objetivo de se obter um desempenho médio do modelo proposto, em cada estudo de caso, foram realizadas 10 execuções do modelo geral (10 experimentos).

Por simplicidade, o AE escolhido para a etapa 1 também foi o utilizado na etapa 2 no módulo de otimização do reservatório. Em cada execução do método geral, foram utilizadas 10 ESNs ($n = 10$) para o cálculo de f_{eval_1} em (74) na etapa 1 e para gerar as redes a serem selecionadas pelos métodos Random e SRG na etapa 2. Com o objetivo de comparar as metodologias Random e SRG, o mesmo conjunto n de redes inicializadas após a etapa 1 pelo AE escolhido é utilizado como entrada para estes métodos. Com isto, do modelo geral são obtidos 6 submodelos a serem testados e comparados nos experimentos: AEIQ-R/Random/ESN, AEIQ-R/SRG/ESN, AEIQ-R/AEIQ-R/ESN, CMA-ES/Random/ESN, CMA-ES/SRG/ESN e CMA-ES/CMA-ES/ESN (tabela 3).

Tabela 3 – Submodelos resultantes do modelo geral testados

Modelo Geral		Submodelos	
Etapa 1	Etapa 2	Nome	Nº
AEIQ-R	Random	AEIQ-R/Random/ESN	1
	SRG	AEIQ-R/SRG/ESN	2
	AEIQ-R	AEIQ-R/AEIQ-R/ESN	3
CMA-ES	Random	CMA-ES/Random/ESN	4
	SRG	CMA-ES/SRG/ESN	5
	CMA-ES	CMA-ES/CMA-ES/ESN	6

6.3.2.

Configurações das ESNs

Como dito anteriormente, para geração e treinamento das ESNs foi utilizado o *toolbox* de Jaeger (Jaeger, 2009). Mantendo-se as configurações do mesmo, as matrizes \mathbf{W}_0^{in} e $\mathbf{W}_0^{\text{back}}$ foram inicializadas entre $[-1;1]$ e \mathbf{W}_0 entre $[-0,5; 0,5]$. Como função de ativação para o reservatório, utilizou-se a *tanh*, enquanto que na camada de saída, foi utilizada a função linear. O método de treinamento foi o da pseudoinversa, como descrito na seção 3.5. As entradas e saídas da rede, em cada estudo de caso, são as mesmas das bases de dados descritas na seção 6.2, isto é não se utilizou nenhuma entrada adicional

com sinais atrasados no tempo das entradas e/ou saídas do sistema. A tabela 4 resume as configurações utilizadas para as ESNs.

Tabela 4 – Configurações das ESNs

Método de Treinamento	Pseudoinversa
Função de ativação do reservatório	<i>Tanh</i>
Tipo de neurônio no reservatório	<i>Leaky Integrator</i>
Função de ativação da saída	<i>Linear</i>
Faixa de \mathbf{W}_0^{in}	[-1; 1]
Faixa de $\mathbf{W}_0^{\text{back}}$	[-1; 1]
Faixa de \mathbf{W}_0	[-0,5; 0,5]
Entradas da rede	Mesmas da base de dados do sistema
Saídas da rede	Mesmas da base de dados do sistema

6.3.3.

Divisão e pré-processamentos das bases de dados

A tabela 5 resume as informações das bases de dados apresentadas na seção 6.2:

Tabela 5 – Informações das bases de dados utilizadas nos estudos de casos

Estudo de Caso	Nome da Base de Dados	Amostras	Entradas	Saídas
1	<i>Trocador de calor a vapor líquido-saturado</i>	4000	1	1
2	<i>Secador de cabelo</i>	1000	1	1
3	<i>Densidade de fluxo de calor através de uma parede de duas camadas</i>	1680	2	1
4	<i>Braço robótico flexível</i>	1024	1	1
5	<i>Reator tanque de agitação contínua</i>	7500	1	2
6	<i>Tanques em cascata</i>	2500	1	2
7	<i>Gerador de Vapor</i>	9600	4	4
8	<i>Braço do leitor de CD</i>	2048	2	2
9	<i>Caixa de prata</i>	131072	1	1

Em cada estudo de caso, as amostras disponíveis das bases de dados foram divididas em 3 conjuntos: treinamento, validação e teste. Visando uma posterior comparação de resultados, a escolha das amostras utilizadas em cada um destes subconjuntos de dados foi condicionada às divisões utilizadas por trabalhos recentes na literatura que utilizaram as mesmas bases de dados. Desta forma, o conjunto de teste em cada base de dados desta dissertação foi o mesmo utilizado por trabalhos recentes,

enquanto que os dados de estimação utilizados nestes trabalhos foram divididos de forma similar, na maioria dos casos, entre treinamento e validação nesta dissertação. A tabela 6 mostra a divisão das bases de dados empregada nesta dissertação em cada estudo de caso, juntamente com o valor do período de *washout* (T_0) para as ESNs, que foi arbitrado e mantido fixo nos experimentos. Os autores com os quais os modelos serão comparados são citados na seção 6.4 de apresentação de resultados.

Tabela 6 - Divisões das bases de dados valores de T_0 para cada estudo de caso

Estudo de caso	Base de dados ¹			T_0
	Treinamento	Validação	Teste	
1	550 (1 a 550)	450 (551 a 1000)	3000 (1001 a 4000)	100
2	150 (1 a 150)	150 (151 a 300)	300 (301 a 1000)	50
3	200 (1 a 200)	300 (201 a 500)	1000 (501 a 1500)	10
4	312 (1 a 312)	200 (313 a 512)	512 (513 a 1024)	30
5	2500 (1 a 2500)	2500 (2501 a 5000)	2500 (5001 a 7500)	100
6	750 (1 a 750)	500 (751 a 1250)	1250 (1251 a 2500)	100
7	5600 (1 a 5600)	2000 (5601 a 7600)	2000 (7601 a 9600)	100
8	200 (1 a 200)	300 (201 a 500)	1000 (501 a 1500)	10
8 (extra)	600 (1 a 600)	600 (601 a 1200)	839 (1210 a 2048)	10
9	51072 (40001 a 91072)	40000 (91073 a 131072)	40000 (1 a 40000)	1000

1 – Número de amostras (amostra inicial a amostra final)

Antes de serem utilizados no processo de identificação, os dados foram linearmente normalizados entre 0 e 1, seguindo-se:

$$X_N = \frac{X - X_{mín}}{X_{máx} - X_{mín}} \quad (87)$$

onde X_N é o valor normalizado da variável X , $X_{máx}$ e $X_{mín}$ são seus valores máximo e mínimo. Assumiu-se para $X_{máx}$ e $X_{mín}$, valores 20% acima e 20% abaixo dos valores máximo e mínimo, respectivamente, da variável correspondente nos dados de estimação (treinamento e validação) de forma que os dados de teste não influenciem a normalização dos dados.

6.3.4.

Configurações dos Algoritmos Evolutivos

Ainda que o modelo proposto defina automaticamente os parâmetros de uma ESN, os parâmetros dos AEs, bem como as faixas de variação de suas variáveis precisam ser ajustados, o que depende de um processo de tentativa e erro. A seguir são mostrados os valores de tais parâmetros definidos em alguns testes iniciais.

O intervalo de variação utilizado em cada variável a ser otimizada pelos AEs nas etapas 1 e 2 em cada experimento é mostrado na tabela 7.

Tabela 7 – Intervalo de variação das variáveis otimizadas pelo AEIQ-R e pelo CMA-ES

Estudo de caso	Etapa 1						Etapa 2
	N	c_W	ρ_W	s_{in}	s_{back}	α	\tilde{w}
1	[40; 250]	[0,1; 1]	[0,1; 0,99]	[0,0001; 1]	[0; 1]	[0,1; 1]	[-0,5; 0,5]
2	[20; 100]	[0,1; 1]	[0,1; 0,99]	[0,0001; 1]	[0; 1]	[0,1; 1]	[-0,5; 0,5]
3	[20; 100]	[0,1; 1]	[0,1; 0,99]	[0,0001; 1]	[0; 1]	[0,1; 1]	[-0,5; 0,5]
4	[20; 150]	[0,1; 1]	[0,1; 0,99]	[0,0001; 1]	[0; 1]	[0,1; 1]	[-0,5; 0,5]
5	[50; 300]	[0,1; 1]	[0,1; 0,99]	[0,0001; 1]	[0; 1]	[0,1; 1]	[-0,5; 0,5]
6	[40; 200]	[0,1; 1]	[0,1; 0,99]	[0,0001; 1]	[0; 1]	[0,1; 1]	[-0,5; 0,5]
7	[50; 500]	[0,1; 1]	[0,1; 0,99]	[0,0001; 1]	[0; 1]	[0,1; 1]	[-0,5; 0,5]
8	[15; 75]	[0,1; 1]	[0,1; 0,99]	[0,0001; 1]	[0; 1]	[0,1; 1]	[-0,5; 0,5]
9	[200; 450]	[0,1; 1]	[0,1; 0,99]	[0,0001; 1]	[0; 1]	[0,1; 1]	[-0,5; 0,5]

Como mostrado pela tabela 7, o intervalo de variação dos parâmetros gerais c_W , ρ_W , s_{in} , s_{back} e α e dos pesos do reservatório foi o mesmo em todos os experimentos. A única faixa de valores que mudou entre os problemas foi o do parâmetro N . Conforme visto no capítulo 3, para evitar problemas de *overfitting*, N deveria figurar entre $T/10$ e $T/2$, porém como verificado em alguns testes iniciais tal faixa de valores superestima os valores de N e em alguns casos, como o dos problemas 7 e 9, que envolvem bases de dados maiores e/ou com mais variáveis, tal estratégia é inviável de ser utilizada devido ao grande custo de se utilizar redes muito grandes. Desta forma, tal faixa foi usada como ponto de partida e conforme as soluções atingidas pelos AEs, restringiu-se tal faixa nos valores da tabela 7 para os experimentos finais.

Para o AEIQ-R, o intervalo de gerações em que a população quântica é atualizada (*upG*) foi mantido fixo em 5 gerações e os valores dos demais parâmetros de configuração foram definidos após alguns testes iniciais, testando-se algumas configurações fixas. Durante a execução dos experimentos, deu-se preferência a valores pequenos de população clássica (no máximo 75) de forma a reduzir o custo computacional. Conseqüentemente, o tamanho da população quântica foi menor do que o da clássica e escolhido de forma que N_C fosse múltiplo de N_Q . Durante os experimentos, a ideia era de se testarem números de gerações de 50, 100, 150 e 200 nas duas etapas, porém, verificou-se que o tempo computacional despendido na segunda etapa era bem menor do que na primeira, então utilizou-se um número maior de gerações na etapa 2. Alguns valores tabelados foram testados para C_r e C_Q . A tabela 8 mostra os valores definidos para os parâmetros de configuração do AEIQ-R para as etapas 1 e 2, em cada experimento, os quais retornaram os melhores valores de aptidão nos testes iniciais:

Tabela 8 – Parâmetros de configuração do AEIQ-R

Estudo de caso	Etapla	N_C	N_Q	C_r	<i>Gen</i>	<i>upG</i>	C_Q
1	1	50	10	0,45	150	5	0,75
	2	75	15	0,45	600	5	0,75
2	1	60	20	0,9	50	5	0,8
	2	50	25	0,85	800	5	0,7
3	1	50	5	0,8	50	5	0,4
	2	40	10	0,3	500	5	0,8
4	1	60	20	0,6	100	5	0,2
	2	40	10	0,3	1000	5	0,8
5	1	30	10	0,3	100	5	0,6
	2	45	15	0,6	700	5	0,4
6	1	30	10	0,9	100	5	0,6
	2	30	5	0,6	800	5	0,4
7	1	30	10	0,6	100	5	0,8
	2	50	10	0,6	100	5	0,8
8	1	75	15	0,85	150	5	0,7
	2	60	15	0,8	500	5	0,6
9	1	15	5	0,3	30	5	0,6
	2	30	10	0,9	30	5	0,4

No caso do CMA-ES, não realizou-se testes iniciais, apenas igualou-se o valor de λ ao de N_C , previamente determinado para o AEIQ-R, e utilizou-se o mesmo número de

gerações utilizadas pelo AEIQ-R, de forma que o número de avaliações de ambos os algoritmos fosse o mesmo (tabela 9). O valor inicial de $\mathbf{m}^{(0)}$, que é uma das entradas para o algoritmo, foi escolhido como um vetor aleatório uniformemente distribuído dentro do intervalo de variação das variáveis do problema. O valor de $\boldsymbol{\sigma}^{(0)}$, outra entrada para o algoritmo, foi escolhido como um vetor com valor de 30% da faixa de variação de cada variável do problema. Os demais parâmetros do algoritmo foram mantidos em seus valores *default*.

Tabela 9 - Parâmetros de configuração do CMA-ES

Estudo de caso	Etapa	λ	μ (default)	Gen
1	1	50	25	150
	2	75	37	600
2	1	60	30	50
	2	50	25	800
3	1	50	25	50
	2	40	20	500
4	1	60	30	100
	2	40	20	1000
5	1	30	15	100
	2	45	22	700
6	1	30	15	100
	2	30	15	800
7	1	30	15	100
	2	50	25	100
8	1	75	37	150
	2	60	30	500
9	1	15	7	30
	2	30	15	30

6.4.

Resultados e discussões

Todos os resultados apresentados a seguir referem-se ao desempenho da ESN durante a simulação livre (previsão de infinitos passos à frente) dos dados e não a previsões de 1 passo à frente. Devido à quantidade de estudos de caso e também ao grande tamanho de algumas bases de dados, os experimentos foram executados em diferentes computadores e com versões paralelizadas do AEIQ-R e sequenciais do CMA-ES, o que inviabilizou uma comparação em termos de custo computacional das abordagens testadas.

Desta forma, somente serão comparados e discutidos somente os resultados de acurácia das 6 diferentes variações do modelo geral testadas.

Quanto à exibição dos resultados, mesmo que se tenham realizados 10 experimentos independentes para cada estudo de caso, somente serão exibidos os gráficos referentes à simulação livre das variáveis de saída e gráficos de razão de separação da configuração de cada submodelo que tiver obtido melhor desempenho durante o teste. Serão omitidos os resultados de otimização dos pesos das ESNs e somente será exibida a configuração de parâmetros globais do submodelo que tiver atingido melhor desempenho entre os testados. Os resultados de todas as soluções de cada submodelo nos 10 experimentos podem ser consultados no apêndice A.

Para cada estudo de caso, foram calculados o desempenho médio nos conjuntos de treinamento, validação e teste da melhor ESN retornada por cada submodelo nos 10 experimentos, conforme figura a seguir:

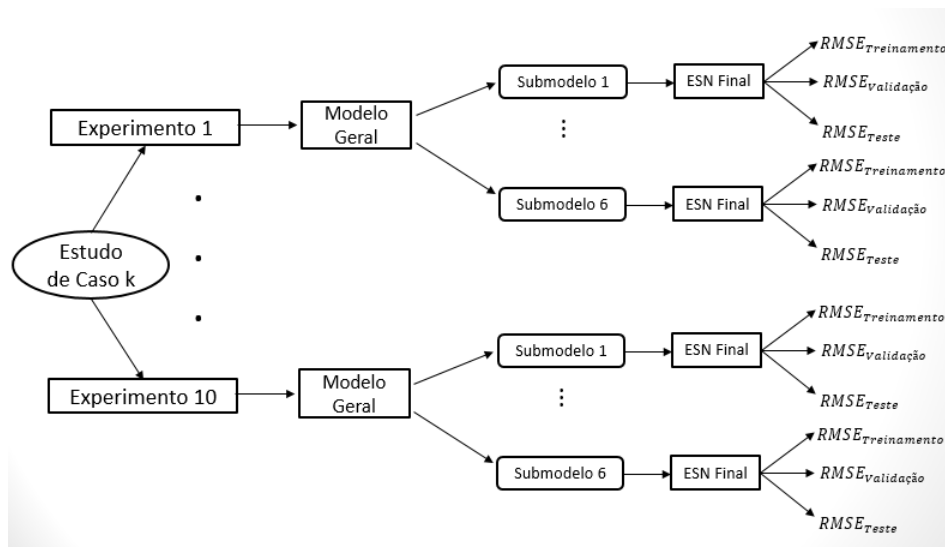


Figura 49 - Organização geral dos experimentos

Como mostrado pela figura 49, para cada estudo de caso, foram realizados 10 experimentos, que correspondem a 10 execuções independentes do modelo geral. Em cada experimento, cada um dos 6 submodelos derivados do modelo geral retornou uma ESN final, a qual foi utilizada para simular os conjuntos de treinamento, validação e teste, retornando em cada um dos casos um valor de RMSE ($RMSE_{treinamento}$, $RMSE_{validação}$ e $RMSE_{teste}$, respectivamente). Desta forma, nas

subseções seguintes, em cada estudo de caso, foi obtido o desempenho médio de cada submodelo nos conjuntos de treinamento, validação e teste, da seguinte forma:

$$RMSE_{m\u00e9dio_{treinamento}_{ik}} = \sum_{j=1}^{10} \frac{RMSE_{treinamento_{ij}}}{10} \quad (88)$$

$$RMSE_{m\u00e9dio_{valida\u00e7\u00e3o}_{ik}} = \sum_{j=1}^{10} \frac{RMSE_{valida\u00e7\u00e3o_{ij}}}{10} \quad (89)$$

$$RMSE_{m\u00e9dio_{teste}_{ik}} = \sum_{j=1}^{10} \frac{RMSE_{teste_{ij}}}{10} \quad (90)$$

onde $k = 1, 2, \dots, 9$ corresponde ao n\u00famero do experimento, $i = 1, 2, \dots, 6$ \u00e9 o n\u00famero do submodelo, $j = 1, 2, \dots, 10$ \u00e9 o n\u00famero do experimento e $RMSE_{m\u00e9dio_{treinamento}_{ik}}$, $RMSE_{m\u00e9dio_{valida\u00e7\u00e3o}_{ik}}$, $RMSE_{m\u00e9dio_{teste}_{ik}}$, correspondem ao $RMSE$ m\u00e9dio nos 10 experimento do submodelo i no estudo de caso k nos conjuntos de treinamento, alida\u00e7\u00e3o e teste, respectivamente.

Cabe observar que na etapa 1 do modelo geral, durante a avalia\u00e7\u00e3o, s\u00e3o treinadas e \u00e9 obtido o desempenho m\u00e9dio de 10 ESNs (coincidentemente com o n\u00famero de experimentos) para cada indiv\u00edduo, que representa um conjunto de par\u00e2metros gerais. Ainda, durante os m\u00e9todos *Random* e SRG na segunda etapa s\u00e3o inicializadas 10 ESNs (tamb\u00e9m coincidentemente com o n\u00famero de experimentos) com os par\u00e2metros determinados na etapa 1, as quais s\u00e3o selecionadas com base no erro de valida\u00e7\u00e3o e coeficiente SRG, respectivamente.

6.4.1.

Estudo de caso 1 - Trocador de calor a vapor l\u00edquido-saturado

A tabela 10 exibe os resultados obtidos durante os 10 experimentos pelos diferentes submodelos derivados do m\u00e9todo geral para o estudo de caso 1 – *Trocador de calor a vapor l\u00edquido-saturado*, com o AEIQ-R e o CMA-ES configurados conforme tabelas 7 e 8 e com a base de dados dividida como mostrado na tabela 6:

Tabela 10 – Resultados obtidos de RMSE pelos diferentes submodelos para o *benchmark Trocador de calor a vapor líquido-saturado* nos 10 experimentos realizados

Submodelo	Treinamento	Validação	Teste			
	média	média	média	melhor	pior	desvio
AEIQ-R/Random/ESN	0,025660	0,150896	0,194123	0,175676	0,240606	0,017544
AEIQ-R/SRG/ESN	0,025511	0,187822	0,236160	0,202707	0,264654	0,020497
AEIQ-R/AEIQ-R/ESN	0,024818	0,077889	0,133451	0,127238	0,139479	0,003544
CMA-ES/Random/ESN	0,030581	0,156905	0,196809	0,179600	0,254880	0,021456
CMA-ES/SRG/ESN	0,030700	0,181703	0,228058	0,191796	0,256434	0,026123
CMA-ES/CMA-ES/ESN	0,025197	0,100066	0,144726	0,131503	0,157982	0,009315

Pelos resultados mostrados pela tabela 10, pode-se ver que para este estudo de caso os submodelos AEIQ-R/AEIQ-R/ESN e CMA-ES/CMA-ES/ESN, que otimizaram tanto os parâmetros globais como os pesos da ESN, foram os mais precisos, apresentando um RMSE médio na simulação do conjunto de teste de 0,133451 e 0,144726, respectivamente. Comparando-se os métodos Random e SRG, pelos resultados obtidos na tabela 10, pode-se ver que tanto quanto utiliza-se o AEIQ-R como o CMA-ES na etapa 1, neste estudo de caso, o método Random na etapa 2 gera, em média, resultados melhores do que as soluções encontradas pelo método SRG na simulação do conjunto de teste. Antes de se comparar o desempenho dos AEs, as figuras 49 e 50 mostram as curvas de evolução dos algoritmos AEIQ-R e CMA-ES nas etapas 1 e 2, exibindo-se a média da aptidão nos 10 experimentos do melhor indivíduo da população a cada geração:

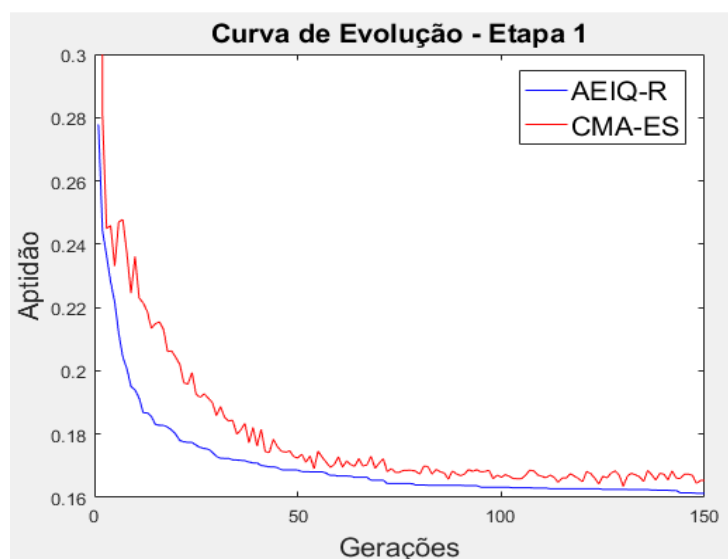


Figura 50 – Curva de evolução da etapa 1 (média da aptidão do melhor indivíduo em cada geração nos 10 experimentos) para o estudo de caso 1

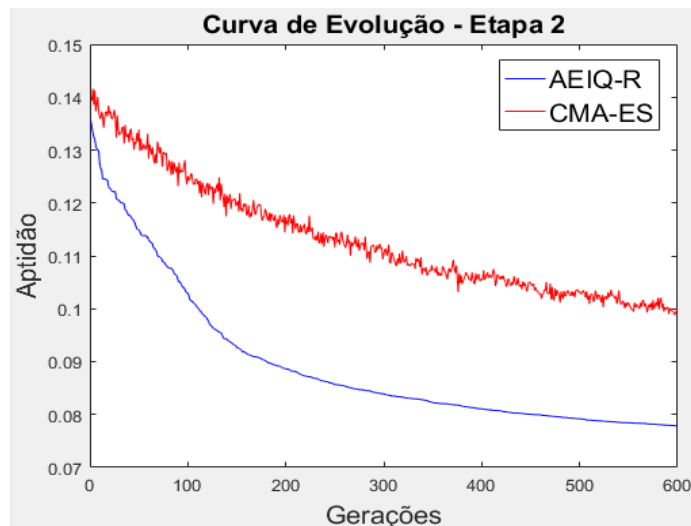


Figura 51 - Curva de evolução da etapa 2 (média da aptidão do melhor indivíduo em cada geração nos 10 experimentos) para o estudo de caso 1

Tabela 11 – Aptidão dos AEs na última geração (média de 10 experimentos) para o estudo de caso 1

Algoritmo	Aptidão na última geração (média de 10 experimentos)	
	Etapa 1	Etapa 2
AEIQ-R	0,161294	0,077889
CMA-ES	0,165404	0,100066

Como mostrado na figura 49, o AEIQ-R conseguiu evoluir, em média, soluções com melhor aptidão durante todas as gerações da etapa 1, gerando resultados ligeiramente melhores na última geração do que o CMA-ES (0,161294 contra 0,165404, respectivamente, conforme tabela 11). Esta ligeira superioridade do AEIQ-R refletiu-se em melhores soluções geradas pelo método Random durante o teste (RMSE médio de 0,194123 para o AEIQ-R/Random/ESN) em comparação com o método Random proveniente do CMA-ES (0,196809 para o CMA-ES/Random/ESN), mas ligeira inferioridade comparando-se ambas abordagens com o método SRG na etapa 2 (AEIQ-R/SRG/ESN - 0,236160 e AEIQ-R/SRG/ESN - 0,228058). Cabe ressaltar que a oscilação de valores de aptidão entre uma geração e outra na curva de evolução do CMA-ES se deve ao mesmo se tratar de uma ES (μ, λ) e, portanto, não elitista, enquanto que o AEIQ-R guarda as melhores soluções a cada geração na população $B(t)$, gerando uma curva de evolução sempre descendente.

Pela figura 50, pode-se ver que há uma superioridade mais acentuada para o AEIQ-R frente ao CMA-ES (aptidão média de 0,077889 contra 0,100066, respectivamente na última geração conforme tabela 11) durante a evolução da etapa 2,

embora ainda, visualmente, ambos algoritmos pareçam não ter convergido. Tal superioridade do AEIQ-R refletiu-se em um melhor desempenho médio durante o teste pelas soluções encontradas pelo método AEIQ-R/AEIQ-R/ESN (RMSE de 0,133451) em comparação com o CMA-ES/CMA-ES/ESN (RMSE de 0,144726).

A tabela 12 mostra a configuração de parâmetros globais que gerou as melhores soluções de cada submodelo (dentre as 10 soluções finais de cada experimento), enquanto que as figuras 51 a 56 mostram a simulação por cada uma dessas soluções dos conjuntos de treinamento, validação e teste. A figura 57 mostra os gráficos de razão de separação para cada uma dessas soluções.

Tabela 12 – Parâmetros globais da melhor solução encontrada por cada submodelo para o estudo de caso 2

Melhor solução	N	c_w	ρ_w	s_{in}	s_{back}	α
AEIQ-R/Random/ESN	197	0,5367	0,6448	0,1121	0,0457	0,8792
AEIQ-R/SRG/ESN	197	0,5367	0,6448	0,1121	0,0457	0,8792
AEIQ-R/AEIQ-R/ESN	227	0,3728	0,4418	0,3426	0,0635	0,7791
CMA-ES/Random/ESN	119	0,5923	0,3414	0,0622	0,0178	0,9634
CMA-ES/SRG/ESN	210	0,6613	0,5655	0,0463	0,0115	0,9052
CMA-ES/CMA-ES/ESN	157	0,8251	0,7874	0,4939	0,7943	0,9940

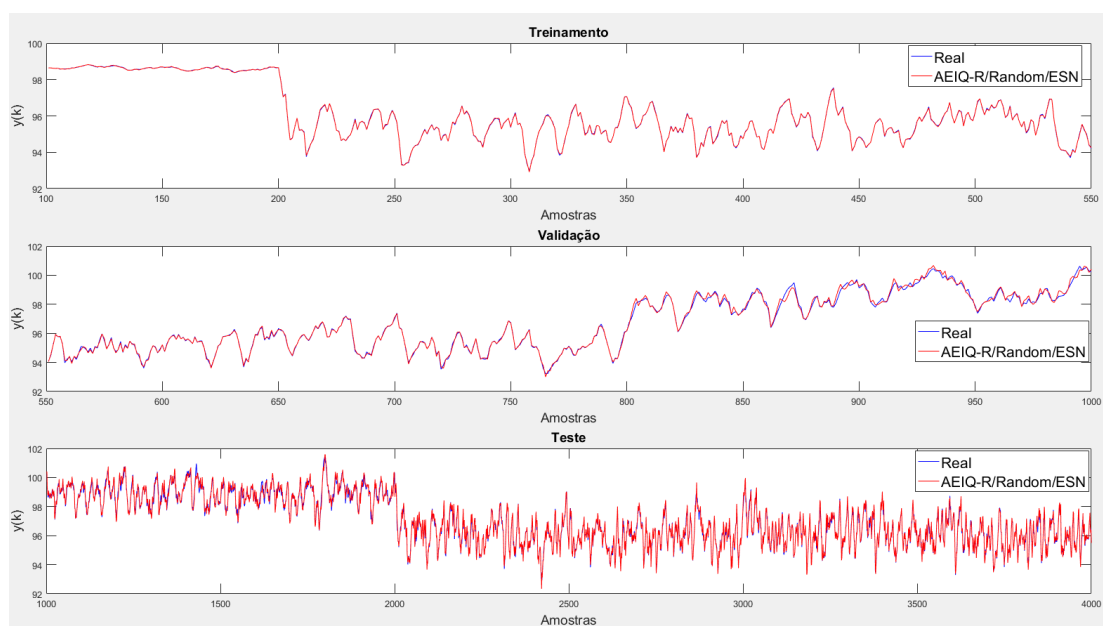


Figura 52 – Simulação da variável de saída y do estudo de caso 1 utilizando a melhor solução do submodelo AEIQ-R/Random/ESN

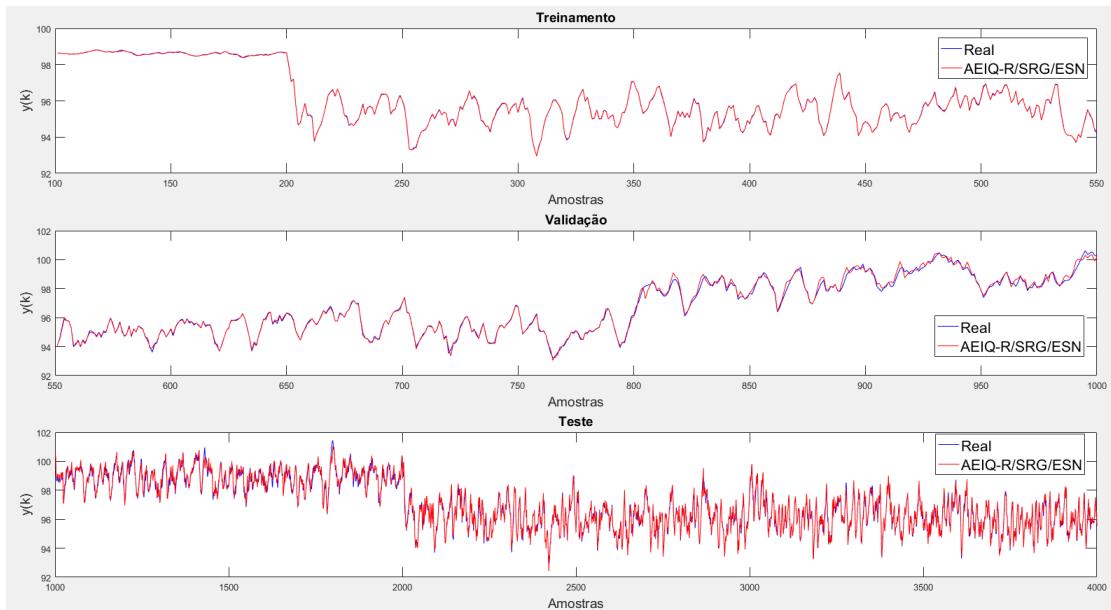


Figura 53 - Simulação da variável de saída y do estudo de caso 1 utilizando a melhor solução do submodelo AEIQ-R/SRG/ESN

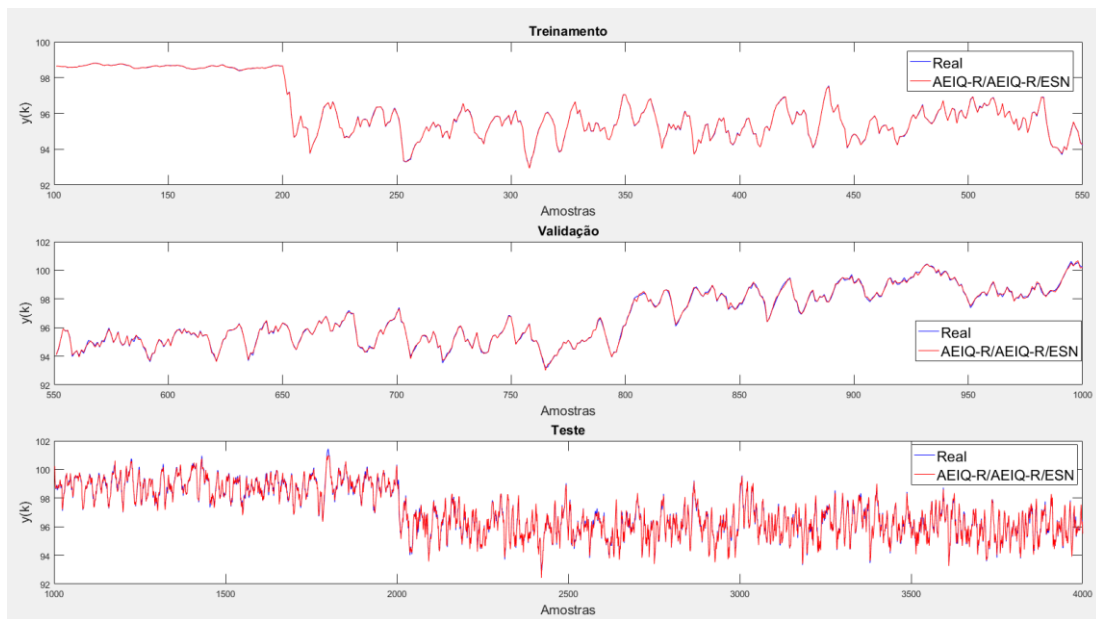


Figura 54 - Simulação da variável de saída y do estudo de caso 1 utilizando a melhor solução do submodelo AEIQ-R/AEIQ-R/ESN

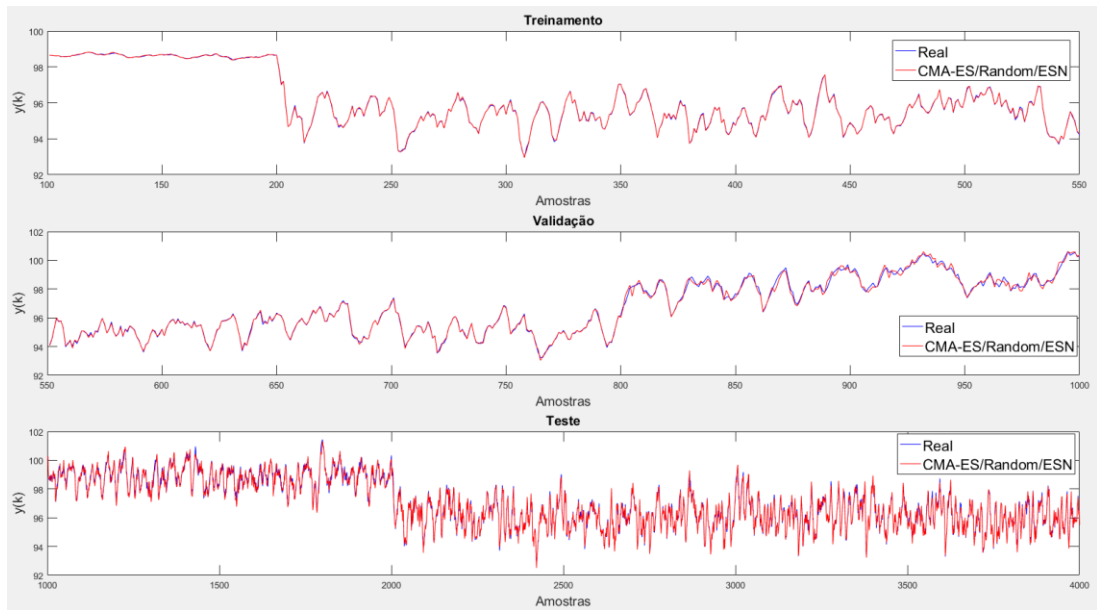


Figura 55 - Simulação da variável de saída y do estudo de caso 1 utilizando a melhor solução do submodelo CMA-ES/Random/ESN

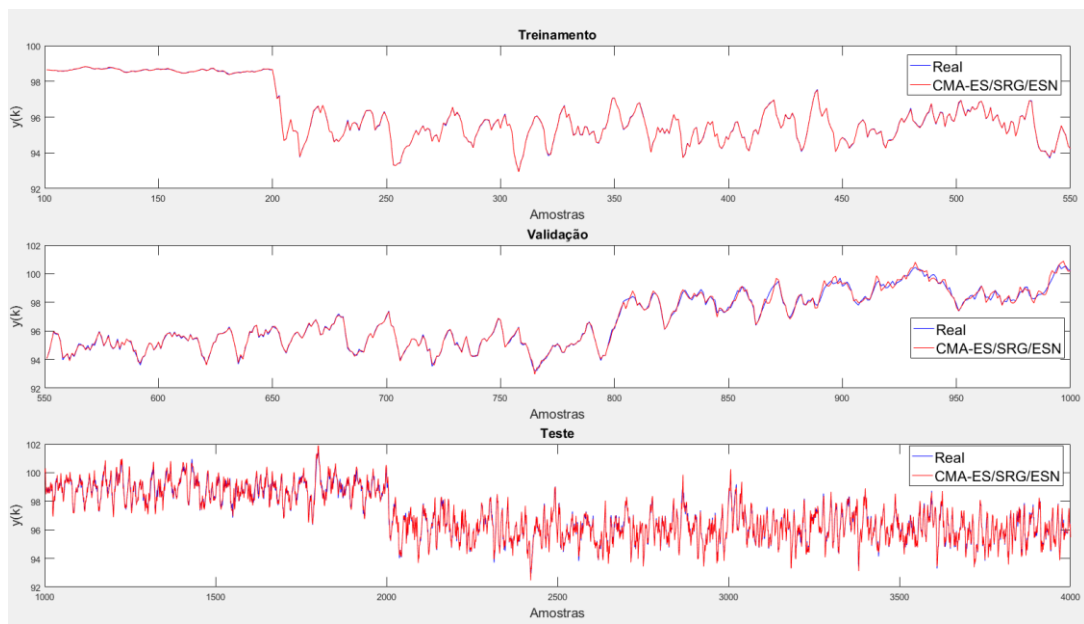


Figura 56 - Simulação da variável de saída y do estudo de caso 1 utilizando a melhor solução do submodelo CMA-ES/SRG/ESN

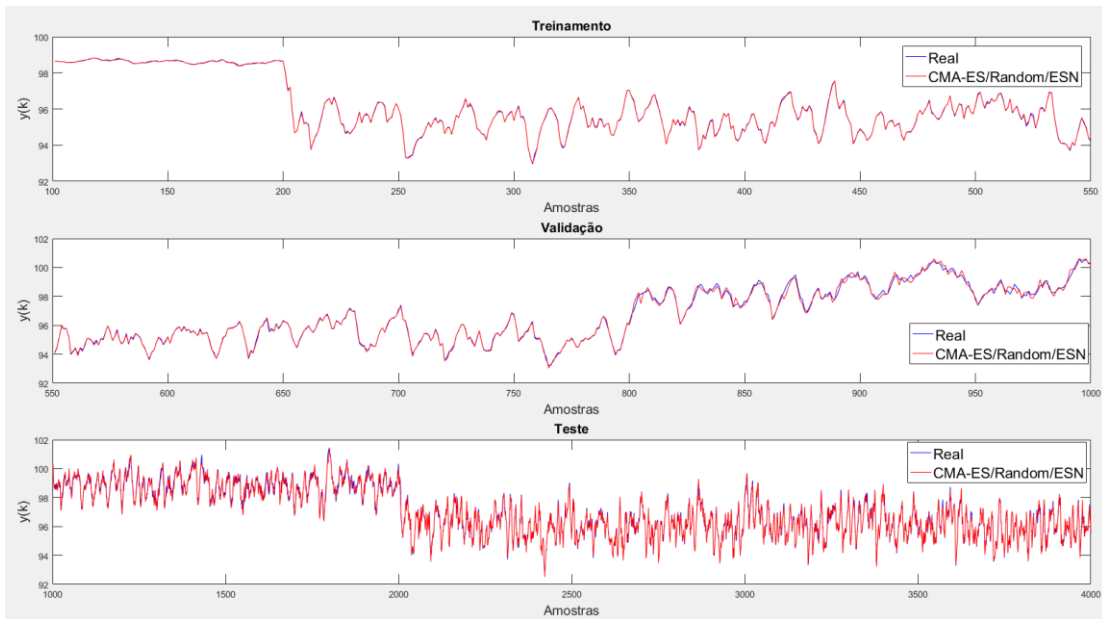


Figura 57 - Simulação da variável de saída y do estudo de caso 1 utilizando a melhor solução do submodelo CMA-ES/CMA-ES/ESN

A figura 57 mostra os gráficos de razão de separação obtidos para a melhor solução encontrada por cada submodelo.

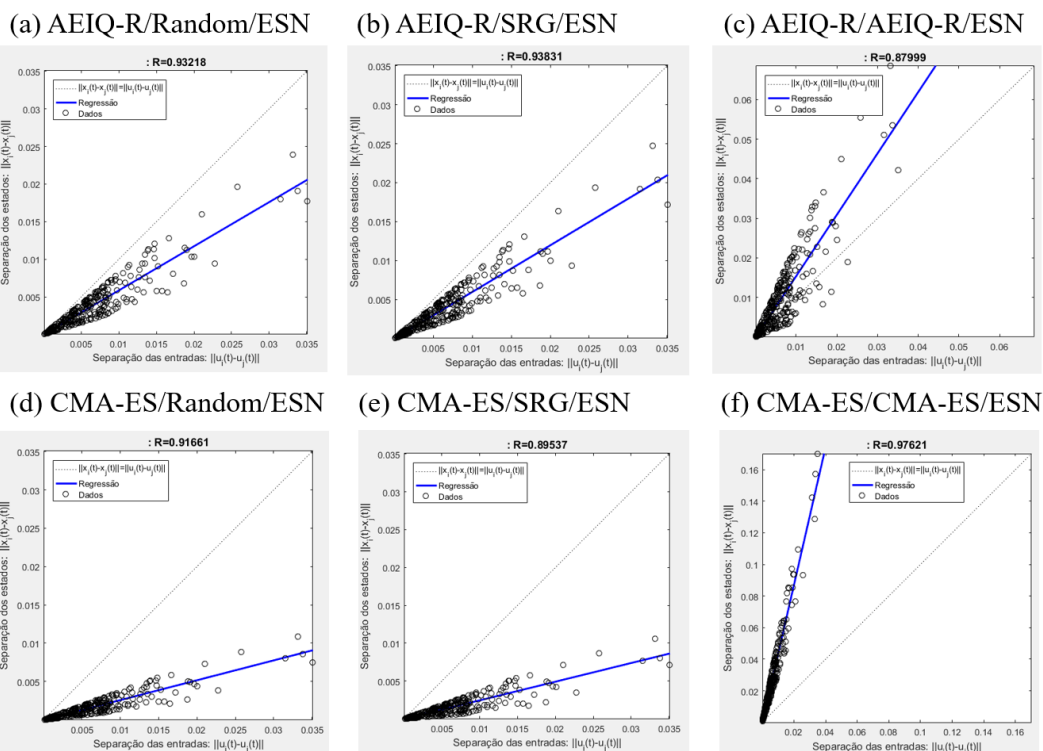


Figura 58 – Gráficos de razão de separação encontrados para as melhores soluções de cada submodelo para o estudo de caso 1

Como mostrado pelas figuras 52 a 57, as melhores soluções de cada submodelo dentre os 10 experimentos conseguiram simular adequadamente a variável de saída do sistema. Como mostrado anteriormente na tabela 10, dentre as melhores soluções de cada submodelo, os métodos AEIQ-R/AEIQ-R/ESN e CMA-ES/CMA-ES/ESN também apresentaram melhor acurácia nesta simulação (RMSEs de 0,127238 e 0,131503, respectivamente). Comparando-se os gráficos de razão de separação, pode-se ver as soluções encontradas pelos métodos AEIQ-R/SRG/ESN e AEIQ-R/Random/ESN foram bastante similares e mantiveram-se dentro da região atratora, enquanto que o reservatório da solução AEIQ-R/AEIQ-R/ESN, que passou por otimização, permaneceu na zona caótica e ainda assim, gerou resultados de acurácia melhores. No caso das soluções encontradas partindo-se do CMA-ES, o mesmo comportamento é observado, entretanto, as respectivas soluções provenientes do AEIQ-R visualmente parecem estar mais próximas da região objetivo.

A tabela 13 compara os resultados obtidos pelas melhores soluções dos submodelos com os melhores resultados dos métodos de Liu et al. (2013) e Hansson et al. (2012), que utilizaram as mesmas amostras de teste. A métrica utilizada para comparação é a fit, cujo objetivo é atingir um valor mais próximo de 100%:

$$fit(\%) = 100(1 - \frac{||\hat{y} - y||}{||y - \bar{y}||}) \quad (91)$$

Tabela 13 – Comparação de resultados com outros métodos para o *benchmark Liquid-Saturated Heat Exchanger*

Método	Métrica Fit
AEIQ-R/AEIQ-R/ESN (melhor)	91,8
CMA-ES/CMA-ES/ESN (melhor)	91,1
AEIQ-R/Random/ESN (melhor)	89,7
CMA-ES/Random/ESN (melhor)	88,9
CMA-ES/SRG/ESN (melhor)	88,2
AEIQ-R/SRG/ESN (melhor)	87,5
Weighted Nuclear Norm (Liu et al., 2013)	76,6
CVA (Hansson et al., 2012)	71,8
IVM (Hansson et al., 2012)	70,8
MOESP (Hansson et al., 2012)	70,8

Método	Métrica
	Fit
NOINSTR (Hansson et al., 2012)	70,8
N4SID (Hansson et al., 2012)	70,8
Regularized Nuclear Norm (Liu et al., 2013)	45,1
NONE (Hansson et al., 2012)	-245

Como mostrado pela tabela 13, os resultados atingidos pelas melhores soluções de cada submodelo deste trabalho são bem superiores às melhores soluções dos métodos de Liu et al. (2013) e Hansson et al. (2012) para este estudo de caso, com destaque para as soluções AEIQ-R/AEIQ-R/ESN e CMA-ES/CMA-ES/ESN que apresentaram valores de *fit* de 91,8% e 91,1%, respectivamente, enquanto que o melhor método de Liu et al. (2013), o *Weighted Nuclear Norm*, atingiu um *fit* de 76,6% no mesmo conjunto de teste.

6.4.2. Estudo de caso 2 – *Secador de cabelo*

A tabela 14 exibe os resultados obtidos durante os 10 experimentos pelos diferentes submodelos derivados do método geral para o estudo de caso 2 – *Secador de cabelo*, com o AEIQ-R e o CMA-ES configurados conforme tabelas 7 e 8 e com a base de dados dividida como mostrado na tabela 6:

Tabela 14 – Resultados obtidos de RMSE pelos diferentes submodelos para o *benchmark Secador de cabelo* nos 10 experimentos realizados

Submodelo	Treinamento	Validação	Teste			
	média	média	média	melhor	pior	desvio
AEIQ-R/ <i>Random</i> /ESN	0,051005	0,104672	0,135961	0,128737	0,143988	0,004907
AEIQ-R/SRG/ESN	0,052977	0,123969	0,152971	0,117701	0,180636	0,021669
AEIQ-R/AEIQ-R/ESN	0,135920	0,074572	0,130780	0,112544	0,180968	0,019675
CMA-ES/ <i>Random</i> /ESN	0,052960	0,102639	0,141140	0,130475	0,151993	0,007717
CMA-ES/SRG/ESN	0,050968	0,110911	0,149405	0,136650	0,164323	0,009816
CMA-ES/CMA-ES/ESN	0,082825	0,076495	0,123333	0,105634	0,142720	0,014679

Como mostrado pela tabela 14, neste segundo estudo de caso também pode-se verificar pelos resultados médios RMSE durante o teste, que os métodos AEIQ-R/AEIQ-R/ESN e CMA-ES/CMA-ES/ESN apresentaram um resultado de acurácia melhor do que

os demais, com um melhor desempenho, neste caso, para o último submodelo. A julgar os resultados de teste dos métodos Random e SRG, vê-se que tanto quanto se utiliza o AEIQ-R como o CMA-ES na etapa 1, o desempenho médio daquele método no teste supera os resultados desta última abordagem. As figuras 59 e 60 mostram as curvas de evolução do AEIQ-R e do CMA-ES nas etapas 1 e 2, exibindo-se a média da aptidão nos 10 experimentos do melhor indivíduo da população, enquanto a tabela evidencia os resultados de aptidão atingidos por cada algoritmo na última geração de ambas etapas:

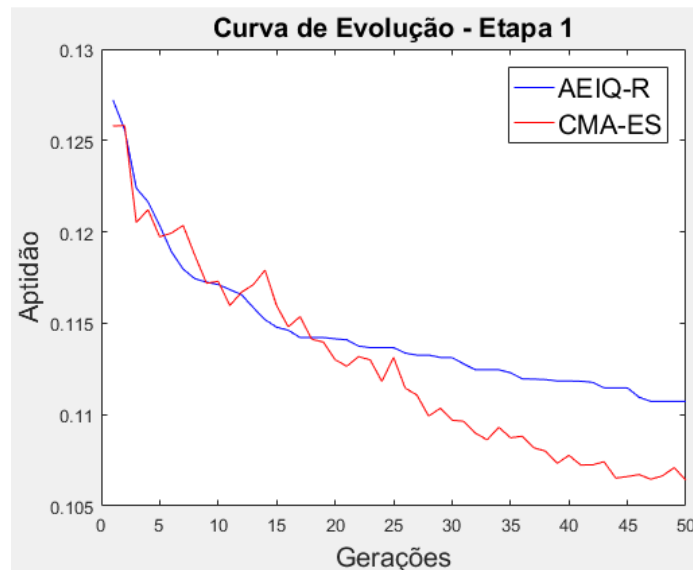


Figura 59 – Curva de evolução da etapa 1 (média da aptidão do melhor indivíduo em cada geração nos 10 experimentos) para o estudo de caso 2

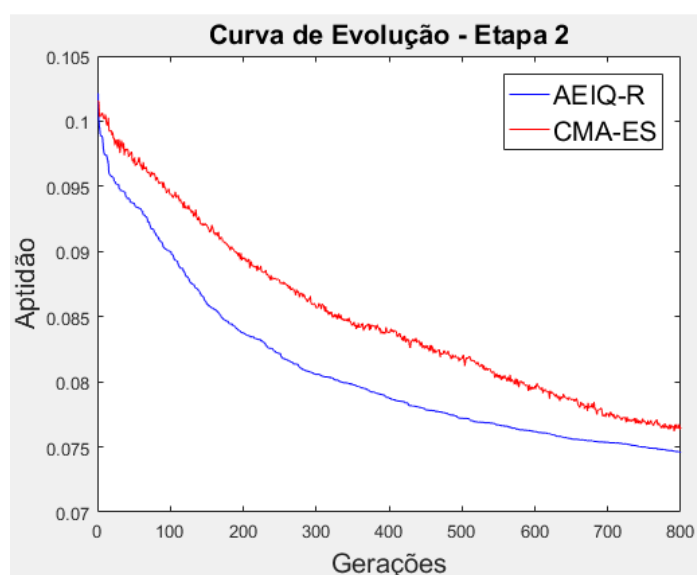


Figura 60 - Curva de evolução da etapa 2 (média da aptidão do melhor indivíduo em cada geração nos 10 experimentos) para o estudo de caso 2

Tabela 15 – Aptidão dos AEs na última geração (média de 10 experimentos) para o estudo de caso 2

Algoritmo	Aptidão na última geração (média de 10 experimentos)	
	Etapa 1	Etapa 2
AEIQ-R	0,110724	0,074572
CMA-ES	0,106414	0,076495

Pelas figuras acima, vê-se que o AEIQ-R obteve desempenho inferior na curva de evolução da etapa 1 e desempenho superior na evolução da etapa 2, ainda que nesta diferença tenha sido pequena em termos numéricos na última geração (0,074572 – AEIQ-R e 0,076495 – CMA-ES), embora ainda, visualmente, ambos algoritmos pareçam não ter convergido. Embora tenha obtido um desempenho superior na otimização da função de aptidão ao fim da etapa 2, as soluções encontradas pelo CMA-ES após a otimização da mesma obtiveram desempenho superior durante o teste, conforme anteriormente analisado.

A tabela 16 mostra a configuração de parâmetros globais que gerou as melhores soluções de cada submodelo (dentre as 10 soluções finais de cada experimento), enquanto que as figuras 61 a 66 mostram a simulação por cada uma dessas soluções dos conjuntos de treinamento, validação e teste. A figura 67 mostra os gráficos de razão de separação para cada uma dessas soluções.

Tabela 16 – Parâmetros globais da melhor solução encontrada por cada submodelo para o estudo de caso 2

Melhor solução	N	c_w	ρ_w	s_{in}	s_{back}	α
AEIQ-R/Random/ESN	23	0,7402	0,8143	0,3526	0,2564	0,7745
AEIQ-R/SRG/ESN	28	0,5009	0,7023	0,4812	0,4241	0,6388
AEIQ-R/AEIQ-R/ESN	24	0,6612	0,5320	0,3851	0,3290	0,6377
CMA-ES/Random/ESN	22	0,4321	0,1776	0,0539	0,0844	0,5812
CMA-ES/SRG/ESN	20	0,2515	0,1000	0,0147	0,0390	0,6060
CMA-ES/CMA-ES/ESN	21	0,2332	0,1248	0,0295	0,0352	0,6187

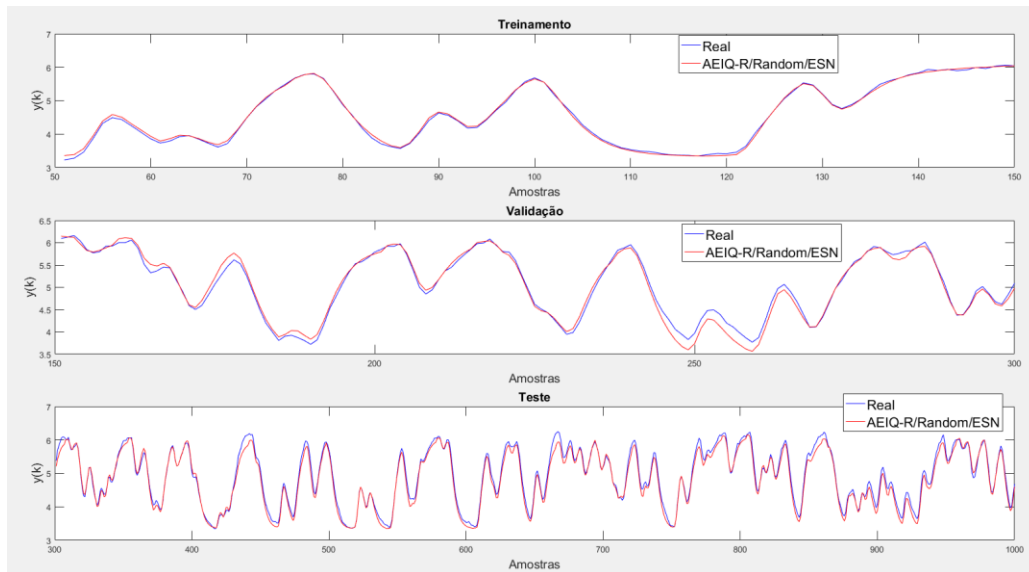


Figura 61 – Simulação da variável de saída y do estudo de caso 2 utilizando a melhor solução do submodelo AEIQ-R/Random/ESN

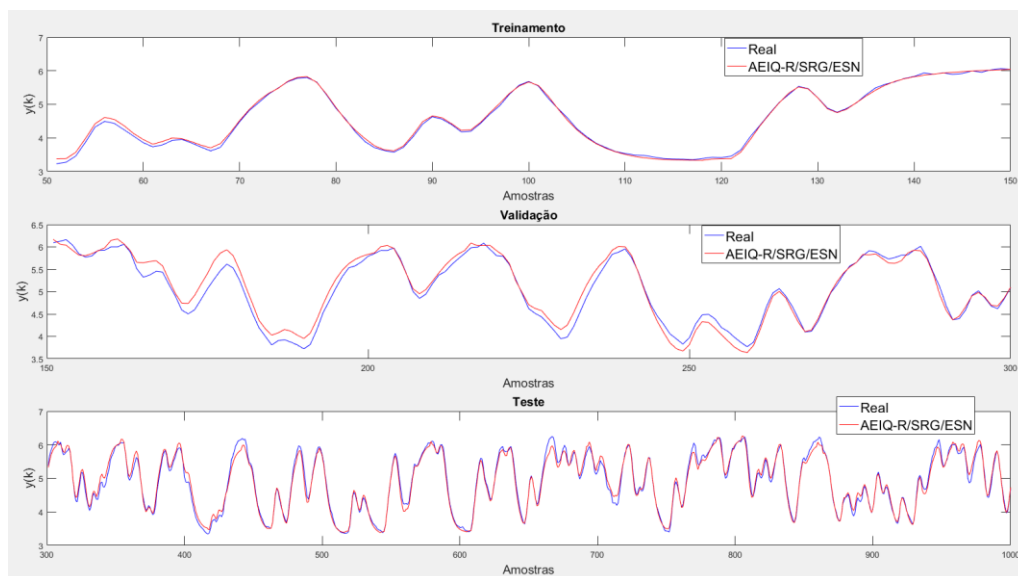


Figura 62 - Simulação da variável de saída y do estudo de caso 2 utilizando a melhor solução do submodelo AEIQ-R/SRG/ESN

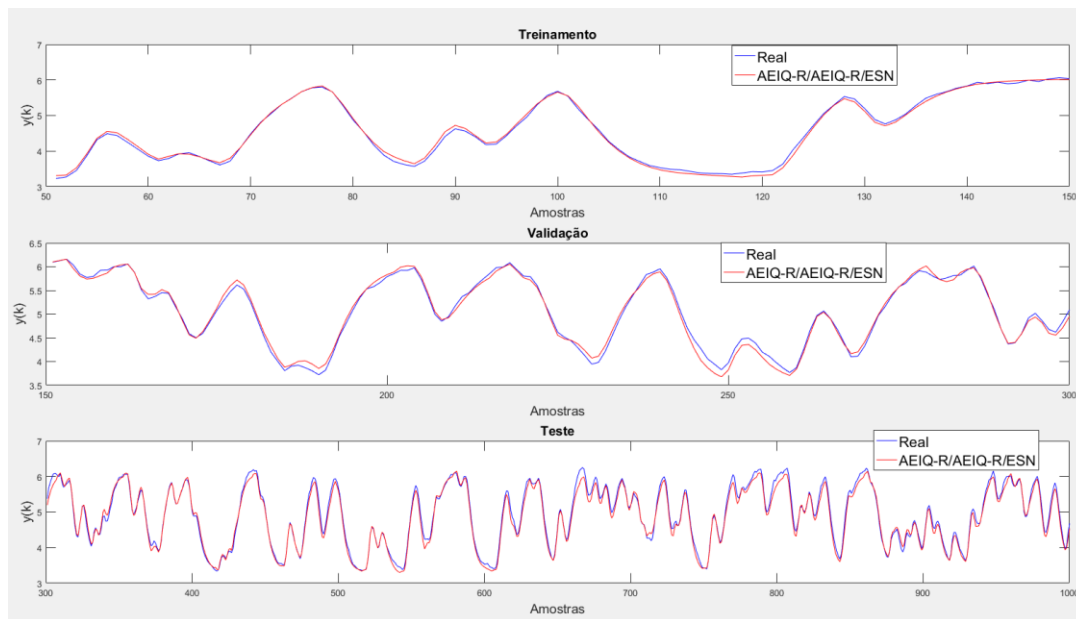


Figura 63 - Simulação da variável de saída y do estudo de caso 2 utilizando a melhor solução do submodelo AEIQ-R/AEIQ-R/ESN

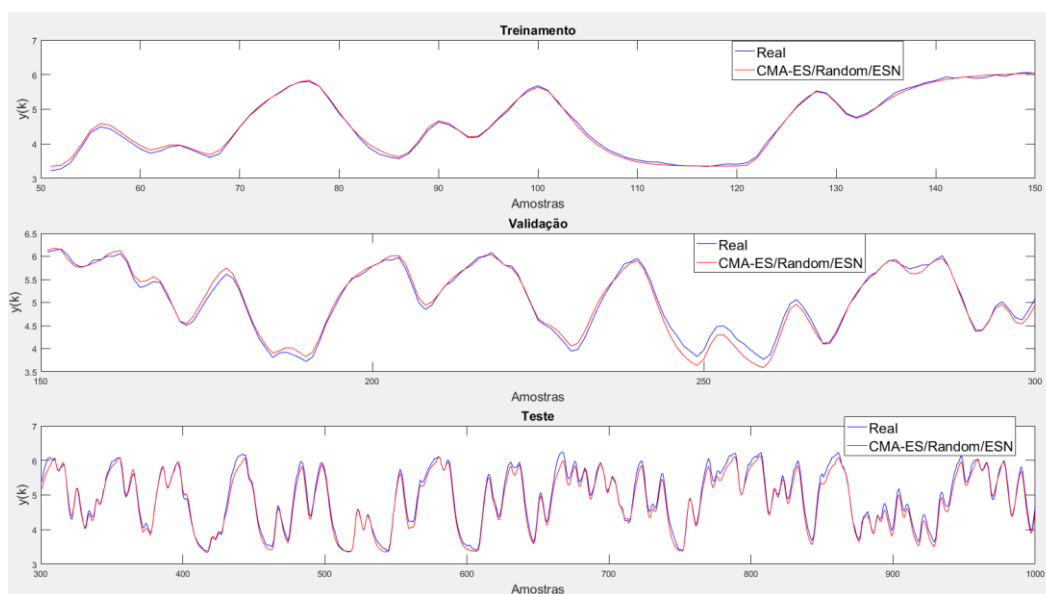


Figura 64 - Simulação da variável de saída y do estudo de caso 2 utilizando a melhor solução do submodelo CMA-ES/Random/ESN

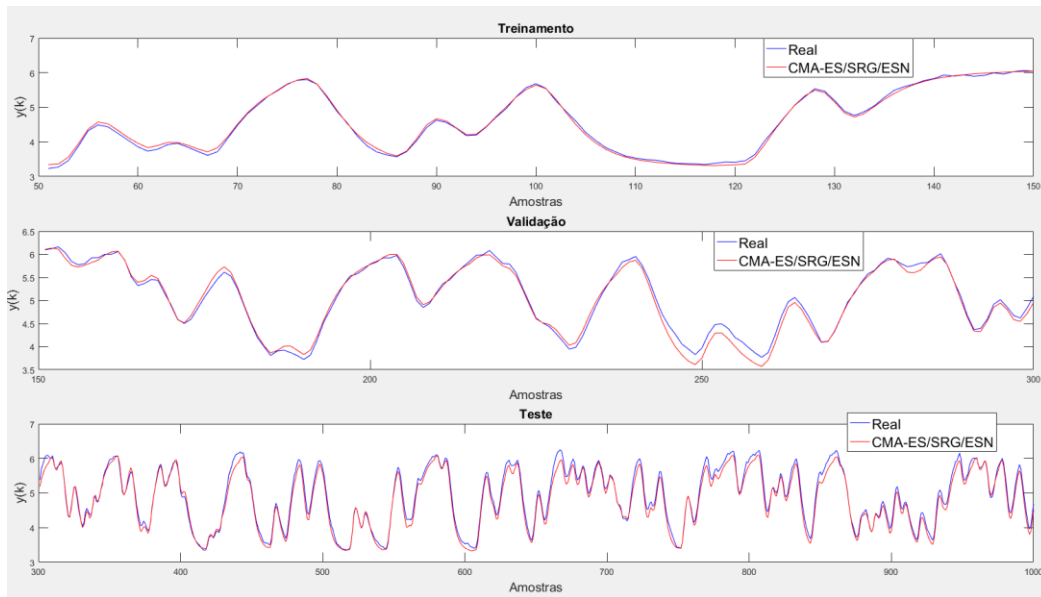


Figura 65 - Simulação da variável de saída y do estudo de caso 2 utilizando a melhor solução do submodelo CMA-ES/SRG/ESN

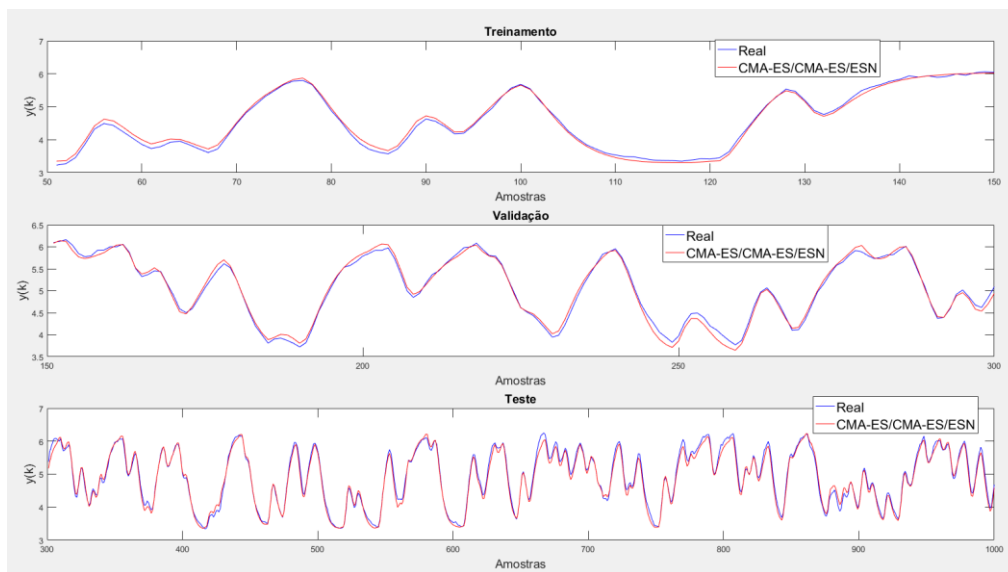


Figura 66 - Simulação da variável de saída y do estudo de caso 2 utilizando a melhor solução do submodelo CMA-ES/CMA-ES/ESN

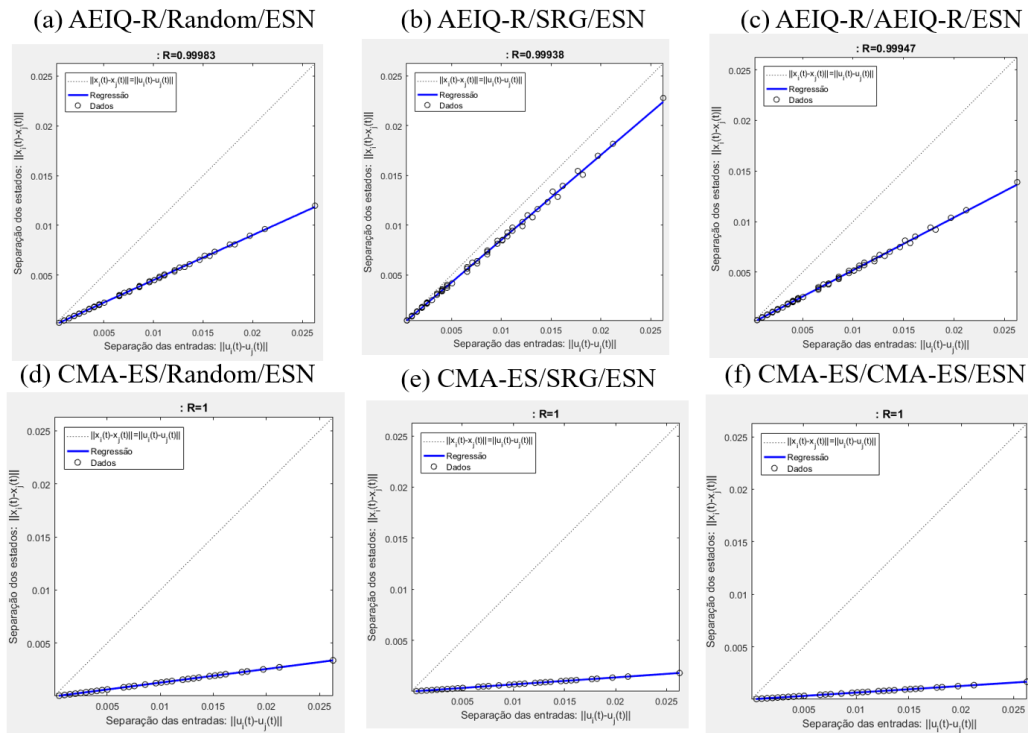


Figura 67 – Gráficos de razão de separação encontrados para as melhores soluções de cada submodelo para o estudo de caso 2

Como mostrado pelas figuras 61 a 66, as melhores soluções de cada submodelo dentre os 10 experimentos conseguiram simular adequadamente a variável de saída do sistema. Como mostrado anteriormente na tabela 14, dentre as melhores soluções de cada submodelo, os métodos AEIQ-R/AEIQ-R/ESN e CMA-ES/CMA-ES/ESN também apresentaram melhor acurácia nesta simulação (RMSEs de 0,112544 e 0,105634, respectivamente).

Comparando-se os gráficos de razão de separação da figura 66, pode-se ver que dos métodos derivados do AEIQ-R, as melhores soluções AEIQ-R/Random/ESN e AEIQ-R/AEIQ-R/ESN apresentaram gráficos similares, dentro da área atratora, com o deste último estando visualmente um pouco mais próximo da região objetivo. A melhor solução encontrada pelo método AEIQ-R/SRG/ESN está claramente com o gráfico mais próximo da região objetivo, e apesar disto, não apresenta o melhor desempenho entre as soluções derivadas do AEIQ-R. As soluções provenientes do CMA-ES, ficaram dentro da zona atratora, porém muito mais distantes da região objetivo.

A tabela 17 compara os resultados obtidos pelas melhores soluções dos submodelos do modelo geral deste trabalho com os melhores resultados dos métodos de Liu et al. (2013) e Hansson et al. (2012), que utilizaram as mesmas amostras de teste.

Tabela 17 – Comparação de resultados com outros métodos para o *benchmark Secador de cabelo*

Método	Métrica Fit
NOINSTR (Hansson et al., 2012)	90,2
Regularized Nuclear Norm (Liu et al., 2013)	88,6
CMA-ES/CMA-ES/ESN (melhor)	87,1
Weighted Nuclear Norm (Liu et al., 2013)	86,3
AEIQ-R/AEIQ-R/ESN (melhor)	86,3
AEIQ-R/SRG/ESN (melhor)	85,7
N4SID (Hansson et al., 2012)	85,0
CVA (Hansson et al., 2012)	84,9
IVM (Hansson et al., 2012)	84,9
MOESP (Hansson et al., 2012)	84,8
AEIQ-R/Random/ESN (melhor)	84,3
NONE (Hansson et al., 2012)	84,2
CMA-ES/Random/ESN (melhor)	84,1
CMA-ES/SRG/ESN (melhor)	83,4

Como mostrado pela tabela 17, a melhor solução deste trabalho, obtida com o modelo CMA-ES/CMA-ES/ESN figura como o 3º melhor em comparação aos métodos de Liu et al. (2013) e Hansson et al. (2012), não conseguindo superar os modelos NOINSTR (Hansson et al., 2012) e Regularized Nuclear Norm (Liu et al., 2013) em termos de *fit*. Uma possível explicação para a inferioridade dos modelos com ESN para este estudo de caso pode estar relacionado à natureza do sinal utilizado como entrada para excitar o sistema. Como mostrado pela figura 32 da seção 6.2.2, o sinal de entrada do sistema é de amplitude binária, com variações na largura dos pulsos que refletem em variações na frequência. Desta forma, a dinâmica excitada por este sinal é predominantemente linear, uma vez que variações na amplitude do sinal de entrada, importantes para evidenciar comportamentos não lineares, não foram muito exploradas. Uma possível evidência de que o sistema está operando em uma região linear pode ser enxergada analisando-se as melhores soluções derivadas do CMA-ES, que possuem fatores de escala da entrada e da realimentação com valores muito pequenos, o que indica

que os neurônios do reservatório estariam operando próximos da região linear. Estes valores pequenos de fatores de escala, também explicariam os gráficos de razão de separação obtidos pelos submodelos derivados do CMA-ES, uma vez que seriam gerados estados com valores absolutos pequenos sempre nesta situação. Desta forma, a ESN, um modelo não linear, poderia ter uma pequena queda de desempenho ao modelar relações lineares de entrada-saída.

Uma vez que frequentemente as RNA são caracterizadas por possuírem mais parâmetros que outros modelos e, portanto, necessitarem de mais dados para treinamento, uma outra questão que pode ter deteriorado o desempenho das ESNs, refere-se aos poucos dados disponíveis para se estimar o modelo, uma vez que 700 amostras foram utilizadas para comparar o desempenho durante o teste com Liu et al. (2013) e Hansson et al. (2012), restando-se apenas 300 para serem divididas entre treinamento e validação.

6.4.3.

Estudo de caso 3 – Densidade de fluxo de calor através de uma parede de duas camadas

A tabela 18 exhibe os resultados obtidos durante os 10 experimentos pelos diferentes submodelos derivados do método geral para o estudo de caso 3 – *Densidade de fluxo de calor através de uma parede de duas camadas*, com o AEIQ-R e o CMA-ES configurados conforme tabelas 7 e 8 e com a base de dados dividida como mostrado na tabela 6:

Tabela 18 – Resultados obtidos de RMSE pelos diferentes submodelos para o *benchmark Densidade de fluxo de calor através de uma parede de duas camadas* nos 10 experimentos realizados

Submodelo	Treinamento	Validação	Teste			
	média	média	Média	melhor	pior	desvio
AEIQ-R/Random/ESN	0,118167	0,152911	0,220903	0,185051	0,272107	0,028755
AEIQ-R/SRG/ESN	0,122683	0,180068	0,314525	0,219314	0,686093	0,139179
AEIQ-R/AEIQ-R/ESN	0,118783	0,126258	0,233609	0,168504	0,453815	0,083194
CMA-ES/Random/ESN	0,117937	0,149296	0,213938	0,185882	0,248120	0,022693
CMA-ES/SRG/ESN	0,119930	0,179283	0,309211	0,210304	0,451434	0,083423
CMA-ES/CMA-ES/ESN	0,121704	0,123116	0,240149	0,172672	0,439663	0,085985

Pelos resultados de RMSE médio durante o teste mostrados pela tabela 18, pode-se ver que os métodos AEIQ-R/AEIQ-R/ESN e CMA-ES/CMA-ES/ESN não conseguiram superar os resultados atingidos pelos métodos AEIQ-R/Random/ESN e CMA-ES/Random/ESN, que obtiveram melhor desempenho entre os submodelos testados, indicando que a otimização dos pesos do reservatório não funcionou para este estudo de caso. Pela mesma tabela, vê-se que as abordagens que utilizaram o método SRG na segunda etapa obtiveram pior desempenho dentre os submodelos testados. As figuras 68 e 69 mostram as curvas de evolução dos algoritmos AEIQ-R e CMA-ES nas etapas 1 e 2, exibindo-se a média da aptidão nos 10 experimentos do melhor indivíduo da população, enquanto a tabela 19 exibe os valores de aptidão na última geração:

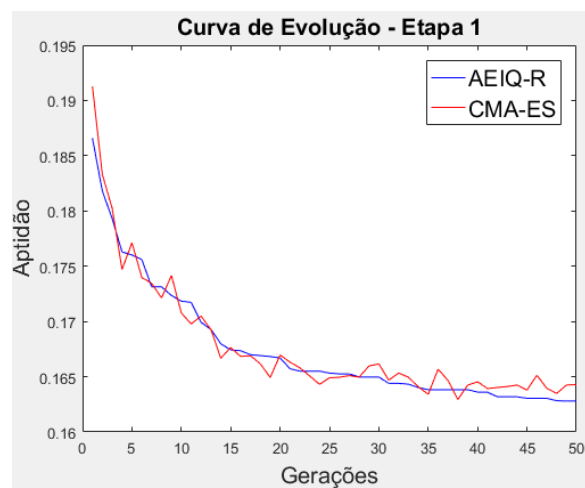


Figura 68– Curva de evolução da etapa 1 (média da aptidão do melhor indivíduo em cada geração nos 10 experimentos) para o estudo de caso 3

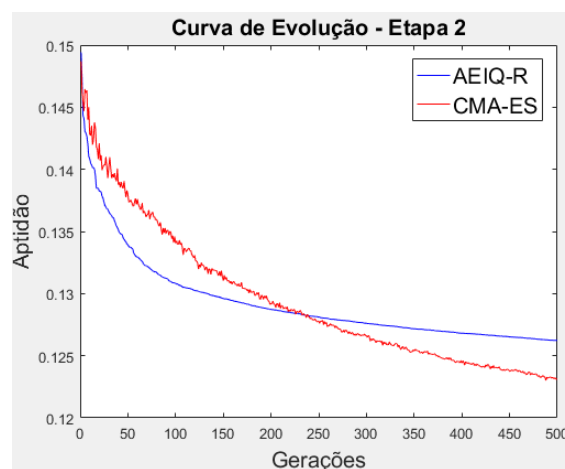


Figura 69 - Curva de evolução da etapa 2 (média da aptidão do melhor indivíduo em cada geração nos 10 experimentos) para o estudo de caso 3

Tabela 19 – Aptidão dos AEs na última geração (média de 10 experimentos) para o estudo de caso 3

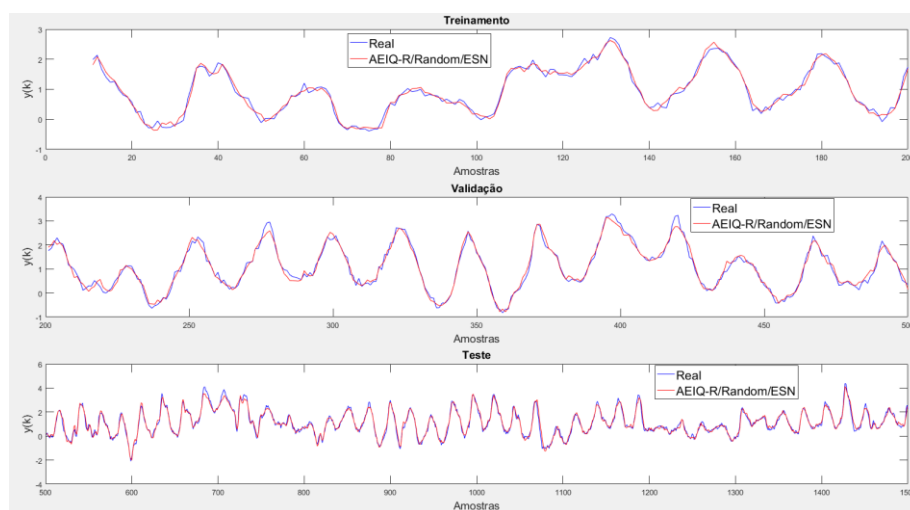
Algoritmo	Aptidão na última geração (média de 10 experimentos)	
	Etapa 1	Etapa 2
AEIQ-R	0,162797	0,126258
CMA-ES	0,164283	0,123116

Pelas figuras 68 e 69, vê-se que ambos AEs apresentaram desempenho similar durante o processo evolutivo da etapa 1, enquanto que na etapa 2, o CMA-ES conduziu melhor a evolução do que o AEIQ-R, ainda que as soluções CMA-ES/CMA-ES/ESN tenham sido em média inferiores às soluções AEIQ-R/AEIQ-R/ESN e ambos algoritmos pareçam não ter convergido.

A tabela 20 mostra a configuração de parâmetros globais que gerou as melhores soluções de cada submodelo (dentre as 10 soluções finais de cada experimento), enquanto que as figuras 70 a 75 mostram a simulação por cada uma dessas soluções dos conjuntos de treinamento, validação e teste. A figura 76 mostra os gráficos de razão de separação para cada uma dessas soluções.

Tabela 20 – Parâmetros globais da melhor solução encontrada por cada submodelo para o estudo de caso 3

Melhor solução	N	c_W	ρ_W	S_{in}	S_{back}	α
AEIQ-R/Random/ESN	23	0,5651	0,6600	0,5661	0,0924	0,4247
AEIQ-R/SRG/ESN	27	0,6536	0,5080	0,6996	0,2677	0,5066
AEIQ-R/AEIQ-R/ESN	23	0,5651	0,6600	0,5661	0,0924	0,4247
CMA-ES/Random/ESN	21	0,7328	0,6040	0,9218	0,1517	0,2790
CMA-ES/SRG/ESN	21	0,3069	0,6776	0,5920	0,0320	0,3967
CMA-ES/CMA-ES/ESN	20	0,7007	0,5688	0,8208	0,0694	0,3304

Figura 70 - Simulação da variável de saída y do estudo de caso 3 utilizando a melhor solução do submodelo AEIQ-R/Random/ESN

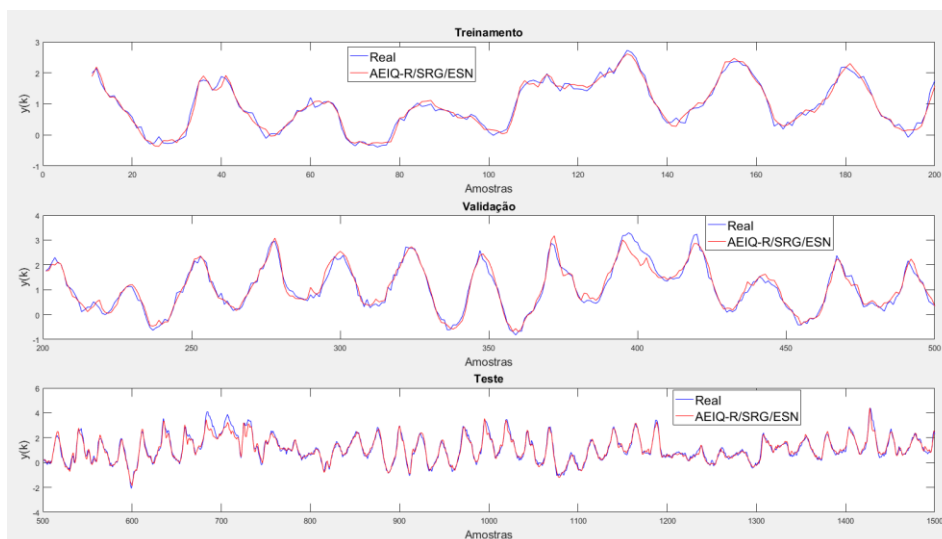


Figura 71 - Simulação da variável de saída y do estudo de caso 3 utilizando a melhor solução do submodelo AEIQ-R/SRG/ESN

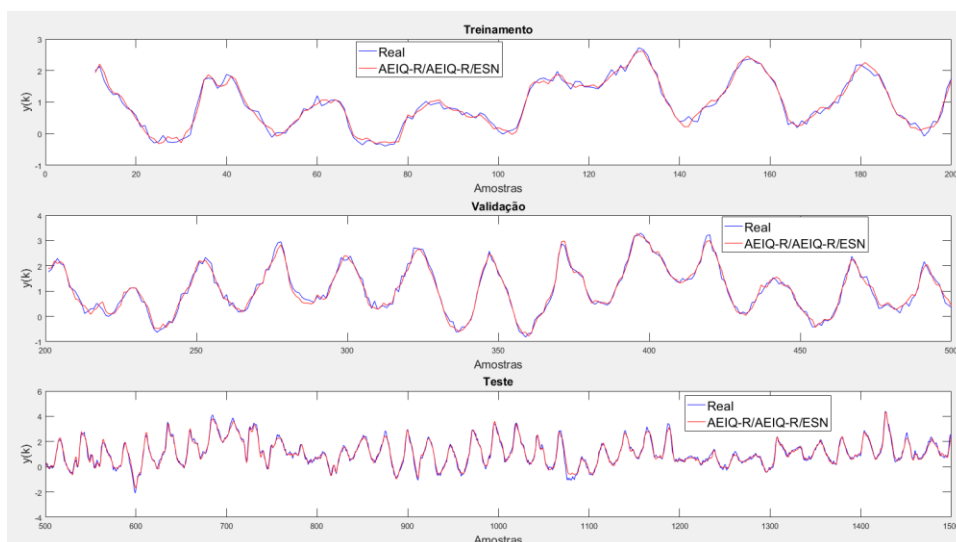


Figura 72 - Simulação da variável de saída y do estudo de caso 3 utilizando a melhor solução do submodelo AEIQ-R/AEIQ-R/ESN

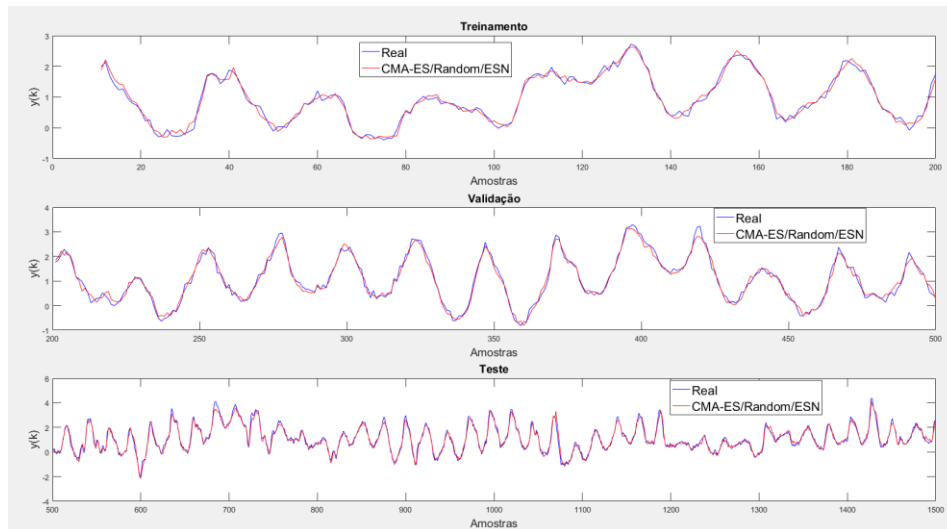


Figura 73 - Simulação da variável de saída y do estudo de caso 3 utilizando a melhor solução do submodelo CMA-ES/Random/ESN

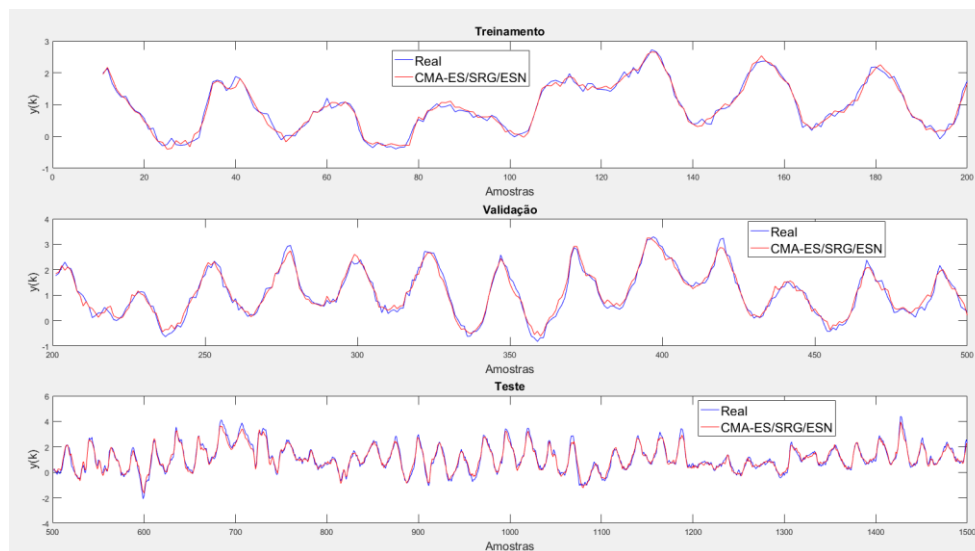


Figura 74 - Simulação da variável de saída y do estudo de caso 3 utilizando a melhor solução do submodelo CMA-ES/SRG/ESN

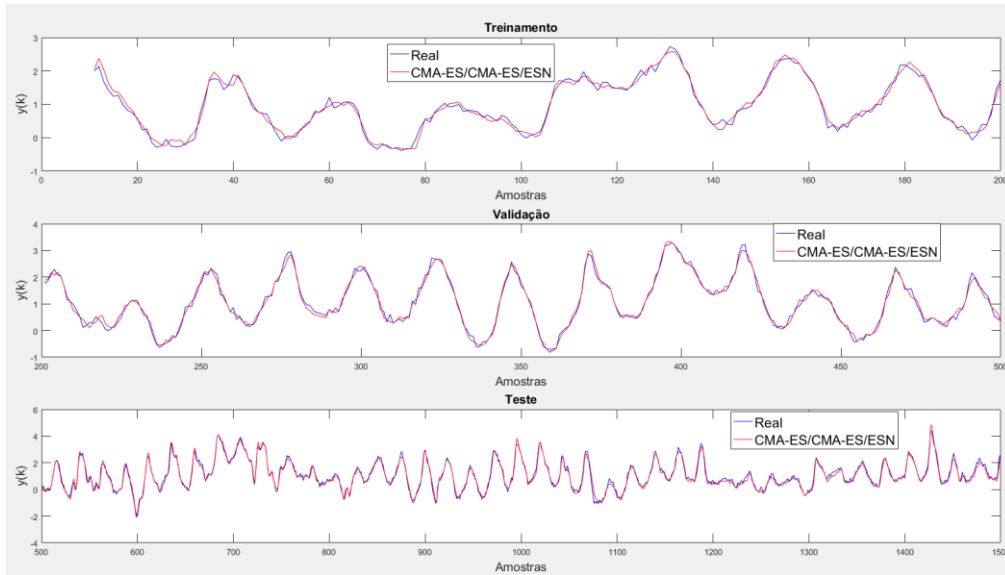


Figura 75 - Simulação da variável de saída y do estudo de caso 3 utilizando a melhor solução do submodelo CMA-ES/CMA-ES/ESN

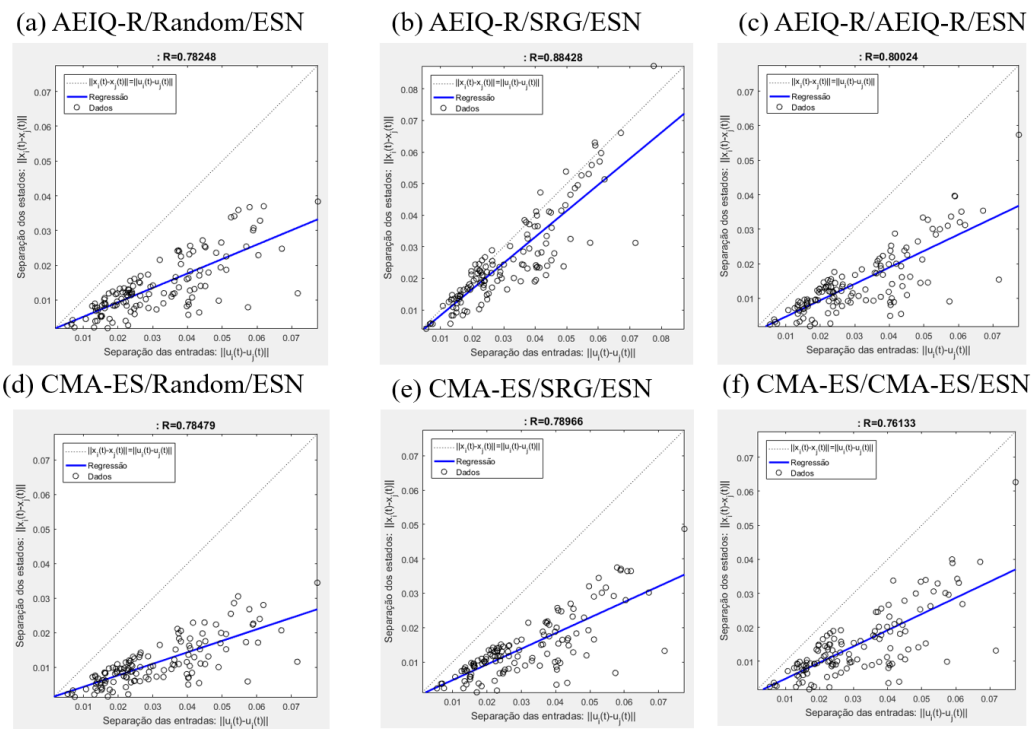


Figura 76 – Gráficos de razão de separação encontrados para as melhores soluções de cada submodelo para o estudo de caso 3

Como mostrado pelas figuras 70 a 75, as melhores soluções de cada submodelo dentre os 10 experimentos conseguiram simular adequadamente a variável de saída do sistema. Ainda que, em média, o desempenho dos métodos CMA-ES/CMA-ES/ESN e AEIQ-R/AEIQ-R/ESN não tenha sido superior aos demais, suas melhores soluções, dentre os 10 experimentos, retornaram os menores valores de erro durante o teste

(RMSEs de 0,172672 e 0,168504, respectivamente) do que as melhores soluções dos demais modelos. Uma possível hipótese para o desempenho inferior de um reservatório otimizado frente a um aleatório pode ser devido a uma espécie de *overfitting* ao se minimizar demais o erro de validação neste estudo de caso. Para tentar verificar esta hipótese, repetiu-se o processo de otimização partindo-se das mesmas soluções encontradas na etapa 1, mas com um número menor de gerações (100). Os resultados deste experimento extra, mostrados na tabela 21, fariam com que os submodelos AEIQ-R/AEIQ-R/ESN e CMA-ES/CMA-ES/ESN atingissem os melhores resultados entre os submodelos testados, com valores médios de RMSE de 0,189200 e 0,180384, respectivamente.

Tabela 21 – Resultados para o experimento extra para o estudo de caso 3 com 100 gerações na etapa 2

Submodelo	Treinamento	Validação	Teste			
	média	média	média	melhor	pior	desvio
AEIQ-R/AEIQ-R/ESN	0,119152	0,130171	0,189200	0,168861	0,226607	0,018869
CMA-ES/CMA-ES/ESN	0,117983	0,134263	0,180384	0,160248	0,201166	0,014333

Dos gráficos de razão de separação da figura 76, destaca-se o do submodelo AEIQ-R/SRG/ESN, que apresentou o reservatório com comportamento mais próximo do ideal, ainda que este método não tenha sido o de melhor desempenho.

A tabela 22 compara os resultados obtidos pelas melhores soluções dos submodelos (sem considerar o experimento extra) com os melhores resultados dos métodos de Liu et al. (2013) e Hansson et al. (2012), que utilizaram as mesmas amostras de teste.

Tabela 22 – Comparação de resultados com outros métodos para o *benchmark Densidade de fluxo de calor através de uma parede de duas camadas*

Método	Métrica Fit
Regularized Nuclear Norm (Liu et al., 2013)	86,7
Weighted Nuclear Norm (Liu et al., 2013)	86,7
CVA (Hansson et al., 2012)	86,2
IVM (Hansson et al., 2012)	86,2
NOINSTR (Hansson et al., 2012)	85,1

Método	Métrica
	Fit
MOESP (Hansson et al., 2012)	84,8
NONE (Hansson et al., 2012)	84,8
N4SID (Hansson et al., 2012)	83,9
AEIQ-R/AEIQ-R/ESN (melhor)	83,6
CMA-ES/CMA-ES/ESN (melhor)	83,2
AEIQ-R/Random/ESN (melhor)	82,0
CMA-ES/Random/ESN (melhor)	81,9
CMA-ES/SRG/ESN (melhor)	79,5
AEIQ-R/SRG/ESN (melhor)	78,7

Como mostrado pela tabela 22, todos as melhores soluções dos submodelos desta dissertação apresentaram desempenho inferior aos métodos de Liu et al. (2013) e Hansson et al. (2012). Uma possibilidade para esse desempenho inferior pode estar na divisão dos dados, que deixa mais amostras para teste (1000) do que para estimação (500) dos modelos, o que pode ser uma desvantagem para as ESNs e uma vantagem para os métodos de Liu et al. (2013) e Hansson et al. (2012). Esta questão voltará a ser discutida na seção 6.4.8.

6.4.4. **Estudo de caso 4 – Braço robótico flexível**

A tabela 23 exhibe os resultados obtidos durante os 10 experimentos pelos diferentes submodelos derivados do método geral para o estudo de caso 4 – *Braço robótico flexível*, com o AEIQ-R e o CMA-ES configurados conforme tabelas 7 e 8 e com a base de dados dividida como mostrado na tabela 6:

Tabela 23 – Resultados obtidos de RMSE pelos diferentes submodelos para o *benchmark Braço robótico flexível* nos 10 experimentos realizados

Submodelo	Treinamento	Validação	Teste			
	Média	Média	média	melhor	pior	desvio
AEIQ-R/Random/ESN	0,118367	0,0430031	0,046168	0,032082	0,086773	0,016241
AEIQ-R/SRG/ESN	73,440947	78,410775	78,4463	0,032606	782,94716	247,53649
AEIQ-R/AEIQ-R/ESN	0,067696	0,005374	0,024391	0,016050	0,032857	0,004588
CMA-ES/Random/ESN	0,113276	0,053482	0,071691	0,038975	0,119122	0,025515
CMA-ES/SRG/ESN	0,250034	0,302367	13,468504	0,053357	128,8476	40,5722
CMA-ES/CMA-ES/ESN	0,047003	0,009030	0,022260	0,013779	0,029793	0,005630

Comparando-se os resultados de simulação do conjunto de teste mostrados na tabela 23, os métodos AEIQ-R/AEIQ-R/ESN e CMA-ES/CMA-ES/ESN foram, em média, os de melhor desempenho. Pela mesma tabela, pode-se ver que os submodelos AEIQ-R/SRG/ESN e CMA-ES/SRG/ESN foram os de pior desempenho durante o teste, uma vez que geraram algumas soluções instáveis, como pode ser visto nos valores altíssimos de RMSE de suas piores soluções nos 10 experimentos. As figuras 77 e 78 mostram as curvas de evolução dos algoritmos AEIQ-R e CMA-ES nas etapas 1 e 2, exibindo-se a média da aptidão nos 10 experimentos do melhor indivíduo da população, enquanto a tabela 24 exhibe os valores de aptidão na última geração:

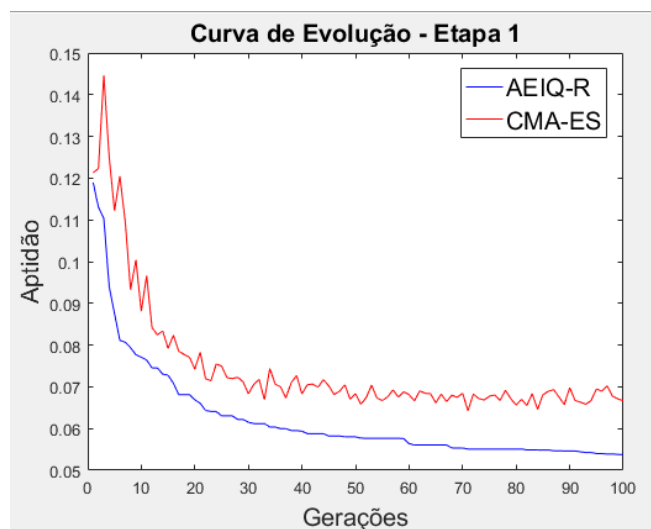


Figura 77 - Curva de evolução da etapa 1 (média da aptidão do melhor indivíduo em cada geração nos 10 experimentos) para o estudo de caso 4

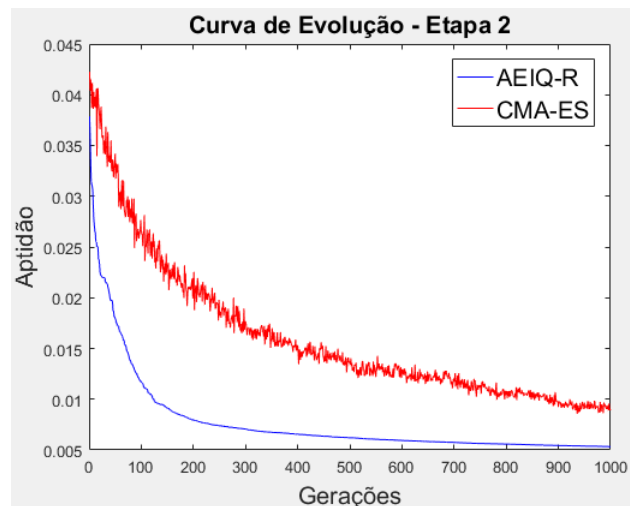


Figura 78 - Curva de evolução da etapa 2 (média da aptidão do melhor indivíduo em cada geração nos 10 experimentos) para o estudo de caso 4

Tabela 24 - Aptidão dos AEs na última geração (média de 10 experimentos) para o estudo de caso 4

Algoritmo	Aptidão na última geração (média de 10 experimentos)	
	Etapa 1	Etapa 2
AEIQ-R	0,053788	0,005374
CMA-ES	0,066701	0,009030

Pelas figuras 77 e 78, vê-se que em ambas etapas, o AEIQ-R conduziu melhor a evolução do que o CMA-ES, ainda que as soluções AEIQ-R/AEIQ-R/ESN tenham sido em média inferiores às soluções CMA-ES/CMA-ES/ESN.

A tabela 25 mostra a configuração de parâmetros globais que gerou as melhores soluções de cada submodelo (dentre as 10 soluções finais de cada experimento), enquanto que as figuras 79 a 84 mostram a simulação por cada uma dessas soluções dos conjuntos de treinamento, validação e teste. A figura 85 mostra os gráficos de razão de separação para cada uma dessas soluções.

Tabela 25 - Parâmetros globais da melhor solução encontrada por cada submodelo para o estudo de caso 4

Melhor solução	N	c_w	ρ_w	s_{in}	s_{back}	α
AEIQ-R/Random/ESN	54	0,9312	0,7447	0,4685	0,0785	0,5872
AEIQ-R/SRG/ESN	85	0,3866	0,5965	0,5133	0,0490	0,5401
AEIQ-R/AEIQ-R/ESN	76	0,5991	0,7906	0,4305	0,1581	0,7129
CMA-ES/Random/ESN	104	0,7866	0,2797	0,3590	0,0061	0,4334
CMA-ES/SRG/ESN	63	0,5430	0,7384	0,4748	0,0620	0,6663
CMA-ES/CMA-ES/ESN	55	0,5075	0,8413	0,3559	0,0751	0,7227

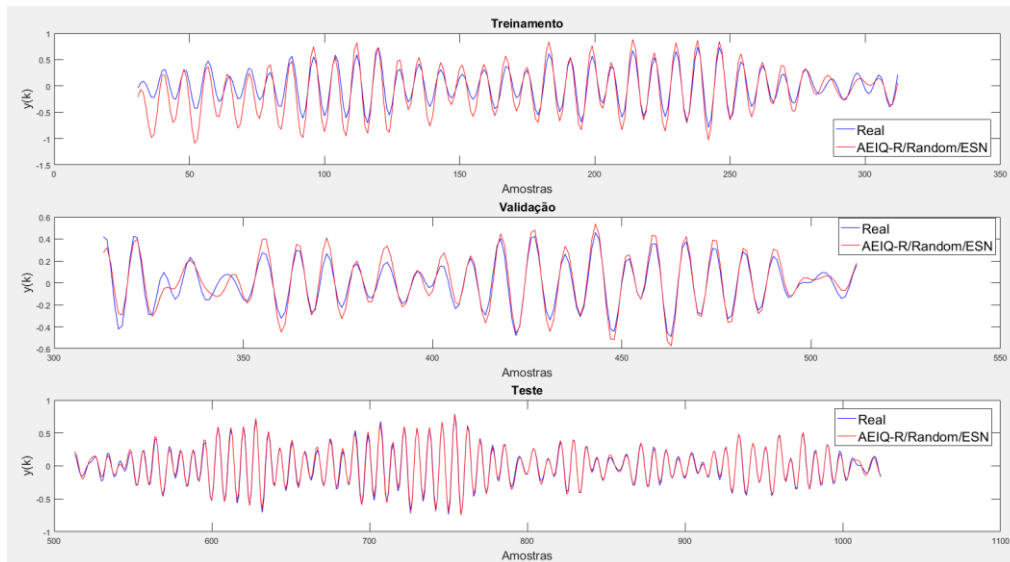


Figura 79 - Simulação da variável de saída y do estudo de caso 4 utilizando a melhor solução do submodelo AEIQ-R/Random/ESN

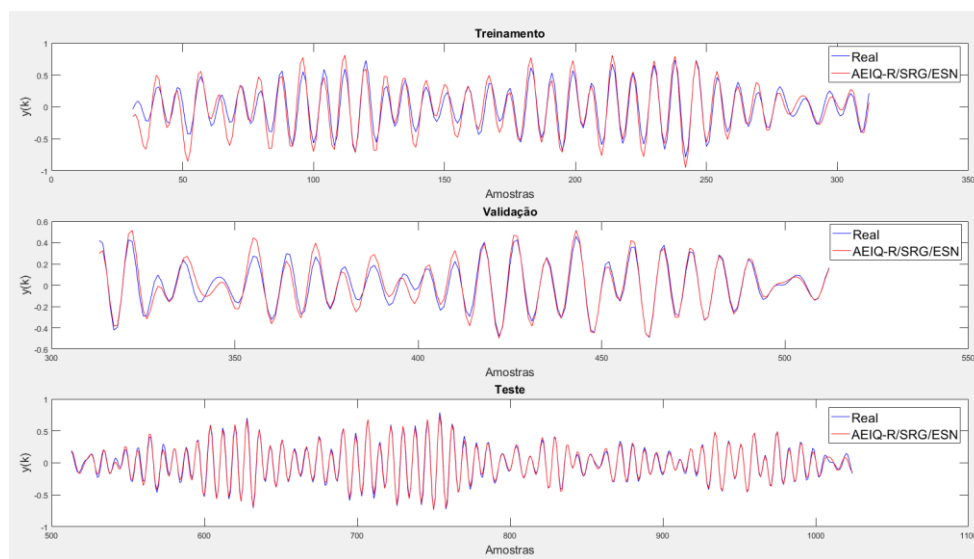


Figura 80 - Simulação da variável de saída y do estudo de caso 4 utilizando a melhor solução do submodelo AEIQ-R/SRG/ESN

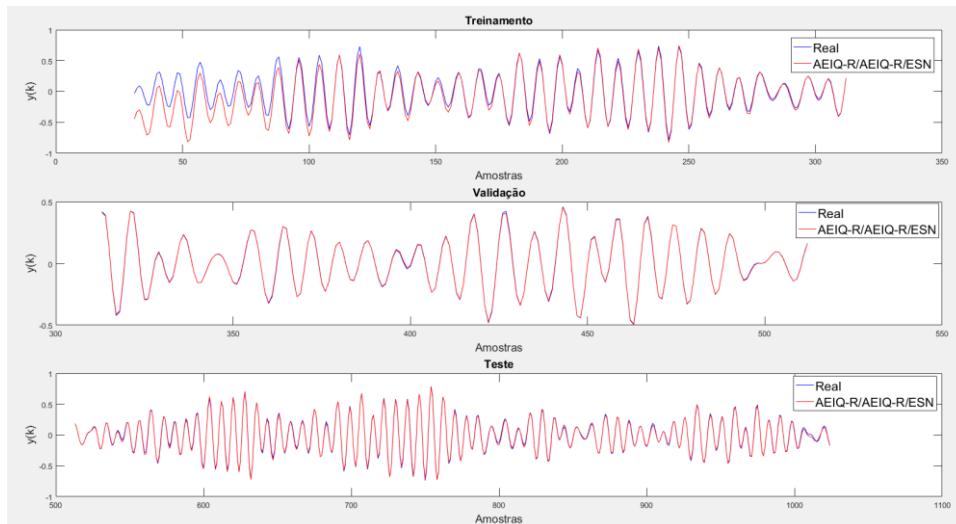


Figura 81 - Simulação da variável de saída y do estudo de caso 4 utilizando a melhor solução do submodelo AEIQ-R/AEIQ-R/ESN

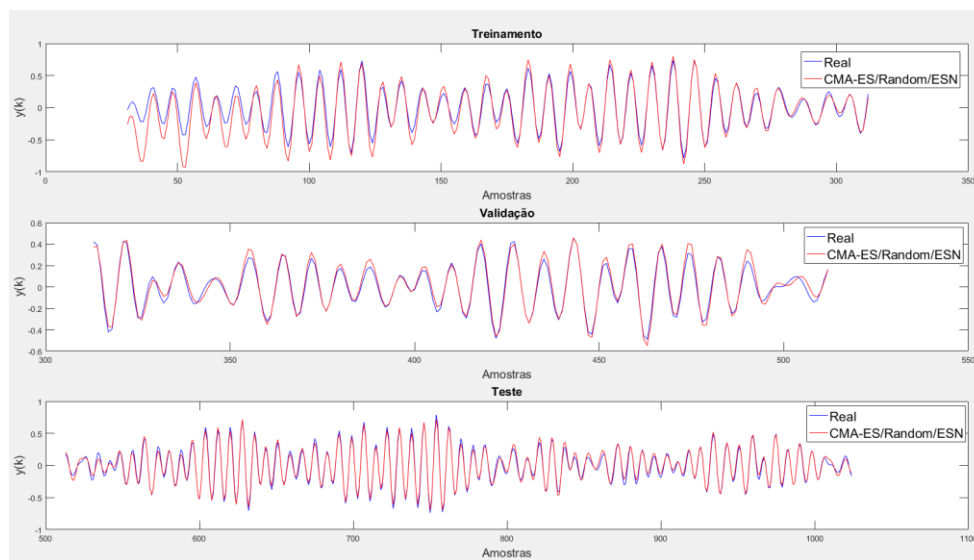


Figura 82 - Simulação da variável de saída y do estudo de caso 4 utilizando a melhor solução do submodelo CMA-ES/Random/ESN

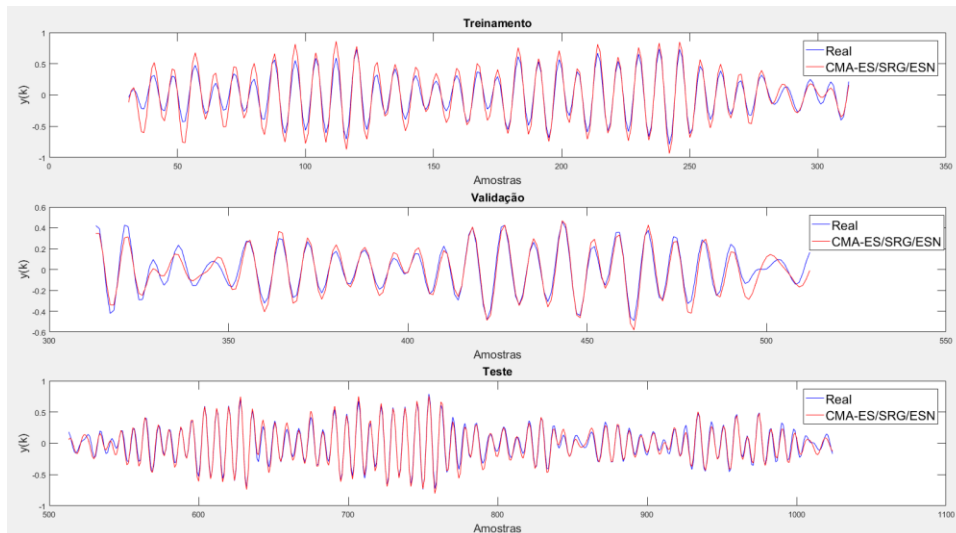


Figura 83 - Simulação da variável de saída y do estudo de caso 4 utilizando a melhor solução do submodelo CMA-ES/SRG/ESN

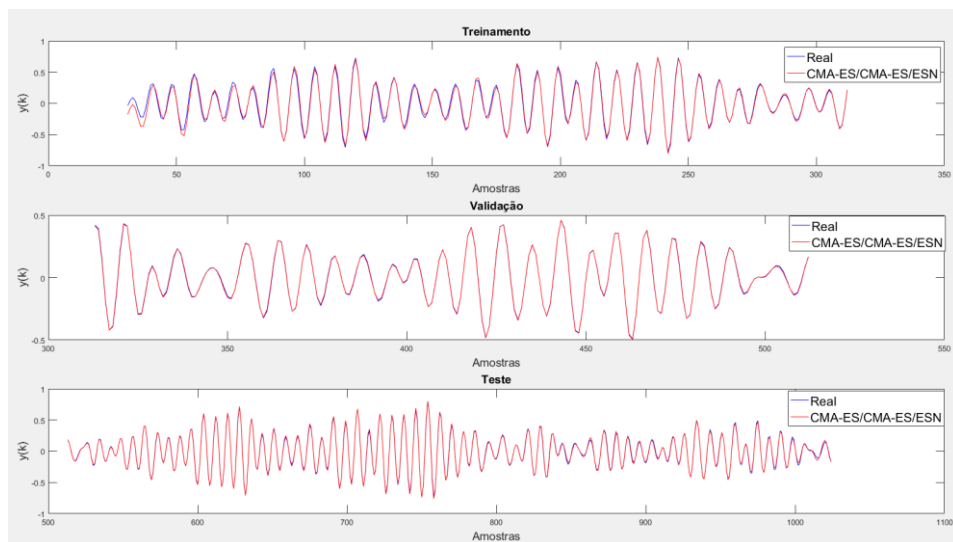


Figura 84 - Simulação da variável de saída y do estudo de caso 4 utilizando a melhor solução do submodelo CMA-ES/CMA-ES/ESN

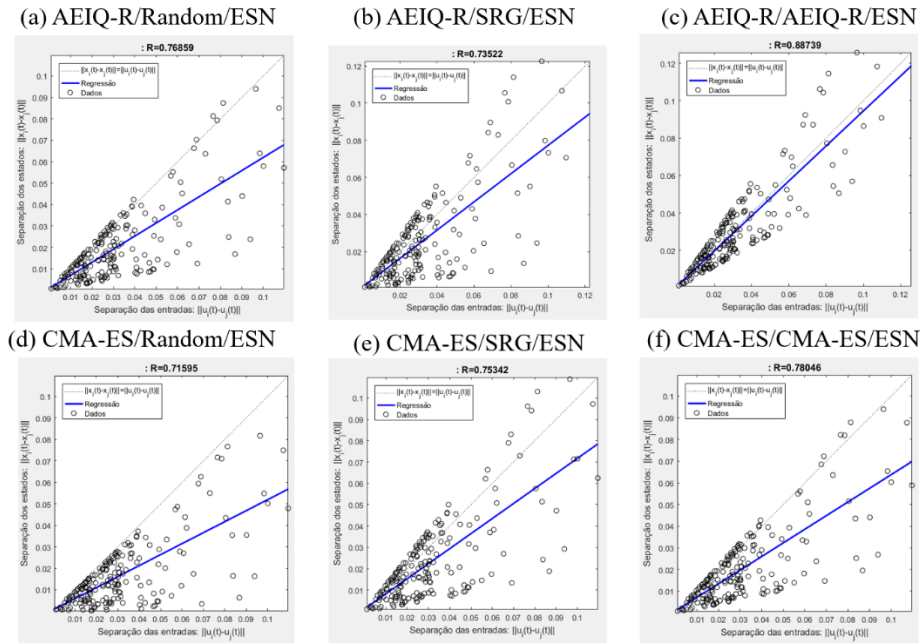


Figura 85 - Gráficos de razão de separação encontrados para as melhores soluções de cada submodelo para o estudo de caso 4

Como mostrado pelas figuras 79 a 84, as melhores soluções dos submodelos CMA-ES/CMA-ES/ESN e AEIQ-R/AEIQ-R/ESN conseguiram simular de maneira mais adequada e precisa a saída do sistema do que as demais soluções.

Dos gráficos de razão de separação da figura 85, destaca-se o do submodelo AEIQ-R/AEIQ-R/ESN, que apresentou o reservatório com comportamento mais próximo do ideal, ainda que esta solução não tenha superado o desempenho da solução CMA-ES/CMA-ES/ESN, cujo reservatório apresenta um comportamento mais distante da região objetivo.

A tabela 26 compara os resultados obtidos das abordagens testadas (sem considerar os métodos AEIQ-R/SRG/ESN e CMA-ES/SRG/ESN com os resultados dos métodos de Azmi et al. (2015) e Schumacher & Oliveira (2015), que utilizaram as mesmas amostras de teste, onde a métrica MSE é definida como:

$$MSE = \sum_{k=1}^{N_d} \frac{(y(k) - \hat{y}(k))^2}{N_d} \quad (92)$$

onde N_d representa o número de amostras.

Tabela 26 - Comparação de resultados com outros métodos para o *benchmark Flexible Arm*

Método	Métrica
	Fit
SMOPE-GSA (Azmi et al., 2015)	92,9
CMA-ES/CMA-ES/ESN (média)	91,8
AEIQ-R/AEIQ-R/ESN (média)	91,0
AEIQ-R/Random/ESN (média)	83,0
CMA-ES/Random/ESN (média)	73,7
	MSE
CMA-ES/CMA-ES/ESN (melhor)	0,0002
AEIQ-R/AEIQ-R/ESN (melhor)	0,0003
AEIQ-R/Random/ESN (melhor)	0,0010
CMA-ES/Random/ESN (melhor)	0,0015
zVF-DT (Schumacher & Oliveira, 2015)	0,0042
zOVF-DT (Schumacher & Oliveira, 2015)	0,0042
VF-DT usando CR (Schumacher & Oliveira, 2015)	0,0588
VF-DT usando IT (Schumacher & Oliveira, 2015)	0,0695

Como mostrado pela tabela 26, ainda que os resultados dos submodelos testados nesta dissertação não tenham superado os resultados do método SMOPE-GSA de Azmi et al. (2015), o valor médio de *fit* em 10 experimentos das abordagens AEIQ-R/AEIQ-R/ESN e CMA-ES/CMA-ES/ESN supera 90,0 %, o que indica que as soluções geradas por estes métodos conseguiram simular de maneira satisfatória o comportamento dinâmico do sistema durante o teste. Em termos de MSE, as melhores soluções dos métodos AEIQ-R/Random/ESN, AEIQ-R/AEIQ-R/ESN, CMA-ES/CMA-ES/ESN superaram os resultados obtidos pelos 4 métodos de Schumacher & Oliveira (2015).

6.4.5. Estudo de caso 5 – Reator tanque de agitação contínua

A tabela 27 exibe os resultados obtidos durante os 10 experimentos pelos diferentes submodelos derivados do método geral para o estudo de caso 5 – *Reator tanque de agitação contínua*, com o AEIQ-R e o CMA-ES configurados conforme tabelas 7 e 8 e com a base de dados dividida como mostrado na tabela 6:

Tabela 27– Resultados obtidos de RMSE pelos diferentes submodelos para o *benchmark Reator tanque de agitação contínua* nos 10 experimentos realizados

Submodelo	Treinamento	Validação	Teste			
	Média	média	média	melhor	Pior	desvio
AEIQ-R/Random/ESN	18,8125	0,010485	0,020766	0,016727	0,032944	0,004699
AEIQ-R/SRG/ESN	10,268014	8,428160	8,420741	0,018163	50,3930	16,3198
AEIQ-R/AEIQ-R/ESN	5,194993	0,006171	0,015463	0,013060	0,018979	0,002022
CMA-ES/Random/ESN	0,015946	0,014303	0,034474	0,016393	0,096019	0,025974
CMA-ES/SRG/ESN	9,639223	9,412652	9,190147	0,027354	36,6394	14,2728
CMA-ES/CMA-ES/ESN	2,210326	0,008941	0,018136	0,014900	0,026651	0,003432

Pelos resultados de RMSE médio durante o teste mostrados pela tabela 27, pode-se ver que os métodos AEIQ-R/AEIQ-R/ESN e CMA-ES/CMA-ES/ESN foram os que apresentaram melhor desempenho entre os testados. Novamente, as abordagens AEIQ-R/SRG/ESN e CMA-ES/SRG/ESN geraram soluções instáveis, apresentando os maiores valores de RMSE. As figuras 86 e 87 mostram as curvas de evolução dos algoritmos AEIQ-R e CMA-ES nas etapas 1 e 2, exibindo-se a média da aptidão nos 10 experimentos do melhor indivíduo da população, enquanto a tabela 28 exibe os valores de aptidão na última geração:

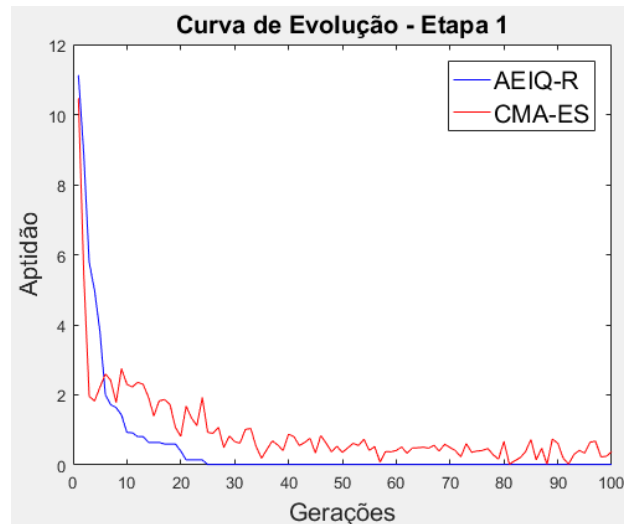


Figura 86 - Curva de evolução da etapa 1 (média da aptidão do melhor indivíduo em cada geração nos 10 experimentos) para o estudo de caso 5

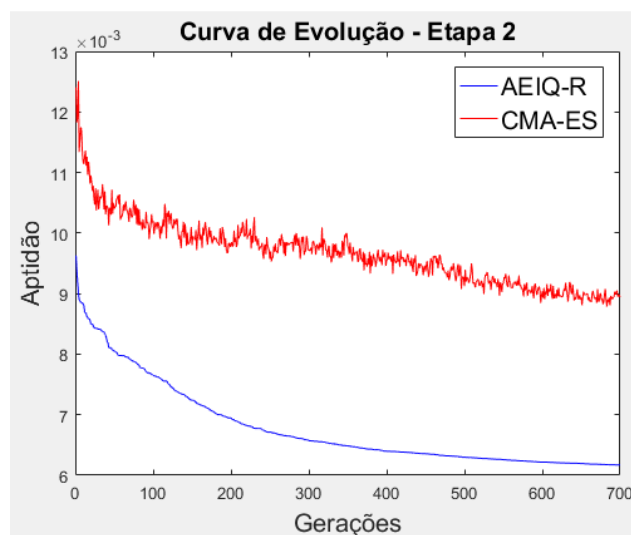


Figura 87 - Curva de evolução da etapa 2 (média da aptidão do melhor indivíduo em cada geração nos 10 experimentos) para o estudo de caso 5

Tabela 28 - Aptidão dos AEs na última geração (média de 10 experimentos) para o estudo de caso 5

Algoritmo	Aptidão na última geração (média de 10 experimentos)	
	Etapa 1	Etapa 2
AEIQ-R	0,011456	0,006171
CMA-ES	0,378984	0,008941

Como mostrado pelas figuras 86 e 87, em ambas as etapas, o AEIQ-R apresentou resultados superiores ao CMA-ES durante a evolução, embora ainda, visualmente, ambos algoritmos pareçam não ter convergido. Neste caso, tal superioridade refletiu-se em

soluções provenientes do AEIQ-R com melhor desempenho do que as soluções geradas pelo CMA-ES.

A tabela 29 mostra a configuração de parâmetros globais que gerou as melhores soluções de cada submodelo (dentre as 10 soluções finais de cada experimento), enquanto que as figuras 88 a 99 mostram a simulação por cada uma dessas soluções dos conjuntos de treinamento, validação e teste. A figura 100 mostra os gráficos de razão de separação para cada uma dessas soluções.

Tabela 29 - Parâmetros globais da melhor solução encontrada por cada submodelo para o estudo de caso 5

Melhor solução	N	c_w	ρ_w	s_{in}	s_{back}	α
AEIQ-R/Random/ESN	100	0,7742	0,7747	0,9593	0,4756	0,8818
AEIQ-R/SRG/ESN	112	0,4442	0,6908	0,7327	0,6869	0,7954
AEIQ-R/AEIQ-R/ESN	186	0,6644	0,8537	0,7098	0,6682	0,9001
CMA-ES/Random/ESN	113	0,6653	0,9900	0,8984	0,6068	0,9816
CMA-ES/SRG/ESN	124	0,5844	0,9900	0,8850	0,3545	0,9494
CMA-ES/CMA-ES/ESN	94	0,4103	0,8559	0,9529	0,0741	0,9827

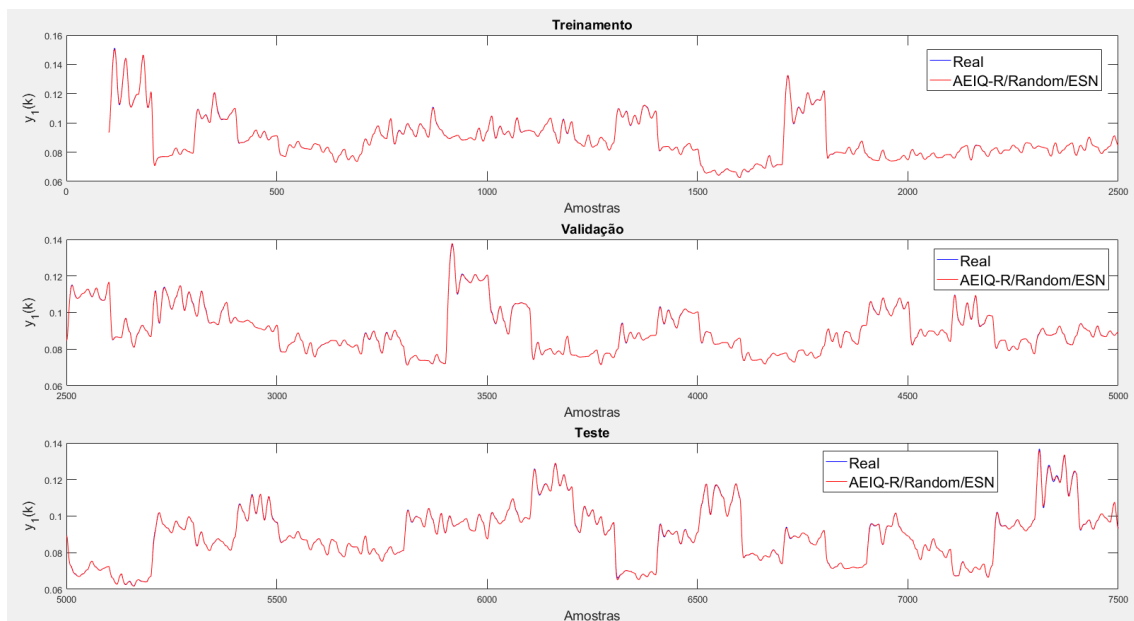


Figura 88 - Simulação da variável de saída y_1 do estudo de caso 5 utilizando a melhor solução do submodelo AEIQ-R/Random/ESN

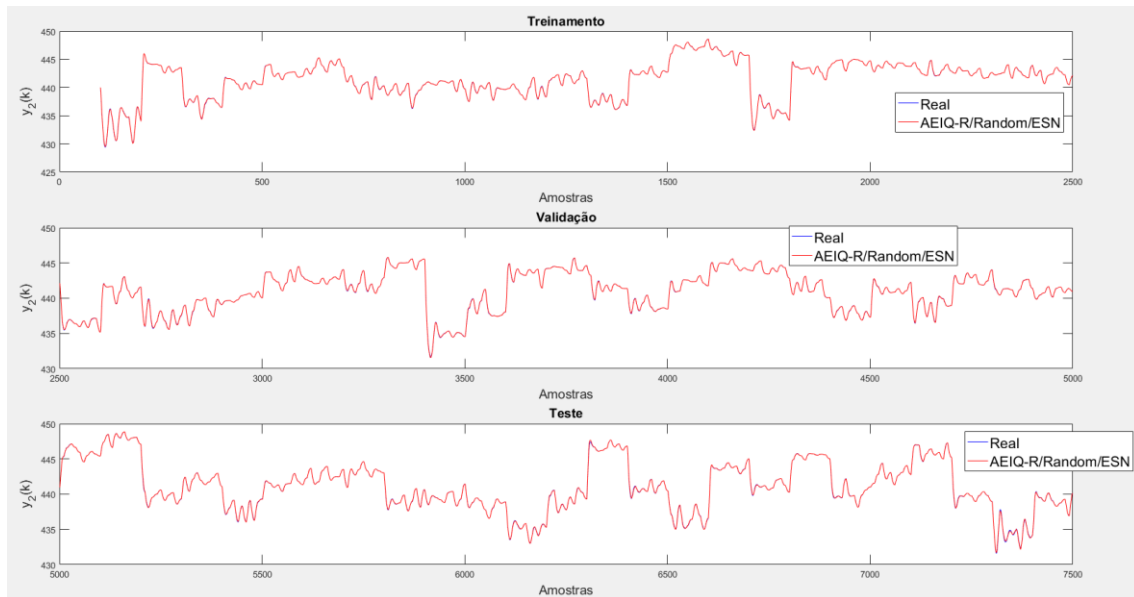


Figura 89 - Simulação da variável de saída y_2 do estudo de caso 5 utilizando a melhor solução do submodelo AEIQ-R/Random/ESN

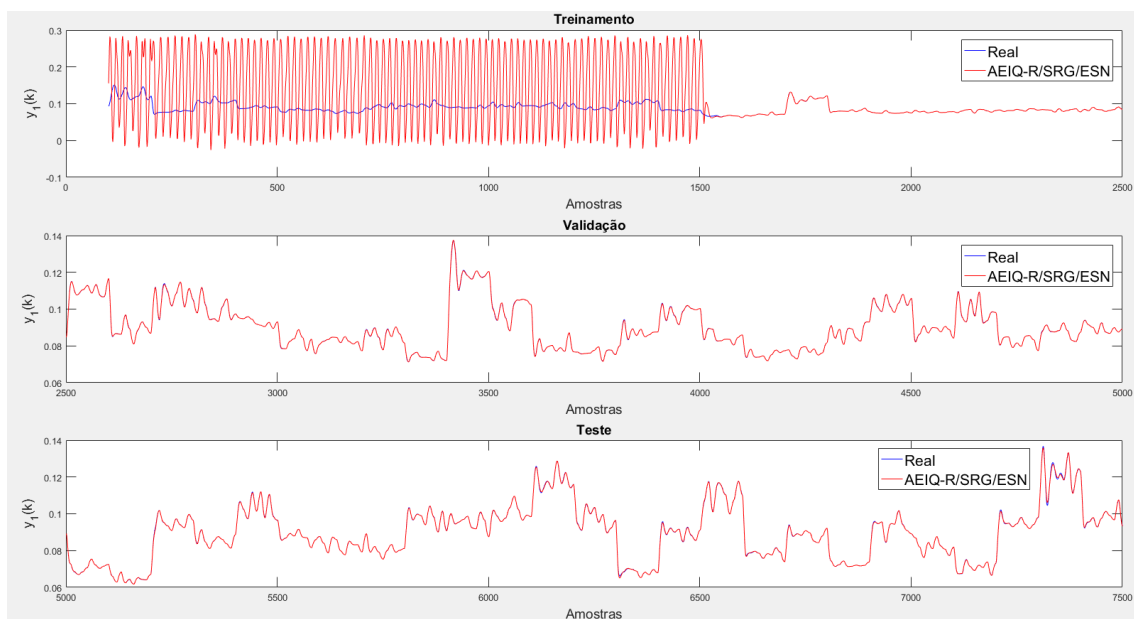


Figura 90 - Simulação da variável de saída y_1 do estudo de caso 5 utilizando a melhor solução do submodelo AEIQ-R/SRG/ESN

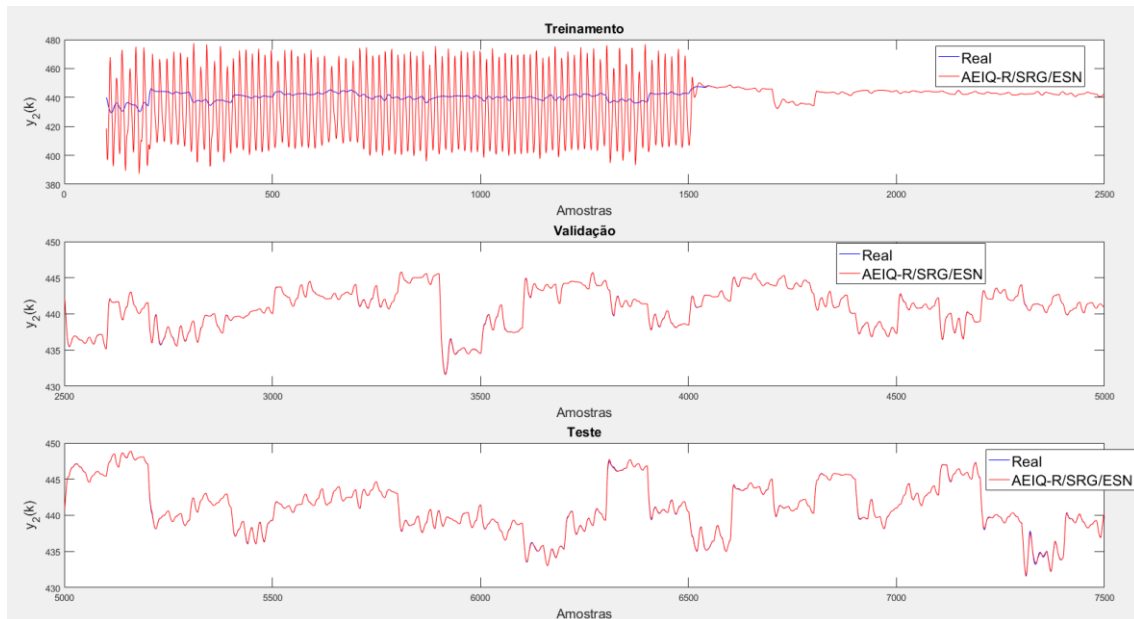


Figura 91 - Simulação da variável de saída y_2 do estudo de caso 5 utilizando a melhor solução do submodelo AEIQ-R/SRG/ESN

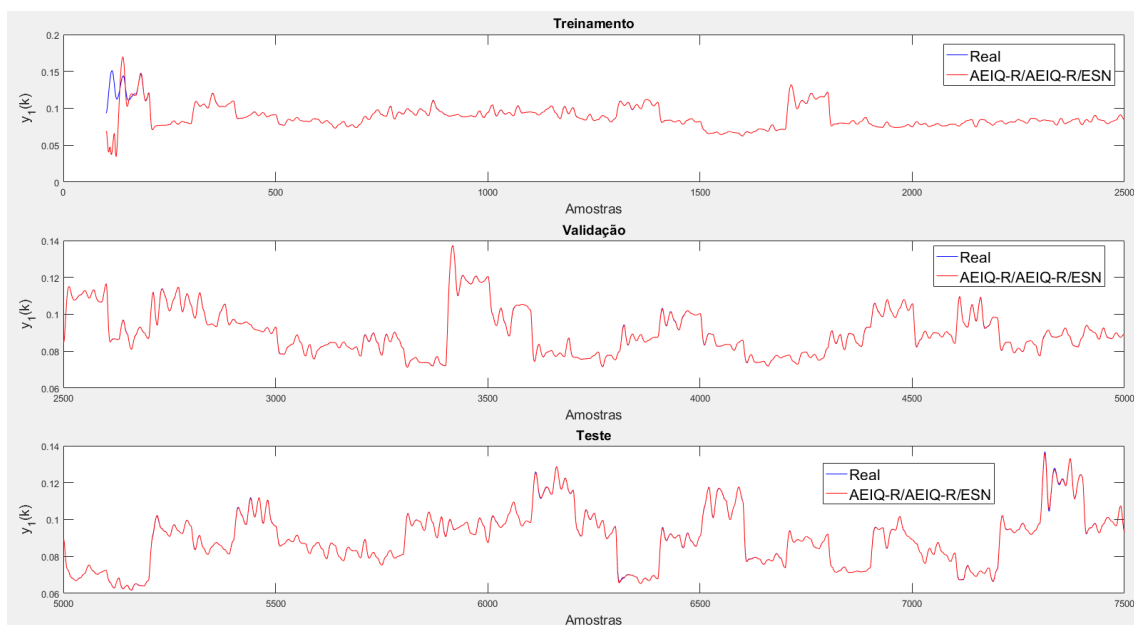


Figura 92 - Simulação da variável de saída y_1 do estudo de caso 5 utilizando a melhor solução do submodelo AEIQ-R/AEIQ-R/ESN

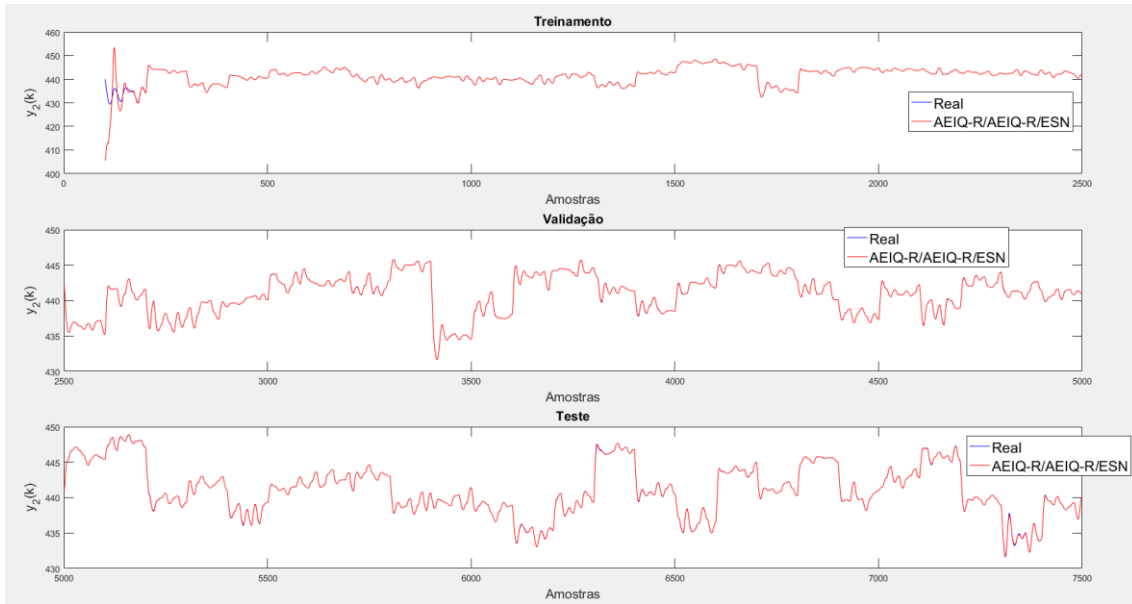


Figura 93 - Simulação da variável de saída y_2 do estudo de caso 5 utilizando a melhor solução do submodelo AEIQ-R/AEIQ-R/ESN

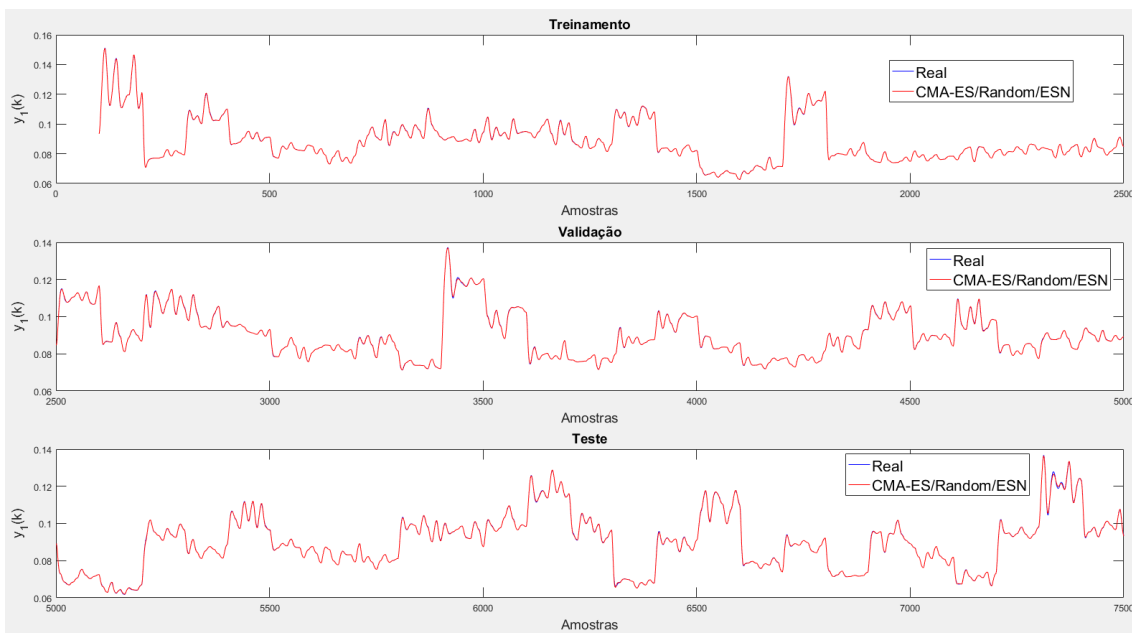


Figura 94 - Simulação da variável de saída y_1 do estudo de caso 5 utilizando a melhor solução do submodelo CMA-ES/Random/ESN

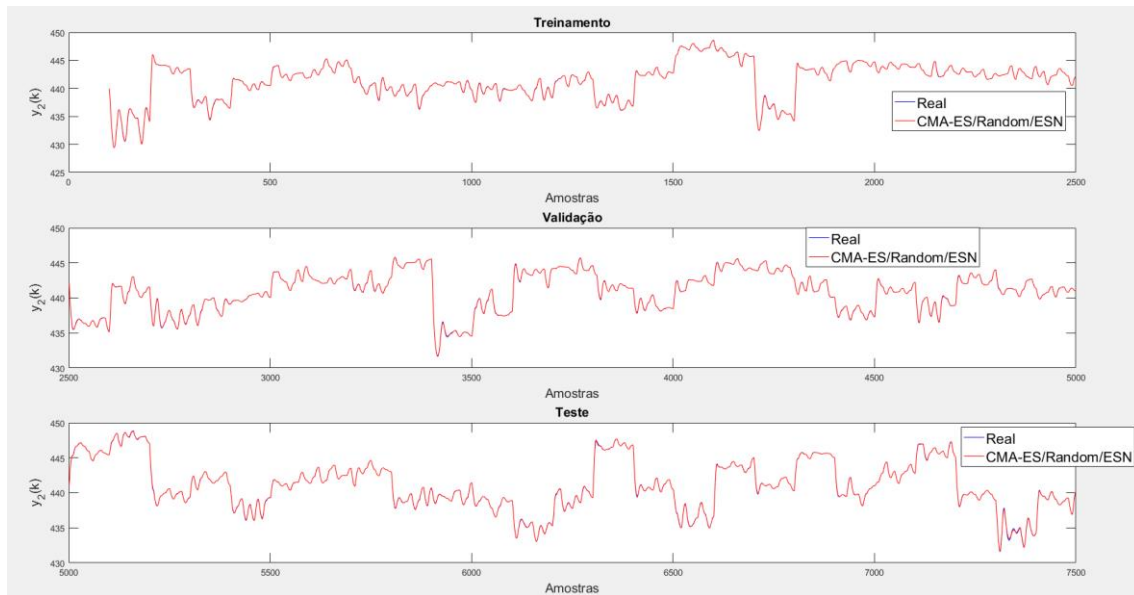


Figura 95 - Simulação da variável de saída y_2 do estudo de caso 5 utilizando a melhor solução do submodelo CMA-ES/Random/ESN

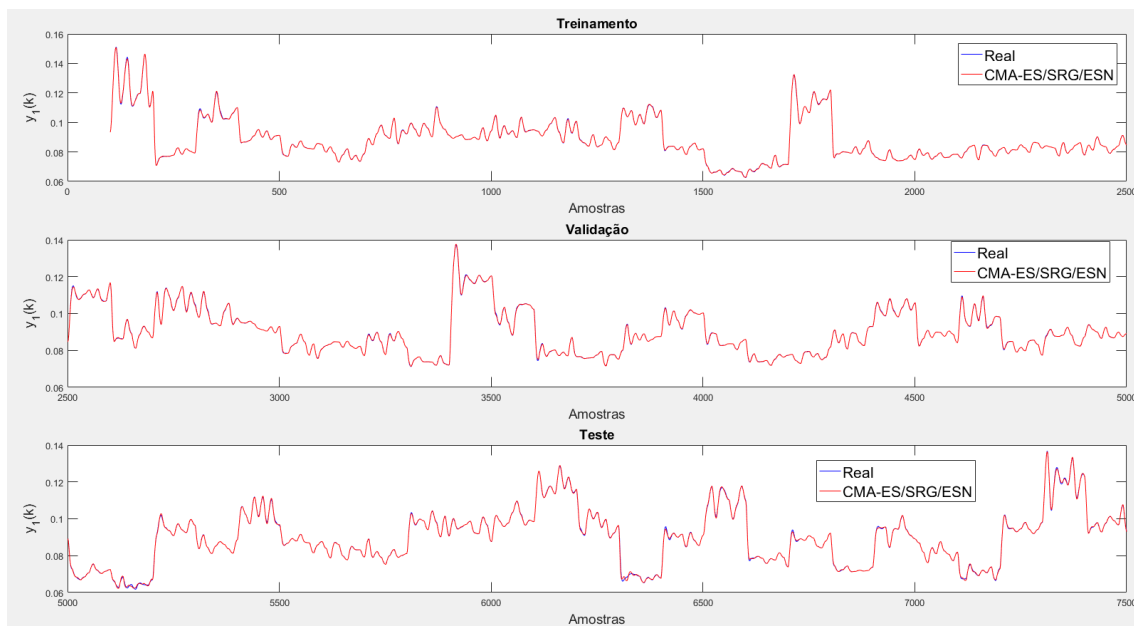


Figura 96 - Simulação da variável de saída y_1 do estudo de caso 5 utilizando a melhor solução do submodelo CMA-ES/SRG/ESN

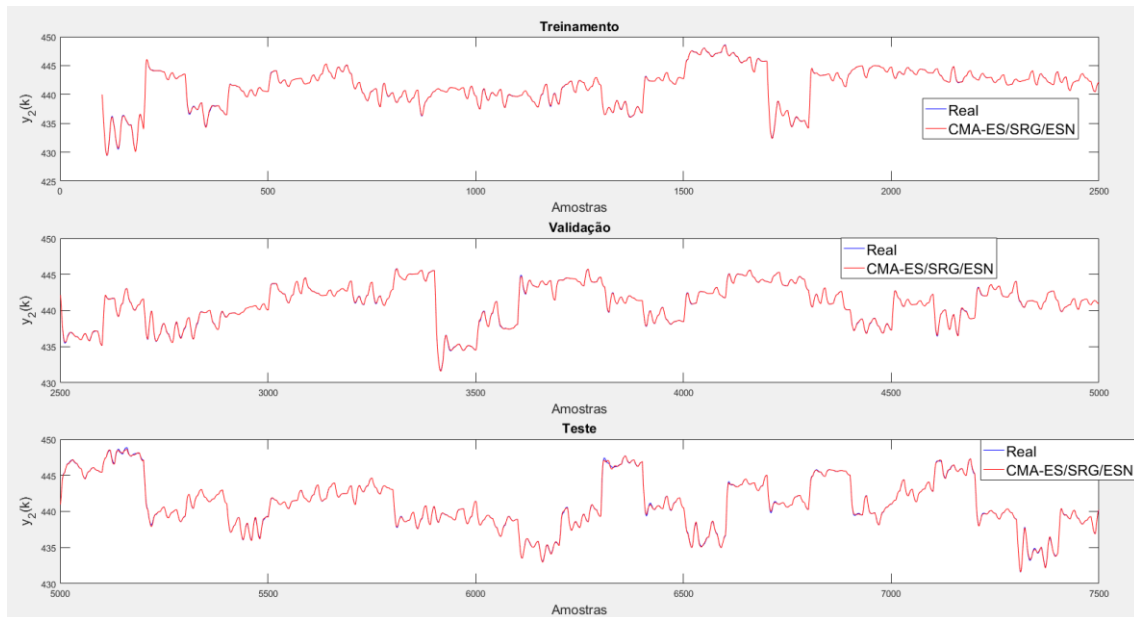


Figura 97 - Simulação da variável de saída y_2 do estudo de caso 5 utilizando a melhor solução do submodelo CMA-ES/SRG/ESN

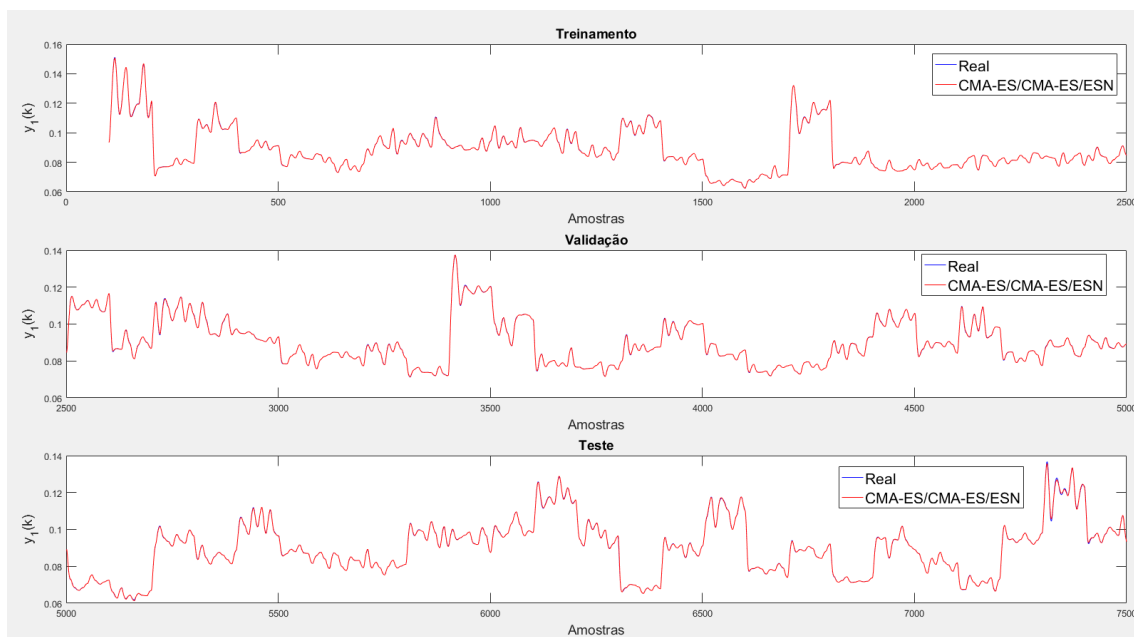


Figura 98 - Simulação da variável de saída y_1 do estudo de caso 5 utilizando a melhor solução do submodelo CMA-ES/CMA-ES/ESN

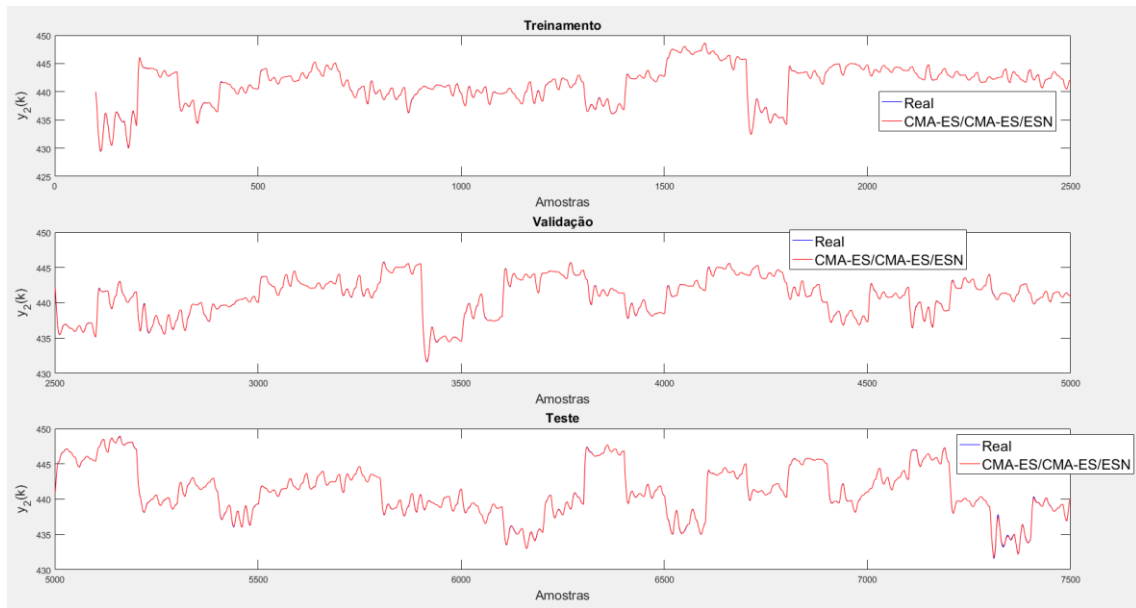


Figura 99 - Simulação da variável de saída y_2 do estudo de caso 5 utilizando a melhor solução do submodelo CMA-ES/CMA-ES/ESN

PUC-Rio - Certificação Digital Nº 1612970/CA

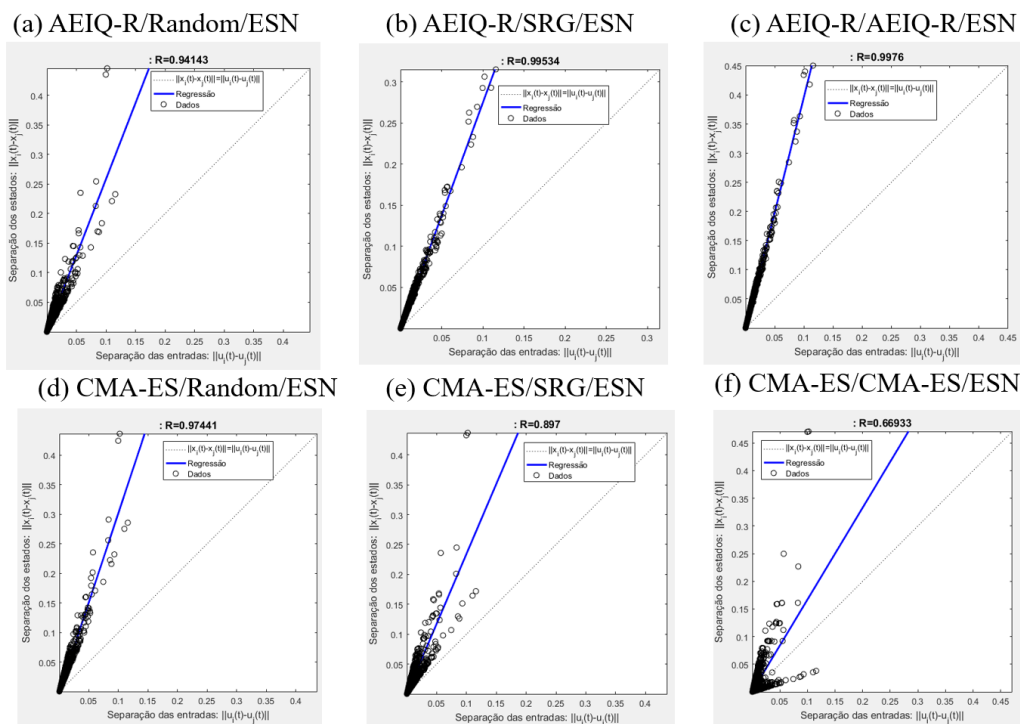


Figura 100 - Gráficos de razão de separação encontrados para as melhores soluções de cada submodelo para o estudo de caso 5

Como mostrado pelas figuras 88 a 99, salvo algumas simulações do conjunto de treinamento, as melhores soluções de cada submodelo dentre os 10 experimentos conseguiram simular adequadamente a variável de saída do sistema.

O comportamento instável durante o treinamento até a amostra 1500 da melhor solução do submodelo AEIQ-R/SRG/ESN pode ser devido a um período transitório inicial maior para esta rede, onde o período de *washout* T_0 pode não ter sido suficiente para dissipar tal efeito. Efeito similar, mas em menor nível e por um período menor de tempo, é visto na simulação das amostras de treinamento pela melhor solução do submodelo AEIQ-R/AEIQ-R/ESN. Nestes casos, especificamente para estas redes, dever-se-ia ter utilizado um valor maior de T_0 de forma que as amostras em que a rede apresentou comportamento instável fossem descartadas e não influenciassem no treinamento.

Os gráficos de razão de separação da figura 100 foram bastante similares (na zona caótica), com o do submodelo CMA-ES/CMA-ES/ESN visualmente um pouco mais próximo da região objetivo.

As tabelas 31 e 32 comparam os resultados obtidos pelas melhores soluções dos submodelos (sem considerar os submodelos AEIQ-R/SRG/ESN e CMA-ES/CMA-ES/ESN, que geraram algumas soluções instáveis) com os resultados dos métodos de Aleksovski et al. (2014, 2016), que utilizaram as mesmas amostras de teste, onde as métricas RRSE e RRMSE são dadas por:

$$RRMSE = \frac{\sqrt{\sum_{k=1}^{N_d} (y(k) - \hat{y}(k))^2}}{\sqrt{\sum_{k=1}^{N_d} (y(k) - \bar{y})^2}} \quad (93)$$

$$RRSE = \sqrt{\frac{\sum_{l=1}^{n_y} \sum_{k=1}^{N_d} (y_l(k) - \hat{y}_l(k))^2}{\sum_{l=1}^{n_y} \sum_{k=1}^{N_d} (y_l(k) - \bar{y}_l)^2}} \quad (94)$$

onde n_y é o número de variáveis de saída, N_d é o número de amostras e \bar{y}_l é o valor médio do vetor de amostras de teste da saída y_l .

Tabela 30 - Comparação 1 de resultados com outros métodos para o *benchmark Reator tanque de agitação contínua*

Método	Métrica
	RRSE
AEIQ-R/AEIQ-R/ESN (média)	0,008708
CMA-ES/CMA-ES/ESN (média)	0,010213
$Bagg_s(SO)$ (Aleksovski et al., 2016)	0,011749
$Bagg_s(MO)$ (Aleksovski et al., 2016)	0,012208
$Lol_s(MO)$ (Aleksovski et al., 2016)	0,012234
$Lol_s(SO)$ (Aleksovski et al., 2016)	0,013023
$Bagg_p(SO)$ (Aleksovski et al., 2016)	0,013708
$Bagg_p(MO)$ (Aleksovski et al., 2016)	0,014389
$Lol_p(SO)$ (Aleksovski et al., 2016)	0,014428
$Lol_p(MO)$ (Aleksovski et al., 2016)	0,014389
CMA-ES/Random/ESN (média)	0,019420
AEIQ-R/Random/ESN (média)	0,020766

Tabela 31 - Comparação 2 de resultados com outros métodos para o *benchmark Reator tanque de agitação contínua*

Método	Métrica	
	RRMSE	
	y_1	y_2
AEIQ-R/AEIQ-R/ESN (melhor)	0,0087	0,0073
CMA-ES/Random/ESN (melhor)	0,0106	0,0084
AEIQ-R/Random/ESN (melhor)	0,0114	0,0094
CMA-ES/CMA-ES/ESN (melhor)	0,0113	0,0097
Bagging MO Model Trees (Aleksovski et al., 2014)	0,0149	0,0128
Bagging SO Model Trees (Aleksovski et al., 2014)	0,0164	0,0127
Model Trees MO (Aleksovski et al., 2014)	0,0195	0,0151
Model Trees SO (Aleksovski et al., 2014)	0,0279	0,0228
Lolimot MO (Aleksovski et al., 2014)	0,0360	0,0238
Lolimot SO (Aleksovski et al., 2014)	0,0384	0,0344

Considerando o desempenho médio nos 10 experimentos em termos de RRSE, pela tabela 30, pode-se ver que ambas as abordagens que passaram pela otimização do reservatório conseguiram gerar resultados superiores aos resultados médios dos métodos apresentados por Aleksovski et al. (2016), ainda que os submodelos AEIQ-R/Random/ESN e CMA-ES/Random/ESN tenham sido os piores nesta comparação. Comparando-se as melhores soluções das abordagens testadas com os resultados dos métodos de Aleksovski et al. (2014) em termos de RRMSE, vê-se que os métodos AEIQ-R/AEIQ-R/ESN e CMA-ES/CMA-ES/ESN foram os que geraram os melhores resultados para ambas as saídas, superando as melhores soluções dos métodos AEIQ-R/Random/ESN e CMA-ES/Random/ESN, que por sua vez, também foram superiores aos métodos de Aleksovski et al. (2014).

6.4.6.

Estudo de caso 6 – Tanques em cascata

A tabela 32 exibe os resultados obtidos durante os 10 experimentos pelos diferentes submodelos derivados do método geral para o estudo de caso 6 – *Tanques em cascata*, com o AEIQ-R e o CMA-ES configurados conforme tabelas 7 e 8 e com a base de dados dividida como mostrado na tabela 6:

Tabela 32– Resultados obtidos de RMSE pelos diferentes submodelos para o *benchmark Tanques em cascata* nos 10 experimentos realizados

Submodelo	Treinamento	Validação	Teste			
	Média	Média	média	melhor	pior	desvio
AEIQ-R/Random/ESN	0,089460	0,133417	7,565423	0,212962	73,0965	23,025382
AEIQ-R/SRG/ESN	0,094467	0,207962	1,1226	0,178916	8,885537	2,728260
AEIQ-R/AEIQ-R/ESN	0,094577	0,059337	0,256585	0,176050	0,345844	0,048943
CMA-ES/Random/ESN	0,090432	0,138407	0,286990	0,208571	0,403794	0,073393
CMA-ES/SRG/ESN	0,087303	0,173237	0,325602	0,210808	0,869377	0,194585
CMA-ES/CMA-ES/ESN	0,088473	0,076477	0,214222	0,167296	0,261731	0,028115

Pelos resultados médios de RMSE nos 10 experimentos mostrados na tabela acima, pode-se ver que os métodos AEIQ-R/AEIQ-R/ESN e CMA-ES/CMA-ES/ESN foram os que apresentaram melhor desempenho durante a simulação do conjunto de teste, com superioridade deste último frente àquele. Pela tabela 32, vê-se que os submodelos AEIQ-R/Random/ESN e AEIQ-R/SRG/ESN foram os que apresentaram os maiores

valores médios de RMSE durante o teste, o que foi devido a soluções instáveis geradas por estes métodos. As figuras 101 e 102 mostram as curvas de evolução dos algoritmos AEIQ-R e CMA-ES nas etapas 1 e 2, exibindo-se a média da aptidão nos 10 experimentos do melhor indivíduo da população, enquanto a tabela 33 exhibe os valores de aptidão na última geração:

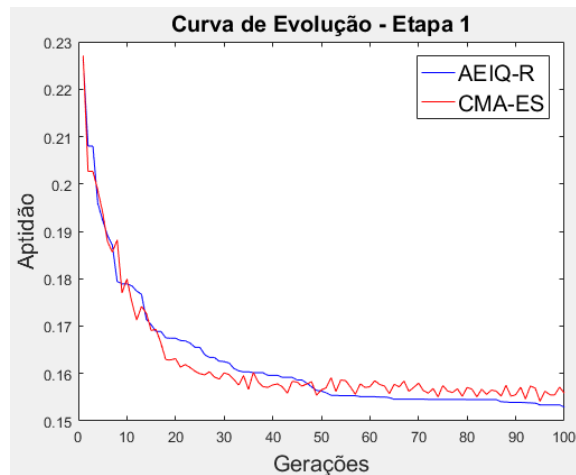


Figura 101 - Curva de evolução da etapa 1 (média da aptidão do melhor indivíduo em cada geração nos 10 experimentos) para o estudo de caso 6

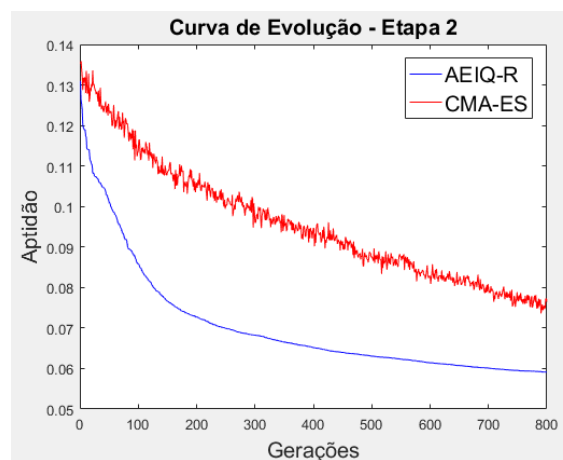


Figura 102 - Curva de evolução da etapa 2 (média da aptidão do melhor indivíduo em cada geração nos 10 experimentos) para o estudo de caso 6

Tabela 33 - Aptidão dos AEs na última geração (média de 10 experimentos) para o estudo de caso 6

Algoritmo	Aptidão na última geração (média de 10 experimentos)	
	Etapa 1	Etapa 2
AEIQ-R	0,152921	0,059337
CMA-ES	0,155831	0,076477

Pelas figuras 101 e 102, vê-se que o AEIQ-R e o CMA-ES apresentaram desempenho similar durante a curva de evolução da etapa 1. Durante a etapa 2, o AEIQ-

R visivelmente conseguiu obter desempenho superior ao CMA-ES durante todo o processo evolutivo, convergindo para valores menores de erro de validação, porém como visto pela tabela 32, as soluções obtidas pelo submodelo CMA-ES/CMA-ES/ESN foram de melhor qualidade, obtendo melhor desempenho durante o teste. Nas duas etapas ambos algoritmos pareceram não ter convergido.

A tabela 34 mostra a configuração de parâmetros globais que gerou as melhores soluções de cada submodelo (dentre as 10 soluções finais de cada experimento), enquanto que as figuras 103 a 114 mostram a simulação por cada uma dessas soluções dos conjuntos de treinamento, validação e teste. A figura 115 mostra os gráficos de razão de separação para cada uma dessas soluções.

Tabela 34 - Parâmetros globais da melhor solução encontrada por cada submodelo para o estudo de caso 6

Melhor solução	N	c_W	ρ_W	s_{in}	s_{back}	α
AEIQ-R/Random/ESN	77	0,6689	0,7188	0,8436	0,1760	0,8835
AEIQ-R/SRG/ESN	69	0,6738	0,6736	0,9178	0,2701	0,8927
AEIQ-R/AEIQ-R/ESN	66	0,4427	0,7856	0,7752	0,3633	0,9057
CMA-ES/Random/ESN	80	0,6317	0,6195	0,9875	0,1700	0,9641
CMA-ES/SRG/ESN	81	0,3767	0,7692	0,9570	0,2167	0,9300
CMA-ES/CMA-ES/ESN	80	0,3805	0,7099	0,9624	0,2069	0,9838

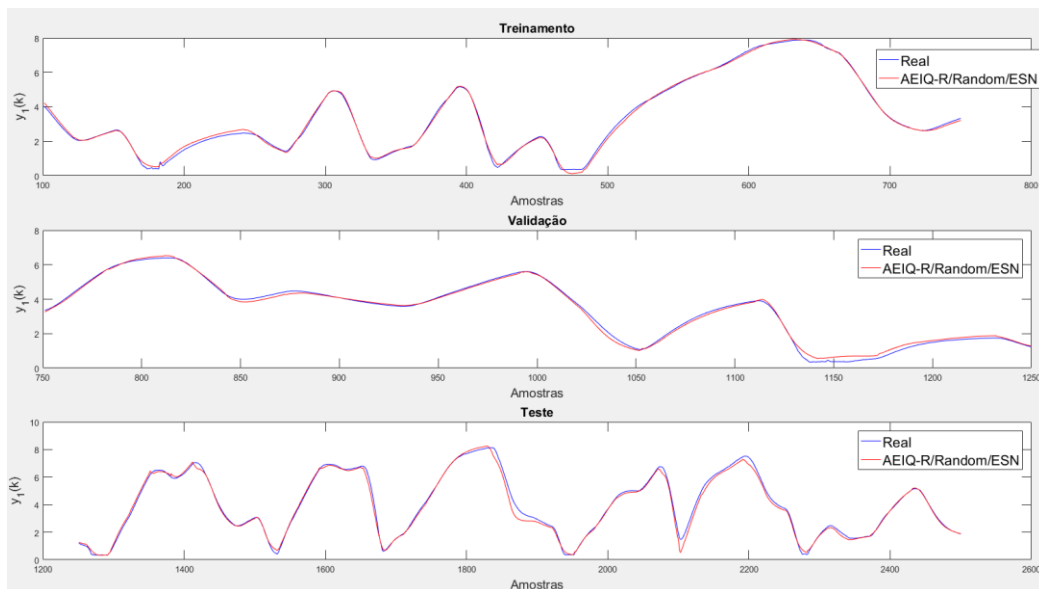


Figura 103 - Simulação da variável de saída y_1 do estudo de caso 6 utilizando a melhor solução do submodelo AEIQ-R/Random/ESN

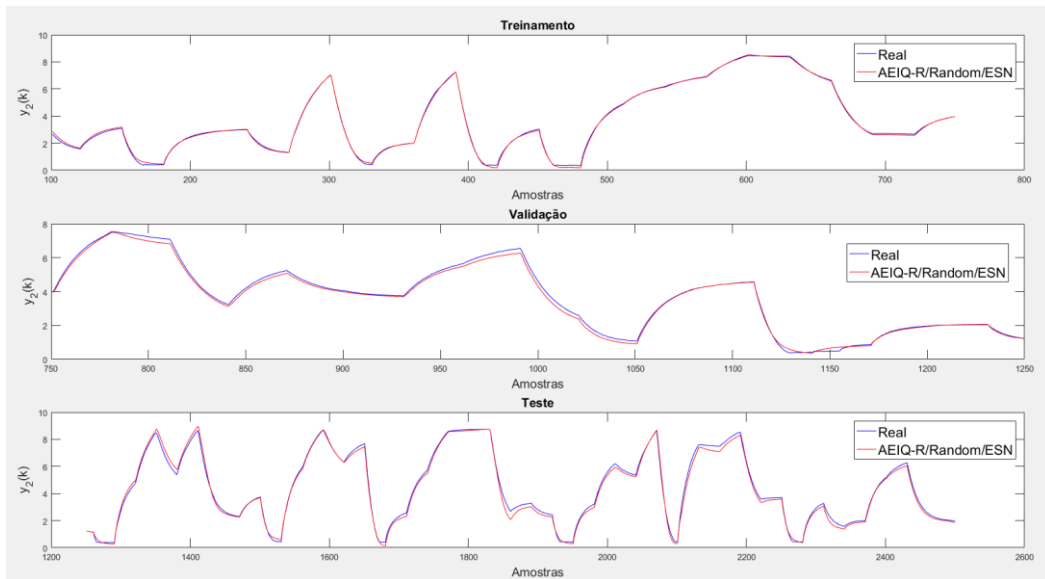


Figura 104 - Simulação da variável de saída y_2 do estudo de caso 6 utilizando a melhor solução do submodelo AEIQ-R/Random/ESN

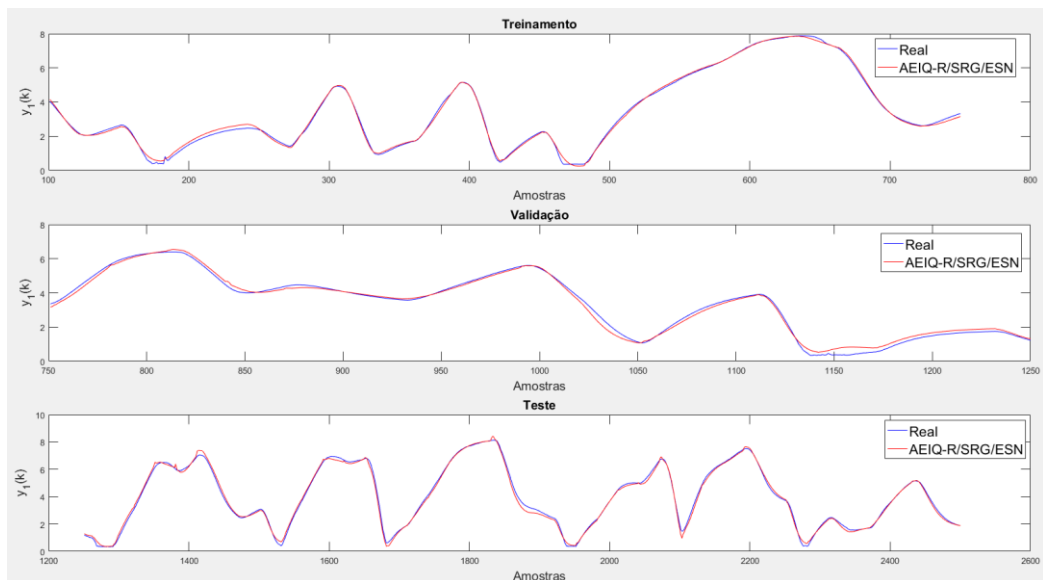


Figura 105 - Simulação da variável de saída y_1 do estudo de caso 6 utilizando a melhor solução do submodelo AEIQ-R/SRG/ESN

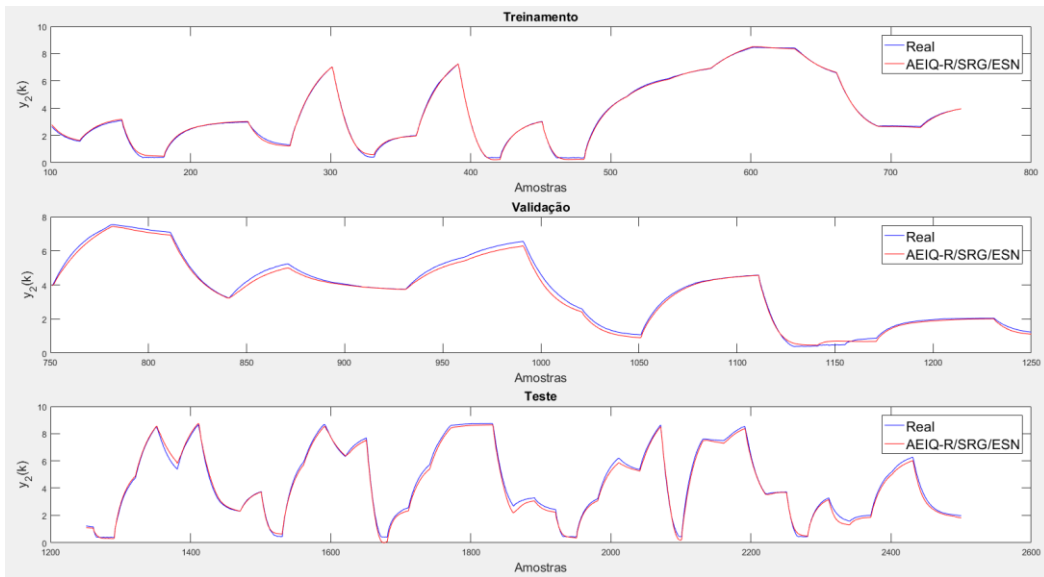


Figura 106 - Simulação da variável de saída y_2 do estudo de caso 6 utilizando a melhor solução do submodelo AEIQ-R/SRG/ESN

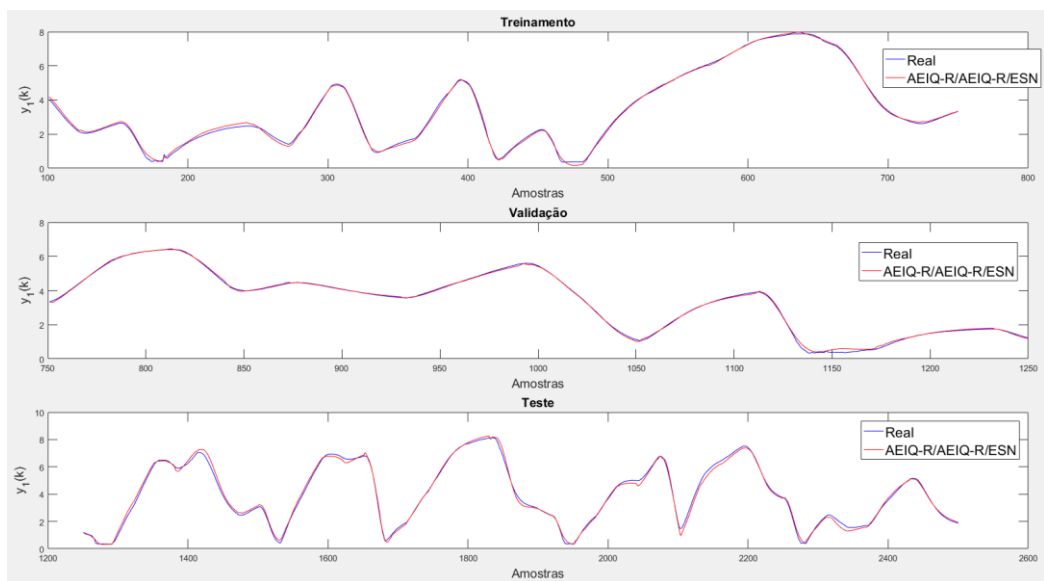


Figura 107 - Simulação da variável de saída y_1 do estudo de caso 6 utilizando a melhor solução do submodelo AEIQ-R/AEIQ-R/ESN

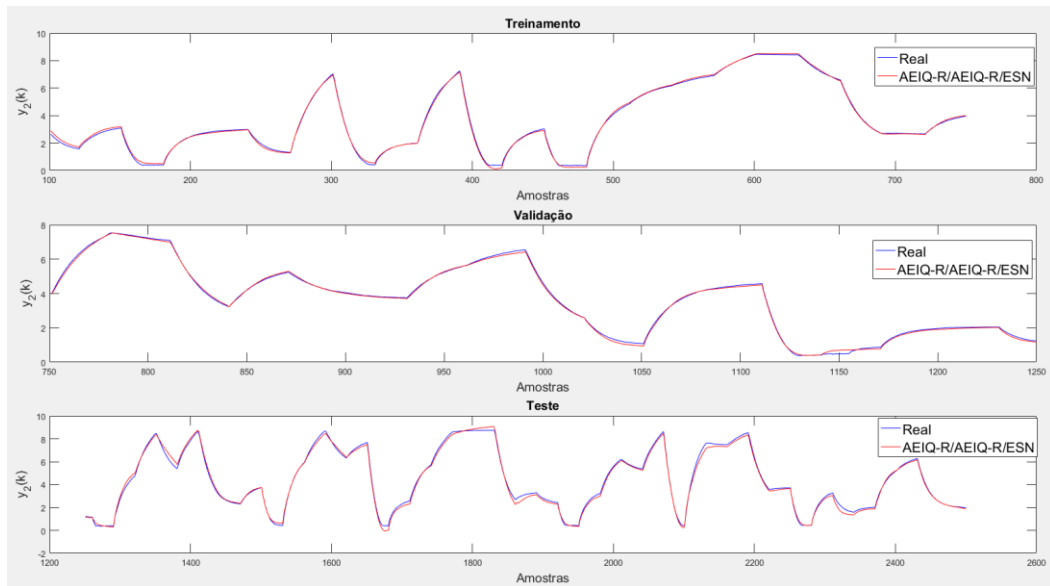


Figura 108 - Simulação da variável de saída y_2 do estudo de caso 6 utilizando a melhor solução do submodelo AEIQ-R/AEIQ-R/ESN

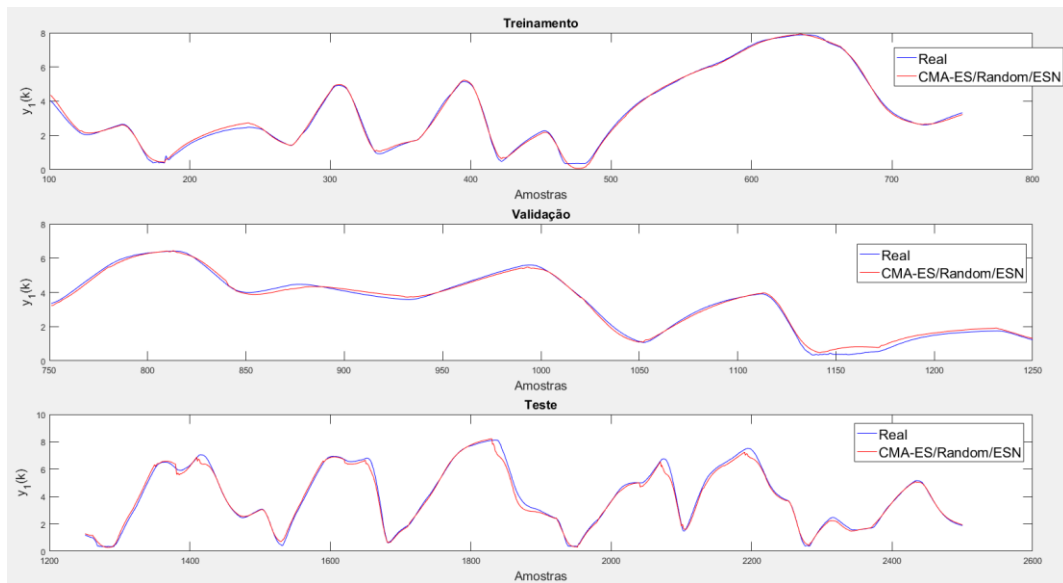


Figura 109 - Simulação da variável de saída y_1 do estudo de caso 6 utilizando a melhor solução do submodelo CMA-ES/Random/ESN

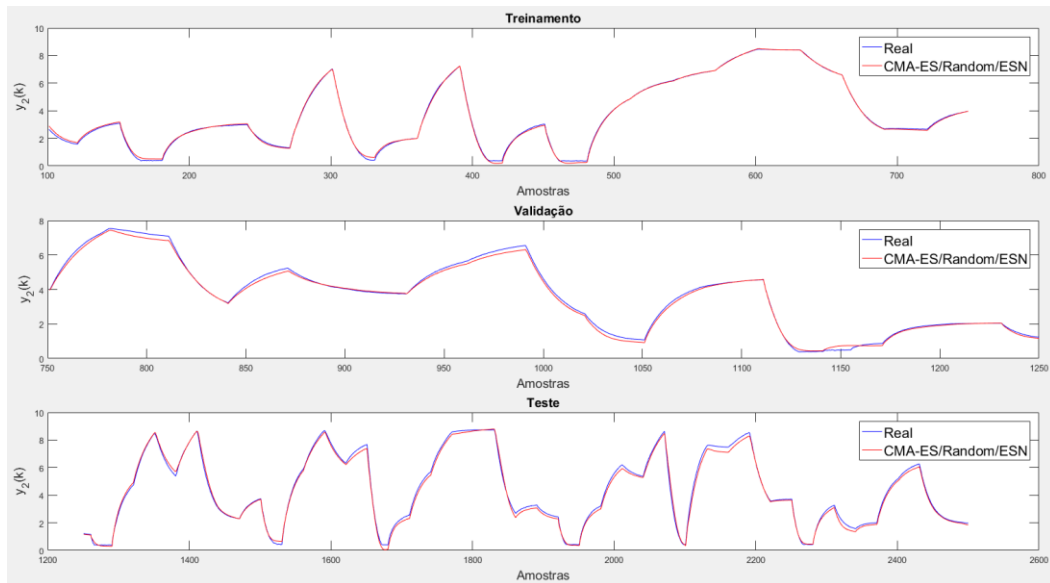


Figura 110 - Simulação da variável de saída y_2 do estudo de caso 6 utilizando a melhor solução do submodelo CMA-ES/Random/ESN

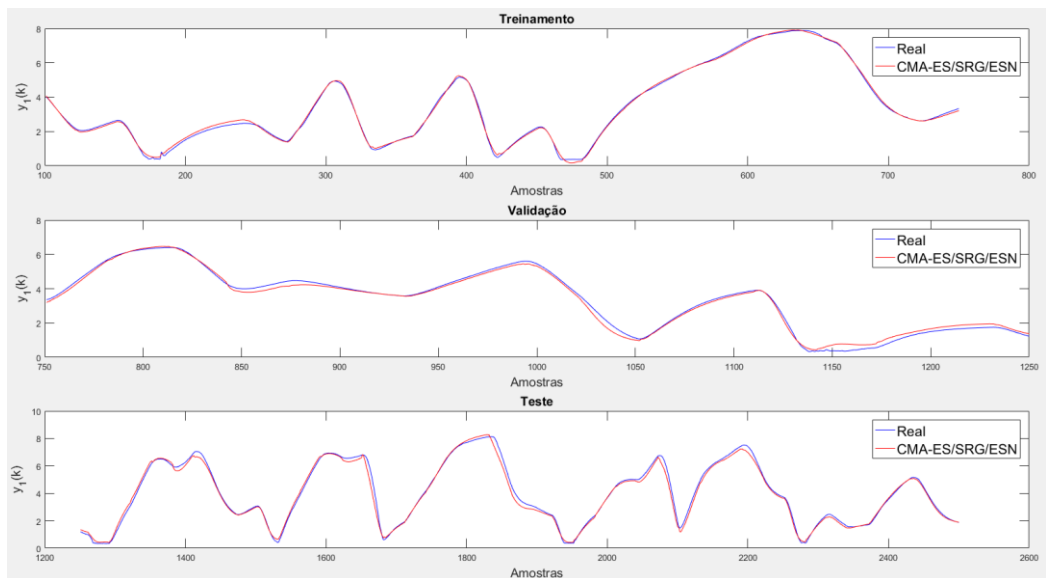


Figura 111 - Simulação da variável de saída y_1 do estudo de caso 6 utilizando a melhor solução do submodelo CMA-ES/SRG/ESN

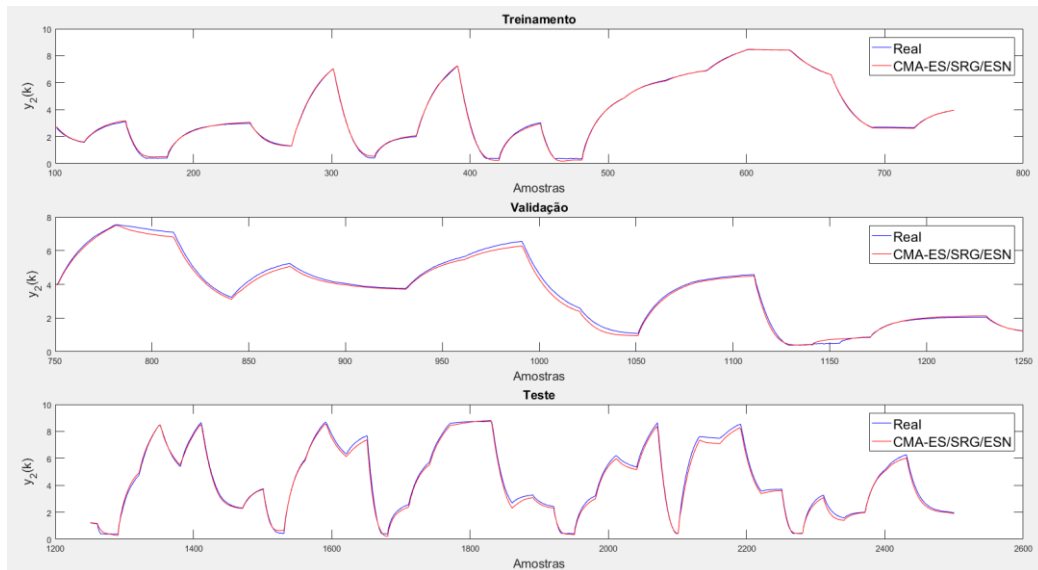


Figura 112 - Simulação da variável de saída y_2 do estudo de caso 6 utilizando a melhor solução do submodelo CMA-ES/SRG/ESN

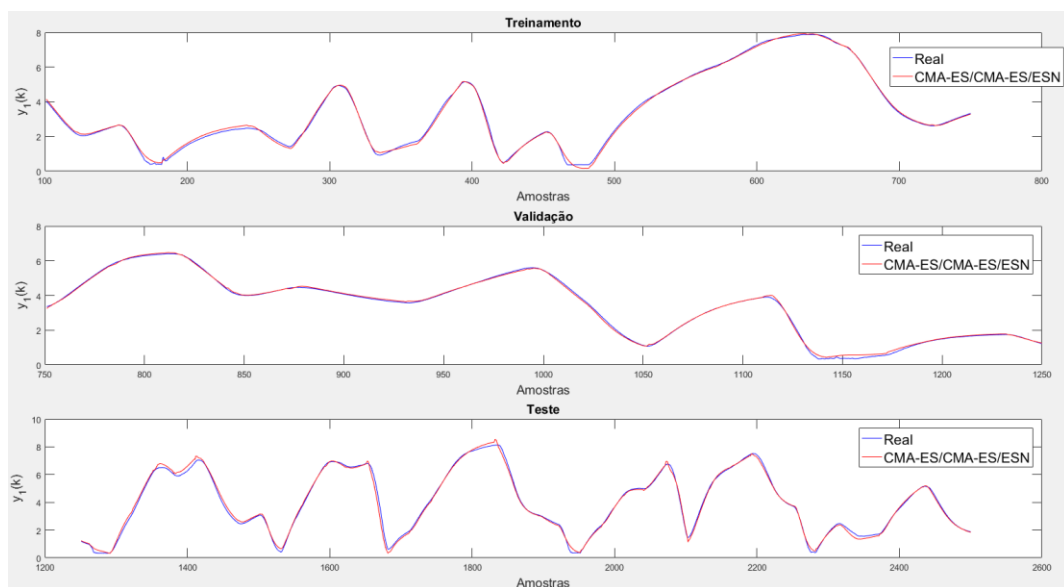


Figura 113 - Simulação da variável de saída y_1 do estudo de caso 6 utilizando a melhor solução do submodelo CMA-ES/CMA-ES/ESN

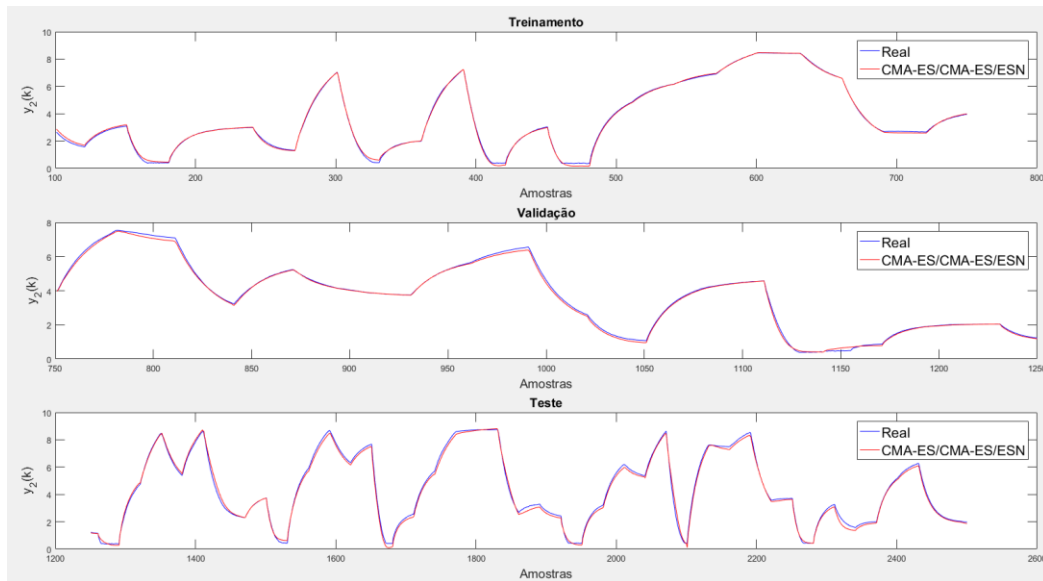


Figura 114 - Simulação da variável de saída y_2 do estudo de caso 6 utilizando a melhor solução do submodelo CMA-ES/CMA-ES/ESN

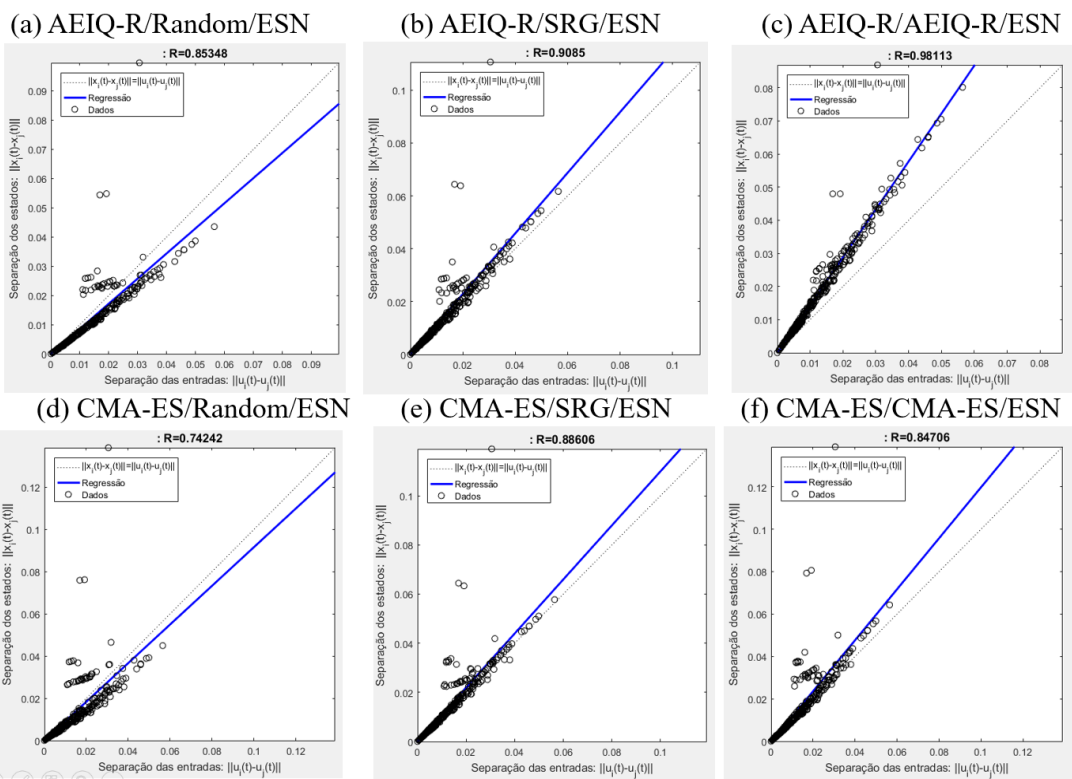


Figura 115 - Gráficos de razão de separação encontrados para as melhores soluções de cada submodelo para o estudo de caso 6

Como mostrado pelas figuras 103 a 114, as melhores soluções de cada submodelo dentre os 10 experimentos conseguiram simular adequadamente as variáveis de saída do sistema. Quanto aos gráficos de razão de separação da figura 115, vê-se que os gráficos

das melhores soluções dos submodelos CMA-ES/CMA-ES/ESN e AEIQ-R/AEIQ-R/ESN ficaram um pouco mais distantes do que os demais da região objetivo e ainda assim atingiram melhor desempenho durante a simulação do conjunto de teste.

A tabela 35 compara os resultados obtidos pelas melhores soluções dos submodelos (sem considerar os submodelos AEIQ-R/Random/ESN e AEIQ-R/SRG/ESN, que produziram soluções instáveis) com os melhores resultados dos métodos de La Cava et al. (2016), que utilizaram as mesmas amostras de teste, onde a métrica R^2 é utilizada (objetivo encontrar valores mais próximos de 1):

$$R^2 = \frac{cov(y, \hat{y})}{var(y)var(\hat{y})} \quad (95)$$

onde $var(.)$ e $cov(.)$ representam as funções de variância e covariância, respectivamente.

Tabela 35 - Comparação de resultados com outros métodos para o *benchmark Tanques em cascata*

Método	Métrica			
	MSE		R^2	
	y_1	y_2	y_1	y_2
CMA-ES/CMA-ES/ESN (melhor)	0,026	0,030	0,995	0,996
AEIQ-R/AEIQ-R/ESN (melhor)	0,028	0,034	0,994	0,995
CMA-ES/Random/ESN (melhor)	0,054	0,034	0,991	0,996
CMA-ES/SRG/ESN (melhor)	0,054	0,036	0,991	0,997
NARX - MLP (La Cava et al., 2016)	0,120	0,182	0,977	0,974
Epigenetic Linear Genetic Programming (La Cava et al., 2016)	0,249	0,288	0,953	0,974
ARX (La Cava et al., 2016)	0,432	0,350	0,919	0,961
Teórico (La Cava et al., 2016)	0,463	5,343	0,907	0,313
NARX – Wavelet Neural Networks (La Cava et al., 2016)	2,960	6,165	0,807	0,748

Como mostrado na tabela 35, as melhores soluções dos submodelos testados nesta dissertação apresentaram resultados mais acurados, considerando tanto o MSE quanto o a métrica R^2 , para as duas saídas do sistema, em comparação aos métodos de La Cava et al. (2016). Porém, se o número de pesos das ESNs finais de cada submodelo for considerado como uma medida de complexidade, tais submodelos podem ser

considerados mais complexos do que os dois melhores métodos de La Cava et al. (2016), NARX - MLP e ELGP, que possuem 60 e 3 parâmetros, de acordo com La Cava et al. (2016).

6.4.7. Estudo de caso 7 – Gerador de Vapor

A tabela 36 exibe os resultados obtidos durante os 10 experimentos pelos diferentes submodelos derivados do método geral para o estudo de caso 7 – *Gerador de Vapor*, com o AEIQ-R e o CMA-ES configurados conforme tabelas 7 e 8 e com a base de dados dividida como mostrado na tabela 6:

Tabela 36 – Resultados obtidos de RMSE pelos diferentes submodelos para o *benchmark Gerador de Vapor* nos 10 experimentos realizados

Submodelo	Treinamento	Validação	Teste			
	Média	média	média	Melhor	pior	desvio
AEIQ-R/Random/ESN	1,136652	1,582134	2,834602	2,700473	3,138800	0,133463
AEIQ-R/SRG/ESN	1,132507	1,897421	153,2818	2,771291	1505,3565	475,07067
AEIQ-R/AEIQ-R/ESN	1,132019	1,081123	2,719567	2,654622	2,779402	0,039592
CMA-ES/Random/ESN	1,141762	1,596429	2,864844	2,534678	3,065028	0,146418
CMA-ES/SRG/ESN	1,154965	1,850932	19,7248	2,793645	170,1615	52,8583
CMA-ES/CMA-ES/ESN	1,095650	1,10756	2,711008	2,552125	2,895655	0,093862

Pelos resultados médios de RMSE mostrados na tabela 36, vê-se que os métodos AEIQ-R/AEIQ-R/ESN e CMA-ES/CMA-ES/ESN foram os que obtiveram melhor desempenho durante a simulação do conjunto de teste. Mais uma vez, os submodelos AEIQ-R/SRG/ESN e CMA-ES/SRG/ESN geraram soluções instáveis, o que contribuiu para que tais abordagens apresentassem o pior desempenho médio considerando os 10 experimentos realizados. Por se tratar de um sistema de comportamento dinâmico complexo, MIMO (com 4 entradas e 4 saídas), além de a base de dados possuir um tamanho relativamente grande, o que limitou o número de gerações usadas durante a evolução da etapa 2, os valores de RMSE apresentados na tabela 36 são relativamente maiores do que os mostrados em estudos de caso anteriores. Talvez por este motivo, a diferença média entre os métodos que utilizaram as abordagens Random e Otimização do

reservatório na segunda etapa tenham sido pequenas neste estudo de caso, com inclusive a melhor solução entre os 10 experimentos tendo sido obtida pela abordagem CMA-ES/SRG/ESN. As figuras 116 e 117 mostram as curvas de evolução dos algoritmos AEIQ-R e CMA-ES nas etapas 1 e 2, exibindo-se a média da aptidão nos 10 experimentos do melhor indivíduo da população, enquanto a tabela 37 exhibe os valores de aptidão na última geração:

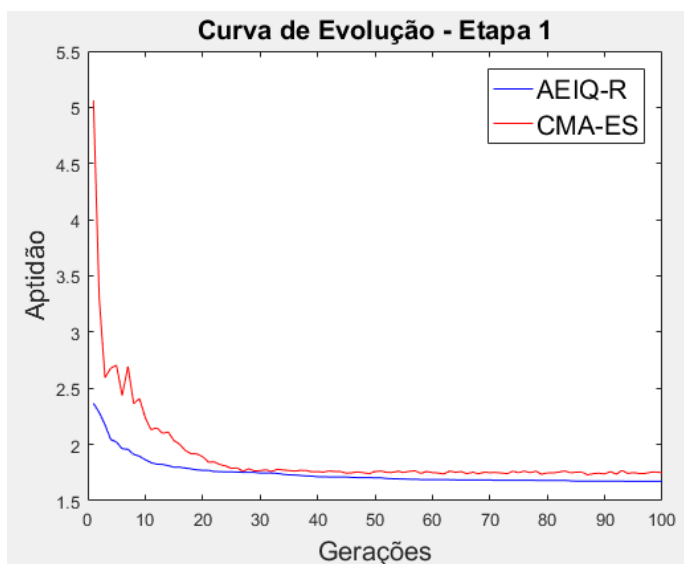


Figura 116 - Curva de evolução da etapa 1 (média da aptidão do melhor indivíduo em cada geração nos 10 experimentos) para o estudo de caso 7

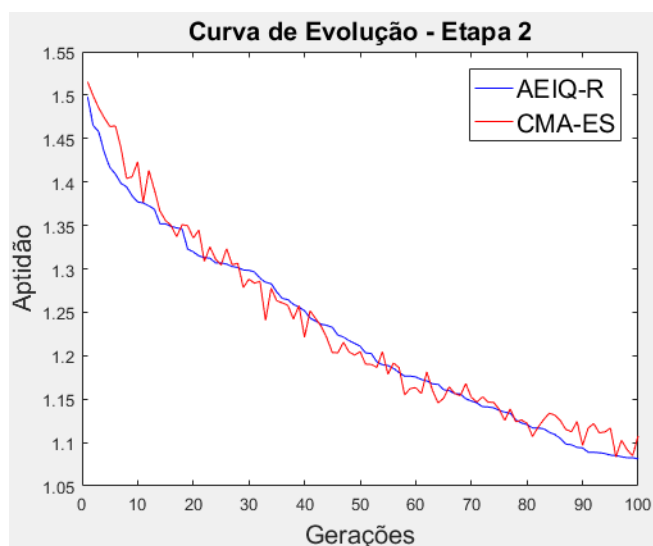


Figura 117 - Curva de evolução da etapa 2 (média da aptidão do melhor indivíduo em cada geração nos 10 experimentos) para o estudo de caso 7

Tabela 37 - Aptidão dos AEs na última geração (média de 10 experimentos) para o estudo de caso 7

Algoritmo	Aptidão na última geração (média de 10 experimentos)	
	Etapa 1	Etapa 2
AEIQ-R	1,671840	1,081123
CMA-ES	1,747469	1,107856

Pelas figuras 116 e 117, vê-se que ambos AEs apresentaram desempenho similar durante o processo evolutivo em ambas etapas, porém, como mostrado pela tabela 36, a abordagem CMA-ES/CMA-ES gerou soluções em média melhores do que a abordagem AEIQ-R/AEIQ-R/ESN e, visualmente, ambos algoritmos pareceram não ter convergido na segunda etapa. Talvez os resultados apresentados na tabela 36 pudessem ser ainda melhorados com o aumento do número de gerações na segunda etapa.

A tabela 38 mostra a configuração de parâmetros globais que gerou as melhores soluções de cada submodelo (dentre as 10 soluções finais de cada experimento), enquanto que as figuras 118 a 141 mostram a simulação por cada uma dessas soluções dos conjuntos de treinamento, validação e teste. A figura 142 mostra os gráficos de razão de separação para cada uma dessas soluções.

Tabela 38 - Parâmetros globais da melhor solução encontrada por cada submodelo para o estudo de caso 7

Melhor solução	N	c_w	ρ_w	s_{in}	s_{back}	α
AEIQ-R/Random/ESN	314	0,4091	0,2882	0,2571	0,0988	0,6646
AEIQ-R/SRG/ESN	264	0,3470	0,3595	0,2985	0,1755	0,6504
AEIQ-R/AEIQ-R/ESN	264	0,3470	0,3595	0,2985	0,1755	0,6504
CMA-ES/Random/ESN	293	0,8463	0,3127	0,2584	0,1170	0,6794
CMA-ES/SRG/ESN	353	0,5849	0,3277	0,3008	0,1263	0,7100
CMA-ES/CMA-ES/ESN	329	0,4974	0,3009	0,2512	0,0902	0,4974

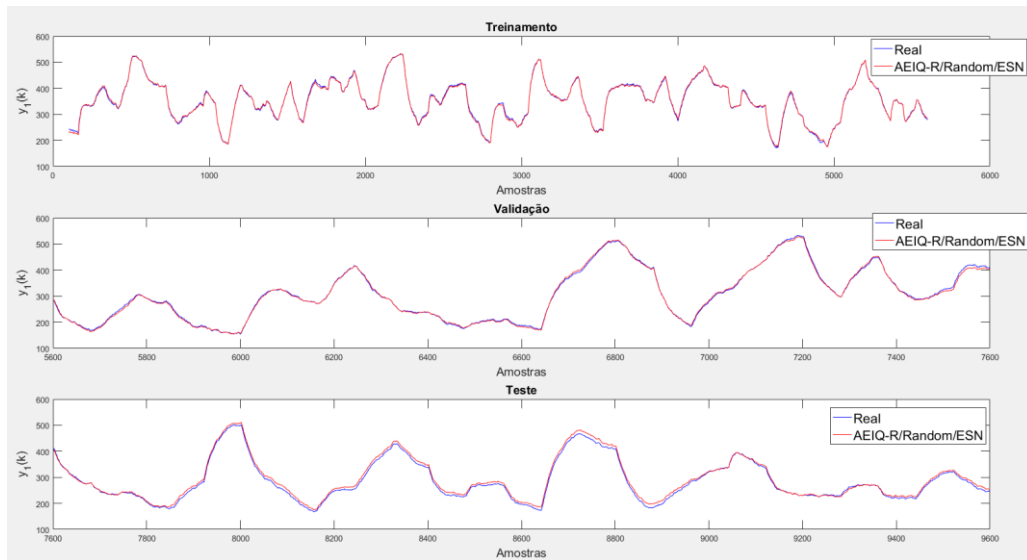


Figura 118 - Simulação da variável de saída y_1 do estudo de caso 7 utilizando a melhor solução do submodelo AEIQ-R/Random/ESN

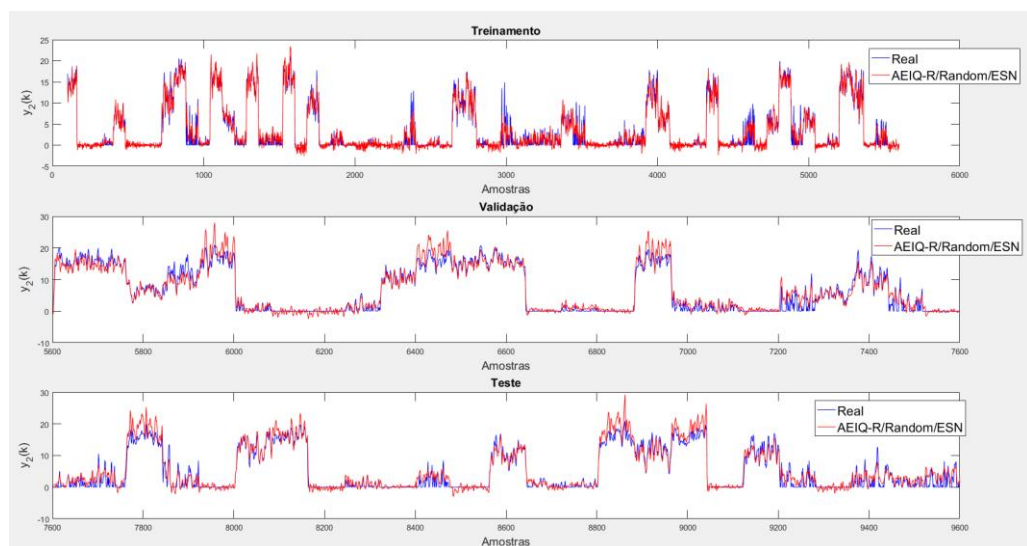


Figura 119 - Simulação da variável de saída y_2 do estudo de caso 7 utilizando a melhor solução do submodelo AEIQ-R/Random/ESN

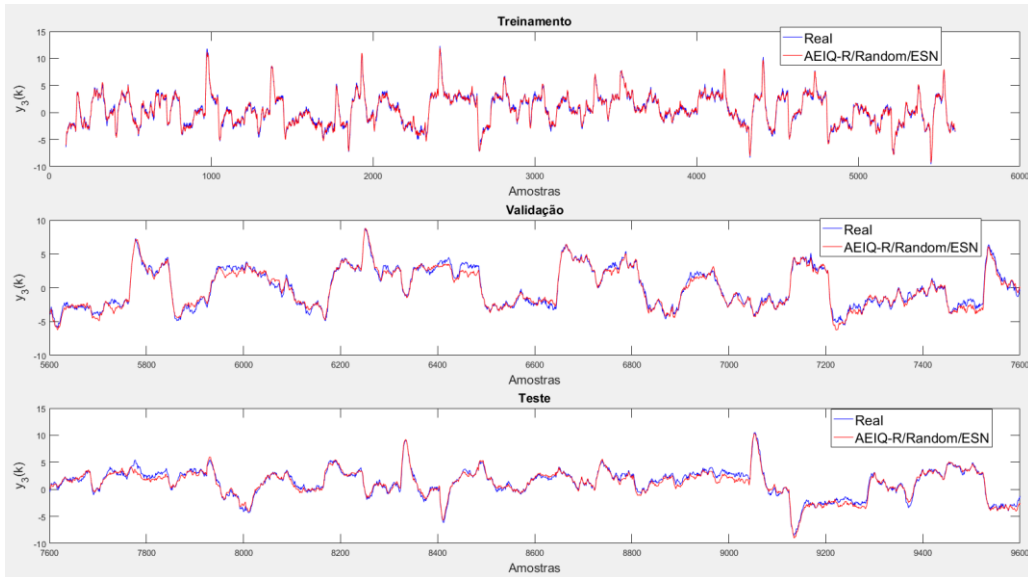


Figura 120 - Simulação da variável de saída y_3 do estudo de caso 7 utilizando a melhor solução do submodelo AEIQ-R/Random/ESN

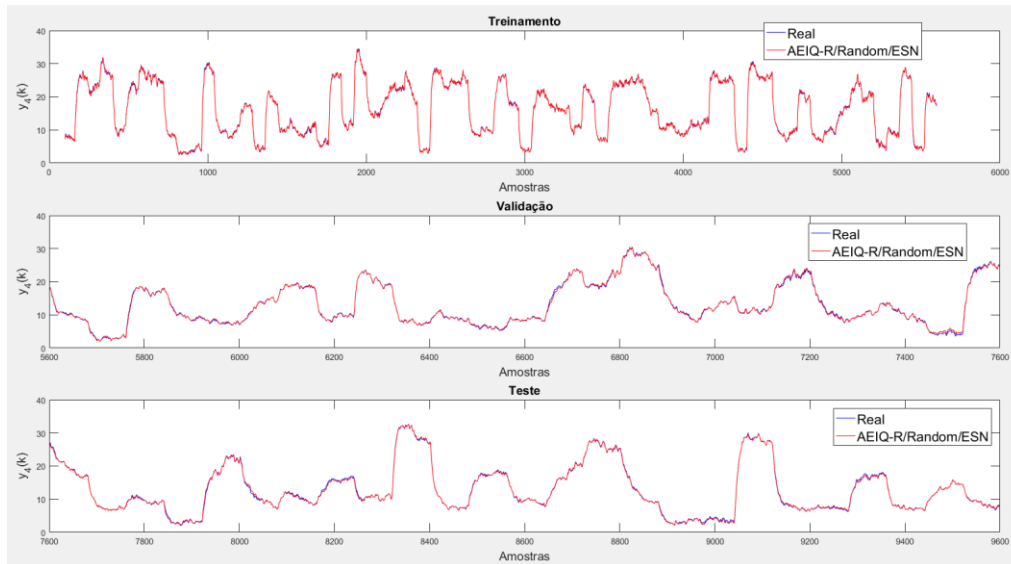


Figura 121 - Simulação da variável de saída y_4 do estudo de caso 7 utilizando a melhor solução do submodelo AEIQ-R/Random/ESN

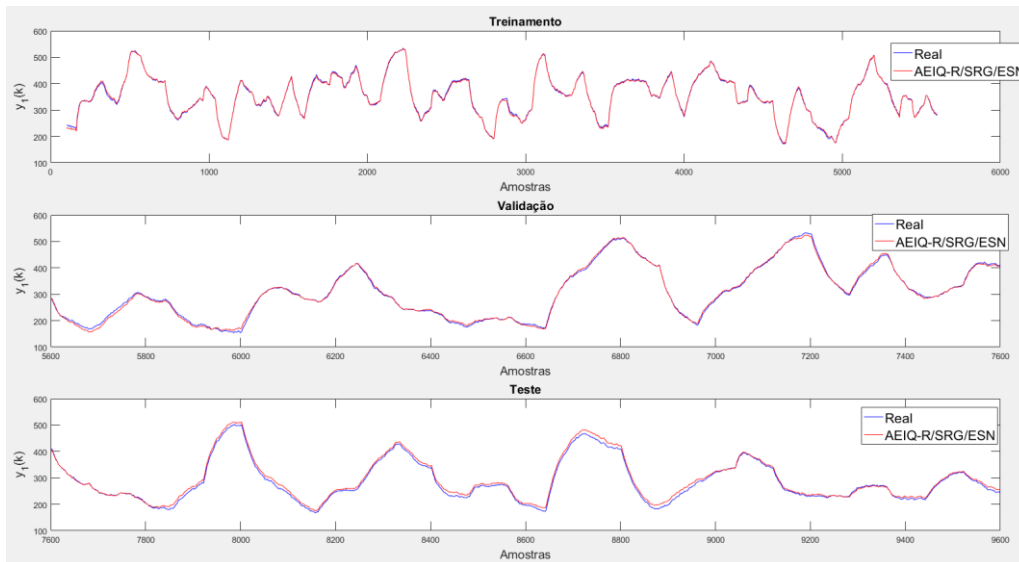


Figura 122 - Simulação da variável de saída y_1 do estudo de caso 7 utilizando a melhor solução do submodelo AEIQ-R/SRG/ESN

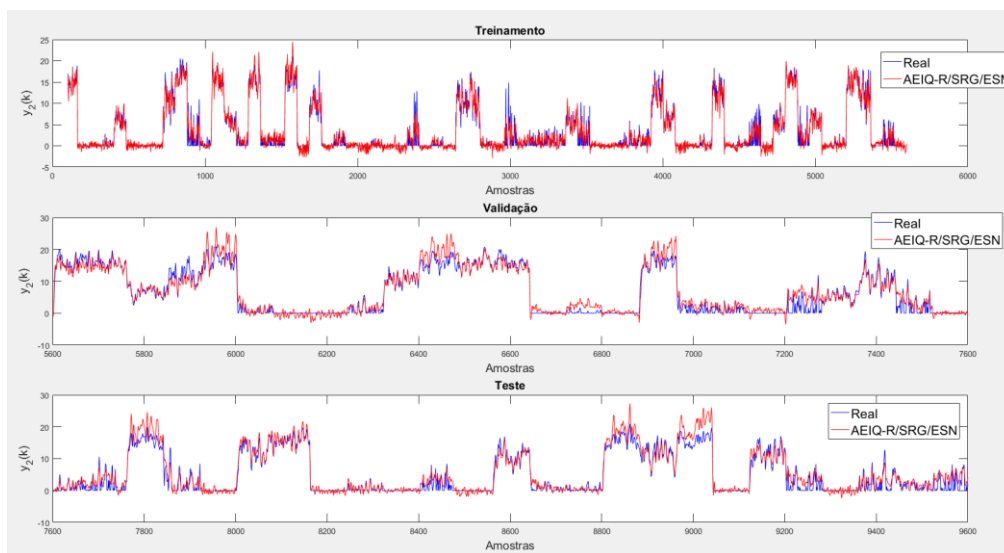


Figura 123 - Simulação da variável de saída y_2 do estudo de caso 7 utilizando a melhor solução do submodelo AEIQ-R/SRG/ESN

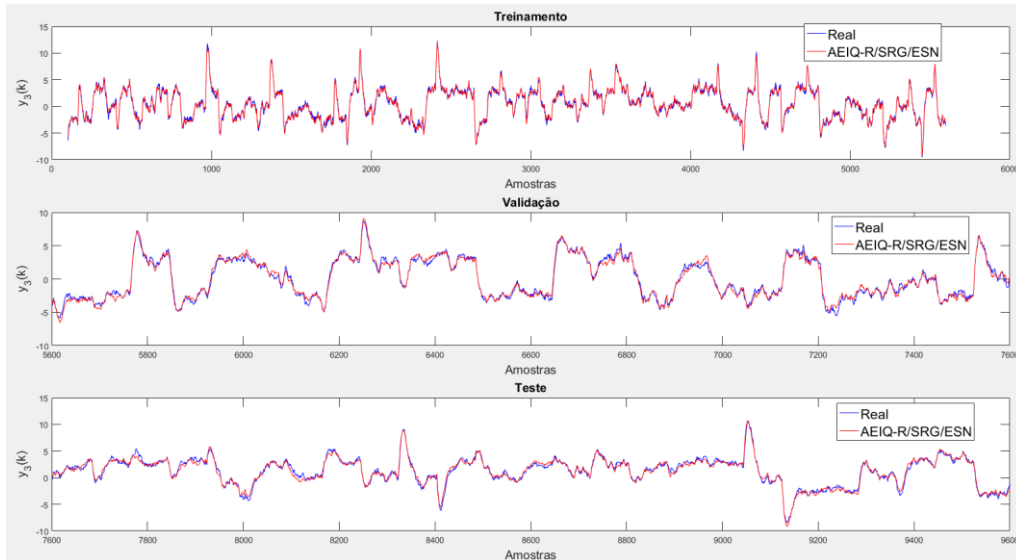


Figura 124 - Simulação da variável de saída y_3 do estudo de caso 7 utilizando a melhor solução do submodelo AEIQ-R/SRG/ESN

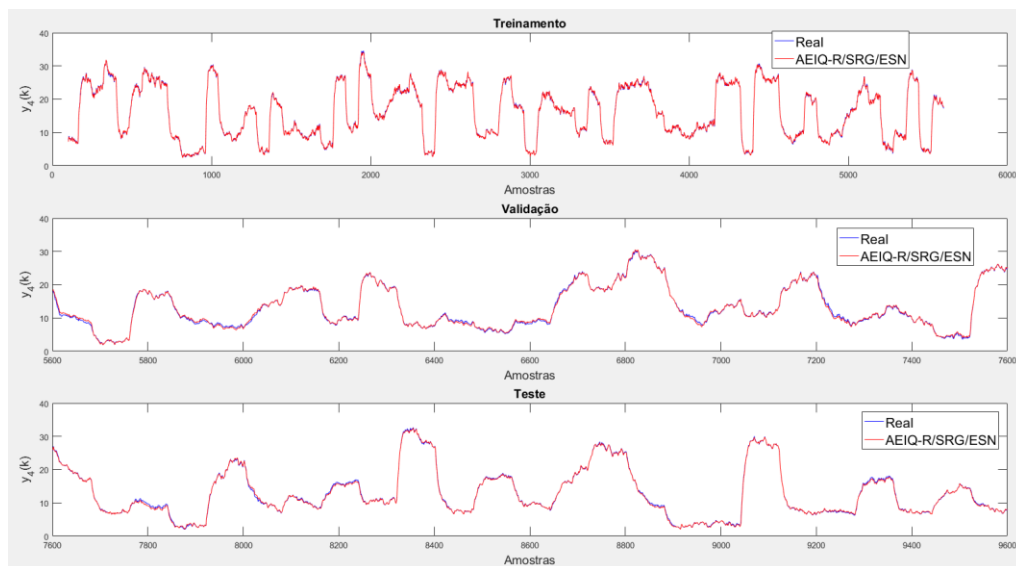


Figura 125 - Simulação da variável de saída y_4 do estudo de caso 7 utilizando a melhor solução do submodelo AEIQ-R/SRG/ESN

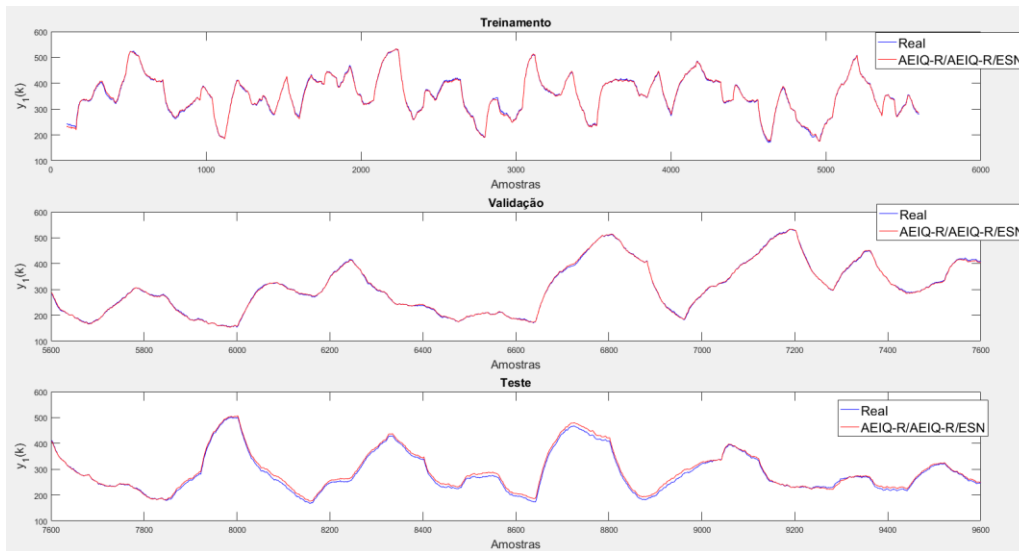


Figura 126 - Simulação da variável de saída y_1 do estudo de caso 7 utilizando a melhor solução do submodelo AEIQ-R/AEIQ-R/ESN

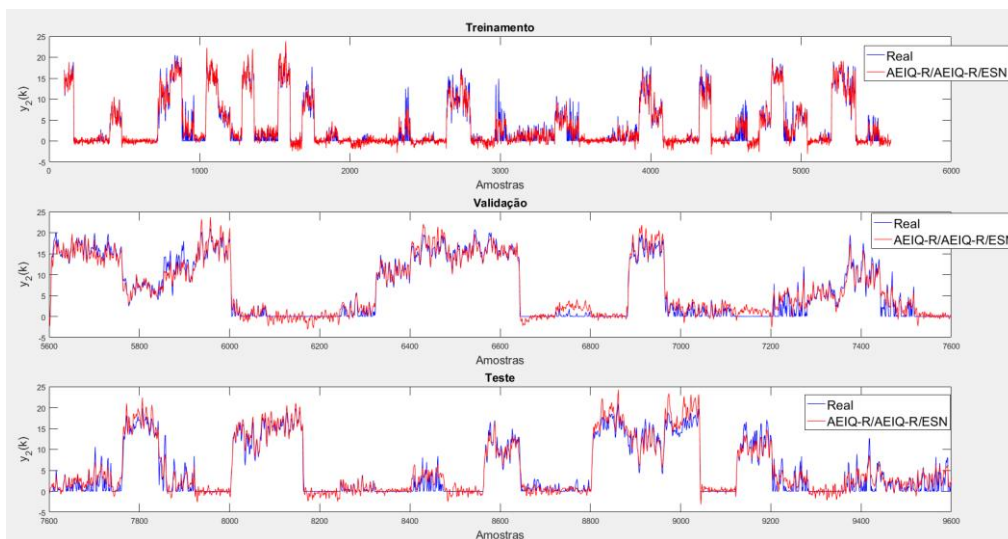


Figura 127 - Simulação da variável de saída y_2 do estudo de caso 7 utilizando a melhor solução do submodelo AEIQ-R/AEIQ-R/ESN

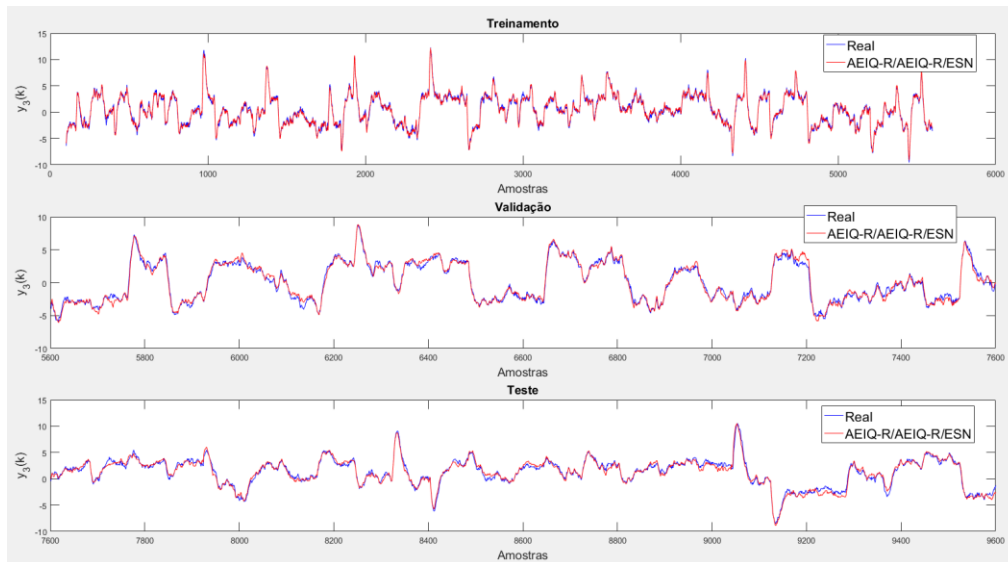


Figura 128 - Simulação da variável de saída y_3 do estudo de caso 7 utilizando a melhor solução do submodelo AEIQ-R/AEIQ-R/ESN

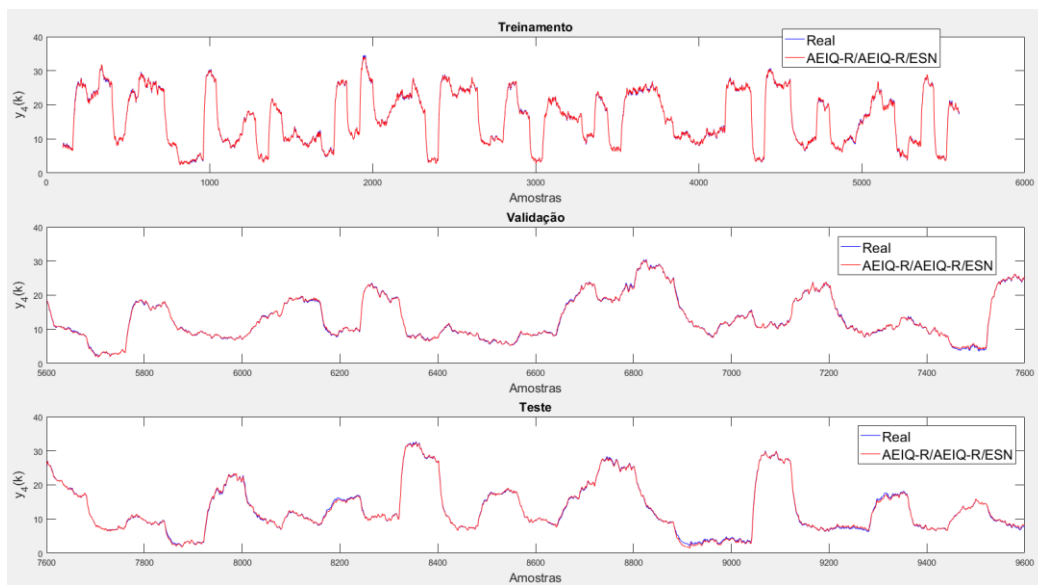


Figura 129 - Simulação da variável de saída y_4 do estudo de caso 7 utilizando a melhor solução do submodelo AEIQ-R/AEIQ-R/ESN

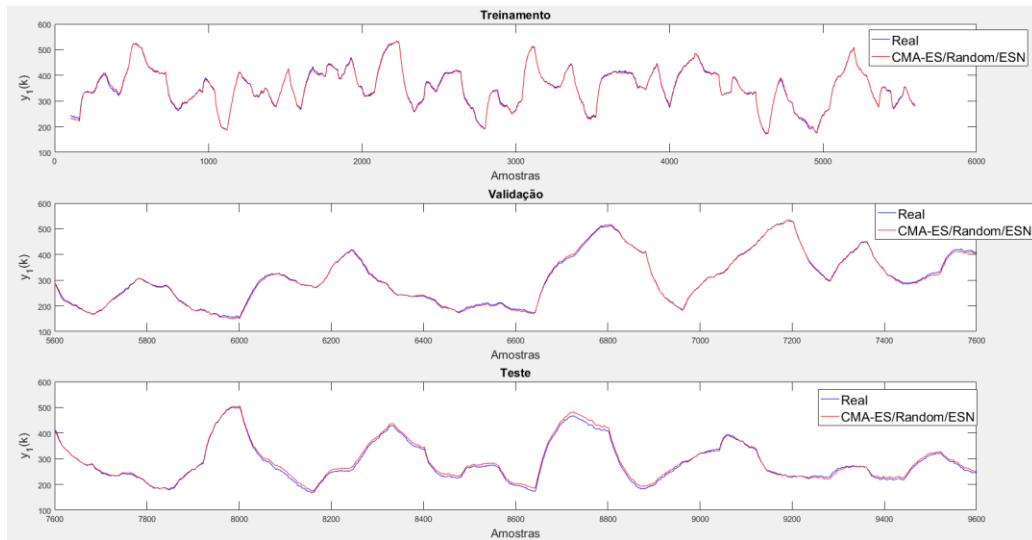


Figura 130 - Simulação da variável de saída y_1 do estudo de caso 7 utilizando a melhor solução do submodelo CMA-ES/Random/ESN

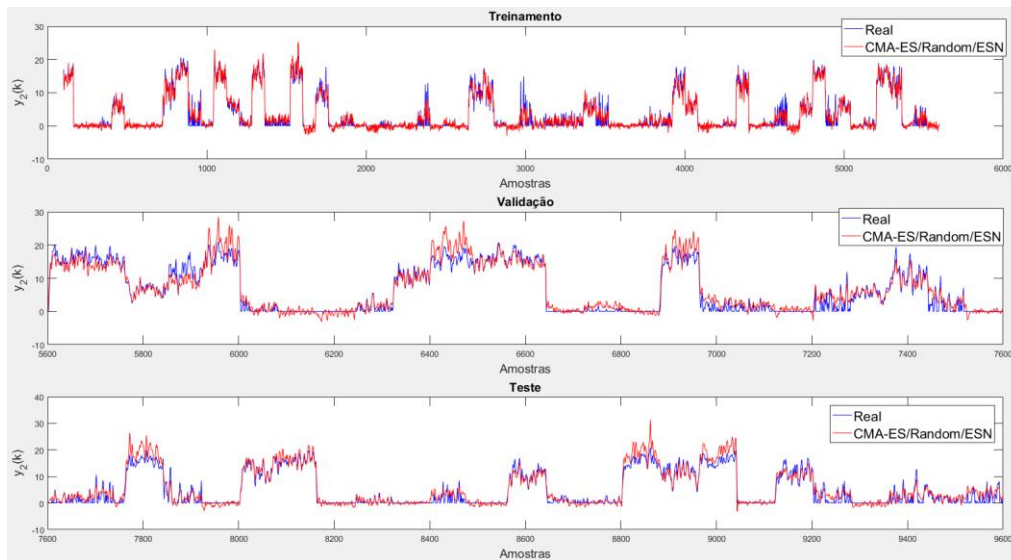


Figura 131 - Simulação da variável de saída y_2 do estudo de caso 7 utilizando a melhor solução do submodelo CMA-ES/Random/ESN

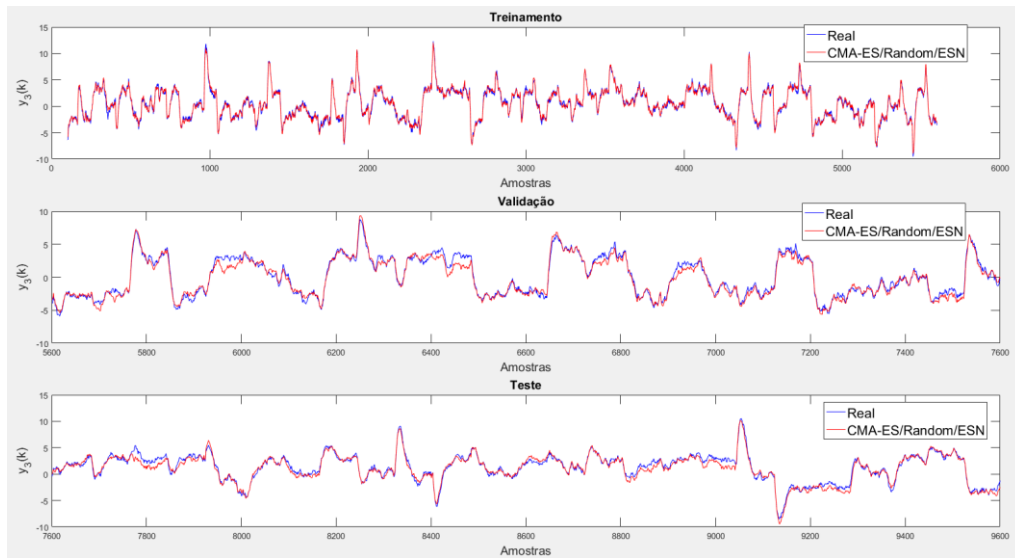


Figura 132 - Simulação da variável de saída y_3 do estudo de caso 7 utilizando a melhor solução do submodelo CMA-ES/Random/ESN

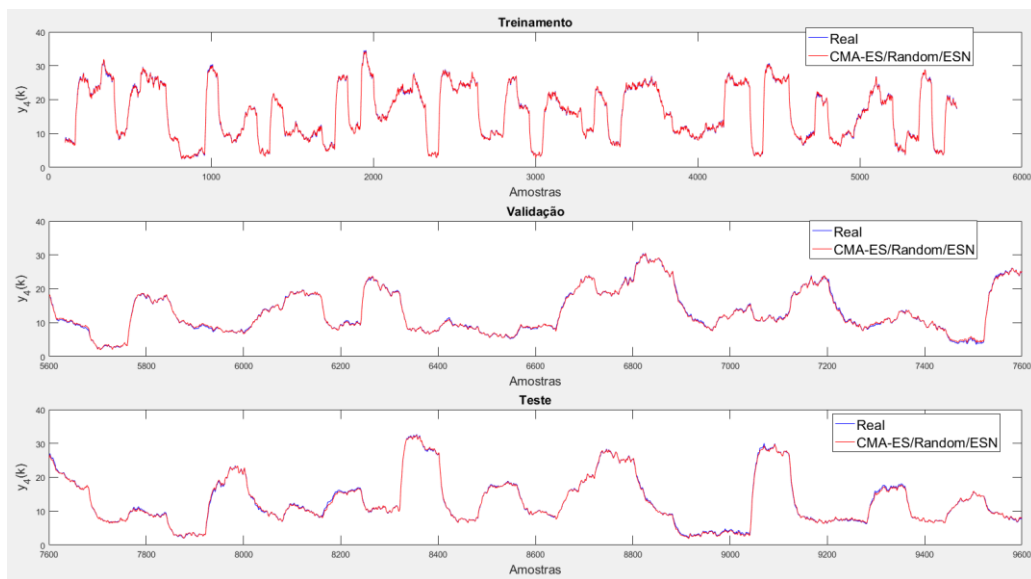


Figura 133 - Simulação da variável de saída y_4 do estudo de caso 7 utilizando a melhor solução do submodelo CMA-ES/Random/ESN

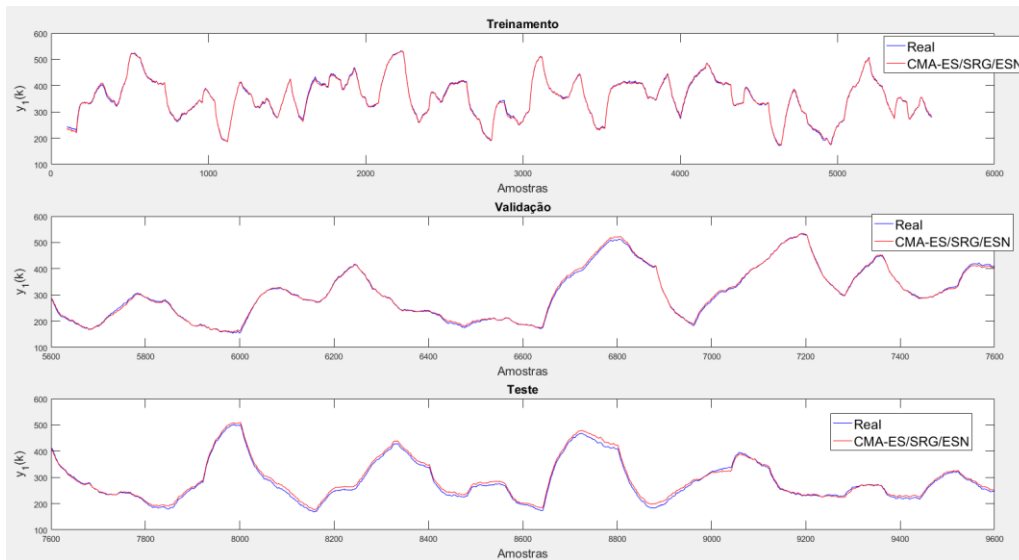


Figura 134 - Simulação da variável de saída y_1 do estudo de caso 7 utilizando a melhor solução do submodelo CMA-ES/SRG/ESN

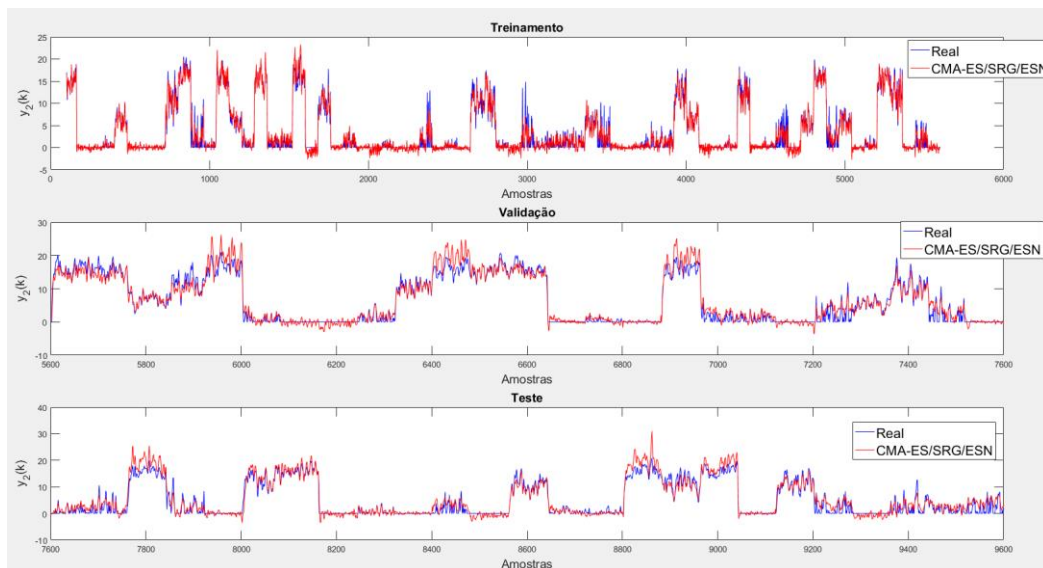


Figura 135 - Simulação da variável de saída y_2 do estudo de caso 7 utilizando a melhor solução do submodelo CMA-ES/SRG/ESN

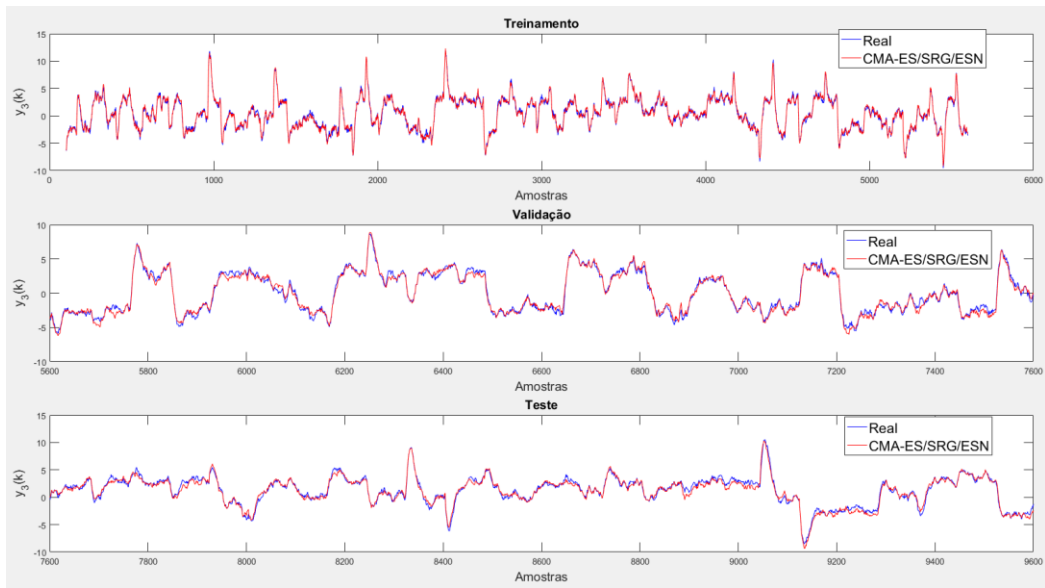


Figura 136 - Simulação da variável de saída y_3 do estudo de caso 7 utilizando a melhor solução do submodelo CMA-ES/SRG/ESN

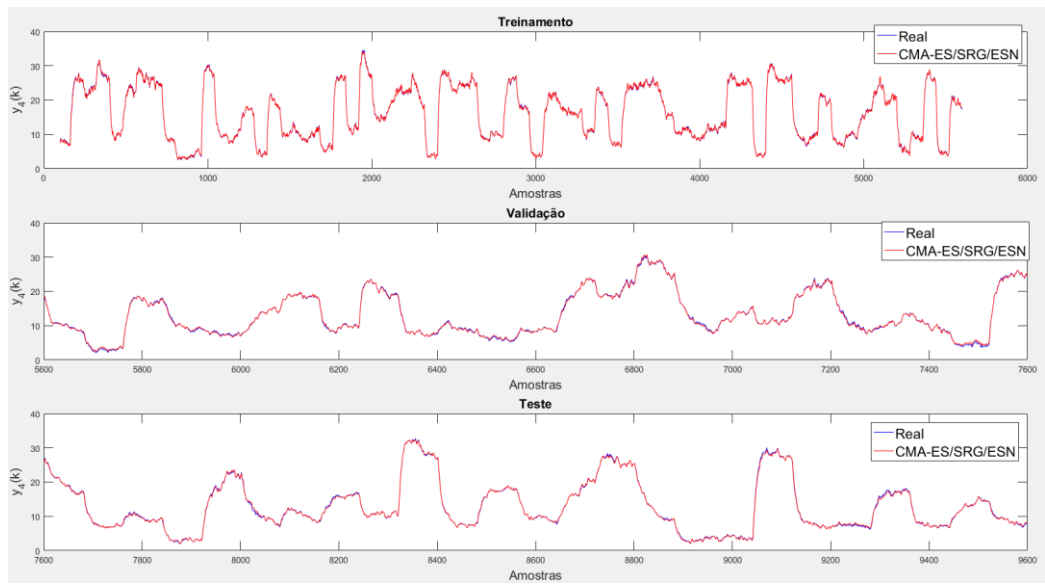


Figura 137 - Simulação da variável de saída y_4 do estudo de caso 7 utilizando a melhor solução do submodelo CMA-ES/SRG/ESN

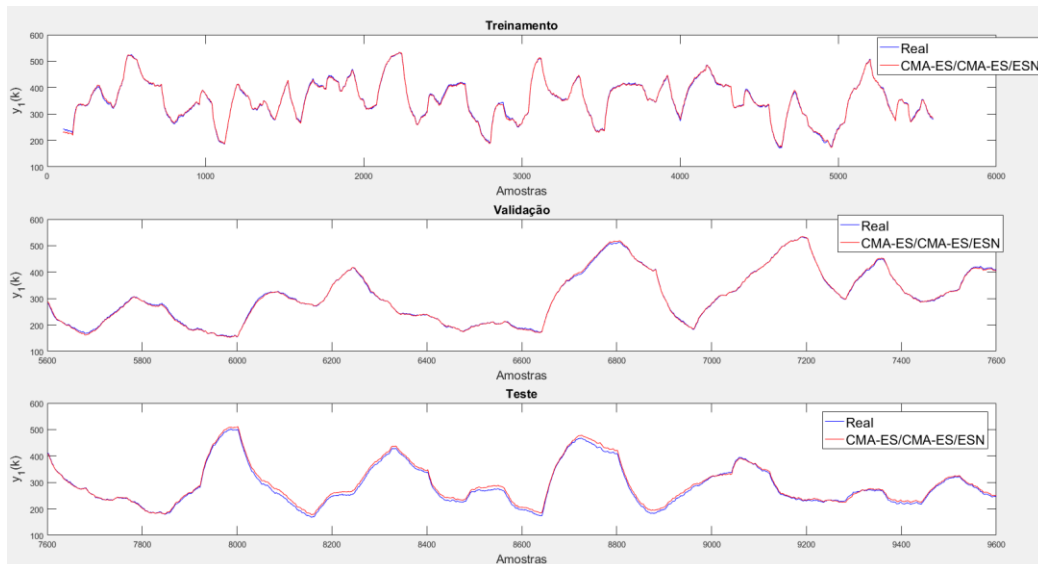


Figura 138 - Simulação da variável de saída y_1 do estudo de caso 7 utilizando a melhor solução do submodelo CMA-ES/CMA-ES/ESN

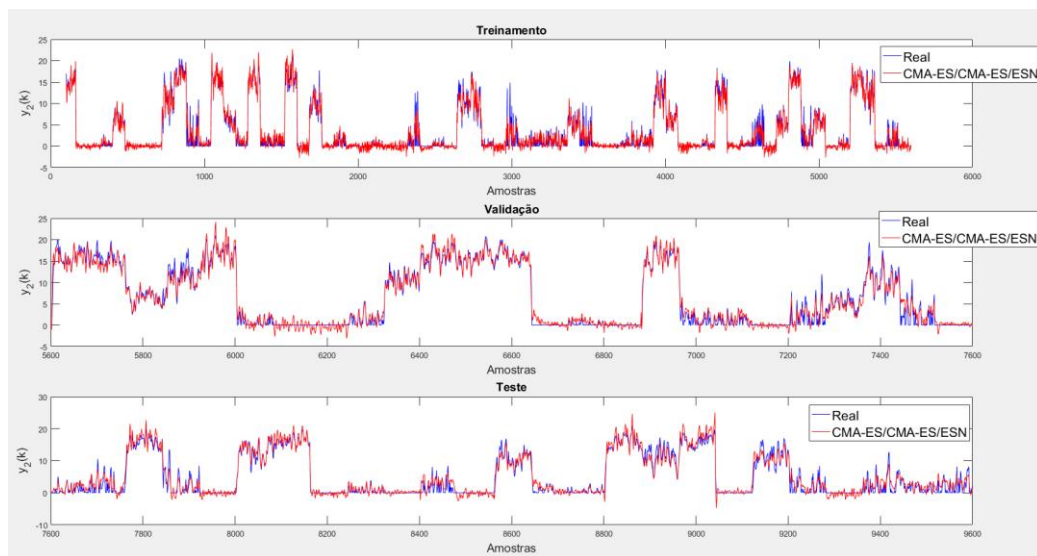


Figura 139 - Simulação da variável de saída y_2 do estudo de caso 7 utilizando a melhor solução do submodelo CMA-ES/CMA-ES/ESN

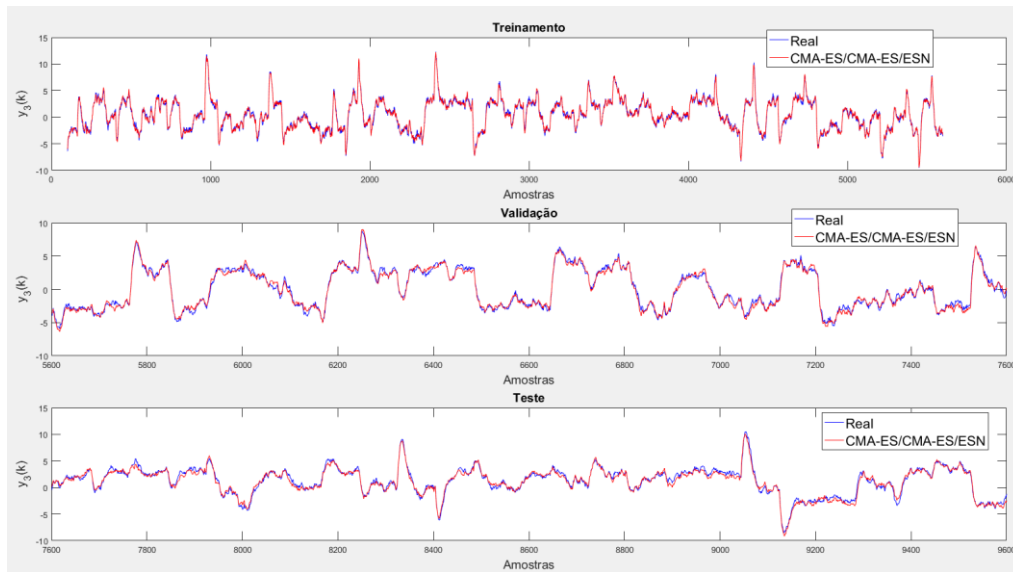


Figura 140 - Simulação da variável de saída y_3 do estudo de caso 7 utilizando a melhor solução do submodelo CMA-ES/CMA-ES/ESN

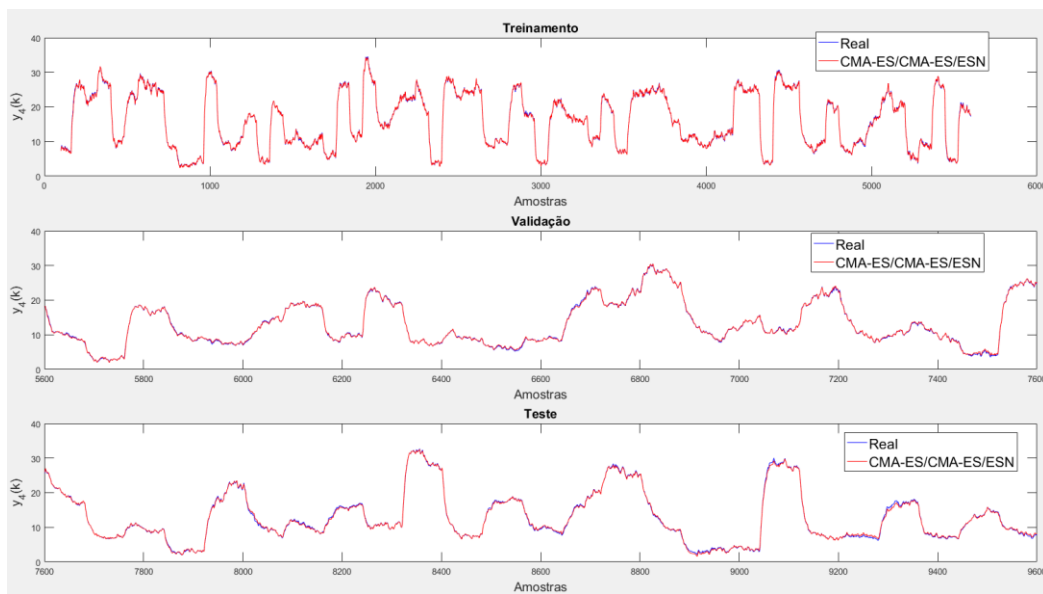


Figura 141 - Simulação da variável de saída y_4 do estudo de caso 7 utilizando a melhor solução do submodelo CMA-ES/CMA-ES/ESN

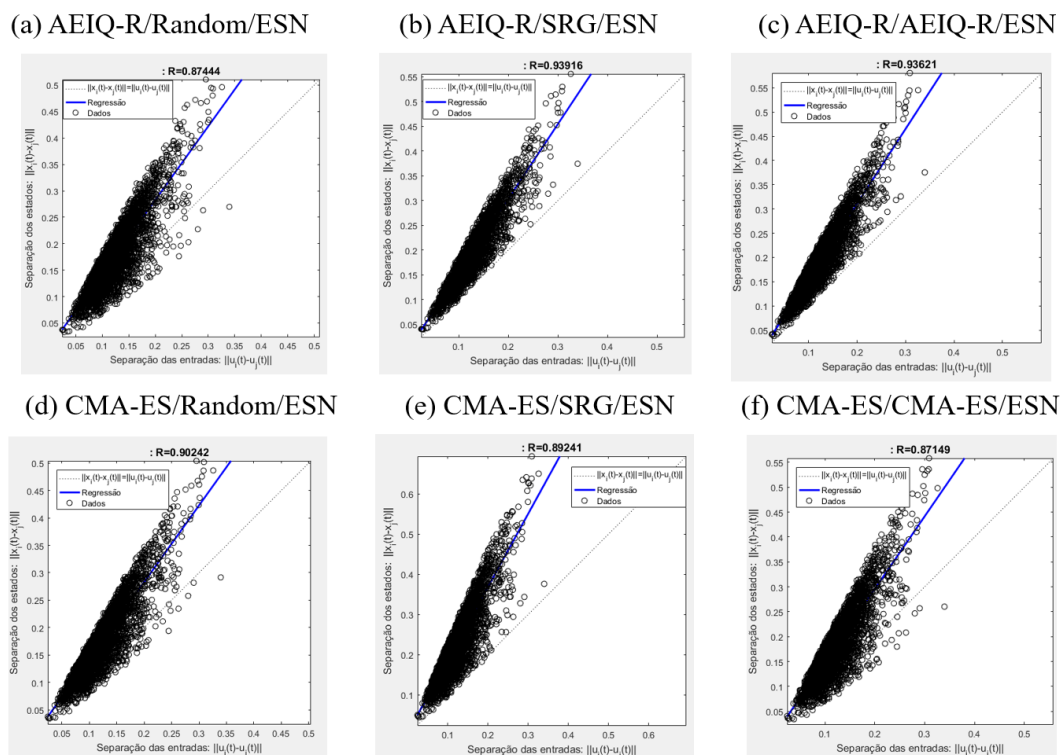


Figura 142 - Gráficos de razão de separação encontrados para as melhores soluções de cada submodelo para o estudo de caso 7

Como mostrado pelas figuras 118 a 141, as melhores soluções de cada submodelo dentre os 10 experimentos conseguiram simular adequadamente as variáveis de saída do sistema, apesar de terem apresentado uma dificuldade maior na simulação das variáveis y_2 e y_3 . Com relação à figura 142, os gráficos de razão de separação apresentaram comportamentos similares dentro da zona caótica.

A tabela 39 compara os resultados obtidos pelas melhores soluções dos submodelos (sem considerar os submodelos AEIQ-R/SRG/ESN e CMA-ES/SRG/ESN, que geraram soluções instáveis) com os resultados dos métodos de Aleksovski et al. (2016), que utilizaram as mesmas amostras de teste.

Tabela 39 - Comparação de resultados com outros métodos para o *benchmark Gerador de Vapor*

Método	Métrica RRSE
AEIQ-R/AEIQ-R/ESN (média)	0,111610
CMA-ES/CMA-ES/ESN (média)	0,112059
AEIQ-R/Random/ESN (média)	0,113800
CMA-ES/Random/ESN (média)	0,115698
$Bagg_s(SO)$ (Aleksovski et al., 2016)	0,195709
$Bagg_p(SO)$ (Aleksovski et al., 2016)	0,197303
$Lol_p(SO)$ (Aleksovski et al., 2016)	0,198113
$Lol_s(SO)$ (Aleksovski et al., 2016)	0,198457
$Lol_s(MO)$ (Aleksovski et al., 2016)	0,201441
$Bagg_s(MO)$ (Aleksovski et al., 2016)	0,202078
$Bagg_p(MO)$ (Aleksovski et al., 2016)	0,209603
$Lol_p(MO)$ (Aleksovski et al., 2016)	0,210321

Como mostrado pela tabela 39, os médios de RRSE durante o teste dos modelos testados nesta dissertação superaram os resultados obtidos pelos métodos de Aleksovski et al. (2016). Pela mesma tabela, vê-se que o submodelo AEIQ-R/AEIQ-R/ESN apresenta desempenho um pouco superior em comparação ao método CMA-ES/CMA-ES/ESN em termos de RRSE, apesar de inicialmente ter sido superado por este último método em termos de RMSE.

6.4.8. Estudo de caso 8 – Braço do leitor de CD

A tabela 40 exibe os resultados obtidos durante os 10 experimentos pelos diferentes submodelos derivados do método geral para o estudo de caso 8 – *Braço do leitor de CD*, com o AEIQ-R e o CMA-ES configurados conforme tabelas 7 e 8 e com a base de dados dividida como mostrado na tabela 6:

Tabela 40 – Resultados obtidos de RMSE pelos diferentes submodelos para o *benchmark Braço do leitor de CD* nos 10 experimentos realizados

Submodelo	Treinamento	Validação	Teste			
	Média	média	média	Melhor	pior	desvio
AEIQ-R/Random/ESN	0,1728227	0,122075	0,147886	0,133926	0,163356	0,007995
AEIQ-R/SRG/ESN	0,584875	4,698573	7,033276	0,148541	55,4144	17,331469
AEIQ-R/AEIQ-R/ESN	0,633507	0,092977	0,136512	0,126708	0,150908	0,007709
CMA-ES/Random/ESN	0,199223	0,127207	0,251889	0,145730	1,023578	0,271572
CMA-ES/SRG/ESN	1,101475	1,073561	1,033491	0,152585	4,745321	1,822788
CMA-ES/CMA-ES/ESN	1,805230	0,074024	0,134590	0,114600	0,153759	0,013112

Pelos resultados mostrados na tabela 40, vê-se que os métodos AEIQ-R/AEIQ-R/ESN e CMA-ES/CMA-ES/ESN foram os que, em média, apresentaram melhor desempenho durante a simulação do conjunto de teste. Os submodelos AEIQ-R/SRG/ESN e CMA-ES/SRG/ESN obtiveram o pior desempenho entre os testados, pois produziram algumas soluções instáveis. As figuras 143 e 144 mostram as curvas de evolução dos algoritmos AEIQ-R e CMA-ES nas etapas 1 e 2, exibindo-se a média da aptidão nos 10 experimentos do melhor indivíduo da população, enquanto a tabela 41 exhibe os valores de aptidão na última geração:

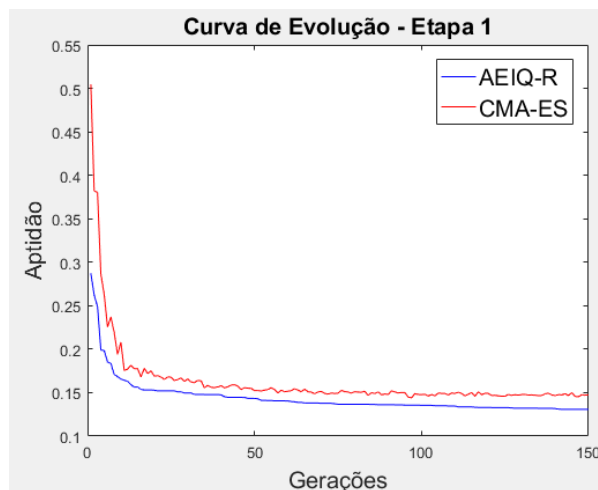


Figura 143 - Curva de evolução da etapa 1 (média da aptidão do melhor indivíduo em cada geração nos 10 experimentos) para o estudo de caso 8

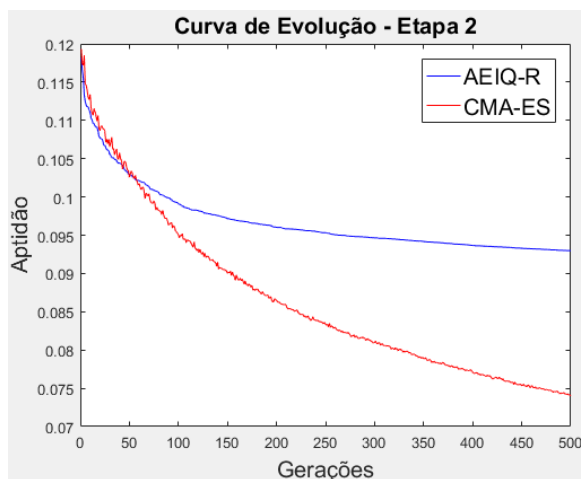


Figura 144 - Curva de evolução da etapa 2 (média da aptidão do melhor indivíduo em cada geração nos 10 experimentos) para o estudo de caso 8

Tabela 41 - Aptidão dos AEs na última geração (média de 10 experimentos) para o estudo de caso 8

Algoritmo	Aptidão na última geração (média de 10 experimentos)	
	Etapa 1	Etapa 2
AEIQ-R	0,130785	0,092977
CMA-ES	0,147421	0,074024

Pelas figuras 143 e 144, vê-se que durante a etapa 1, o AEIQ-R apresentou melhor desempenho durante a etapa 1, enquanto que na etapa 2, o CMA-ES exibiu desempenho superior, e gerando, neste caso, em média, soluções de melhor qualidade. Visualmente, ambos os algoritmos parecem não ter convergido na segunda etapa.

A tabela 42 mostra a configuração de parâmetros globais que gerou as melhores soluções de cada submodelo (dentre as 10 soluções finais de cada experimento), enquanto que as figuras 145 a 156 mostram a simulação por cada uma dessas soluções dos conjuntos de treinamento, validação e teste. A figura 157 mostra os gráficos de razão de separação para cada uma dessas soluções.

Tabela 42 - Parâmetros globais da melhor solução encontrada por cada submodelo para o estudo de caso 8

Melhor solução	N	c_w	ρ_w	s_{in}	s_{back}	α
AEIQ-R/Random/ESN	21	0,5432	0,6844	0,8243	0,4566	0,8684
AEIQ-R/SRG/ESN	18	0,4597	0,5168	0,7955	0,4141	0,6371
AEIQ-R/AEIQ-R/ESN	21	0,5432	0,6844	0,8243	0,4566	0,8684
CMA-ES/Random/ESN	17	0,4470	0,5737	0,9151	0,4002	0,6958
CMA-ES/SRG/ESN	17	0,8951	0,4162	0,9754	0,3410	0,6345
CMA-ES/CMA-ES/ESN	18	0,7250	0,5606	0,9731	0,2424	0,6075

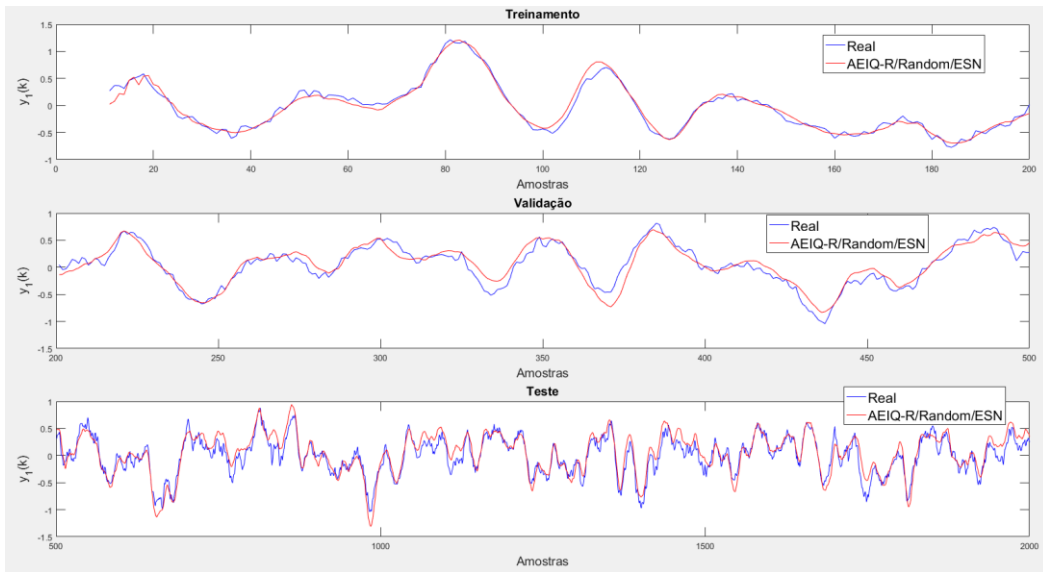


Figura 145 - Simulação da variável de saída y_1 do estudo de caso 8 utilizando a melhor solução do submodelo AEIQ-R/Random/ESN

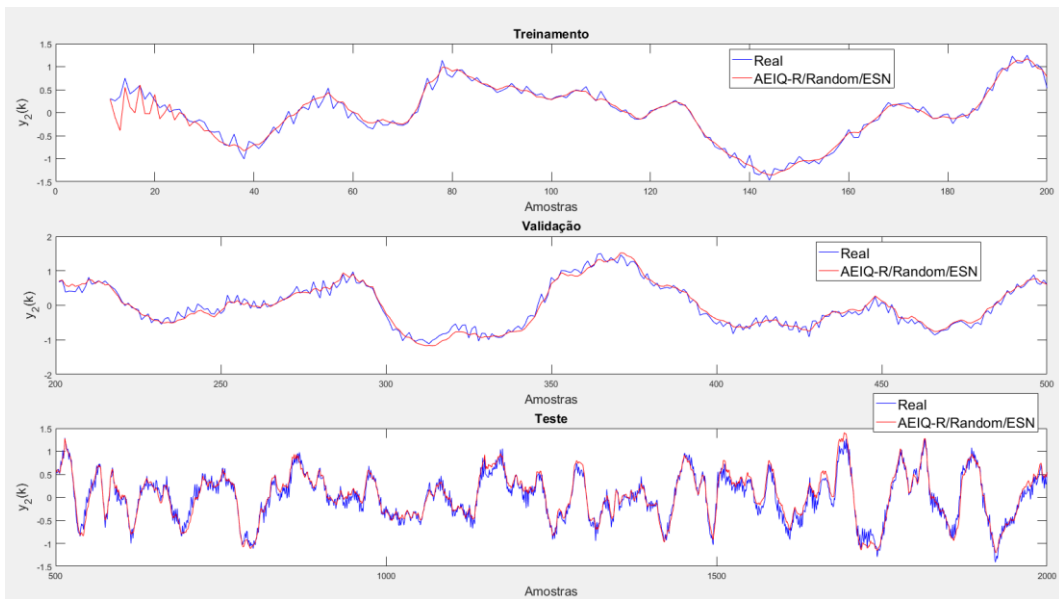


Figura 146 - Simulação da variável de saída y_2 do estudo de caso 8 utilizando a melhor solução do submodelo AEIQ-R/Random/ESN

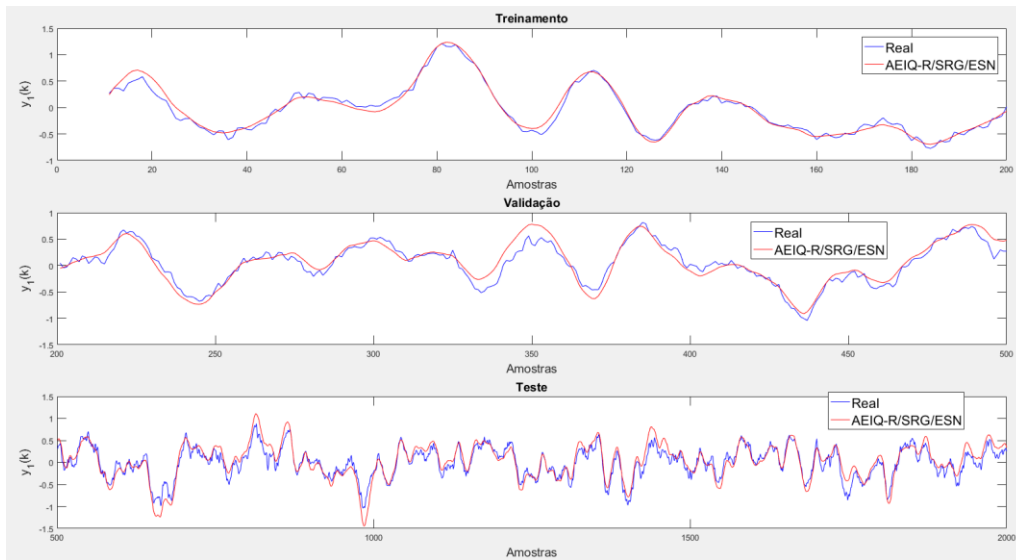


Figura 147 - Simulação da variável de saída y_1 do estudo de caso 8 utilizando a melhor solução do submodelo AEIQ-R/SRG/ESN

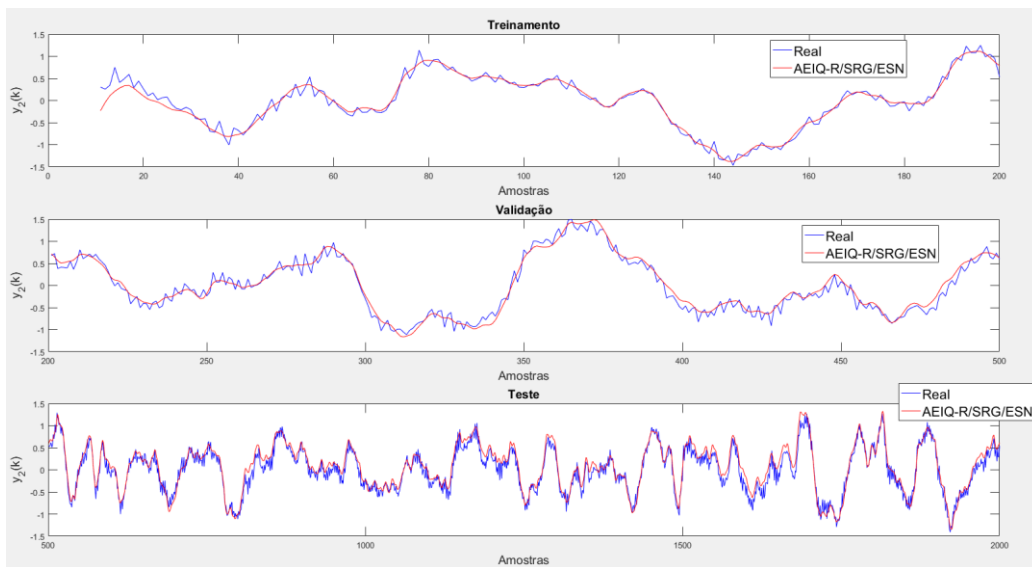


Figura 148 - Simulação da variável de saída y_2 do estudo de caso 8 utilizando a melhor solução do submodelo AEIQ-R/SRG/ESN

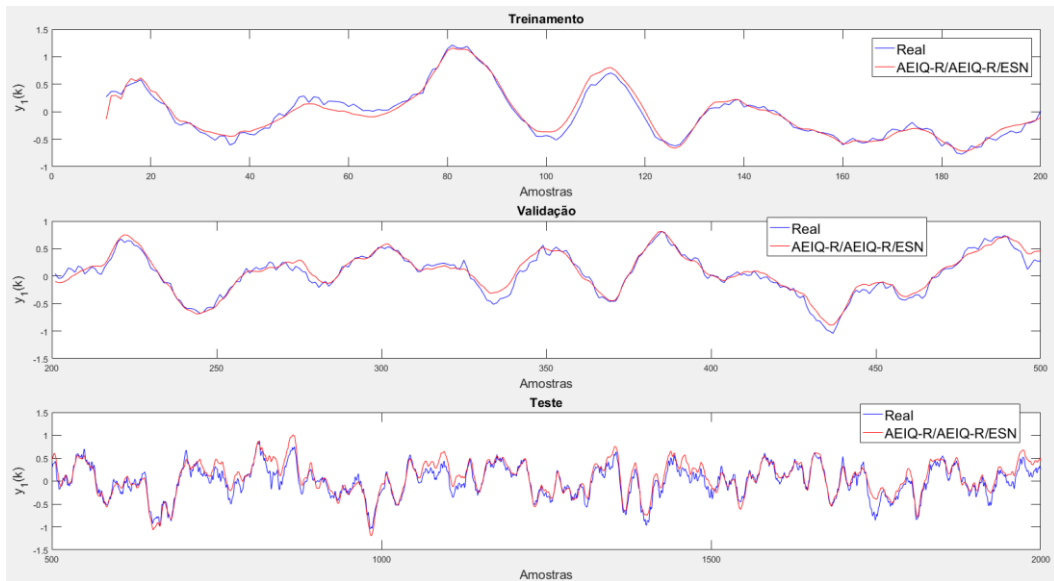


Figura 149 - Simulação da variável de saída y_1 do estudo de caso 8 utilizando a melhor solução do submodelo AEIQ-R/AEIQ-R/ESN

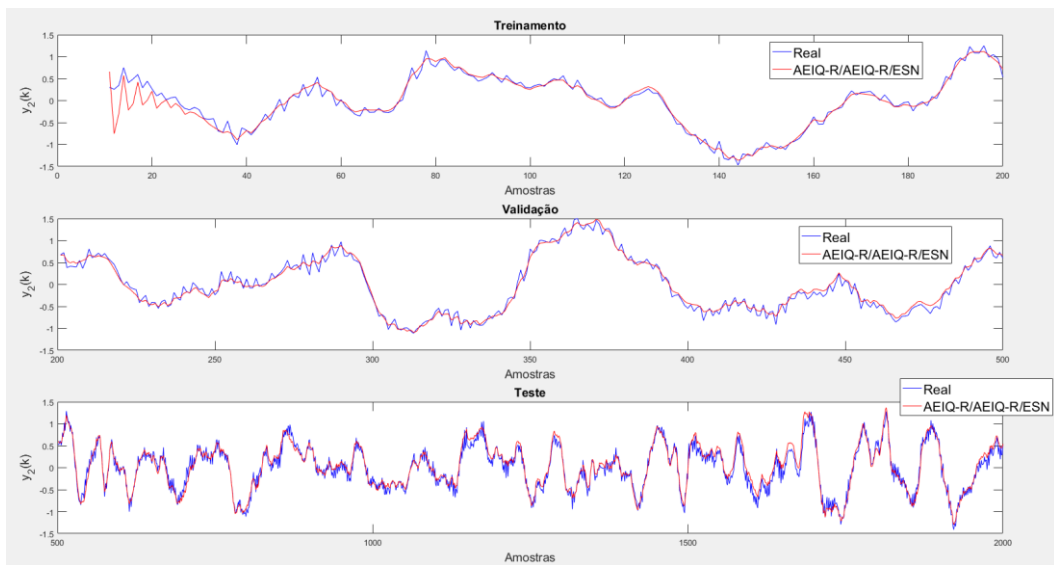


Figura 150 - Simulação da variável de saída y_2 do estudo de caso 8 utilizando a melhor solução do submodelo AEIQ-R/AEIQ-R/ESN

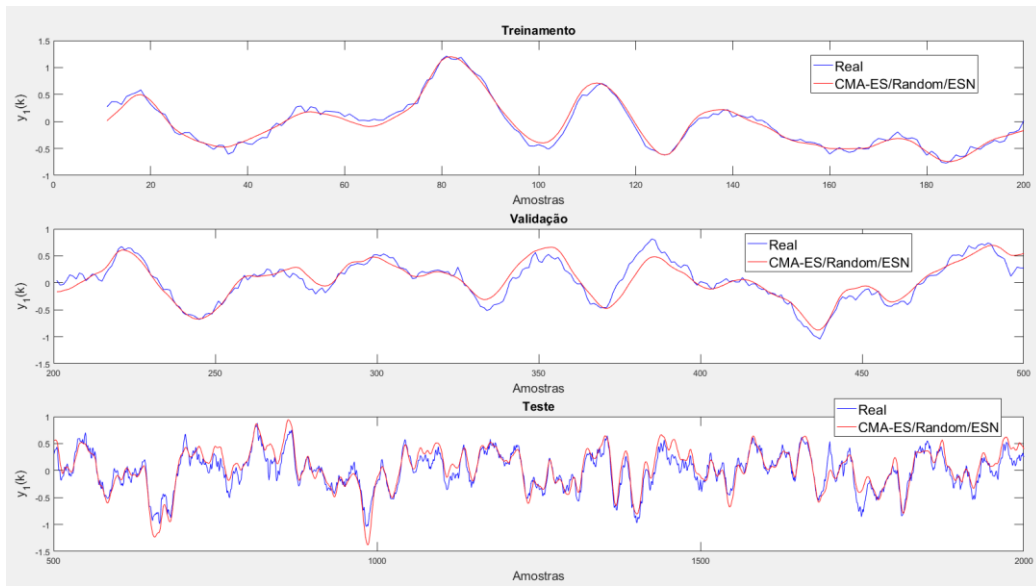


Figura 151 - Simulação da variável de saída y_1 do estudo de caso 8 utilizando a melhor solução do submodelo CMA-ES/Random/ESN

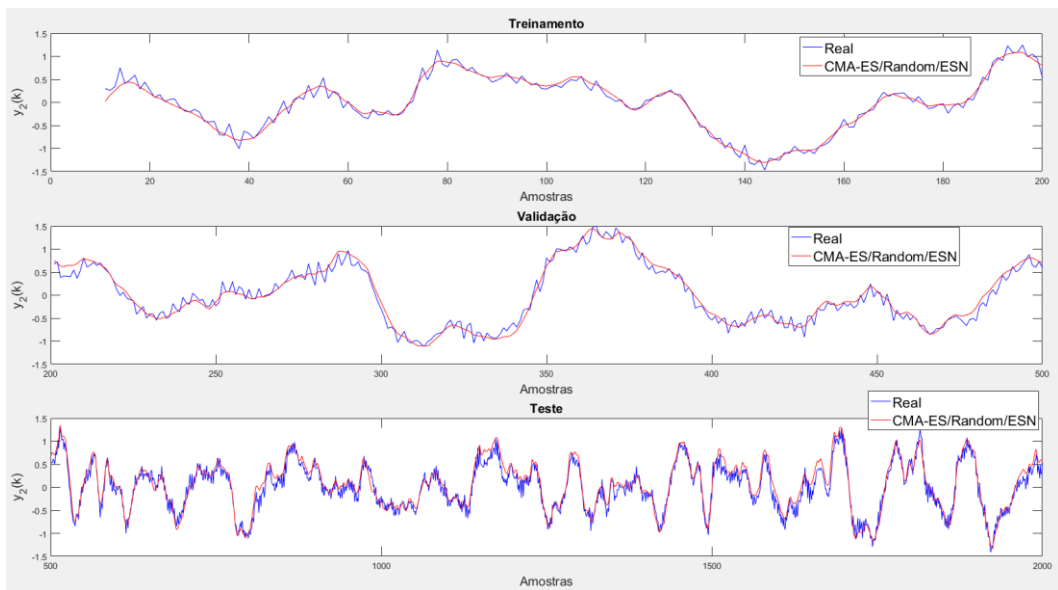


Figura 152 - Simulação da variável de saída y_2 do estudo de caso 8 utilizando a melhor solução do submodelo CMA-ES/Random/ESN

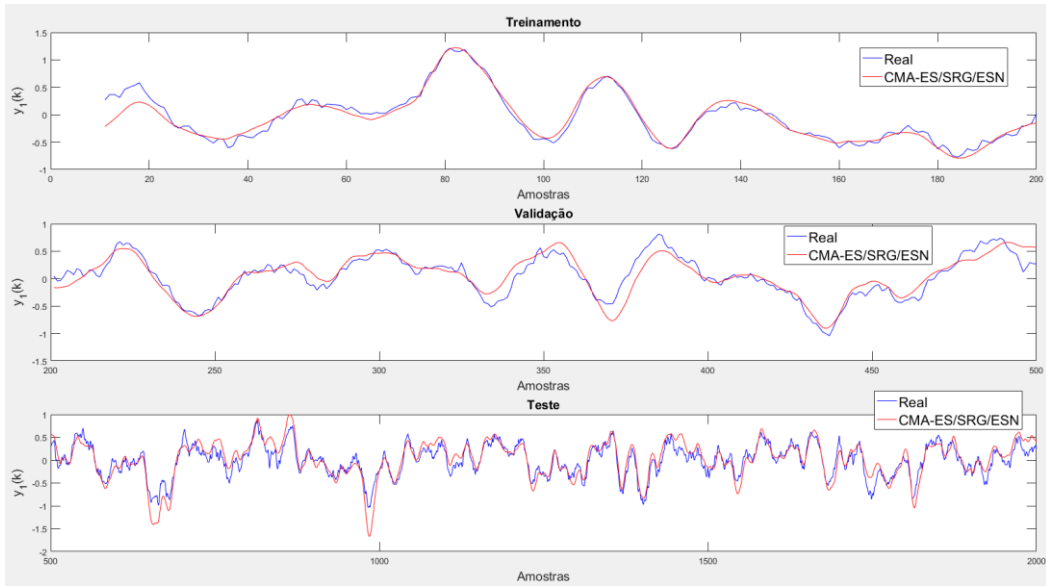


Figura 153 - Simulação da variável de saída y_1 do estudo de caso 8 utilizando a melhor solução do submodelo CMA-ES/SRG/ESN

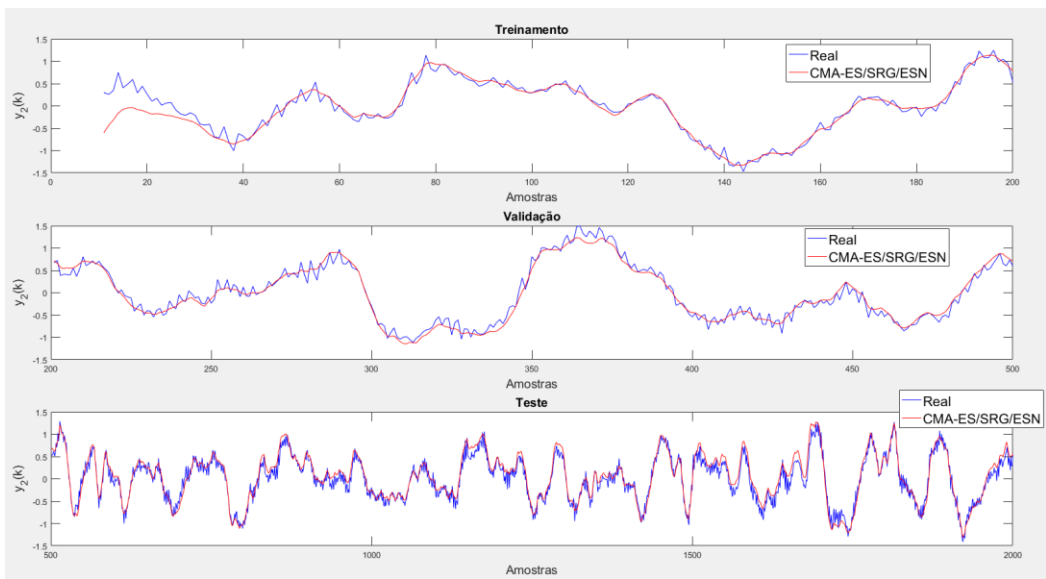


Figura 154 - Simulação da variável de saída y_2 do estudo de caso 8 utilizando a melhor solução do submodelo CMA-ES/SRG/ESN

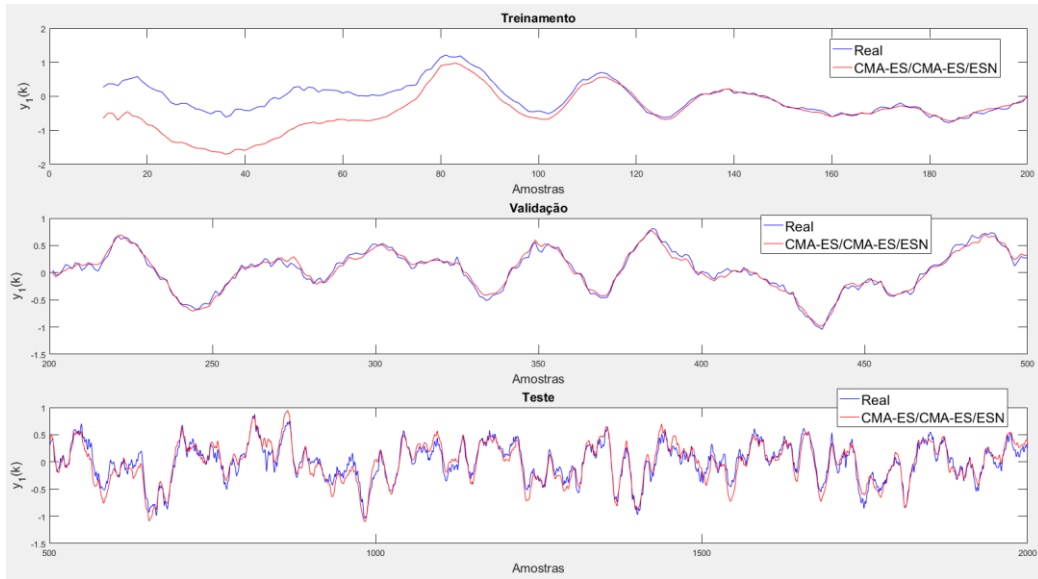


Figura 155 - Simulação da variável de saída y_1 do estudo de caso 8 utilizando a melhor solução do submodelo CMA-ES/CMA-ES/ESN

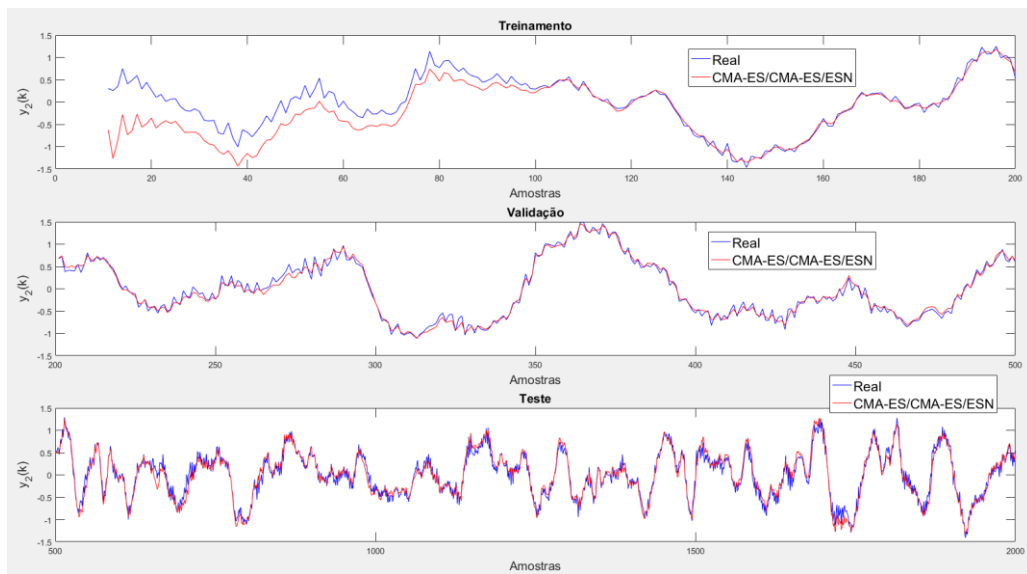


Figura 156 - Simulação da variável de saída y_2 do estudo de caso 8 utilizando a melhor solução do submodelo CMA-ES/CMA-ES/ESN

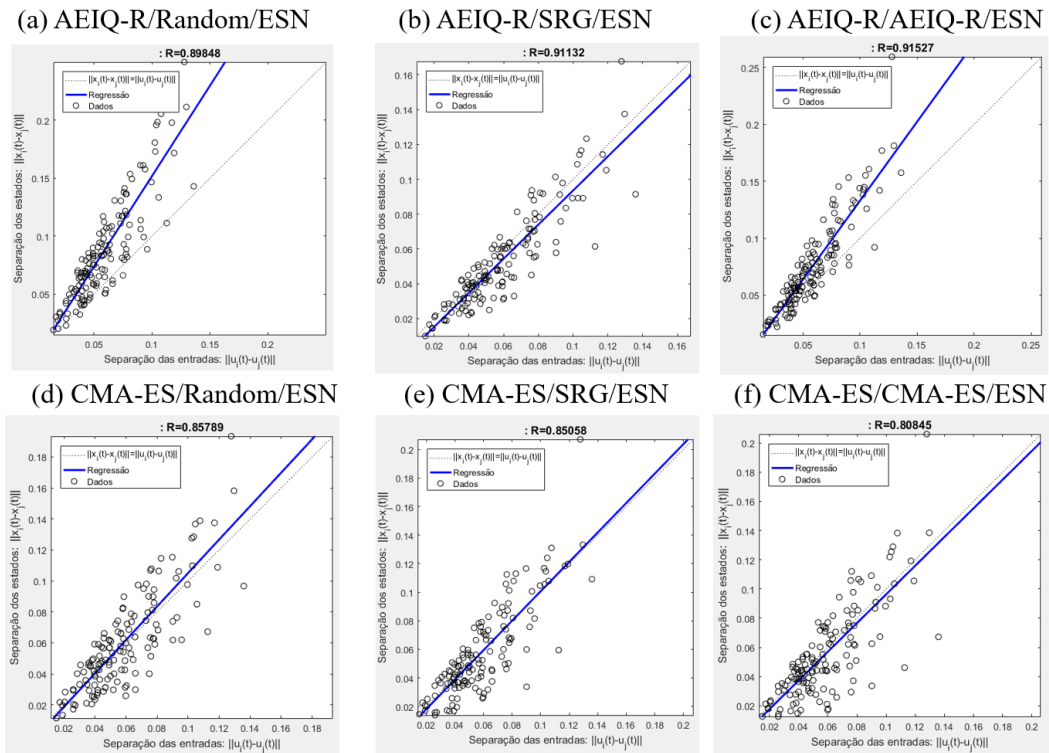


Figura 157 - Gráficos de razão de separação encontrados para as melhores soluções de cada submodelo para o estudo de caso 8

Pelas figuras 145 a 156, vê-se que as melhores soluções de cada submodelo tiveram uma maior dificuldade, em comparação aos problemas anteriores, de simular o comportamento dinâmico das variáveis de saída do sistema *Braço do leitor de CD*. O submodelo CMA-ES/CMA-ES/ESN, que retornou a solução mais precisa, conseguiu simular mais adequadamente os conjuntos de validação e teste, embora o desempenho inicial no treinamento não tenha sido bom, provavelmente pelo período de *washout* não ter sido suficiente para esta rede “esquecer” os estados iniciais. Com relação aos gráficos de razão de separação da figura 157, com exceção dos submodelos AEIQ-R/Random/ESN e AEIQ-R/AEIQ-R/ESN, todas as soluções ficaram muito próximas da região objetivo, com a solução CMA-ES/SRG/ESN praticamente cravada em cima da reta objetivo.

A tabela 43 compara os resultados obtidos pelas melhores soluções dos submodelos (sem considerar os métodos AEIQ-R/SRG/ESN e CMA-ES/SRG/ESN, que geraram soluções instáveis) com os melhores resultados dos métodos de Liu et al. (2013) e Hansson et al. (2012), que utilizaram as mesmas amostras de teste.

Tabela 43 - Comparação de resultados com outros métodos para o *benchmark Braço do leitor de CD*

Método	Métrica Fit
N4SID (Hansson et al., 2012)	73,5
Weighted Nuclear Norm (Liu et al., 2013)	73,3
NONE (Hansson et al., 2012)	72,8
MOESP (Hansson et al., 2012)	72,5
Regularized Nuclear Norm (Liu et al., 2013)	72,5
NOINSTR (Hansson et al., 2012)	72,3
CMA-ES/CMA-ES/ESN (melhor)	71,4
IVM (Hansson et al., 2012)	71,0
CVA (Hansson et al., 2012)	70,0
AEIQ-R/AEIQ-R/ESN (melhor)	68,2
AEIQ-R/Random/ESN (melhor)	66,5
CMA-ES/Random/ESN	64,0

Como mostrado pela tabela 43, somente a melhor solução do submodelo CMA-ES/CMA-ES/ESN conseguiu apresentar resultados comparáveis aos métodos de Liu et al. (2013) e Hansson et al. (2012), mas ainda assim, ficou distante do método de melhor desempenho destes autores, o N4SID, *fits* de 71,4 e 73,5, respectivamente. Os autores Liu et al. (2013) e Hansson et al. (2012) são os mesmos das comparações nos estudos de caso 1, 2 e 3. No estudo de caso 3, também os métodos testados nesta dissertação apresentaram desempenho inferior aos métodos de Liu et al. (2013) e Hansson et al. (2012). Naquela ocasião, levantou-se a possibilidade de a divisão da base de dados, que para comparação com Liu et al. (2013) e Hansson et al. (2012), possui mais amostras para teste (1000) do que para estimação do modelo (500, que foi dividida entre treinamento – 200 amostras e teste – 300 amostras), possa ter influenciado nos resultados. Como uma forma de comparar o desempenho dos submodelos propostos com o desempenho de outro trabalho na literatura (Dudul & Ghatol, 2003), que utiliza uma divisão de dados diferente, com 1200 dados para estimação, que foram igualmente divididos entre treinamento e validação, e 839 dados para teste, um experimento extra, mantendo as mesmas configurações dos AEs, foi realizado, cujos resultados estão mostrados nas tabelas 44 e 45.

Tabela 44 – Resultados para o estudo de caso 8 – experimento extra

Submodelo	Treinamento	Validação	Teste			
	Média	média	média	melhor	pior	desvio
AEIQ-R/Random/ESN	0,273799	0,124428	0,129247	0,118935	0,143824	0,007824
AEIQ-R/SRG/ESN	1,255270	1,123069	1,115917	0,121995	9,934716	3,098630
AEIQ-R/AEIQ-R/ESN	0,295671	0,083815	0,119050	0,114114	0,127612	0,004606
CMA-ES/Random/ESN	0,087551	0,135294	0,139023	0,117933	0,196253	0,021827
CMA-ES/SRG/ESN	0,089474	0,164546	0,138609	0,128259	0,156201	0,009546
CMA-ES/CMA-ES/ESN	0,079655	0,079425	0,114247	0,105883	0,121897	0,004774

Tabela 45 - Comparação de resultados com os métodos de (Dudul; Ghatol, 2003) para o estudo de caso 8 – experimento extra

Método	Métrica	
	fit	
	y_1	y_2
CMA-ES/CMA-ES/ESN (melhor)	68,65	79,89
MLP (Dudul & Ghatol, 2003)	66,07	78,48
AEIQ-R/AEIQ-R/ESN (melhor)	65,81	78,56
CMA-ES/Random/ESN (melhor)	67,35	76,03
AEIQ-R/Random/ESN (melhor)	67,07	75,92
State Space (Dudul & Ghatol, 2003)	61,63	75,42

Como mostrado pela tabela 44, vê-se que com esta nova divisão de dados, o desempenho médio de todos os submodelos melhorou durante a simulação do conjunto de teste. Ainda, o método CMA-ES/CMA-ES/ESN continuou sendo o melhor dentre os testados, sendo que sua melhor solução superou os resultados obtidos pelos métodos MLP e State Space de Dudul & Ghatol, (2003) levando em consideração o *fit* para as duas saídas, conforme mostrado na tabela 45. Fazendo-se uma média dos *fits* das duas saídas atingidos pela melhor solução do método CMA-ES/CMA-ES/ESN, chegaria-se a um valor de *fit* de 74,27, que superaria o valor de *fit* de 73,5 do método N4SID de Liu et al. (2013). Tais resultados indicam que a subdivisão empregada pelos autores Liu et al. (2013) e Hansson et al. (2012) degradou o desempenho das ESNs, o que pode ter levado ao desempenho inferior dos submodelos desta dissertação frente aos métodos de Liu et al. (2013) e Hansson et al. (2012), como mostrado na tabela 43. Para finalizar esta comparação e confirmar tais hipóteses, deve-se verificar o desempenho dos métodos de

Liu et al. (2013) e Hansson et al. (2012) nesta nova divisão de dados, o que não é possível de ser feito neste trabalho.

Uma outra hipótese para tentar justificar os resultados inferiores das ESNs neste estudo de caso poderia ser a de que os resultados apresentados por Liu et al. (2013) e Hansson et al. (2012) seriam de uma previsão de 1 passo à frente e não de uma simulação livre. Tal hipótese, muito provavelmente, está descartada, uma vez que o pior modelo da tabela 45, o *State Space*, consegue atingir valores de *fit* muito maiores que os de simulação, 85,99 e 79,65 para as saídas y_1 e y_2 , respectivamente, como mostrado no trabalho de Dudul & Ghatol (2003), o que também indica que os resultados de previsão de 1 passo à frente não são confiáveis em uma comparação de diferentes métodos, como também é discutido por Dudul & Ghatol (2003).

6.4.9. Estudo de caso 9 – Caixa de prata

A tabela 46 exibe os resultados obtidos durante os 10 experimentos pelos diferentes submodelos derivados do método geral para o estudo de caso 9 – *Caixa de prata*, com o AEIQ-R e o CMA-ES configurados conforme tabelas 7 e 8 e com a base de dados dividida como mostrado na tabela 6:

Tabela 46 – Resultados obtidos de RMSE pelos diferentes submodelos para o *benchmark Caixa de prata* nos 10 experimentos realizados

Submodelo	Treinamento	Validação	Teste			
	Média	média	média	melhor	pior	desvio
AEIQ-R/Random/ESN	0,202894	0,210172	0,409211	0,331630	0,602864	0,087813
AEIQ-R/SRG/ESN	0,203319	0,212544	0,379212	0,327616	0,471601	0,041063
AEIQ-R/AEIQ-R/ESN	0,202563	0,208189	0,363232	0,328428	0,404311	0,029158
CMA-ES/Random/ESN	0,200760	0,209077	0,392883	0,324921	0,553672	0,078926
CMA-ES/SRG/ESN	0,201272	0,210714	0,354804	0,307495	0,399120	0,030595
CMA-ES/CMA-ES/ESN	0,200756	0,207639	0,385372	0,343025	0,461988	0,041075

Pelos resultados mostrados pela tabela 46, neste estudo de caso, o submodelo CMA-ES/SRG/ESN obteve, em média, o melhor desempenho durante a simulação do conjunto de teste. Tais resultados divergem dos encontrados nos estudos de caso anteriores, onde, na maioria das situações, as abordagens Random e Otimização do

reservatório superaram as abordagens que escolhiam o reservatório na segunda etapa com base no método SRG. Tal fato pode ser explicado pelo fato de que, devido ao tamanho enorme da base de dados (131072 amostras), limitou-se o número de gerações utilizadas durante a evolução (30), bem como o tamanho da população (15 na etapa 1 e 30 na etapa 2), o que pode ter sido insuficiente para que ambos AEs explorassem bem o espaço de busca e encontrassem soluções ainda melhores. As figuras 158 e 159 mostram as curvas de evolução dos algoritmos AEIQ-R e CMA-ES nas etapas 1 e 2, exibindo-se a média da aptidão nos 10 experimentos do melhor indivíduo da população, enquanto a tabela 47 exhibe os valores de aptidão na última geração:

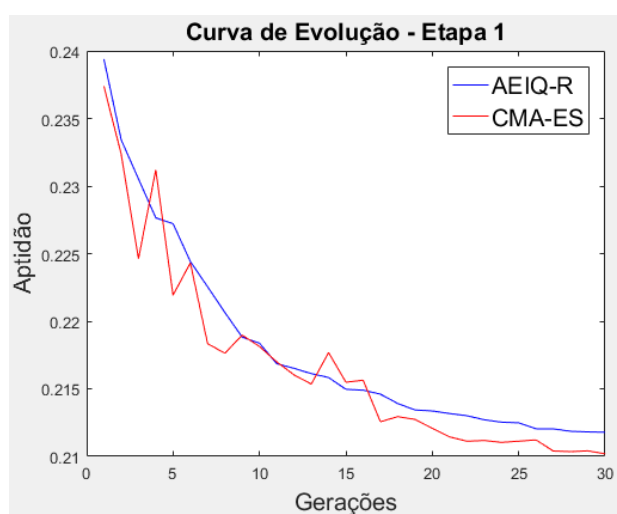


Figura 158 - Curva de evolução da etapa 1 (média da aptidão do melhor indivíduo em cada geração nos 10 experimentos) para o estudo de caso 9

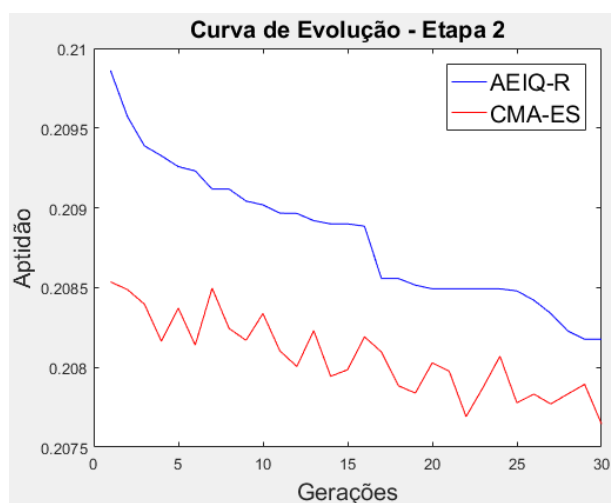


Figura 159 - Curva de evolução da etapa 2 (média da aptidão do melhor indivíduo em cada geração nos 10 experimentos) para o estudo de caso 9

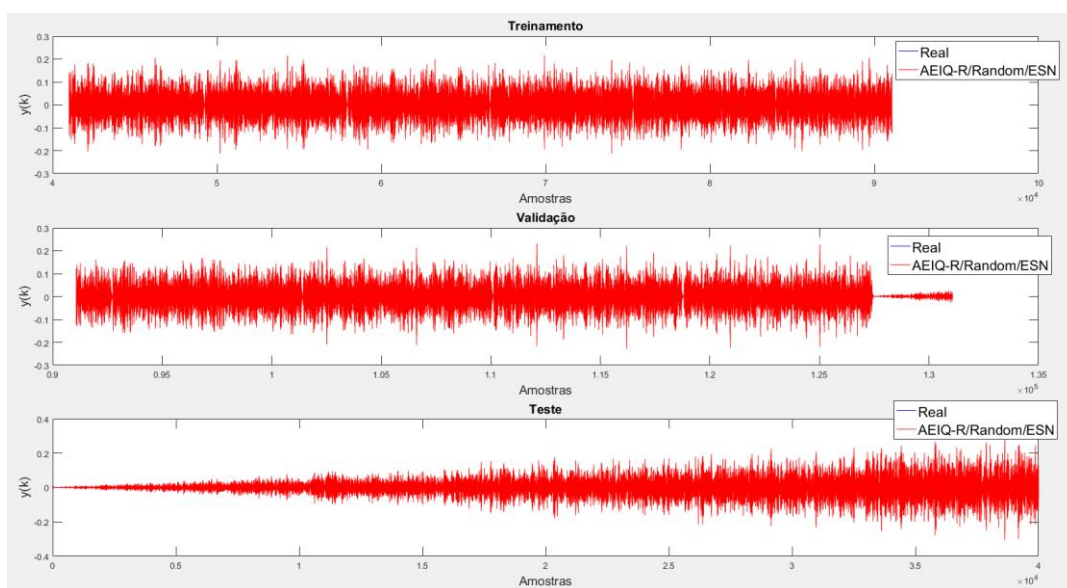
Tabela 47 - Aptidão dos AEs na última geração (média de 10 experimentos) para o estudo de caso 9

Algoritmo	Aptidão na última geração (média de 10 experimentos)	
	Etapa 1	Etapa 2
AEIQ-R	0,211784	0,208189
CMA-ES	0,210194	0,207639

Pelas figuras 158 e 159, vê-se que o número de gerações, em ambos métodos, foi insuficiente para a convergência em ambas as etapas. A tabela 48 mostra a configuração de parâmetros globais que gerou as melhores soluções de cada submodelo (dentre as 10 soluções finais de cada experimento), enquanto que as figuras 160 a 165 mostram a simulação por cada uma dessas soluções dos conjuntos de treinamento, validação e teste em todas as amostras e as figuras 166 a 172 mostram as 400 últimas amostras desta simulação. A figura 173 mostra os gráficos de razão de separação para cada uma dessas soluções.

Tabela 48 - Parâmetros globais da melhor solução encontrada por cada submodelo para o estudo de caso 9

Melhor solução	N	c_w	ρ_w	S_{in}	S_{back}	α
AEIQ-R/Random/ESN	430	0,7794	0,3137	0,1137	0,5592	0,9748
AEIQ-R/SRG/ESN	426	0,4878	0,2292	0,9000	0,9824	0,8492
AEIQ-R/AEIQ-R/ESN	385	0,7244	0,4795	0,2962	0,7941	0,9201
CMA-ES/Random/ESN	423	0,1616	0,4728	0,1443	0,7702	1,0000
CMA-ES/SRG/ESN	399	0,4233	0,4624	0,2296	0,9119	0,9918
CMA-ES/CMA-ES/ESN	416	0,5320	0,4846	0,1785	1,0000	1,0000

Figura 160 - Simulação da variável de saída y do estudo de caso 9 utilizando a melhor solução do submodelo AEIQ-R/Random/ESN

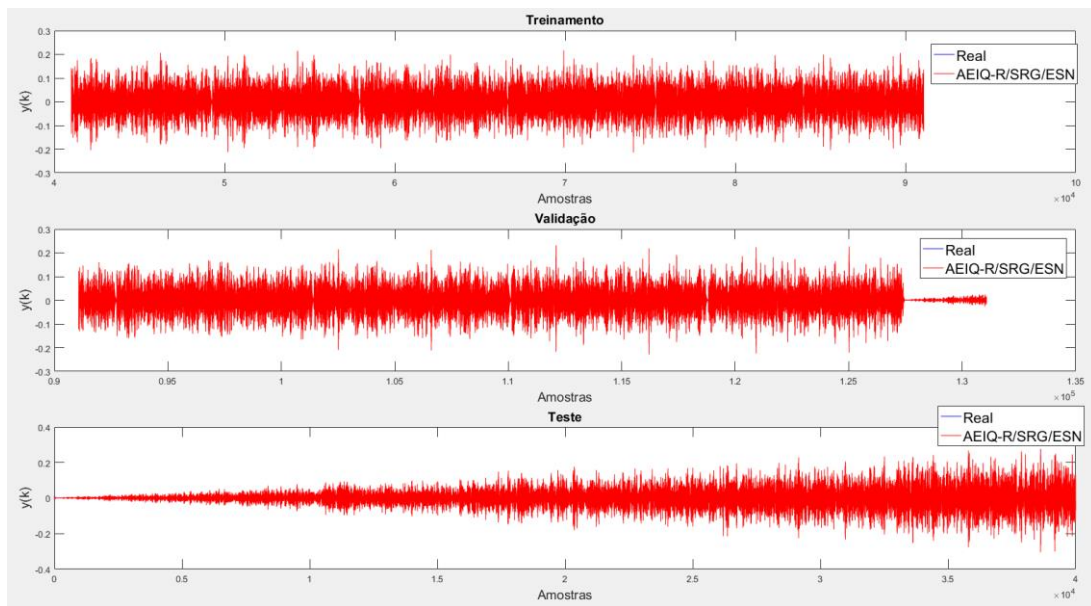


Figura 161 - Simulação da variável de saída y do estudo de caso 9 utilizando a melhor solução do submodelo AEIQ-R/SRG/ESN

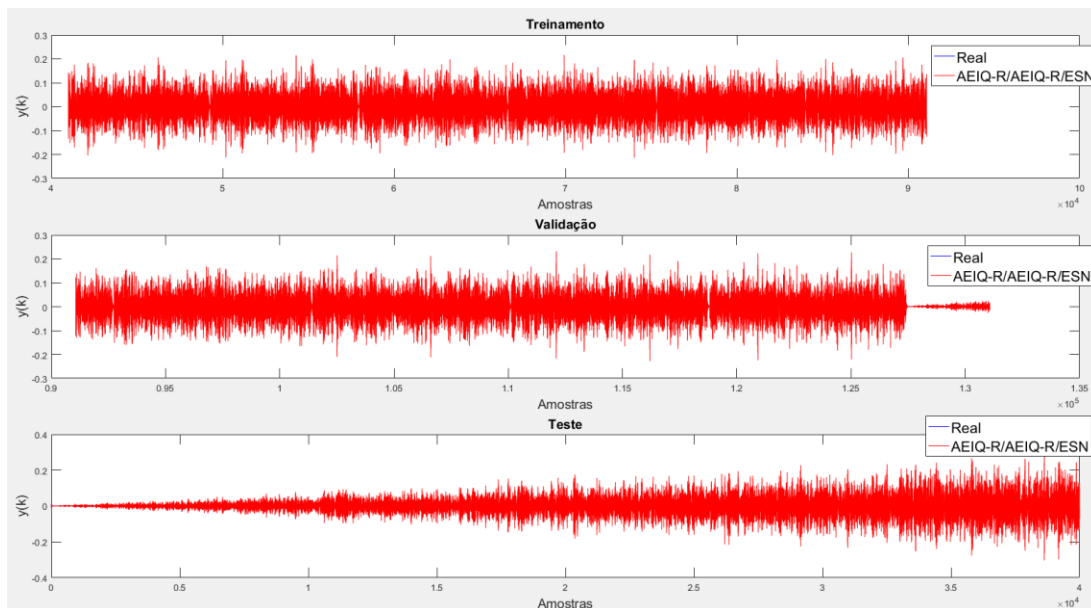


Figura 162 - Simulação da variável de saída y do estudo de caso 9 utilizando a melhor solução do submodelo AEIQ-R/AEIQ-R/ESN

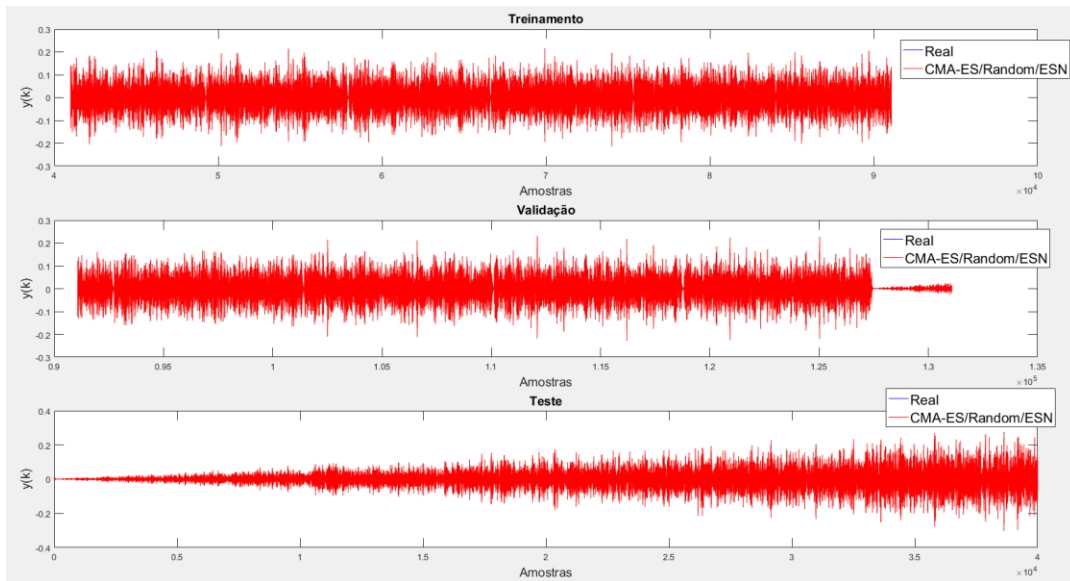


Figura 163 - Simulação da variável de saída y do estudo de caso 9 utilizando a melhor solução do submodelo CMA-ES/Random/ESN

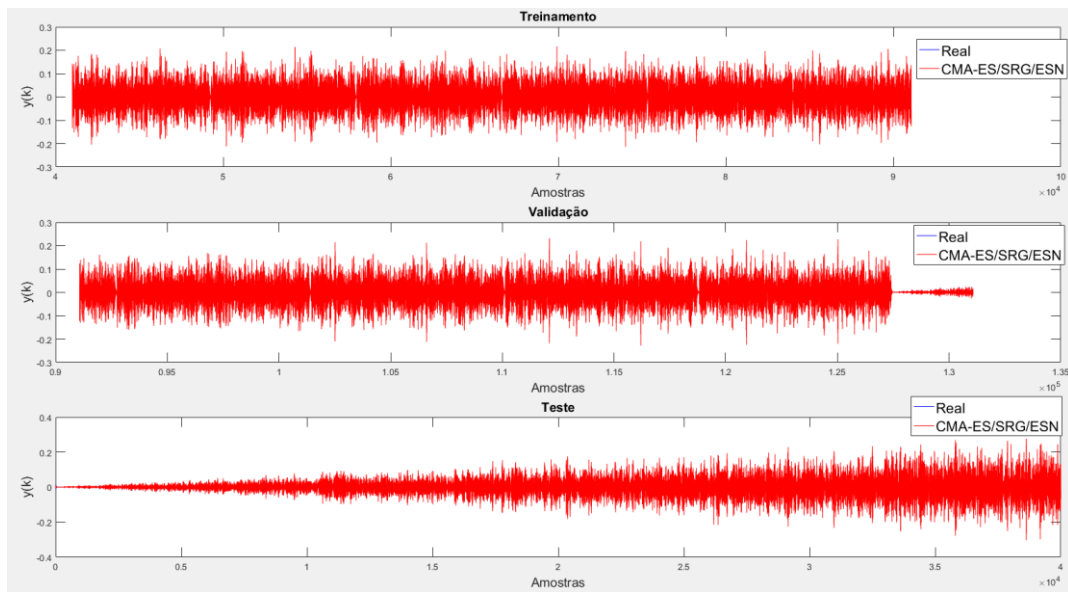


Figura 164 - Simulação da variável de saída y do estudo de caso 9 utilizando a melhor solução do submodelo CMA-ES/SRG/ESN

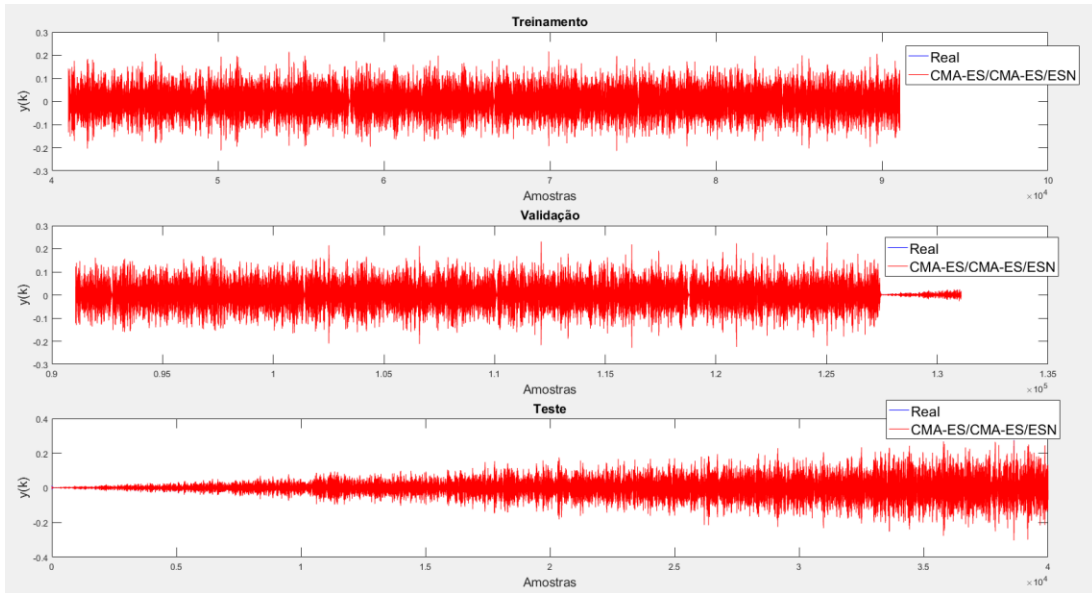


Figura 165 - Simulação da variável de saída y do estudo de caso 9 utilizando a melhor solução do submodelo CMA-ES/CMA-ES/ESN

PUC-Rio - Certificação Digital Nº I612970/CA

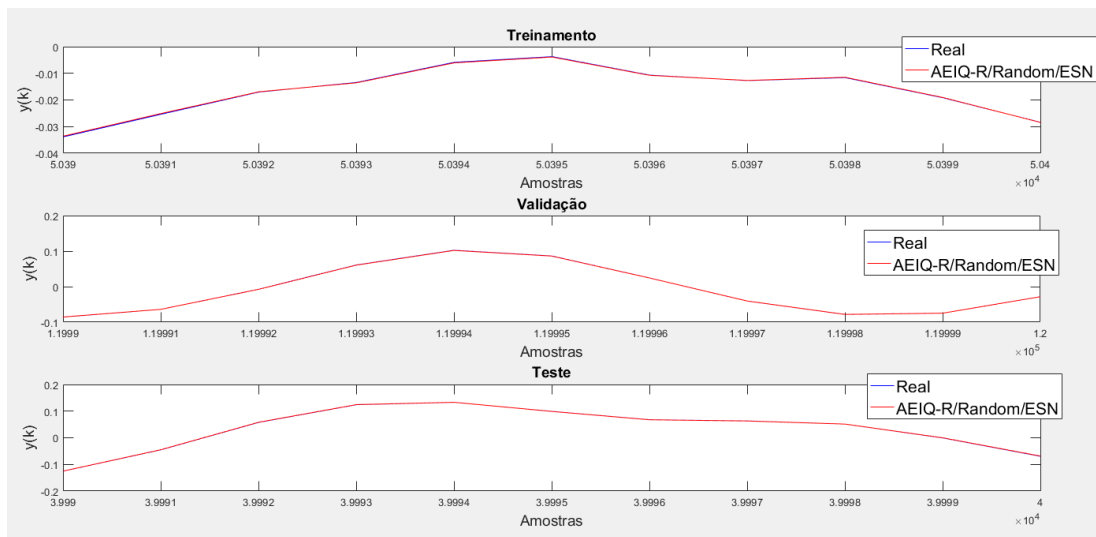


Figura 166 - Simulação da variável de saída y do estudo de caso 9 utilizando a melhor solução do submodelo AEIQ-R/Random/ESN (zoom em algumas amostras

)

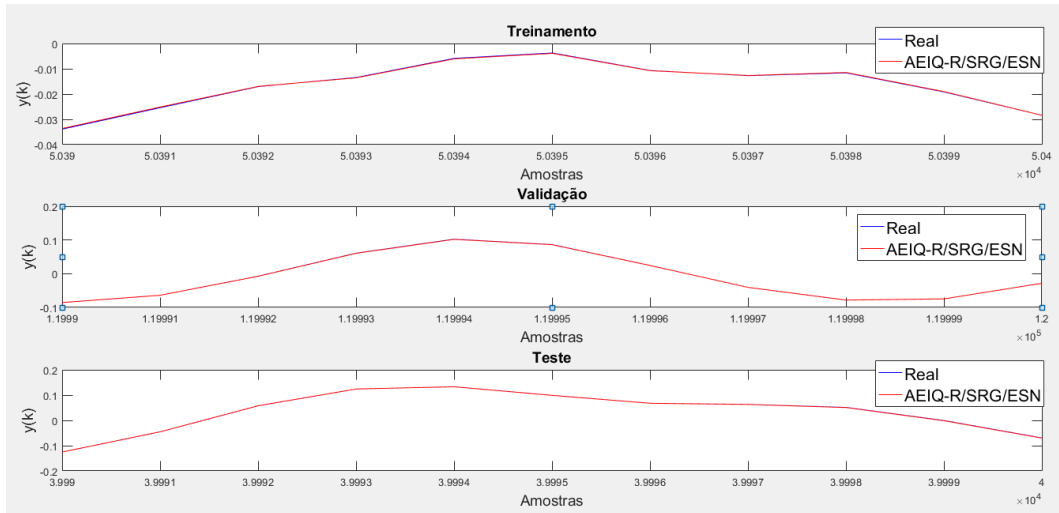


Figura 167 - Simulação da variável de saída y do estudo de caso 9 utilizando a melhor solução do submodelo AEIQ-R/SRG/ESN (zoom em algumas amostras)

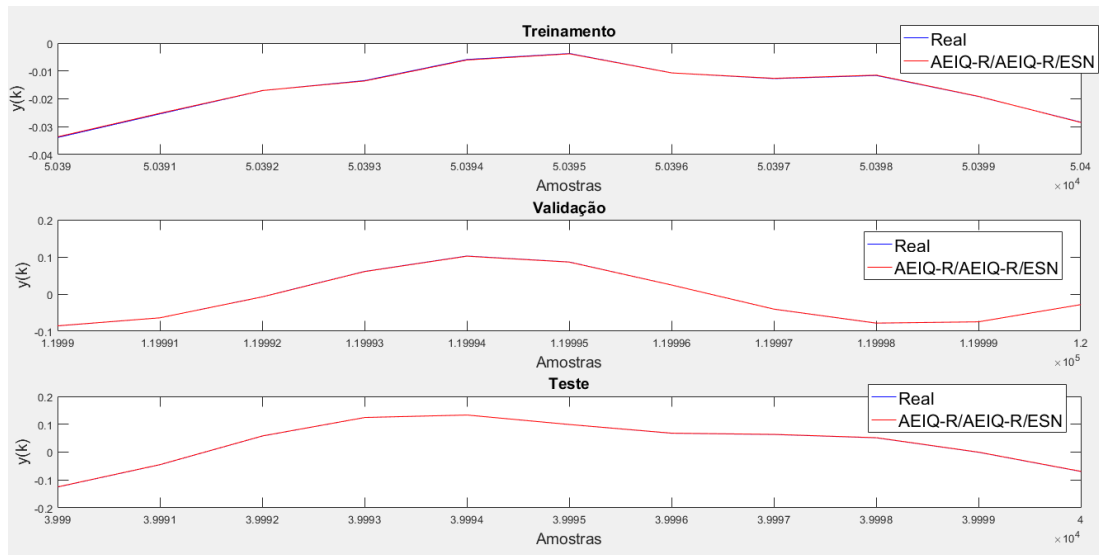


Figura 168 - Simulação da variável de saída y do estudo de caso 9 utilizando a melhor solução do submodelo AEIQ-R/AEIQ-R/ESN (zoom em algumas amostras)

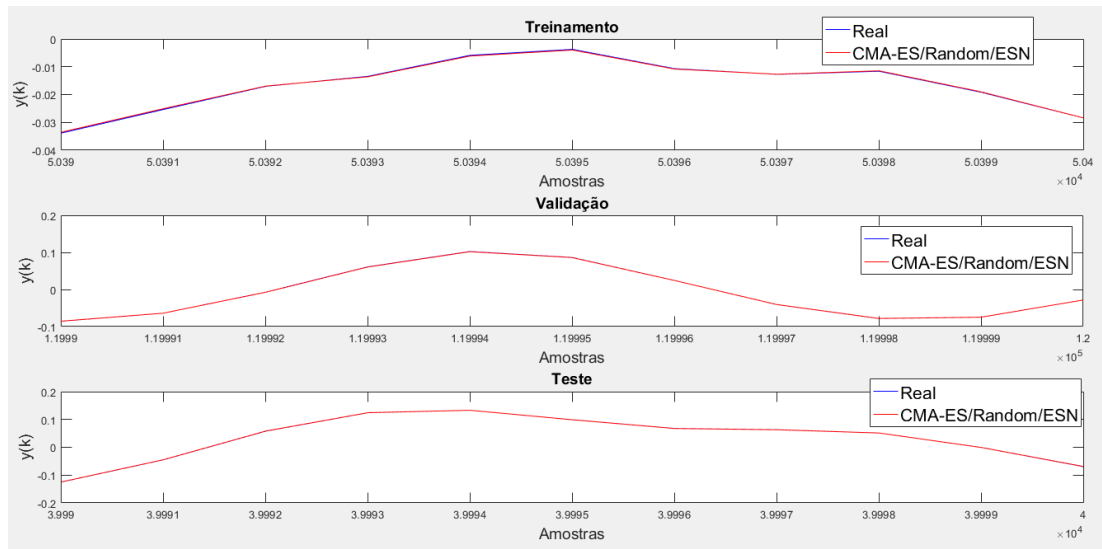


Figura 169 - Simulação da variável de saída y do estudo de caso 9 utilizando a melhor solução do submodelo CMA-ES/Random/ESN (zoom em algumas amostras)

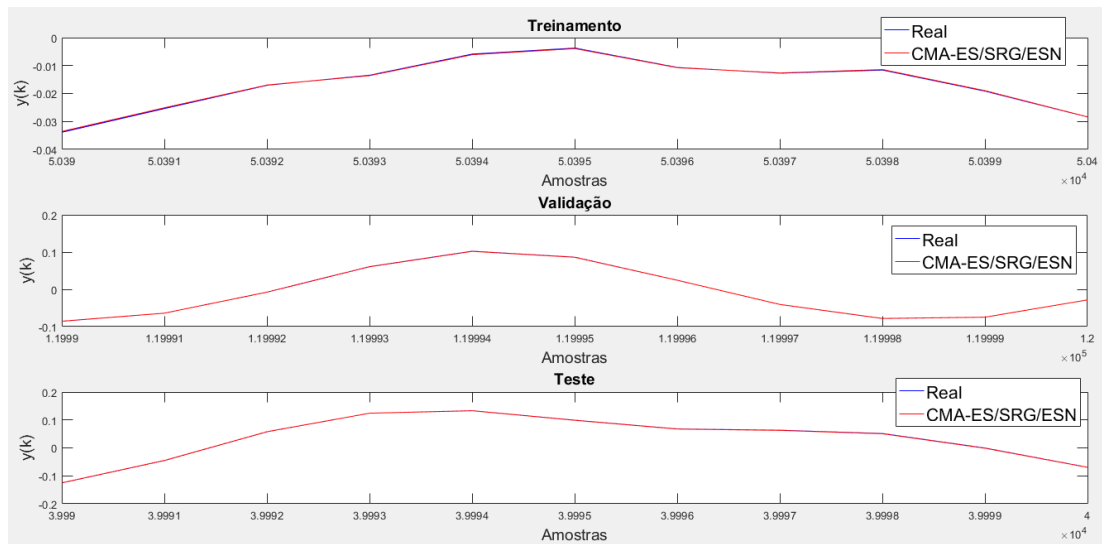


Figura 170 - Simulação da variável de saída y do estudo de caso 9 utilizando a melhor solução do submodelo CMA-ES/SRG/ESN (zoom em algumas amostras)

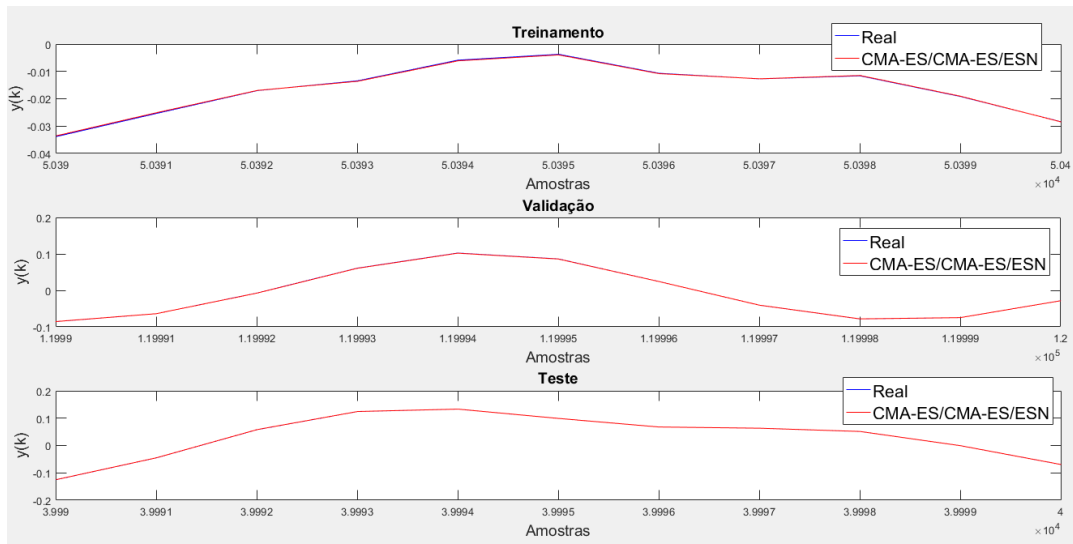


Figura 171 - Simulação da variável de saída y do estudo de caso 9 utilizando a melhor solução do submodelo CMA-ES/CMA-ES/ESN (zoom em algumas amostras)

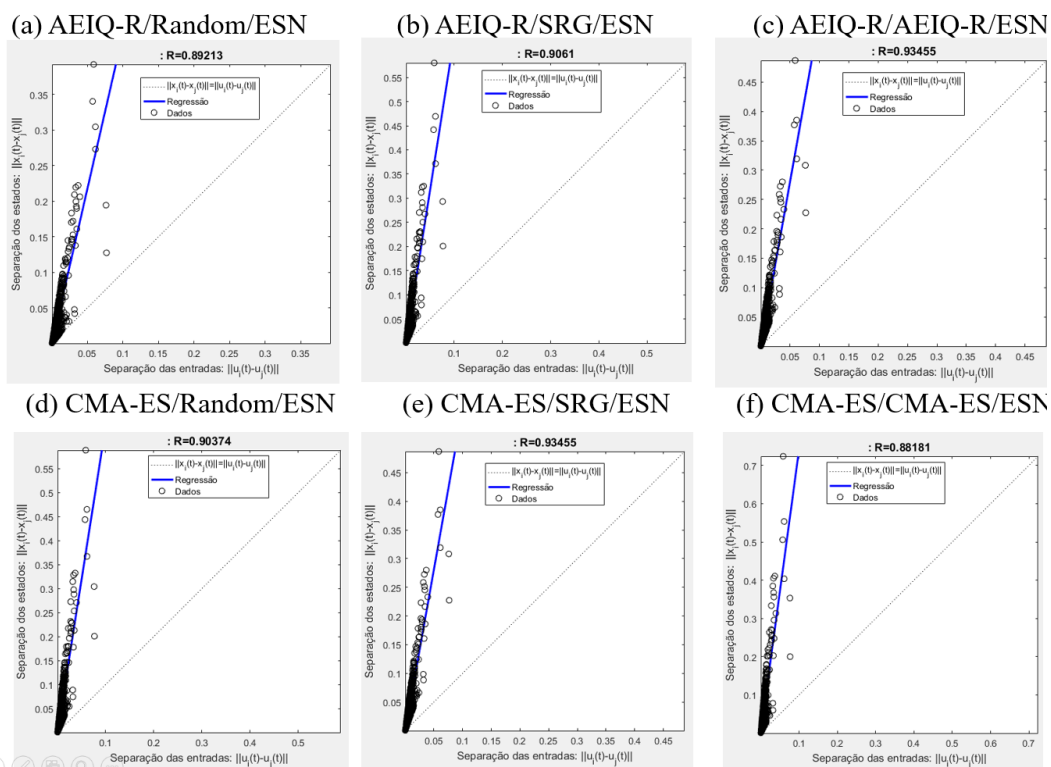


Figura 172 - Gráficos de razão de separação encontrados para as melhores soluções de cada submodelo para o estudo de caso 9

Como mostrado pelas figuras 160 a 171, as melhores soluções de cada submodelo dentre os 10 experimentos conseguiram simular adequadamente a variável de saída do

sistema. Os gráficos de razão de separação da figura 172 foram bastante similares, indicando reservatórios com comportamentos caóticos e distantes da região objetivo

As tabelas 49 e 50 comparam os resultados obtidos pelas melhores soluções dos submodelos testados nesta dissertação com os melhores resultados dos métodos de diversos autores que utilizaram as mesmas amostras de teste.

Tabela 49 - Comparação 1 de resultados com outros métodos para o *benchmark Caixa de prata*

Método	RMSE (mV)
Polynomial Nonlinear State Space (Paduart et al., 2010)	0,26
PLLS-SVM (Espinoza et al., 2005)	0,27
NARX – Sigmoid Neural Network (Ljung et al., 2004)	0,30
CMA-ES/SRG/ESN (melhor)	0,31
FS-PLS (Espinoza et al., 2004)	0,32
CMA-ES/Random/ESN (melhor)	0,32
AEIQ-R/Random/ESN (melhor)	0,33
AEIQ-R/SRG/ESN (melhor)	0,33
AEIQ-R/AEIQ-R/ESN (melhor)	0,33
CMA-ES/CMA-ES/ESN (melhor)	0,34
Nonlinear State-Space (Marconato et al., 2012)	0,34
Poly-LFR (Van Mulders et al., 2013)	0,35
Physical block-oriented (Paduart et al., 2004)	0,38
Physical block-oriented (Hjalmarsson & Schoukens, 2004)	0,96
LLM undirected (Verdult, 2004)	1,3

Tabela 50 - Comparação 2 de resultados com outros métodos para o *benchmark Caixa de prata*

Método	Métrica fit
AEIQ-R/SRG/ESN (melhor)	99,4
CMA-ES/SRG/ESN (melhor)	99,4
AEIQ-R/AEIQ-R/ESN (melhor)	99,4
AEIQ-R/Random/ESN (melhor)	99,4
CMA-ES/CMA-ES/ESN (melhor)	99,3
CMA-ES/Random/ESN (melhor)	99,3
Ahmadi and Karrari proposed algorithm (Ahmadi & Karrari, 2012)	94,0
Neural Network (Ahmadi & Karrari, 2012)	94,0
Wavelet ANOVA expansion (Ahmadi & Karrari, 2012)	92,4
PWA-LFR (Pepona & Paoletti, 2011)	92,3
LOLIMOT (Ahmadi & Karrari, 2012)	85,3
Global quadratic polnomial (Ahmadi & Karrari, 2012)	75,5

Pelos resultados mostrados na tabela 49, vê-se que mesmo com as limitações impostas, as melhores soluções de cada submodelo testado nesta dissertação apresentaram desempenho comparável aos melhores resultados de diferentes autores da literatura, e ainda, tais abordagens conseguiram gerar ESNs que conseguiram alcançar resultados de *fit* acima de 99 % durante a simulação do conjunto de teste desta base de dados, como mostrado pela tabela 50.

6.5. Análise geral dos resultados

6.5.1. Resultados gerais

A tabela 51 resume os resultados de RMSE durante o teste (média nos 10 experimentos) apresentados na seção 6.4 para os 9 estudos de caso utilizando os 6 diferentes submodelos extraídos do modelo geral e sem considerar os experimentos extras realizados nos estudos de caso 3 e 8:

Tabela 51 – Resultados de RMSE no teste (média de 10 experimentos) para a simulação do conjunto de teste utilizando as diferentes variações do método geral nos 9 estudos de caso

	AEIQ-R/Random/ESN	AEIQ-R/SRG/ESN	AEIQ-R/AEIQ-R/ESN	CMA-ES/Random/ESN	CMA-ES/SRG/ESN	CMA-ES/CMA-ES/ESN
1	0,194123	0,236160	0,133451	0,196809	0,228058	0,144726
2	0,135961	0,152971	0,130780	0,141140	0,149405	0,123333
3	0,220903	0,314525	0,233609	0,213938	0,309211	0,240149
4	0,046168	78,4463	0,024391	0,071691	13,468504	0,022260
5	0,020766	8,420741	0,015463	0,034474	9,190147	0,018136
6	7,565423	1,1226	0,256585	0,286990	0,325602	0,214222
7	2,834602	153,2818	2,719567	2,864844	19,7248	2,711008
8	0,147886	7,033276	0,136512	0,251889	1,033491	0,134590
9	0,409211	0,379212	0,363232	0,392883	0,354804	0,385372

Como pode ser visto na tabela 51, com exceção dos estudos de caso 3 e 9, os melhores resultados em cada problema foram obtidos pelas abordagens CMA-ES/CMA-ES/ESN e AEIQ-R/AEIQ-R/ESN, as quais otimizaram alguns pesos do reservatório na segunda etapa do modelo geral proposto. Com relação a estes dois últimos submodelos, o CMA-ES/CMA-ES/ESN foi o melhor em 5 problemas (2, 4, 6, 7, 8) enquanto o AEIQ-R/AEIQ-R/ESN foi o de melhor desempenho em dois estudos de caso (1 e 5). Considerando estes resultados, pode-se dizer que a otimização dos pesos do reservatório, no geral, funcionou bem e melhorou o desempenho de uma ESN de reservatório aleatório (métodos Random e SRG), mas com os mesmos hiperparâmetros, que foram inicialmente otimizados na etapa 1. Entretanto, não se pode afirmar que tal abordagem é perfeita, como visto nos resultados obtidos nos estudos de caso 3, onde ocorreu uma espécie de *overfitting* ao aumentar o número de gerações na etapa 2, como discutido na seção 6.4.3, e no caso 9, onde, devido ao grande número de amostras da base de dados, optou-se por reduzir drasticamente o número de gerações dos AEs e, então, a otimização dos pesos do reservatório praticamente não surtiu efeito.

Comparando-se o desempenho dos AEs empregados, na seção 6.4 viu-se que mesmo em algumas situações nas quais o AEIQ-R tenha convergido para soluções teoricamente melhores, com menores erros de simulação na validação, na prática, as melhores soluções, com melhor desempenho de simulação do conjunto de teste, em geral, foram obtidas pelo CMA-ES, baseando-se numa comparação dos submodelos AEIQ-R/AEIQ-R/ESN e CMA-ES/CMA-ES/ESN. Cabe observar que enquanto que para configurar o AEIQ-R, foram realizados testes iniciais, no caso do CMA-ES praticamente todos os seus parâmetros foram deixados em seus valores automáticos e *default*, com a distribuição inicial de média gerada aleatoriamente e passo global fixo em um terço dos intervalos de variação das variáveis e ajustados apenas o tamanho da população e o número de gerações para que igualasse o número de avaliações do AEIQ-R. Considerando uma das ideias deste trabalho, que é o de conceber um modelo neuroevolutivo automático para identificação de sistemas, com pouca ou nenhuma intervenção do usuário para ajustes de parâmetros, o emprego do CMA-ES neste modelo o deixa mais próximo deste objetivo. Com isso, mesmo que, posteriormente, se verifique que não há diferença significativa no desempenho dos dois algoritmos, pelo fato de fornecer uma maior automatização ao modelo geral com desempenho similar ou superior, o CMA-ES pode ser considerado como uma melhor opção de AE frente ao AEIQ-R.

Comparando-se as abordagens Random e SRG, vê-se que, na maioria das situações, a escolha do reservatório pelo menor erro de validação gerou melhores resultados do que a escolha baseada no menor coeficiente SRG, dentro do mesmo subconjunto de ESNs com os mesmos hiperparâmetros. Uma possível justificativa para tal afirmação pode ser pelo fato de que, como a escolha do reservatório pelo método SRG é não supervisionada e, muitas vezes, tal método selecionou uma ESN com erro de simulação no conjunto de validação muito alto. Como no conjunto de teste também é realizada uma simulação livre (previsão de infinitos passos à frente), uma ESN com um erro de simulação muito grande na validação pode ter um desempenho não muito bom durante o teste e até mesmo gerar soluções instáveis, devido à acumulação de erro a cada passo de simulação. Um exemplo de tal caso é mostrado na tabela 52, onde são exibidos os erros de simulação dos conjuntos de validação e teste das 10 soluções encontradas por cada submodelo para o estudo de caso *Braço do leitor de CD* utilizando o AEIQ-R na etapa 1.

Tabela 52 – Resultados por experimentos métodos AEIQ-R/Random/ESN e AEIQ-R/SRG/ESN para o estudo de caso *Braço do leitor de CD*

<i>Exp</i>	<i>Random</i>		<i>SRG</i>	
	<i>Validação</i>	<i>Teste</i>	<i>Validação</i>	<i>Teste</i>
1	0,123811	0,152534	0,493936	10,906988
2	0,119933	0,148890	0,131793	0,164899
3	0,133489	0,163356	0,140921	0,176874
4	0,122974	0,148541	0,122974	0,148541
5	0,118435	0,144348	0,180614	0,195387
6	0,116464	0,143869	43,109019	55,414491
7	0,118079	0,133926	0,148817	0,157435
8	0,118811	0,140180	0,138264	0,159593
9	0,123972	0,152673	2,394618	2,858014
10	0,124777	0,150540	0,124777	0,150540
<i>Média</i>	0,122075	0,147886	4,698573	7,033276

Como mostrado pela tabela 52, em 3 experimentos o método SRG selecionou ESNs com valores altos de erro de simulação no conjunto de validação, o que conduziu a erros altos de simulação também no conjunto de teste. Uma outra questão relacionada ao método SRG é que não foi encontrada uma relação direta entre reservatórios com gráficos de razão de separação mais próximos da região objetivo e seu correspondente melhor desempenho durante o teste, como visto na seção 6.4.

Como mostrado na seção 6.4, o desempenho dos submodelos derivados do modelo proposto nesta dissertação foram melhores (1, 5, 6, 7), comparáveis (2, 4 e 9) e até piores (3 e 8) em algumas situações frente a alguns métodos da literatura que utilizaram as mesmas bases de dados com as mesmas divisões entre dados de estimação e teste, o que enfatiza que o modelo proposto não é perfeito e nem o melhor a ser aplicado em todas as situações/problemas. Em alguns problemas (3 e 8) em que o método proposto foi inferior, como discutido na seção 6.4.8, há a hipótese de que a subdivisão com mais dados para teste do que para estimação possa ter deteriorado o desempenho das ESNs.

Uma vez que os resultados tenham sido apresentados e discutidos nesta seção e na seção 6.4, na subseção seguinte serão realizados testes estatísticos, que permitirão verificar se as diferenças encontradas entre os métodos e se as afirmações, conclusões e hipóteses levantadas são suportadas do ponto de vista estatístico.

6.5.2. Análise estatística

Como não há garantia de normalidade dos dados e de outras condições necessárias para a utilização de testes paramétricos, para a comparação dos diferentes submodelos foram utilizados os seguintes testes não paramétricos:

- Wilcoxon Signed Ranks Test (Derrac et al., 2011; Trawinski et al., 2012);
- Teste de Friedman e em seguida testes *post-hoc*: Holm, Shaffer e Bergman (Derrac et al., 2011; Trawinski et al., 2012).

O teste de Wilcoxon Signed Ranks é utilizado para uma comparação par a par entre os métodos. O teste de Friedman é empregado para comparar simultaneamente todos os métodos e caso a hipótese de igualdade entre os algoritmos seja rejeitada, testes *post-hoc* são aplicados para detectar eventuais diferenças entre os pares de algoritmos utilizando os valores de p corrigidos em função das múltiplas comparações entre os modelos, fato que não é considerado no teste de Wilcoxon Signed Ranks. O nível de significância (α) considerado nos testes foi de 5% (máximo p-valor aceitado para rejeitar a hipótese nula de igualdade entre os modelos). Todos estes métodos estatísticos estão disponíveis no KEEL *Software Tool* (KEEL, 2018).

A tabela 53 mostra os resultados do teste de Wilcoxon Signed Ranks aplicado aos valores médios de RMSE no conjunto de teste (tabela 51) dos submodelos testados nesta dissertação:

Tabela 53 – Teste de Wilcoxon Signed Ranks aplicado aos submodelos derivados do método geral

Comparação		R ⁺	R ⁻	p-valor	Conclusão
Método 1	Método 2				
CMA-ES/CMA-ES/ESN	AEIQ-R/SRG/ESN	45,0	0,0	0,003906	Método 1 é melhor
AEIQ-R/AEIQ-R/ESN	AEIQ-R/SRG/ESN	44,0	1,0	0,007812	Método 1 é melhor
CMA-ES/CMA-ES/ESN	CMA-ES/SRG/ESN	44,0	1,0	0,007812	Método 1 é melhor
AEIQ-R/AEIQ-R/ESN	CMA-ES/SRG/ESN	43,0	2,0	0,011718	Método 1 é melhor
AEIQ-R/Random/ESN	AEIQ-R/SRG/ESN	37,0	8,0	0,09766	Equivalência
CMA-ES/CMA-ES/ESN	CMA-ES/Random/ESN	41,0	4,0	0,02734	Método 1 é melhor
CMA-ES/Random/ESN	AEIQ-R/SRG/ESN	36,0	0,0	0,007812	Método 1 é melhor
CMA-ES/CMA-ES/ESN	AEIQ-R/Random/ESN	41,0	4,0	0,02734	Método 1 é melhor
AEIQ-R/AEIQ-R/ESN	CMA-ES/Random/ESN	42,0	3,0	0,019532	Método 1 é melhor
AEIQ-R/Random/ESN	CMA-ES/SRG/ESN	37,0	8,0	0,09766	Equivalência
AEIQ-R/AEIQ-R/ESN	AEIQ-R/Random/ESN	41,0	4,0	0,02734	Método 1 é melhor
CMA-ES/Random/ESN	CMA-ES/SRG/ESN	43,0	2,0	0,011718	Método 1 é melhor
CMA-ES/SRG/ESN	AEIQ-R/SRG/ESN	37,0	8,0	0,09766	Equivalência
AEIQ-R/AEIQ-R/ESN	CMA-ES/CMA-ES/ESN	15,0	30,0	≥0,2	Equivalência
AEIQ-R/Random/ESN	CMA-ES/Random/ESN	28,0	17,0	≥0,2	Equivalência

Pela tabela 53, vê-se que, considerando um nível de significância $\alpha = 0,05$, pelo teste de Wilcoxon Signed Ranks, o método CMA-ES/CMA-ES/ESN foi significativamente melhor do que os submodelos AEIQ-R/SRG/ESN, CMA-

ES/SRG/ESN, CMA-ES/*Random*/ESN e AEIQ-R/*Random*/ESN. Pela mesma tabela, conclui-se que o método AEIQ-R/AEIQ-R/ESN apresentou melhorias significativas frente às abordagens AEIQ-R/SRG/ESN, CMA-ES/SRG/ESN, CMA-ES/*Random*/ESN e AEIQ-R/*Random*/ESN. Com estes resultados, pode ver que em uma comparação par a par pelo método de Wilcoxon Signed Ranks, considerando $\alpha = 0,05$, a otimização dos pesos do reservatório resultou em melhorias significativas em relação à utilização de reservatórios aleatórios. Considerando as abordagens *Random* e SRG, o método aponta para melhorias significativas somente para o CMA-ES/*Random*/ESN frente ao AEIQ-R/SRG/ESN e ao CMA-ES/SRG/ESN. Não foi encontrada diferença significativa entre os métodos AEIQ-R/AEIQ-R/ESN e CMA-ES/CMA-ES/ESN e nem entre os métodos AEIQ-R/*Random*/ESN e CMA-ES/*Random*/ESN. A tabela 54 mostra os resultados obtidos pelo teste de Friedman.

Tabela 54 - Resultados para o teste de Friedman para comparação entre os submodelos

Método	Ranking
AEIQ-R/ <i>Random</i> /ESN	3,3333
AEIQ-R/SRG/ESN	5,5556
AEIQ-R/AEIQ-R/ESN	2,1111
CMA-ES/ <i>Random</i> /ESN	3,6667
CMA-ES/SRG/ESN	4,6667
CMA-ES/CMA-ES/ESN	1,6667
p-valor = 0,000034622	
Conclusão:	
Há diferença significativa entre os métodos	

Pela tabela 54 vê-se que a partir dos *rankings* obtidos a partir da tabela 51, o método de Friedman chega a um p-valor de 0,000034622, donde pode-se concluir que há diferença significativa entre os modelos e métodos *post-hoc* podem ser aplicados. A

tabela 55 mostra os resultados de aplicação dos métodos post-hoc de Holm, Shaffer e Bergman.

Tabela 55 – Resultados de aplicação dos testes *post-hoc*

Comparação		p não ajustado	Holm	Shaffer	Bergman	Conclusão
Método1	Método 2					
AEIQ-R/SRG/ESN	CMA-ES/CMA-ES/ESN	0,00001	0,000155	0,000155	0,000155	Método 2 é melhor
AEIQ-R/SRG/ESN	AEIQ-R/AEIQ-R/ESN	0,000094	0,001316	0,00094	0,00094	Método 2 é melhor
CMA-ES/SRG/ESN	CMA-ES/CMA-ES/ESN	0,00067	0,008706	0,006697	0,006697	Método 2 é melhor
AEIQ-R/AEIQ-R/ESN	CMA-ES/SRG/ESN	0,003759	0,045105	0,037588	0,022553	Método 1 é melhor
AEIQ-R/Random/ESN	AEIQ-R/SRG/ESN	0,011743	0,129177	0,117434	0,082204	Equivalência
CMA-ES/Random/ESN	CMA-ES/CMA-ES/ESN	0,023342	0,233422	0,233422	0,163395	Equivalência
AEIQ-R/SRG/ESN	CMA-ES/SRG/ESN	0,03221	0,289887	0,233422	0,193258	Equivalência
AEIQ-R/Random/ESN	CMA-ES/CMA-ES/ESN	0,058782	0,470254	0,411472	0,235127	Equivalência
AEIQ-R/AEIQ-R/ESN	CMA-ES/Random/ESN	0,07776	0,544319	0,544319	0,31104	Equivalência
AEIQ-R/Random/ESN	CMA-ES/SRG/ESN	0,13057	0,78342	0,78342	0,52228	Equivalência
AEIQ-R/Random/ESN	AEIQ-R/AEIQ-R/ESN	0,165787	0,828933	0,78342	0,52228	Equivalência
CMA-ES/Random/ESN	CMA-ES/SRG/ESN	0,256839	1,0	1,0	0,52228	Equivalência
AEIQ-R/SRG/ESN	CMA-ES/SRG/ESN	0,3135	1,0	1,0	0,9405	Equivalência
AEIQ-R/AEIQ-R/ESN	CMA-ES/CMA-ES/ESN	0,614295	1,0	1,0	1,0	Equivalência
AEIQ-R/Random/ESN	CMA-ES/Random/ESN	0,705457	1,0	1,0	1,0	Equivalência

Pelos resultados da tabela 55, vê-se que, considerando $\alpha = 0,05$, os testes *post-hoc* apenas confirmam o melhor desempenho do método CMA-ES/CMA-ES/ESN frente aos submodelos AEIQ-R/SRG/ESN e CMA-ES/SRG/ESN e do AEIQ-R/AEIQ-R/ESN sobre as abordagens AEIQ-R/SRG/ESN e CMA-ES/SRG/ESN.

A comparação com os resultados de outros trabalhos em termos de testes estatísticos, foi realizada somente com um grupo pequeno de autores (Liu et al., 2013; Hansson et al., 2012) e em um subconjunto de problemas (1, 2, 3 e 8), pois na literatura não há resultados que envolvam a utilização de todas as bases de dados pelos mesmos autores e quando as mesmas são utilizadas, geralmente não se segue a mesma divisão entre dados de estimação e teste, nem sempre as mesmas métricas são empregadas e dificilmente são informados se os resultados foram obtidos durante uma previsão de 1 passo à frente ou através de uma simulação livre. A tabela 56 resume os melhores resultados dos métodos testados nesta dissertação em termos de *fit*, excluindo-se as abordagens AEIQ-R/SRG/ESN e CMA-ES/SRG/ESN que geraram soluções instáveis no estudo de caso 8, frente aos métodos de Liu et al. (2013) e Hansson et al. (2012) nos problemas 1, 2, 3 e 8, enquanto as tabelas 57 e 58 mostram os resultados obtidos pela aplicação dos testes de Wilcoxon Signed Ranks e Friedman, respectivamente.

Tabela 56 – Resumo dos resultados de *fit* de teste para os estudos de caso 1, 2, 3 e 8

Método	Problema			
	1	2	3	8
AEIQ-R/AEIQ-R/ESN	91,8	86,3	83,6	68,2
CMA-ES/CMA-ES/ESN	91,1	87,1	83,2	71,2
AEIQ-R/Random/ESN	89,7	84,3	82,0	66,5
CMA-ES/Random/ESN	88,9	87,1	81,9	64,0
Weighted Nuclear Norm (Liu et al., 2013)	76,6	86,3	86,7	73,3
CVA (Hansson et al., 2012)	71,8	84,9	86,2	70,0
IVM (Hansson et al., 2012)	70,8	84,9	86,2	71,0
MOESP (Hansson et al., 2012)	70,8	84,8	84,8	72,5
NOINSTR (Hansson et al., 2012)	70,8	90,2	85,1	72,3
N4SID (Hansson et al., 2012)	70,8	85,0	83,9	73,5
Regularized Nuclear Norm (Liu et al., 2013)	45,1	88,6	86,7	72,5
NONE (Hansson et al., 2012)	-245	84,2	84,8	72,8

Tabela 57 – Resultados do teste de Wilcoxon Signed Ranks para os modelos da tabela 56

Comparação		R ⁺	R ⁻	p-valor	Conclusão
Método 1	Método 2				
AEIQ-R/AEIQ-R/ESN	Weighted Nuclear Norm (Liu et al., 2013)	5,0	1,0	≥0,2	Equivalência
AEIQ-R/AEIQ-R/ESN	CVA (Hansson et al., 2012)	5,0	5,0	≥0,2	Equivalência
AEIQ-R/AEIQ-R/ESN	IVM (Hansson et al., 2012)	5,0	5,0	≥0,2	Equivalência
AEIQ-R/AEIQ-R/ESN	MOESP (Hansson et al., 2012)	6,0	4,0	≥0,2	Equivalência

Comparação		R ⁺	R ⁻	p-valor	Conclusão
Método 1	Método 2				
AEIQ-R/AEIQ-R/ESN	NOINSTR (Hansson et al., 2012)	4,0	6,0	≥0,2	Equivalência
AEIQ-R/AEIQ-R/ESN	N4SID (Hansson et al., 2012)	6,0	4,0	≥0,2	Equivalência
AEIQ-R/AEIQ-R/ESN	Regularized Nuclear Norm (Liu et al., 2013)	4,0	6,0	≥0,2	Equivalência
AEIQ-R/AEIQ-R/ESN	NONE (Hansson et al., 2012)	6,0	4,0	≥0,2	Equivalência
CMA-ES/CMA-ES/ESN	Weighted Nuclear Norm (Liu et al., 2013)	5,0	5,0	≥0,2	Equivalência
CMA-ES/CMA-ES/ESN	CVA (Hansson et al., 2012)	6,0	4,0	≥0,2	Equivalência
CMA-ES/CMA-ES/ESN	IVM (Hansson et al., 2012)	7,0	3,0	≥0,2	Equivalência
CMA-ES/CMA-ES/ESN	MOESP (Hansson et al., 2012)	8,0	2,0	≥0,2	Equivalência
CMA-ES/CMA-ES/ESN	NOINSTR (Hansson et al., 2012)	4,0	6,0	≥0,2	Equivalência
CMA-ES/CMA-ES/ESN	N4SID (Hansson et al., 2012)	7,0	3,0	≥0,2	Equivalência
CMA-ES/CMA-ES/ESN	Regularized Nuclear Norm (Liu et al., 2013)	4,0	6,0	≥0,2	Equivalência
CMA-ES/CMA-ES/ESN	NONE (Hansson et al., 2012)	7,0	3,0	≥0,2	Equivalência
AEIQ-R/Random/ESN	Weighted Nuclear Norm (Liu et al., 2013)	4,0	6,0	≥0,2	Equivalência
AEIQ-R/Random/ESN	CVA (Hansson et al., 2012)	4,0	6,0	≥0,2	Equivalência
AEIQ-R/Random/ESN	IVM (Hansson et al., 2012)	4,0	6,0	≥0,2	Equivalência
AEIQ-R/Random/ESN	MOESP (Hansson et al., 2012)	4,0	6,0	≥0,2	Equivalência
AEIQ-R/Random/ESN	NOINSTR (Hansson et al., 2012)	4,0	6,0	≥0,2	Equivalência
AEIQ-R/Random/ESN	N4SID (Hansson et al., 2012)	4,0	6,0	≥0,2	Equivalência
AEIQ-R/Random/ESN	Regularized Nuclear Norm (Liu et al., 2013)	4,0	6,0	≥0,2	Equivalência
AEIQ-R/Random/ESN	NONE (Hansson et al., 2012)	5,0	5,0	≥0,2	Equivalência
CMA-ES/Random/ESN	Weighted Nuclear Norm (Liu et al., 2013)	5,0	5,0	≥0,2	Equivalência
CMA-ES/Random/ESN	CVA (Hansson et al., 2012)	5,0	5,0	≥0,2	Equivalência

Comparação		R ⁺	R ⁻	p-valor	Conclusão
Método 1	Método 2				
CMA-ES/Random/ESN	IVM (Hansson et al., 2012)	5,0	5,0	≥0,2	Equivalência
CMA-ES/Random/ESN	MOESP (Hansson et al., 2012)	5,0	5,0	≥0,2	Equivalência
CMA-ES/Random/ESN	NOINSTR (Hansson et al., 2012)	4,0	6,0	≥0,2	Equivalência
CMA-ES/Random/ESN	N4SID (Hansson et al., 2012)	6,0	4,0	≥0,2	Equivalência
CMA-ES/Random/ESN	Regularized Nuclear Norm (Liu et al., 2013)	4,0	6,0	≥0,2	Equivalência
CMA-ES/Random/ESN	NONE (Hansson et al., 2012)	5,5	4,5	≥0,2	Equivalência

Tabela 58 – Resultados do teste de Friedman para os modelos da tabela 56

Método	Ranking
AEIQ-R/AEIQ-R/ESN	6,375
CMA-ES/CMA-ES/ESN	5,625
AEIQ-R/Random/ESN	9
CMA-ES/Random/ESN	7,875
Weighted Nuclear Norm (Liu et al., 2013)	3,75
CVA (Hansson et al., 2012)	6,75
IVM (Hansson et al., 2012)	7,125
MOESP (Hansson et al., 2012)	7,375
NOINSTR (Hansson et al., 2012)	5,125
N4SID (Hansson et al., 2012)	6,375
Regularized Nuclear Norm (Liu et al., 2013)	4,75
NONE (Hansson et al., 2012)	7,875
p-valor = 0,75229	
Conclusão: Não há diferença significativa entre os métodos	

Pelos resultados mostrados nas tabelas 57 e 58, vê-se que tanto pelo teste de Friedman quanto pelo Wilcoxon Signed Ranks Test, a conclusão é de que não há diferença significativa entre os métodos considerados nesta dissertação e os algoritmos de (Liu et al., 2013) e (Hansson et al., 2012) nos 4 problemas considerados. Talvez o número relativamente pequeno de problemas considerados nesta comparação tenha sido insuficiente para ambos os métodos estatísticos detectarem diferenças entre as abordagens.

7 Conclusão

Neste trabalho foi proposto um modelo neuroevolutivo composto por ESNs e por um algoritmo evolutivo, que pode ser o AEIQ-R ou o CMA-ES. O modelo permite aproveitar o treinamento rápido e a natureza naturalmente dinâmica das ESNs, o que reduz consideravelmente o espaço de entradas pela não utilização de linhas de atraso temporais na entrada da rede, além de explorar a capacidade teórica de obter boas soluções com poucas avaliações do AEIQ-R ou as credenciais de otimizador numérico estado da arte e de algoritmo auto ajustável do CMA-ES. Tal modelo define automaticamente uma ESN para problemas de Identificação de Sistemas, focando inicialmente na otimização de seus parâmetros gerais, onde pode-se optar pela utilização do AEIQ-R ou do CMA-ES, e em um segundo momento atuando na seleção do reservatório mais adequado para uma rede com estes hiperparâmetros otimizados fixos. Para a seleção do reservatório, o modelo permite optar pelos métodos Random e SRG, os quais selecionam a melhor ESN dentre um subconjunto de n redes com reservatórios inicializados aleatoriamente com base nos critérios de menor erro de validação e menor coeficiente SRG, respectivamente. Também é permitido pelo modelo obter a melhor ESN através de uma segunda otimização centrada no ajuste de alguns pesos de seu reservatório.

As diferentes variações do modelo proposto (6 testadas), denominadas nesta dissertação de submodelos, foram avaliadas em 9 problemas *benchmark* da área de Identificação de Sistemas. Salvo algumas exceções, como em um caso em que ocorreu *overfitting* e em outro em que a base de dados era muito grande para se ter um processo evolutivo longo o suficiente, viu-se uma superioridade de desempenho dos submodelos que otimizaram o reservatório (CMA-ES/CMA-ES/ESN e AEIQ-R/AEIQ-R/ESN) em relação aos que somente passaram pela otimização dos parâmetros gerais da rede (AEIQ-R/Random/ESN, AEIQ-R/SRG/ESN, CMA-ES/Random/ESN e CMA-ES/SRG/ESN). Tal superioridade dos métodos que otimizaram o reservatório foi confirmada estatisticamente pelo Wilcoxon Signed Ranks Test e por testes *post-hoc*, nestes últimos

somente verificado o melhor desempenho do CMA-ES/CMA-ES/ESN e do AEIQ-R/AEIQ-R/ESN sobre as abordagens que utilizaram o SRG (AEIQ-R/SRG/ESN e CMA-ES/SRG/ESN).

Comparando-se os resultados de simulação no conjunto de teste, viu-se que as soluções provenientes do CMA-ES, na maioria das situações, obtiveram melhor desempenho do que as advindas do AEIQ-R, embora estas diferenças não tenham sido significativas do ponto de vista estatístico. A escolha do CMA-ES como AE para o modelo neuroevolutivo proposto o torna mais “automático” em relação ao uso do AEIQ-R, pois enquanto o primeiro pode ser utilizado com valores de parâmetros *default* e capazes de serem auto ajustados e ainda assim apresentar bons resultados, o segundo necessita de mais testes iniciais por possuir mais parâmetros a serem configurados.

Pela comparação em termos de acurácia com os resultados obtidos por outros métodos que utilizaram as mesmas bases de dados, viu-se que os submodelos testados nesta dissertação apresentaram desempenho comparável ou superior, na maioria das vezes, mas também inferior, em alguns casos, o que indica que a abordagem neuroevolutiva proposta é possível de ser aplicada em um procedimento automático para Identificação de Sistemas, mas, definitivamente, não é perfeita e, possivelmente, pode ser aprimorada.

Como continuação imediata a este trabalho, pode-se sugerir a inclusão no conjunto de parâmetros gerais a serem evoluídos o período de *washout* T_0 , cuja influência não foi avaliada neste trabalho, e no conjunto de pesos a serem otimizados, pode-se avaliar também a inclusão de alguns ou de todos os pesos das matrizes \mathbf{W}^{in} e \mathbf{W}^{back} , como já foi considerado em alguns trabalhos (Ishu et al., 2004; Chouikhi et al., 2015, 2017). Posteriormente, pode-se pensar em uma abordagem que permita otimizar simultaneamente os hiperparâmetros e pesos da rede, reescalando ou não \mathbf{W} por ρ_W como já foi explorado com bons resultados em por Ferreira (2011) e Ferreira & Ludermir (2011). Durante o desenvolvimento desta dissertação, chegou-se a implementar uma abordagem com o AEIQ-R que considerava este cenário de otimização simultânea de hiperparâmetros e pesos, porém, a mesma não chegou a ser testada em todos os problemas.

Acredita-se que por utilizar uma métrica não supervisionada, o método SRG selecionou algumas soluções instáveis que acabaram prejudicando o desempenho dos

submodelos AEIQ-R/SRG/ESN e CMA-ES/SRG/ESN. De forma a tentar explorar mais este método, pode-se pensar em uma estratégia de otimização multiobjetivo, onde se buscaria minimizar não somente o coeficiente SRG do reservatório, mas também o erro de simulação da rede no conjunto de validação.

Referências bibliográficas

AGUIRRE, L. A. **Introdução à Identificação de Sistemas - Técnicas Lineares e Não Lineares: Teoria e Aplicação**. 4ª Ed. Editora UFMG, Belo Horizonte, MG, Brasil, 2015.

AGUIRRE, L. A.; MENDES, E. M. A. M. **Global nonlinear polynomial models: Structure, term clusters and fixed points**. International Journal of Bifurcation and Chaos, vol. 6, issue 2, p. 279-294, 1996.

AGUIRRE, L. A.; RODRIGUES, G. G.; JÁCOME, C. R. F. **Identificação de Sistemas Não-Lineares utilizando modelos NARMAX Polinomiais – Uma Revisão e Novos Resultados**. SBA Controle & Automação, Vol. 9, no. 2, 1998.

AHMADI, A.; KARRARI, M. **An Iterative Approach to Determine the Complexity of Local Models for Robust Identification of Nonlinear Systems**. International Journal of Control, Automation and Systems, vol. 10, issue 1, p. 1-10, 2012.

AL-DHAIFALLAH, M.; NISAR, K. S.; AGARWAL, P.; ELSAYYAD, A. **Modeling and Identification of heat exchanger process using least squares support vector machines**. Termal Science Year 2017, vol. 21, issue 6, p. 2859-2869, 2015.

ALEKSOVSKI, D.; KOCIJAN, J.; DZEROSKI, S. **Ensembles of Fuzzy Linear Model Trees for the Identification of Multioutput Systems**. IEEE Transactions on Fuzzy Systems, vol. 24, no. 4, 2016.

ALEKSOVSKI, D.; KOCIJAN, J.; DZEROSKI, S. **Model Tree Ensembles for the Identification of Multiple-Output Systems**. Proceedings of the European Control Conference (ECC), Strasbourg, France, 2014.

ANTONELO, E. A.; CAMPONOGARA, E; PLUCENIO, A. **System Identification of a Vertical Riser Model with Echo State Networks**. IFAC-Papers OnLine, vol. 48, issue 6, p. 304-310, 2015.

AZMI, K. Z. M.; PEBRIANTI, D.; IBRAHIM, Z.; SUDIN, S.; NAWAWI, W. **Simultaneous Computation of Model Order and Parameter Estimation for System Identification Based on Gravitational Search Algorithm**. Proceedings of the 6 th International Conference on Intelligent Systems, Modelling and Simulation (ISMS), Kuala Lumpur, Malaysia, 2015.

BABINEC, S.; POSPÍČHAL, J. **Improving the prediction accuracy of Echo State Networks by Anti-Ojas's learning**. In: DE SÁ et al (eds), *Artificial Neural Networks – ICANN 2007*. Lecture Notes in Computer Science, vol. 4688, Springer-Verlag, Berlin, Heidelberg, 2007a.

BABINEC, S.; POSPÍČHAL, J. **Optimization of Echo State Neural Networks for Electrical Load Forecasting**. *Neural Network World*, 17, 2, Technology Collection, p. 133, 2007b.

BACK, T.; HAMMEL, U.; SCHWEFEL, H. P. **Evolutionary computation: comments on the history and current state**. *IEEE Transactions on Evolutionary Computation*, vol. 1, issue 1, p. 3–17, 1997.

BARRY, T., **A Data Driven Approach to Constrained Control**. Master Thesis - School of Electrical and Computer Engineering, RMIT University, Melbourne, Australia, 2004.

BASTERRECH, S.; ALBA, E.; SNASEL, V. **An Experimental Analysis of the Echo State Network Initialization Using the Particle Swarm Optimization**. *Proceedings of the Sixth World Congress on Nature and Biologically Inspired Computing*, p. 214-219, Porto, Portugal, 2014.

BECERRA, V. M.; CALADO, J. M. F.; SILVA, P. M.; GARCES, F. **System Identification Using Dynamic Neural Networks: Training and initialization aspects**. *IFAC Proceedings Volumes*, vol. 35, issue 1, p. 235-240, 2002.

BERTSCHINGER, N.; NATSCHLÄGER, T. **Real-time computation at the edge of chaos in recurrent neural networks**. *Neural Computation*, vol. 16, issue 7, p. 1413-1436, 2004.

BIAN, X.; MOU, C. **Identification of Non-linear Dynamic Model of UUV Based on ESN Neural Network**. *Proceedings of the 30 th Chinese Control Conference*, Yantai, China, 2011.

BIANCHI, F. M.; SCARDAPANE, S.; UNCINI, A.; RIZZI, A.; SADEGHIAN, A. **Prediction of telephone calls load using Echo State Network with exogenous variables**. *Neural Networks*, vol. 71, p. 204-213, 2015.

BILLINGS, S. A. **Nonlinear System Identification: NARMAX Methods in Time, Frequency, and Spatio-Temporal Domains**. First Ed. John Wiley and Sons, United Kingdom, 2013.

BIRPOUTSOUKIS, G.; CSURCSIA, P. Z.; SHOUKENS, J. **Efficient multidimensional regularization for Volterra Series estimation**. *Mechanical Systems and Signal Processing*, vol 104, p. 896-914, 2018.

BITTANTI, S.; PIRODDI, L. **Nonlinear identification and control of a heat exchanger: a neural network approach**. Journal of the Franklin Institute, vol. 334 B, n. 1, p. 135-153, 1997.

BLOEM, J. **System Identification Competition, Benchmark tests for estimation methods of thermal characteristics of buildings and building components**. Joint Research Centre, Ispra, Italy, 1994.

BOCCATO, L.; LOPES, A.; ATTUX, R.; VON ZUBEN, J. **An Echo State Network Architecture Based on Volterra Filtering and PCA with Application to the Channel Equalization Problem**. Proceedings of the International Joint Conference on Neural Networks, San Jose, California, 2011.

BOTELHO, A. C. O. P. **Algoritmo Evolutivo com Inspiração Quântica, Representação Mista e Decaimento de Pesos Aplicado a Neuroevolução**. Internal Research Reports, n. 38, Departamento de Engenharia Elétrica, Pontifícia Universidade Católica do Rio de Janeiro (PUC-Rio), Rio de Janeiro, 2014.

BRAGA, A. P.; CARVALHO, A. P. L.; LUDERMIR, T. B. **Redes Neurais Artificiais: Teoria e Aplicações**. 1ª Ed, Ed. LTC, Rio de Janeiro, RJ, Brasil, 2000.

CAO, S.; XU, W.; HU, X. **Dual Adaptive Control of Nonlinear Stochastic Systems based on Echo State Network**. Proceedings of the 27 th Chinese Control and Decision Conference (CCDC), Qingdao, China, p. 4579 - 4584, 2015.

CASTELLANOS, O. F. A. **Identificación de sistemas utilizando lógica difusa**. Tese (Mestrado em Engenharia Eletrônica) - Centro Nacional de Investigación y Desarrollo Tecnológico, Cuernavaca, Morelos, México, 2008.

CHATZIDIMITRIOU, K. C.; MITKAS, P. A. **A NEAT Way for Evolving Echo State Networks**. Proceedings of the European Conference on Artificial Intelligence, p. 909-914, 2010.

CHATZIDIMITRIOU, K. C.; MITKAS. **Adaptive reservoir computing through evolution and learning**. Neurocomputing, vol. 103, p. 198-209, 2013.

CHEN, S.; BILLINGS, S. A. **Neural networks for nonlinear dynamic system modelling and identification**. International Journal of Control, vol. 56, issue 2, p. 319-346, 1992.

CHINARRO, D. **System Engineering Applied to Fuenmayor Karst Aquifer (San Julián de Banzo, Huesca) and Collins Glacier (King George Island, Antarctica)**. Springer Theses, Outstanding Ph. D. thesis by the University of Zaragoza, Spain, 2014.

CHITSAZAN, M. A.; FADALI, M. S.; NELSON, A. K.; TRZYNADLOWSKI, A. M. **Wind Speed Forecasting Using an Echo State Network with Nonlinear Output Functions**. American Control Conference (ACC), Seattle, USA, 2017.

CHOUIKHI, N.; AMMAR, B.; ROKBANI, N.; ALIM, A. M. **PSO-based analysis of Echo State Network parameters for time series forecasting**. Applied Soft Computing, vol. 55, p. 211-225, 2017.

CHOUIKHI, N.; AMMAR, B.; ROKBANI, N.; ALIM, A. M.; ABRAHAM, A. **A Hybrid Approach Based on Particle Swarm Optimization for Echo State Network Initialization**. Proceedings of IEEE International Conference on Systems, Man and Cybernetics, Kowloon, China, p. 2896-2901, 2015.

COBAN, R. **A context layered locally recurrent neural network for dynamic system identification**. Engineering Applications of Artificial Intelligence, vol. 26, issue 1, p. 241–250, 2013.

COELHO, A. A. R.; COELHO, L. S. **Identificação de Sistemas Dinâmicos Lineares**. Ed. UFSC, Universidade Federal de Santa Catarina, Florianópolis, Santa Catarina, 2004.

CRUZ, A. V. A. **Algoritmos Evolutivos com Inspiração Quântica para Problemas com Representação Numérica**. Tese (Doutorado em Engenharia Elétrica) - Departamento de Engenharia Elétrica, Centro Técnico Científico, Pontifícia Universidade Católica do Rio de Janeiro (PUC-Rio), Rio de Janeiro, RJ, Brasil, 2007.

CYBENKO, G. V. **Approximation by superpositions of a sigmoidal function**. Mathematics of Control, Signals and Systems, vol.2, no. 4, p. 303-314, 1989.

DE MOOR, B.L.R. (ed.). **DalSy: Database for the Identification of Systems**. Department of Electrical Engineering, ESAT/STADIUS, KU Leuven, Belgium. Disponível em: <<http://homes.esat.kuleuven.be/~smc/daisy/>>. Acesso em: 31 maio 2018.

DEFOIN-PLATEL, M. S.; SCHILIEBS, S.; KASABOV, N. **Quantum-Inspired Evolutionary Algorithm: a Multimodel EDA**. IEEE Transactions on Evolutionary Computation, vol. 13, issue 6, p. 1218-1232, 2009.

DENG, Z.; ZHANG, Y. **Complex Systems Modeling Using Scale-Free Highly-Clustered Echo State Network**. Proceedings of the International Joint Conference on Neural Networks, Vancouver, BC, Canada, p. 3128-3135, 2006.

DERRAC, J.; GARCÍA, S.; MOLINA, D.; HERRERA, F. **A practical tutorial on the use of nonparametric statistical tests as a methodology for comparing evolutionary and swarm intelligence algorithms**. Swarm and Evolutionary Computation, vol. 1, p. 3-18, 2011.

DUAN, H.; WANG, X. **Echo State Networks With Orthogonal Pigeon-Inspired Optimization for Image Restoration**. IEEE Transactions on Neural Networks and Learning Systems, vol. 27, issue 11, p. 2413-2425, 2016.

DUDUL, S. V.; GHATOL, A. A. **Identification of a Typical CD Player Arm Using A Two-layer Perceptron Neural Network Model**. Proceedings of the International Joint Conference on Neural Networks, Portland, OR, USA, p. 1157-1162, 2003.

EMARA-SHABAİK, H.E.; AHMED, M. S.; AL-AJMI, K. H. **Parameter Estimation of Wiener-Hammerstein models**. JSME International Journal Series C, vol. 44, issue 1, p. 118-124, 2001.

ESPINOZA, M.; PELCKMANS, K.; HOEGAERTS, L.; SUYKENS, J.; MOOR, B. D. **A comparative study of ls-svms applied to the silver box identification problem**. Proceedings of the IFAC Symposium on Nonlinear Control Systems, Stuttgart, Germany, p. 369-374, 2004.

ESPINOZA, M.; SUYKENS, J. A. K.; De Moor, B. **Kernel based partially linear models and nonlinear identification**. IEEE Transactions on Automatic Control, no. 10, p. 1602-1606, 2005.

FERREIRA, A. A. **Um método para Design e Treinamento de Reservoir Computing Aplicado à Previsão de Séries Temporais**. Tese (Doutorado em Ciência da Computação) – Centro de Informática, Universidade Federal de Pernambuco, Recife, Pernambuco, Brasil, 2011.

FERREIRA, A. A.; LUDERMIR, T. B. **Comparing Evolutionary Methods for Reservoir Computing Pre-training**. Proceedings of the International Joint Conference on Neural Networks, San Jose, California, p. 283-290, 2011.

FERREIRA, A. A.; LUDERMIR, T. B. **Evolutionary strategy for simultaneous optimization of parameters, topology and reservoir weights in Echo State Networks**. International Joint Conference on Neural Networks, Barcelona, Spain, 2010.

FERREIRA, A. A.; LUDERMIR, T. B. **Genetic Algorithm for Reservoir Computing Optimization**. Proceedings of the International Joint Conference on Neural Networks, Atlanta, Georgia, USA, 2009.

FERREIRA, A. A.; LUDERMIR, T. B.; AQUINO, R. R. B. **An approach to reservoir computing design and training**. Expert Systems with Applications, vol. 40, issue 10, p. 4172-4182, 2013.

FERREIRA, C. C.; SERRA, G. L. O. **An Approach for Fuzzy Frequency Response Estimation of Flexible Robot Arm From Experimental Data**. Proceedings of IEEE International Conference on Industrial Technology, Athens, Greece, p. 166-171, 2012.

FOGEL, L. J.; OWENS, A. J.; WALSH, M. J. **Artificial Intelligence through Simulated Evolution**. John Wiley, 1966.

GANG, L.; LI, B. J.; YU, X. G.; CHENG, C. T. **Echo State Network with Bayesian Regularization for Forecasting Short-Term Power Production of Small Hydropower Plants**. *Energies*, v. 8, no. 10, p. 12228-12241, 2015.

GIBBONS, T. E. **Unifying quality metrics for reservoir networks**. Proceedings of International Joint Conference on Neural Networks, Barcelona, Spain, p. 1-7, 2010.

GOODMAN, E.; VENTURA, D. **Spatiotemporal Pattern Recognition via Liquid State Machines**. Proceedings of the International Joint Conference on Neural Networks, Vancouver, BC, Canada, p. 3848 – 3853, 2006.

GOUDARZI, A.; STEFANOVIC, D. **Towards a calculus of echo state networks**. *Procedia Computer Science*, vol. 41, p. 176-181, 2014.

HAN, K. H.; KIM, J. H. **Genetic quantum algorithm and its application to combinatorial optimization problem**. Proceedings of the Congress on Evolutionary Computation, La Jolla, CA, USA, p. 1354-1360, 2000.

HAN, K. H.; KIM, J. H. **Quantum-inspired evolutionary algorithm for a class of combinatorial optimization**. *IEEE Transactions on Evolutionary Computation*, vol. 6, issue 6, p. 580–593, 2002.

HAN, M.; WANG, X. **Weighted Support Vector Echo State Machine for Multivariate Dynamic System Modeling**. Proceedings of American Control Conference (ACC), Portland, Oregon, USA, p. 4824-4828, 2014.

HANSEN, N. **CMA-ES Applications**. 2009. Online. Disponível em: <<https://www.lri.fr/~hansen/cmaapplications.pdf>>. Acesso em: 31 maio 2018.

HANSEN, N. **CMA-ES version 3.61 for MATLAB**. 2012. Disponível em: <https://www.lri.fr/~hansen/cmaes_inmatlab.html>. Acesso em: 31 maio 2018.

HANSEN, N. **The CMA Evolution Strategy: A Tutorial**. ArXiv e-prints, [arXiv:1604.00772v1](https://arxiv.org/abs/1604.00772v1), 2016.

HANSEN, N.; AUGER, A.; ROS, R.; FINCK, S.; POSIK, P. **Comparing Results of 31 Algorithms from the Black-box Optimization Benchmarking BBOB-2009**. Proceedings of the 12th Annual Conference Companion on Genetic and Evolutionary Computation, GECCO, Portland, Oregon, USA, p. 1689–1696, 2010.

HANSEN, N.; KERN, S. **Evaluating the CMA evolution strategy on multimodal test functions**. In YAO et al. (eds) *Parallel Problem Solving from*

Nature – PPSN VIII, Lecture Notes in Computer Science, vol. 3242, p. 282–291, Springer-Verlag, Berlin, Heidelberg, 2004.

HANSEN, N.; OSTERMEIER, A. **Completely derandomized self-adaptation in evolution strategies**. Evolutionary Computation, vol. 9, issue 2, p. 159–195, 2001.

HANSEN, N.; OSTERMEIER. **Adapting arbitrary normal mutation distributions in evolution strategies: The covariance matrix adaptation**. Proceedings of IEEE International Conference on Evolutionary Computation, Nagoya, Japan, p. 312-317, 1996.

HANSSON, A.; LIU, Z.; VANDENBERGHE, L. **Subspace system identification via weighted nuclear norm optimization**. Proceedings of the 51st IEEE Conference on Decision and Control, Maui, Hawaii, USA, p. 3439-3444, 2012.

HAYKIN, S. **Neural Networks and Learning Machines**. 3 rd ed., Prentice Hall, New Jersey, USA, 2009.

HAYKIN, S. **Neural Networks: A Comprehensive Foundation**. 2 nd ed., Prentice Hall, 1999.

HJALMARSSON, H.; SCHOUKENS, J. **On direct identification of physical parameters in nonlinear models**. Proceedings of the IFAC Symposium on Nonlinear Control Systems, Stuttgart, Germany, p. 519-524, 2004.

HOLLAND, J. H. **Adaptation in Natural and Artificial Systems**. University of Michigan Press, Ann Arbor, 1975.

ISHU, K.; VAN DER ZANT, T.; BECANOVIC, V.; Plöger, P. **Identification of motion with echo state network**. Proceedings of MTTs/IEEE TECHNO-OCEAN'04, Kobe, Japan, vol. 3, p. 1205-1210, 2004.

JAEGER, H. **A tutorial on training recurrent neural networks, covering BPPT, RTRL, EKF and the echo state network approach**. GMD Report 159, German National Research Center for Information Technology, 5th revision, 2013.

JAEGER, H. **Simple and very simple Matlab toolbox for Echo State Networks**. 2009. Online. Disponível em <<http://reservoir-computing.org/node/129>>. Acesso em 17/01/2017.

JAEGER, H. **The “echo state” approach to analysing and training recurrent neural networks** – with an Erratum note. GMD Report (148), p. 1-47, 2010.

JAEGER, H.; HAAS, H. **Harnessing Nonlinearity: Predicting Chaotic Systems and Saving Energy in Wireless Communication**. Science, vol. 304, issue 5667, p. 78-80, 2004.

JAEGER, H.; LUKOSEVICIUS, M.; POPOVIC, D.; SIEWERT, U. **Optimization and applications of echo state networks with leaky integrator neurons.** Neural Networks, vol. 20, issue 3, p. 335-352, 2007.

JANG, J. S.; HAN, K. H.; KIM, J. H. **Face detection using quantum inspired evolutionary algorithm.** Proceedings of the Congress on Evolutionary Computation, Portland, OR, USA, p. 21002106, 2004.

JIANG, F.; BERRY, H.; SCHOENAUER, M. **Supervised and Evolutionary Learning of Echo State Networks.** In: RUDOLPH et al. (eds) Parallel Problem Solving from Nature – PPSN X. Lecture Notes in Computer Science, vol. 5199, Springer-Verlag, Berlin, Heidelberg, 2008.

JOHNSON, R. A.; WICHERN, D. W. **Applied Multivariate Statistical Analysis.** 6 ed., Prentice-Hall, Upper Saddle River, New Jersey, USA, 2007.

JOSE, T. J.; MYTHILI, P. **FMRI Segmentation Using Echo State Neural Network.** International Journal of Soft Computing, vol. 3 (1), p. 38-43, 2008.

KANDROODI, M. R.; MOSHIRI, B. **Identification and Model Predictive Control of Continuous Stirred Tank Reactor Based on Artificial Neural Networks.** Proceedings of the 2 th International Conference on Control, Instrumentation and Automation, Shiraz, Iran, p. 338-343, 2011.

KEEL. **KEEL Software Tool.** Disponível em: <<http://keel.es/>>. Acesso em: 02 junho 2018.

KEESMAN, K. J. **System Identification: An Introduction.** Advanced Textbooks in Control and Signal Processing, Springer London, New York, 2011.

KIM, S.; BILLARD, A. **Estimating the non-linear dynamics of free-flying objects.** Robotics and Autonomous Systems, vol. 60, issue 9, p. 1108-1122, 2012.

KOLMOGOROV, A. N. **On the representation of continuous functions of several variables by superposition of continuous functions of one variable and addition.** Doklady Akademii. Nauk USSR, vol. 114, pp. 679-681, 1957.

KOPRINKOVA-HRISTOVA, P.; ANGELOVA, D. **Clustering of Spectral Images using Echo State Networks.** IEEE International Symposium on Innovations in Intelligent Systems and Applications, Albena, Bulgaria, 2013.

KOPRINKOVA-HRISTOVA, P.; OUBATI, M.; PALM, G. **Adaptive Critic Design with Echo State Network.** IEEE International Conference on Systems Man and Cybernetics, Istanbul, Turkey, p. 1010-1015, 2010.

LA CAVA, W.; DANAI, K.; SPECTOR, L. **Inference of compact nonlinear dynamic models by epigenetic local search**. Engineering Applications of Artificial Intelligence, vol. 55, p. 292-306, 2016.

LEGENSTEIN, R.; MAASS, W. **Edge of chaos and prediction of computational performance for neural circuit models**. Neural Networks, vol. 20, p. 323-334, 2007.

LEVA, A.; PAPADOULOS, A. V. **Tuning of event-based industrial controllers with simple stability guarantees**. Journal of Process Control, vol. 23, p. 1251-1260, 2013.

LI, G.; NIU, P.; ZHANG, W.; ZHANG, Y. **Control of discrete chaotic systems based on echo state network modeling with an adaptive noise canceler**. Knowledge-Based Systems, vol. 35, p. 35-40, 2012.

LIGHTBODY, G.; IRWIN, G. W. **Nonlinear Control Structures Based on Embedded Neural System Models**. IEEE Transactions on Neural Networks, vol. 8, issue 3, p. 553-567, 1997.

LIN, X. F.; ZHANG, M. Q. **Modelling of the Vertical Raw Cement Mill Grinding Process Based on the Echo State Network**. 12 th World Congress on Intelligent Control and Automation (WCICA), Guilin, China, p. 2498 - 2502, 2016.

LIN, X.; DING, Q.; KONG, W.; SONG, C.; HUANG, Q. **Adaptive dynamic programming-based optimal tracking control for nonlinear systems using general value iteration**. IEEE Symposium on Adaptive Dynamic Programming and Reinforcement Learning (ADPRL), Orlando, FL, USA, 2014.

LINDEN, R. **Algoritmos Genéticos**. 3ª ed, Ciência Moderna, Rio de Janeiro, RJ, Brasil, 2012.

LIU, D.; WANG, J.; WANG, H. **Short-term wind speed forecasting based on spectral clustering and optimised echo state networks**. Renew. Energy 2015, 11, p. 68-75, 2015.

LIU, J.; WENBO, X.; SUN, J. **Nonlinear System Identification of Hammerstein and Wiener Model Using Swarm Intelligence**. Proceedings of the IEEE International Conference on Information Acquisition, Weihai, Shandong, China, p.1219-1223, 2006.

LIU, Z.; HANSSON, A.; VANDENBERGHE, L. **Nuclear norm system identification with missing inputs and outputs**. System and Control Letters, vol. 62, issue 8, p. 605-612, 2013.

LJUNG, L. **System Identification- Theory for the User**. Prentice Hall, Englewood Cliffs, 1987.

LJUNG, L.; ZHANG, Q.; LINDSKOG, P.; JUDITSKI, A. **Estimation of grey box and black box models for non-linear circuit data**. Proceedings of the IFAC Symposium on Nonlinear Control Systems, Stuttgart, Germany, p. 399-404, 2004.

LOVLID, R. **A novel method for training an echo state network with feedback-error learning**. Advances in Artificial Intelligence, vol. 2013, Hindawi Publishing Corporation, p. 1-9, 2013.

LUKOSEVICIUS, M. **A Practical Guide to Applying Echo State Networks**. In: MONTAVON, G.; ORR, G. B.; MÜLLER, K. R. (eds) Neural Networks: Tricks of the Trade. Lecture Notes in Computer Science, vol. 7700, Springer, Berlin, Heidelberg, 2012.

LUKOSEVICIUS, M.; JAEGER, H. **Reservoir Computing Approaches to Recurrent Neural Network Training**. Computer Science Review, vol. 3, issue 3, p. 127-149, 2009.

LUKOSEVICIUS, M.; JAEGER, H.; SCHRAUWEN, B. **Reservoir Computing Trends**. KI – Künstliche Intelligenz, vol. 26, issue 4, Springer-Verlag, p. 365-371, 2012.

LUN, S.; WANG, S.; GUO, T.; DU, C. **An I-V model based on time warp invariant echo state network for photovoltaic array with shaded solar cells**. Solar Energy, vol. 105, p. 529-541, 2014.

MAASS, W.; NATSCHLÄGER, T.; MARKRAM, H. **Real-time computing without stable states: A new framework for neural computation based on perturbations**. Neural Computation, vol. 14, issue 11, p. 2531-60, 2002.

MAGERL, M.; CEPERIC, V.; BARIC, A. **Echo State Networks for Black Box Modelling of Integrated Circuits**. IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems, vol. 35, issue 8, p. 1309-1317, 2015.

MAGERL, M.; STOCKREITER, C.; EISENBERGER, O.; BARIC, A. **Adaptive algorithm for sampling nonlinear circuit behavior in time-domain**. IEEE 20th Workshop on Signal and Power Integrity (SPI), Turin, Italy, p. 1-4, 2016.

MARCONATO, A.; SJÖBERG, J.; SUYKENS, J.; Schoukens, J. **Identification of the Silverbox Benchmark Using Nonlinear State-Space Models**. Proceedings of the 16th IFAC Symposium on System Identification, Brussels, Belgium, p. 632-637, 2012.

MARTINS, G. P. M. **Modelo Híbrido Baseado em Redes Neurais Echo State Network Otimizadas por Algoritmos Evolucionários para Identificação Automática de Sistemas Dinâmicos e a sua Aplicação à Identificação do Nível do Gerador de Vapor de uma Usina Nuclear PWR**. Dissertação (Mestrado em Engenharia Elétrica) - Departamento

de Engenharia Elétrica, Centro Técnico Científico, Pontifícia Universidade Católica do Rio de Janeiro (PUC-Rio), Rio de Janeiro, RJ, Brasil, 2016.

MARTINS, G.; VELLASCO, M.; SCHIRRU, R. **Identificação em malha fechada do nível de água de um gerador de vapor de usina nuclear utilizando *echo state network* sintonizada por algoritmos genéticos.** XII Simpósio Brasileiro de Automação Inteligente (SBAI), Natal, RN, Brasil, p. 491 – 496, 2015.

MARTINS, G.; VELLASCO, M.; SCHIRRU, R.; VELLASCO, P. **Closed Loop Identification of Nuclear Steam Generator Water Level Using ESN Network Tuned by Genetic Algorithm.** In: ILIADIS, L; JAYNE, C. (eds), Engineering Applications of Neural Networks, Communications in Computer and Information Science, vol. 517, Springer, 2015.

MATLAB 2016 b. **Página do MATLAB versão 2016b.** Online. Disponível em: <https://www.mathworks.com/products/new_products/release2016b.html>. Acesso em: 31 maio 2018.

MEFTAH, B.; LÉZORAY, O.; BENYETTOU, A. **Novel Approach Using Echo State Networks for Microscopic Cellular Image Segmentation.** Cognitive Computation, vol. 8, issue 2, p. 237-245, 2016.

MICHALEWICZ, Z. **Genetic algorithms + data structures = evolution programs (2nd, extended ed.).** Springer-Verlag, Berlin, Heidelberg, 1994.

MOON, U. C.; LIM, J.; LEE, K. Y. **A Comparison Study of MIMO Water Wall Model with Linear, MFNN and ESN Models.** Journal of Electrical Engineering and Technology, 11 (2); p. 265-273, 2016.

MOORE, M.; NARAYANAN, A. **Quantum-inspired computing.** 1995.

MOREIRA, W. J. **Identificação Linear a Parâmetros Variantes no Tempo de Sistemas Não-Lineares.** Dissertação (Mestrado em Engenharia Elétrica) - Instituto Militar de Engenharia, Rio de Janeiro, RJ, Brasil, 2008.

NARAYANAN, A.; MOORE, M. **Quantum-inspired genetic algorithms.** Proceedings of IEEE International Conference on Evolutionary Computation, Nagoya, Japan, p. 61-66, 1996.

NARENDRA, K. S. S.; PARTHASARATHY, K. **Identification and Control of Dynamical Systems Using Neural Networks.** IEEE transactions on neural networks, v.1, n.1, p. 4-27, 1990.

NELLES, O. **Nonlinear System Identification: From Classical Approaches to Neural Networks and Fuzzy Models.** Springer, Germany, 2001.

NIÑO, C. H. V. **Modelo Híbrido ESN-GA-SRG: uma abordagem de otimização e seleção de topologias em Echo State Networks para previsão de séries**

temporais. Tese (Doutorado em Engenharia Elétrica) – Departamento de Engenharia Elétrica, Centro Técnico Científico, Pontifícia Universidade Católica do Rio de Janeiro (PUC-Rio), Rio de Janeiro, RJ, Brasil, 2016.

NORTON, D.; VENTURA, D. **Improving liquid state machines through iterative refinement of the reservoir.** Neurocomputing, vol. 73, p. 2893-2904, 2010.

NORTON, D.; VENTURA, D. **Improving the Separability of a Reservoir Facilitates Learning Transfer.** Proceedings of the International Joint Conference on Neural Networks, Atlanta, GA, USA, p. 2288-2293, 2009.

NOZARI, H. A.; BANADAKI, H. D. **Intelligent computationally efficient modelling of multi-input multi-output non-linear dynamical process plants: An industrial steam generator case study.** Proceedings of the Institution of Mechanical Engineers, Part I: J. Systems and Control Engineering, vol. 228, issue 5, p. 278-294, 2014.

OZTURK, M; XU, D; PRINCIPE, J. **Analysis and Design of Echo State Networks for Function Approximation.** Neural Computation, 19, 2007.

PADUART, J.; HORVATH, G.; SCHOUKENS, J. **Fast identification of systems with nonlinear feedback.** Proceedings of the IFAC Symposium on Nonlinear Control Systems, Stuttgart, Germany, p. 381-385, 2004.

PADUART, J.; LAUWERS, L.; SWEVERS, J.; SMOLDERS, K.; SCHOUKENS, J; PINTELON, R. **Identification of nonlinear systems using Polynomial Nonlinear State Space models.** Automatica, vol. 46, no. 4, p. 647-656, 2010.

PAN, Y; WANG, J. **Model Predictive Control of Unknown Nonlinear Dynamical Systems Based on Recurrent Neural Networks.** IEEE Transactions on Industrial Electronics, vol. 59, no. 8, p. 3089-3101, 2012.

PARK, J.; LEE, B.; KANG, S.; KIM, P. Y.; KIM, H. J. **Online Learning Control of Hydraulic Excavators Based on Echo-State Networks.** IEEE Transactions on Automation Science and Engineering, vol. 14, no. 1, p. 249-259, 2017.

PELLEGRINETTI, G.; BENTSMAN, J. **Nonlinear Control Oriented Boiler Modeling – A Benchmark Problem for Controller Design.** IEEE Transactions on Control Systems Technology, vol. 4, no. 1, p. 57-64, 1996.

PEPONA, E.; PAOLETTI, S. **Identification of Piecewise Affine LFR Models of Interconnected Systems.** IEEE Transactions On Control Systems Technology, vol. 19, no. 1, p. 148-155, 2011.

PESSANHA, J. F. M. **Normal Multivariada: Função densidade conjunta e contorno de probabilidade.** Pdf online. Disponível em: <<http://professorjf.webs.com/Normal%20multivariada%202013.pdf>>. Acesso em: 02 junho 2018.

PETROVIC, I; BAOTIC, M.; PERIC, N. **Model structure selection for nonlinear system identification using feedforward neural networks**. Proceedings of the International Joint Conference on Neural Networks, Como, Italy, p. 53-57, 2000.

PINHO, A. G. **Algoritmo Evolucionário com Inspiração Quântica e Representação Mista Aplicado a Neuroevolução**. Dissertação (Mestrado em Engenharia Elétrica) - Departamento de Engenharia Elétrica, Centro Técnico Científico, Pontifícia Universidade Católica do Rio de Janeiro (PUC-Rio), Rio de Janeiro, RJ, Brasil, 2010.

PLÖGER, P. G; ARGHIR, A.; GÜNTHER, T.; HOSSEINY, R. **Echo State Networks for Mobile Robot Modeling and Control**. In: POLANI et al. (Eds.): Robocup 2003, LNAI 3020 p., Springer-Verlag Berlin Heidelberg, p. 157-168, 2004.

RECHENBERG I. **Evolutionsstrategie: Optimierung technischer systeme nach prinzipen der biologischen evolution**. Fromman-Holzboog Verlag, Alemanha, 1973.

RECHENBERG, I. **Cybernetic solution path of an experimental problem**. Royal Aircraft Establishment, Library Translation 1122, Farnborough, 1965.

REGO, J. B. A; MARTINS, A. M.; COSTA, E. B. **Deterministic System Identification Using RBF Networks**. Mathematical Problems in Engineering, vol. 2014, p. 1-10, 2014.

RIEFFEL, E.; POLAK, W. **An introduction to quantum computing for non-physicists**. ACM Computing Surveys, vol. 32, issue 3, p. 300–335, 2000.

ROESCHIES, B.; IGEL, C. **Structure optimization of reservoir networks**. Logic Journal of the Igpl, vol. 18, issue 5, p. 635-669, 2010.

ROSENSTEIN, M. T.; COLLINS, J. J.; DE LUCA; C. J. **A practical method for calculating largest Lyapunov exponents from small data sets**. *Phys. D*, vol. 65, p. 117-134, 1993.

RUEDA, C. V. **EsnPredictor: Ferramenta de Previsão de Séries Temporais Baseada em Echo State Networks Otimizadas por Algoritmos Genéticos e Particle Swarm Optimization**. Dissertação (Mestrado em Engenharia Elétrica) - Departamento de Engenharia Elétrica, Centro Técnico Científico, Pontifícia Universidade Católica do Rio de Janeiro (PUC-Rio), Rio de Janeiro, RJ, Brasil, 2014.

SALMEN, M.; PLÖGER, P. G. **Echo State Networks used for Motor Control**. Proceedings of IEEE International Conference on Robotics and Automation, Spain, p. 1953-1958, 2005.

SANTOS, M. M.; TINÓS, R. **Investigação sobre mutações q-gaussinas na estratégia evolutiva com adaptação da matriz de covariância**. Anais do X Congresso Brasileiro de Inteligência Computacional, Fortaleza- CE, Brasil, p. 1-8, 2011.

SCHMIDHUBER, J; WIERSTRA, D.; GAGLILOLO, M.; GOMEZ, F. J. **Training recurrent networks by Evolino**. Neural Computation 19 (3), p. 757–779, 2007.

SCHRAUWEN, B.; BÜSING, L.; LEGENSTEIN, R. **On computational power and the order-chaos phase transition in reservoir computing**. Advances in Neural Information Processing Systems, vol. 21, p. 1-8, 2008.

SCHRAUWEN, B.; VERSTRAETEN, D.; CAMPENHOUT, J. V. **An overview of reservoir computing: theory, applications and implementations**. Proceedings of European Symposium on Artificial Neural Networks, Bruges, Belgium, p. 471-482, 2007.

SCHUMACHER, R.; OLIVEIRA, G. H. C., **Uma nova abordagem vector fitting para identificação de sistemas com dados no domínio do tempo**. XII Simpósio Brasileiro de Automação Inteligente, 2015, Natal – RN. Anais do XII Simpósio Brasileiro de Automação Inteligente, p. 283-288, 2015.

SCWEFEL, H. P. **Kybernetische Evolution als Strategie der experimentellen Forschung in der Strömungstechnik**. Master's thesis - Technical University of Berlin, Berlin, 1965.

SCHWEFEL. **Evolutionsstrategie und numerische Optimierung**. Technical University of Berlin, 1975.

SERGIO, A. T.; LUDERMIR, T. B. **PSO for Reservoir Computing Optimization**. In: VILLA et al. (eds) Artificial Neural Networks and Machine Learning – ICANN 2012. Lecture Notes in Computer Science, vol. 7552, Springer-Verlag, Berlin, Heidelberg, 2012.

SILVA, A. F. **Optimização de Métodos de Núcleo Utilizando Algoritmos de Enxame**. Tese (Doutorado em Informática) - Departamento de Informática, Universidade de Évora, Évora, Portugal, 2014.

SILVA, I.; SPATTI, D.; FLAUZINO, R. **Redes Neurais Artificiais para engenharia e ciências aplicadas: curso prático**. Ed. Artliber, São Paulo, Brasil, 2010.

SILVA, M. A. B. **Modelos Neuro-Evolucionários de Redes Neurais Spiking Aplicados ao Pré-Diagnóstico de Envelhecimento Vocal**. Tese (Doutorado em Engenharia Elétrica) - Departamento de Engenharia Elétrica, Centro Técnico Científico, Pontifícia Universidade Católica do Rio de Janeiro (PUC-Rio), Rio de Janeiro, RJ, Brasil, 2014.

SKOWRONSKI, M. D.; HARRIS, J. G. **Automatic speech recognition using a predictive echo state network classifier**. Neural Networks, vol. 20, issue 3, p. 414-423, 2007a.

SKOWRONSKI, M. D.; HARRIS, J. G. **Noise-Robust Automatic Speech Recognition Using a Predictive Echo State Network**. IEEE Transactions on Audio, Speech and Language Processing, Vol. 15, no. 5, p. 1771-1774, 2007b.

SOLEIMANI, H.; LUCAS, C.; ARAABI, B. N. **Recursive Gath–Geva clustering as a basis for evolving neuro-fuzzy modeling**. Proceedings of IEEE International Conference on Fuzzy Systems, Barcelona, Spain, 2010.

SONG, Q.; ZHAO, X.; FENG, Z. **Improved Recursive Least Squares Algorithm Based on Echo State Neural Network for Nonlinear System Identification**. Proceedings of the 30 th Chinese Control Conference, Yantai, China, p. 1692-1695, 2011.

SQUARTINI, S.; CECCHI, S.; ROSSINI, M.; PIAZZA, F. **Echo State Networks for Real-Time Audio Applications**. In: Liu et al. (eds) Advances in Neural Networks – ISNN 2007, Lecture Notes in Computer Science, vol. 4493, Springer, Berlin, Heidelberg, 2007.

SRAGNER, L.; SCHOUKENS, J.; HORVATH, G. **Modeling of slightly nonlinear systems: a neural network approach**. Proceedings of the IFAC Symposium on Nonlinear Control Systems, Stuttgart, Germany, p. 387-392, 2004.

STEIL, J. J. **Backpropagation-decorrelation: Recurrent learning with $O(N)$ complexity**. Proceedings of the IEEE International Joint Conference on Neural Networks, vol. 2, Budapest, Hungary, p. 843-848, 2004.

STEIL, J. J. **Online reservoir adaptation by intrinsic plasticity for backpropagation-decorrelation and echo state learning**. Neural Networks, vol. 20, issue 3, p. 353-364, 2007.

STRAUB, T.; WUSTLICH, W.; LABAHN, R. **Design strategies for weight matrices of Echo State Networks**. Neural Computation 24, issue 12, 2012.

STRICKLER, A. **Análise do impacto das estratégias de seleção de tradicionais MOEAS em MOEDAS: CMA-ES e UMDA**. Dissertação (Mestrado em Informática) – Setor de Ciências Exatas, Universidade Federal do Paraná, Curitiba, PR, Brasil, 2017.

TAVAKOLPOUR-SALEH, A. R.; NASSIB, S. M.; SEPASYAN, A.; HASHEMI, S. M. **Parametric and nonparametric system identification of an experimental turbojet engine**. Aerospace Science and Technology, vol. 43, p. 21-29, 2015.

TRAWINSKI, B.; SMETEK, M; TELEC, Z.; LASOTA, T. **Nonparametric Statistical Analysis for Multiple Comparison of Machine Learning Regression Algorithms**. International Journal of Applied Mathematics and Computer Science, vol. 22, no. 4, p. 867-881.

TREBATICKÝ, P. **Recurrent Neural Network Training with the Extended Kalman Filter**. IIT. SRC, p.57-64, 2005.

VAN MULDER, A.; SCHOUKENS, J.; VANBEYLEN, L. **Identification of systems with localised nonlinearity: From state-space to block-structured models**. Automatica, vol. 49, issue 5, p. 1392-1396, 2013.

VERDULT, V. **Identification of local linear state-space models: the silver-box case study**. Proceedings of the IFAC Symposium on Nonlinear Control Systems, Stuttgart, Germany, p. 393-398, 2004.

VERSTRAETEN, D.; SCHRAUWEN, B.; D'HAENE, M.; STROOBANDT, D. **An experimental unification of reservoir computing methods**. Neural Networks, vol. 20, issue 3, p. 391-403, 2007.

VERSTRAETEN, D.; SCHRAUWEN, B.; STROOBANDT, D. **Adapting reservoirs to get Gaussian Distributions**. Proceedings of the European Symposium on Artificial Neural Networks, Bruges, Belgium, p. 495-500, 2007.

VERSTRAETEN, D.; SCHRAUWEN, B.; STROOBANDT. **Reservoir-based techniques for speech recognition**. Proceedings of the International Joint Conference on Neural Networks, Vancouver, BC, Canada, p. 1050-1053, 2006.

WANG, J. S.; CHEN, Y. P. **A Fully Automated Recurrent Neural Network for Unknown Dynamic System Identification and Control**. IEEE Transactions on Circuits and System-I: regular papers, vol. 53, no. 6, p. 1363-1372, 2006.

WERBOS, P. **Backpropagation through time: what it does and how to do it**. Proceedings of IEEE, vol. 78, issue 10, p. 1550–1560, 1990.

WIGREN, T. **Three free data sets for development and benchmarking in nonlinear system identification**. Proceedings of the European Control Conference (ECC), Zürich, Switzerland, p. 2933-2938, 2013.

WILLIAMS, R. J. **Some Observations on the Use of the Extended Kalman Filter as a Recurrent Network Learning Algorithm**. 1992.

WILLIAMS, R. J.; ZIPSER, D. **A learning algorithm for continually running fully recurrent neural networks**. Neural Computation, vol. 1, issue 2, p. 270–280, 1989.

WILLS, A. SCHÖN, T.; LJUNG, L.; NINNESS, B. **Identification of Wiener-Hammerstein models**. Automatica, vol. 49, issue 1, p. 70-81, 2013.

WOLF, A.; SWIFT, J. B.; SWINNEY, H. L.; VASTANO, J. A. **Determining Lyapunov exponents from a Time Series.** Phys. Nonlinear Phenom. , v. 16, p. 285–317, 1985.

WU, J.; WANG, Y.; HUANG, J.; ZHOU, H. **Nonlinear Internal Model Control Using Echo State Network for Pneumatic Muscle System.** Journal of Computers, vol. 7, no. 12, p. 3060-3067, 2012.

XI, J.; SHI, Z.; HAN, M. **Analyzing the State Space Property of Echo State Networks for Chaotic System Prediction.** Proceedings of the IEEE International Joint Conference on Neural Networks, Montreal, Canada, p. 1412-1417, 2005.

XING, K.; WANG, Y.; ZHU, Q.; ZHOU, H. **Modeling and control of McKibben artificial muscle enhanced with echo state networks.** Control Engineering Practice, vol. 20, issue 5, p. 477- 488, 2012.

XU, D.; LAN, J.; PRINCIPE, J. C. **Direct Adaptive Control: An Echo State Network and Genetic Algorithm Approach.** Proceedings of International Joint Conference on Neural Networks, Montreal, Canada, p. 1483-1486, 2005.

YANG, Y.; HARLEY, R.; DIVAN, D.; HABETLER, T. **Adaptive Echo State Network to maximize Overhead Power Line Dynamic Thermal Rating.** IEEE Energy Conversion Congress and Exposition, p. 2247-2254, 2009.

YUENYONG, S. **Fast and Effective Tuning of Echo State Network Reservoir Parameters Using Evolutionary Algorithms And Template Matrices.** Proceedings of the International Computer Science and Engineering Conference, Chiang Mai, Thailand, 2015.

YUSOFF, M. H.; CHROL-CANNON, J.; JIN, Y. **Modeling neural plasticity in echo state networks for classification and regression.** Information Sciences, vol. 364-365, p. 184-196, 2016.

ZHOU, H.; WANG, Y.; XING, K. **Modeling of McKibben Pneumatic Artificial Muscles Using Optimized Echo State Networks.** Proceedings of the 8 th World Congress on Intelligent Control and Automation, p. 1723-1728, 2010.

Apêndice A – Valores de RMSE, SRG e de parâmetros globais das soluções de cada submodelo por experimento

Nas tabelas seguintes considerou-se:

Submodelo 1 – AEIQ-R/Random/ESN

Submodelo 2 – AEIQ-R/SRG/ESN

Submodelo 3 – AEIQ-R/AEIQ-R/ESN

Submodelo 4 – CMA-ES/Random/ESN

Submodelo 5 – CMA-ES/SRG/ESN

Submodelo 6 - CMA-ES/CMA-ES/ESN

a) Estudo de Caso 1

- Melhores parâmetros globais

Tabela 59 – Melhores parâmetros globais encontrados pelo AEIQ-R em cada experimento no estudo de caso 1

AEIQ-R						
Experimento	N	ρ_w	s_{in}	s_{back}	α	c_w
1	217	0,7064	0,2531	0,2378	0,9604	0,1659
2	193	0,7140	0,2711	0,1898	0,8925	0,7400
3	172	0,6532	0,1811	0,1087	0,9201	0,6510
4	197	0,6855	0,1243	0,0468	0,8324	0,2316
5	115	0,4530	0,1801	0,1010	0,9421	0,2984
6	122	0,5099	0,2664	0,2083	0,9954	0,6739
7	227	0,4418	0,3426	0,0635	0,7791	0,3728
8	243	0,4782	0,4348	0,0966	0,7935	0,3706
9	197	0,6448	0,1121	0,0457	0,8792	0,5367
10	172	0,7128	0,2226	0,1718	0,9100	0,6095

Tabela 60 – Melhores parâmetros globais encontrados pelo CMA-ES em cada experimento no estudo de caso 1

CMA-ES						
Experimento	N	ρ_w	s_{in}	s_{back}	α	c_w
1	234	0,6496	0,0390	0,0130	0,9064	0,6750
2	210	0,5655	0,0463	0,0115	0,9052	0,6613
3	157	0,7874	0,4939	0,7943	0,9940	0,8251
4	157	0,5693	0,0847	0,0537	0,9891	0,9298
5	125	0,4453	0,0826	0,0431	0,9928	0,3490
6	199	0,6508	0,1165	0,0568	0,9359	0,9739
7	119	0,3414	0,0622	0,0178	0,9634	0,5923
8	110	0,3901	0,1012	0,0428	0,9366	0,5849
9	139	0,5094	0,1072	0,0601	0,9918	0,1782
10	123	0,4297	0,1004	0,0445	0,9623	0,3030

- Resultados de RMSE

Tabela 61 – Resultados de RMSE da solução final de cada submodelo por experimento no estudo de caso 1 (parte 1)

Exp	Submodelo								
	AEIQ-R/Random/ESN			AEIQ-R/SRG/ESN			AEIQ-R/AEIQ-R/ESN		
	Trein.	Val.	Teste	Trein.	Val.	Teste	Trein.	Val.	Teste
1	0,020492	0,154902	0,194123	0,022108	0,188256	0,254717	0,021786	0,074969	0,139479
2	0,025450	0,147293	0,200024	0,022898	0,227005	0,264654	0,022933	0,078916	0,137574
3	0,028605	0,152580	0,195619	0,030100	0,175671	0,237590	0,026987	0,077616	0,129467
4	0,025067	0,145725	0,185994	0,024202	0,202595	0,249158	0,022480	0,078773	0,132009
5	0,036515	0,156663	0,188609	0,037739	0,178170	0,212438	0,033260	0,083068	0,133345
6	0,033881	0,160481	0,207925	0,034838	0,194507	0,232014	0,033481	0,082746	0,132722
7	0,017940	0,142162	0,209298	0,018450	0,159372	0,215291	0,020131	0,074317	0,127238
8	0,015138	0,153350	0,240606	0,017205	0,207210	0,253755	0,019024	0,071219	0,134064
9	0,023374	0,136642	0,175676	0,022051	0,154570	0,202707	0,022201	0,079306	0,133689
10	0,030139	0,159159	0,200894	0,025521	0,190867	0,239281	0,025895	0,077956	0,134921

Tabela 62 – Resultados de RMSE da solução final de cada submodelo por experimento no estudo de caso 1 (parte 2)

Exp	Submodelo								
	CMA-ES/Random/ESN			CMA-ES/SRG/ESN			CMA-ES/CMA-ES/ESN		
	Trein.	Val.	Teste	Trein.	Val.	Teste	Trein.	Val.	Teste
1	0,022422	0,154863	0,193465	0,019415	0,214956	0,256434	0,016723	0,103944	0,157815
2	0,021806	0,141787	0,180794	0,020902	0,152885	0,191796	0,018799	0,110298	0,157982
3	0,030693	0,196262	0,254880	0,034033	0,197265	0,248053	0,023552	0,086157	0,131503
4	0,030106	0,157193	0,189121	0,031363	0,196656	0,243923	0,026440	0,097133	0,142190
5	0,034679	0,159130	0,192669	0,034048	0,168752	0,216645	0,027429	0,098822	0,146080
6	0,026273	0,154355	0,200617	0,025495	0,200296	0,259417	0,022824	0,107689	0,155639
7	0,032186	0,141702	0,179600	0,037197	0,159698	0,204695	0,030009	0,101406	0,139906
8	0,038292	0,156534	0,194390	0,034015	0,174517	0,213347	0,030785	0,100906	0,138797
9	0,033907	0,148304	0,186310	0,035085	0,193082	0,250019	0,027029	0,095365	0,138428
10	0,035444	0,158920	0,196246	0,035444	0,158920	0,196246	0,028382	0,098943	0,138923

- Resultados de SRG

Tabela 63 – Resultados de SRG dos submodelos em cada experimento no estudo de caso 1

Exp	Submodelo					
	1	2	3	4	5	6
1	2,198535	1,948867	2,176024	1,859995	1,855817	1,868797
2	1,508113	1,358321	1,427917	1,843111	1,807939	1,847439
3	0,412582	0,326194	0,391246	7,795206	7,362775	7,841197
4	1,023047	0,941102	0,919523	1,236006	1,182218	1,144100
5	0,510686	0,408286	0,473231	1,418964	1,373769	1,438770
6	1,090413	0,954655	1,087932	0,767594	0,747243	0,775160
7	1,454127	1,360663	1,626199	1,776073	1,729163	1,786264
8	2,722879	2,317907	2,229160	1,402934	1,341600	1,384129
9	0,992967	0,969463	1,009516	1,074158	0,948967	1,041014
10	0,826375	0,667797	0,810443	1,221327	1,221327	1,306341

b) Estudo de Caso 2

- Melhores parâmetros globais

Tabela 64 – Melhores parâmetros globais encontrados pelo AEIQ-R em cada experimento no estudo de caso 2

AEIQ-R						
Experimento	N	ρ_W	s_{in}	s_{back}	α	c_W
1	24	0,7446	0,4083	0,3067	0,6884	0,4289
2	24	0,5320	0,3851	0,3290	0,6377	0,6612
3	25	0,3922	0,0511	0,1341	0,6170	0,8077
4	28	0,7023	0,4812	0,4241	0,6388	0,5009
5	23	0,6345	0,3821	0,3193	0,6895	0,6293
6	25	0,6011	0,4937	0,4001	0,5718	0,3886
7	24	0,6517	0,3328	0,3673	0,6033	0,3564
8	23	0,8143	0,3696	0,2220	0,6664	0,7402
9	23	0,6701	0,0575	0,0451	0,7745	0,4460
10	24	0,6379	0,3526	0,2564	0,6453	0,5002

Tabela 65 – Melhores parâmetros globais encontrados pelo CMA-ES em cada experimento no estudo de caso 2

CMA-ES						
Experimento	N	ρ_W	s_{in}	s_{back}	α	c_W
1	21	0,1264	0,0306	0,0353	0,6133	0,3794
2	22	0,6330	0,0257	0,0167	0,7720	0,5657
3	22	0,1051	0,0447	0,0721	0,9123	0,6715
4	21	0,1248	0,0295	0,0352	0,6187	0,2332
5	20	0,1000	0,0147	0,0390	0,6060	0,2515
6	21	0,1694	0,0545	0,0677	0,6057	0,3106
7	22	0,1776	0,0539	0,0844	0,5812	0,4321
8	23	0,1175	0,0501	0,1046	0,9497	0,4949
9	22	0,1709	0,0514	0,0500	0,6021	0,4839
10	23	0,1329	0,0583	0,1151	0,8823	0,7812

- Resultados de RMSE

Tabela 66 – Resultados de RMSE da solução final de cada submodelo por experimento no estudo de caso 2 (parte 1)

Exp	Submodelo								
	AEIQ-R/Random/ESN			AEIQ-R/SRG/ESN			AEIQ-R/AEIQ-R/ESN		
	Trein.	Val.	Teste	Trein.	Val.	Teste	Trein.	Val.	Teste
1	0,055065	0,099682	0,143988	0,051738	0,1105970	0,140927	0,086069	0,064449	0,139122
2	0,051140	0,103804	0,137580	0,046420	0,130577	0,172711	0,063081	0,081441	0,112544
3	0,052179	0,106229	0,141132	0,047269	0,109909	0,139748	0,070112	0,083844	0,121639
4	0,052065	0,110329	0,133219	0,054669	0,153182	0,117701	0,636023	0,071907	0,180968
5	0,050732	0,100857	0,137331	0,060619	0,144063	0,180636	0,083442	0,077607	0,126262
6	0,052943	0,109912	0,140559	0,048930	0,146074	0,180297	0,067234	0,077752	0,124074
7	0,045300	0,104514	0,131682	0,047700	0,111595	0,141364	0,091600	0,074641	0,118499
8	0,047823	0,107152	0,128737	0,056069	0,113712	0,165889	0,119833	0,055283	0,140705
9	0,051599	0,100975	0,133175	0,065154	0,121338	0,158230	0,067945	0,085269	0,119738
10	0,051206	0,103268	0,132206	0,051206	0,103268	0,132206	0,073865	0,073531	0,124247

Tabela 67 – Resultados de RMSE da solução final de cada submodelo por experimento no estudo de caso 2 (parte 2)

Exp	Submodelo								
	CMA-ES/Random/ESN			CMA-ES/SRG/ESN			CMA-ES/CMA-ES/ESN		
	Trein.	Val.	Teste	Trein.	Val.	Teste	Trein.	Val.	Teste
1	0,050548	0,102396	0,141009	0,051177	0,110416	0,138830	0,076744	0,078209	0,112436
2	0,056792	0,105670	0,147353	0,047559	0,116197	0,139428	0,084650	0,061935	0,140421
3	0,052384	0,104107	0,147647	0,053165	0,114404	0,162343	0,076419	0,082163	0,142720
4	0,052467	0,101921	0,140581	0,052866	0,109149	0,153479	0,079039	0,078571	0,105634
5	0,056484	0,101213	0,136650	0,056484	0,101213	0,136650	0,085468	0,077773	0,127025
6	0,047743	0,100231	0,132657	0,047082	0,113050	0,155503	0,084416	0,072696	0,110576
7	0,055076	0,101015	0,130475	0,055172	0,105847	0,144168	0,100824	0,071979	0,112674
8	0,053112	0,103654	0,149604	0,046265	0,109563	0,152962	0,095122	0,085166	0,139380
9	0,050560	0,101823	0,133432	0,046918	0,107587	0,146368	0,070166	0,079733	0,109330
10	0,054432	0,104358	0,151993	0,052988	0,121688	0,164323	0,075406	0,076724	0,133130

- Resultados de SRG

Tabela 68 – Resultados de SRG dos submodelos em cada experimento no estudo de caso 2

Exp	Submodelo					
	1	2	3	4	5	6
1	0,287623	0,194485	0,244180	0,578530	0,574710	0,576077
2	0,305356	0,226400	0,295713	0,593144	0,590465	0,591303
3	0,472129	0,452742	0,457970	0,511096	0,492963	0,499129
4	0,092624	0,091377	0,193472	0,578264	0,573454	0,575560
5	0,250645	0,204747	0,293984	0,572359	0,572359	0,573021
6	0,242489	0,170655	0,171020	0,554299	0,534017	0,530568
7	0,248004	0,207787	0,240474	0,535452	0,525113	0,532069
8	0,337478	0,337399	0,373230	0,440029	0,418874	0,450369
9	0,553828	0,550185	0,560405	0,567735	0,559513	0,565837
10	0,291376	0,291376	0,345606	0,439633	0,422127	0,427444

c) Estudo de Caso 3

- Melhores parâmetros globais

Tabela 69 – Melhores parâmetros globais encontrados pelo AEIQ-R em cada experimento no estudo de caso 3

AEIQ-R						
Experimento	N	ρ_W	S_{in}	S_{back}	α	c_W
1	26	0,8325	0,4441	0,1494	0,4780	0,5742
2	25	0,6361	0,6557	0,3169	0,5725	0,7617
3	23	0,6600	0,5661	0,0924	0,4247	0,5651
4	30	0,6399	0,5530	0,2331	0,5420	0,6176
5	27	0,7376	0,5776	0,1646	0,5628	0,5246
6	27	0,5080	0,6996	0,2677	0,5066	0,6536
7	31	0,4770	0,6312	0,2399	0,5527	0,4374
8	27	0,6473	0,5717	0,2661	0,5572	0,6552
9	32	0,7398	0,7986	0,3762	0,5291	0,5701
10	21	0,5971	0,8106	0,1316	0,2727	0,4114

Tabela 70 – Melhores parâmetros globais encontrados pelo CMA-ES em cada experimento no estudo de caso 3

CMA-ES						
Experimento	N	ρ_W	S_{in}	S_{back}	α	c_W
1	25	0,7042	0,7003	0,1924	0,4666	0,2564
2	28	0,6284	0,6247	0,1810	0,5860	0,3801
3	26	0,7036	0,8098	0,1711	0,4498	0,5417
4	25	0,5740	0,7384	0,2011	0,5540	0,6960
5	27	0,6424	0,5440	0,1263	0,5069	0,2580
6	23	0,6799	0,9108	0,2743	0,2931	0,7312
7	20	0,5688	0,8208	0,0694	0,3304	0,7007
8	21	0,6776	0,5920	0,0320	0,3967	0,3069
9	21	0,6040	0,9218	0,1517	0,2790	0,7328
10	22	0,6336	0,7506	0,0937	0,3993	0,5084

- Resultados de RMSE

Tabela 71 – Resultados de RMSE da solução final de cada submodelo por experimento no estudo de caso 3 (parte 1)

Exp	Submodelo								
	AEIQ-R/Random/ESN			AEIQ-R/SRG/ESN			AEIQ-R/AEIQ-R/ESN		
	Trein.	Val.	Teste	Trein.	Val.	Teste	Trein.	Val.	Teste
1	0,112048	0,160600	0,227598	0,123787	0,166696	0,279301	0,174364	0,136071	0,176185
2	0,121619	0,148092	0,272107	0,132497	0,174201	0,686093	0,147936	0,134553	0,440225
3	0,117207	0,150037	0,185051	0,120719	0,160245	0,336617	0,151415	0,139325	0,163458
4	0,120707	0,151465	0,207232	0,124429	0,194168	0,337098	0,152469	0,136052	0,231884
5	0,119021	0,151015	0,188464	0,126366	0,242015	0,316761	0,150189	0,132155	0,258989
6	0,122468	0,155425	0,258110	0,120120	0,190552	0,219314	0,147529	0,135254	0,220459
7	0,116741	0,158196	0,237305	0,115674	0,179000	0,298666	0,157275	0,136491	0,181047
8	0,115179	0,143416	0,199400	0,125281	0,166411	0,220437	0,160922	0,135403	0,185732
9	0,119910	0,164924	0,224038	0,119910	0,164924	0,224038	0,151209	0,133127	0,187080
10	0,116775	0,145942	0,209721	0,118044	0,162472	0,226922	0,153797	0,139621	0,221075

Tabela 72 – Resultados de RMSE da solução final de cada submodelo por experimento no estudo de caso 3 (parte 2)

Exp	Submodelo								
	CMA-ES/Random/ESN			CMA-ES/SRG/ESN			CMA-ES/CMA-ES/ESN		
	Trein.	Val.	Teste	Trein.	Val.	Teste	Trein.	Val.	Teste
1	0,118162	0,157014	0,248120	0,118162	0,157014	0,248120	0,119104	0,121431	0,439663
2	0,123670	0,151305	0,247178	0,119836	0,170130	0,228504	0,127211	0,119610	0,219536
3	0,110678	0,138866	0,222707	0,125514	0,174699	0,418825	0,128217	0,121276	0,229606
4	0,122454	0,150310	0,202807	0,120913	0,176101	0,250173	0,126311	0,122195	0,190229
5	0,118997	0,150999	0,230749	0,118266	0,177576	0,451434	0,120343	0,120414	0,320788
6	0,112777	0,143529	0,195593	0,125952	0,208258	0,332960	0,115011	0,127251	0,172738
7	0,116770	0,152416	0,203384	0,114976	0,157377	0,254503	0,122151	0,124532	0,172672
8	0,116506	0,149624	0,214376	0,116128	0,167019	0,210304	0,117759	0,124326	0,181182
9	0,117556	0,147449	0,185882	0,118934	0,202660	0,338360	0,120368	0,125658	0,285584
10	0,121797	0,151452	0,188583	0,120621	0,201995	0,358923	0,120563	0,124463	0,189494

- Resultados de SRG

Tabela 73 – Resultados de SRG dos submodelos em cada experimento no estudo de caso 3

Exp	Submodelo					
	1	2	3	4	5	6
1	3,089883	2,740753	3,339267	1,738040	1,738040	2,226135
2	1,392546	1,027787	1,021055	1,375209	1,115328	1,476997
3	3,062714	2,786411	2,933389	1,520857	1,451242	1,522825
4	1,564471	1,340410	1,934911	1,216033	1,122071	1,252506
5	1,660196	1,484993	1,868680	2,123443	1,988478	2,354420
6	1,210917	1,103719	1,339875	2,779677	2,577982	2,999383
7	1,043399	0,988785	1,198851	2,783329	2,686070	2,890057
8	1,838637	1,128405	1,545326	3,199109	2,999712	3,102172
9	0,895218	0,895218	0,977905	3,518049	2,894592	2,792981
10	3,340098	3,278211	3,437252	2,264754	2,148407	2,775920

d) Estudo de Caso 4

- Melhores parâmetros globais

Tabela 74 – Melhores parâmetros globais encontrados pelo AEIQ-R em cada experimento no estudo de caso 4

AEIQ-R						
Experimento	N	ρ_W	s_{in}	s_{back}	α	c_W
1	95	0,4644	0,6336	0,0376	0,4487	0,6229
2	53	0,7526	0,4215	0,1495	0,6440	0,3663
3	98	0,9327	0,4450	0,1941	0,8234	0,5738
4	85	0,5965	0,5133	0,0490	0,5401	0,3866
5	87	0,4991	0,5070	0,0638	0,5280	0,6244
6	95	0,5483	0,5379	0,0604	0,4601	0,4744
7	98	0,3898	0,4199	0,0167	0,4614	0,3199
8	76	0,7906	0,4305	0,1581	0,7129	0,5991
9	97	0,8077	0,4014	0,1267	0,8680	0,3134
10	54	0,7447	0,4685	0,0785	0,5872	0,9312

Tabela 75 – Melhores parâmetros globais encontrados pelo CMA-ES em cada experimento no estudo de caso 4

CMA-ES						
Experimento	N	ρ_W	s_{in}	s_{back}	α	c_W
1	82	0,7885	0,4524	0,0659	0,8802	0,6784
2	64	0,7882	0,4319	0,0804	0,7386	0,4801
3	62	0,7139	0,4884	0,0690	0,7011	0,5835
4	104	0,2797	0,3590	0,0061	0,4334	0,7866
5	55	0,8413	0,3559	0,0751	0,7227	0,5075
6	78	0,7916	0,4494	0,0752	0,8174	0,5463
7	63	0,7384	0,4748	0,0620	0,6663	0,5430
8	56	0,8098	0,5013	0,1363	0,6886	0,6141
9	71	0,8340	0,4382	0,0833	0,8104	0,7444
10	66	0,8047	0,5187	0,0913	0,7173	0,5159

- Resultados de RMSE

Tabela 76 – Resultados de RMSE da solução final de cada submodelo por experimento no estudo de caso 4 (parte 1)

Exp	Submodelo								
	AEIQ-R/Random/ESN			AEIQ-R/SRG/ESN			AEIQ-R/AEIQ-R/ESN		
	Trein.	Val.	Teste	Trein.	Val.	Teste	Trein.	Val.	Teste
1	0,117242	0,041624	0,056295	0,218249	0,090116	0,053504	0,069154	0,007673	0,021986
2	0,100127	0,036642	0,039035	0,117996	0,099051	0,095318	0,088945	0,006063	0,024836
3	0,081819	0,052601	0,086773	0,097654	0,061069	0,064211	0,032570	0,005600	0,025575
4	0,072412	0,035790	0,037082	0,150130	0,061858	0,032606	0,042068	0,005584	0,026611
5	0,074677	0,033277	0,042769	0,154321	0,072609	0,092470	0,055838	0,006034	0,032857
6	0,141403	0,046365	0,043611	732,581838	782,887312	782,947168	0,051918	0,004712	0,027478
7	0,153355	0,040974	0,039473	0,153355	0,040974	0,039473	0,103779	0,004151	0,024242
8	0,115873	0,039300	0,032337	0,140361	0,119859	0,134931	0,133521	0,004610	0,016050
9	0,121710	0,038402	0,052219	0,622553	0,599861	0,890810	0,040104	0,004244	0,019294
10	0,205053	0,065338	0,032082	0,173016	0,075044	0,112840	0,059058	0,005068	0,024986

Tabela 77 – Resultados de RMSE da solução final de cada submodelo por experimento no estudo de caso 4 (parte 2)

Exp	Submodelo								
	CMA-ES/Random/ESN			CMA-ES/SRG/ESN			CMA-ES/CMA-ES/ESN		
	Trein.	Val.	Teste	Trein.	Val.	Teste	Trein.	Val.	Teste
1	0,137688	0,060775	0,099632	0,194041	0,086950	0,058951	0,043230	0,010984	0,021764
2	0,133187	0,044338	0,043755	0,652483	1,377097	128,847618	0,045225	0,010716	0,027500
3	0,131476	0,051955	0,059166	0,135341	0,073074	0,093227	0,048155	0,009014	0,023929
4	0,138009	0,038917	0,038975	0,114347	0,064546	0,115677	0,066921	0,010299	0,028197
5	0,131737	0,050259	0,066293	0,074215	0,083779	0,075218	0,042352	0,005266	0,013779
6	0,078794	0,057200	0,119122	0,790607	0,993192	5,214498	0,046894	0,009696	0,020077
7	0,113983	0,059813	0,053357	0,113983	0,059813	0,053357	0,030305	0,006316	0,016510
8	0,098979	0,069543	0,065368	0,144564	0,089746	0,092263	0,054949	0,007516	0,015677
9	0,078706	0,048463	0,089365	0,128338	0,098641	0,061594	0,020946	0,010792	0,025375
10	0,090196	0,053553	0,081873	0,152423	0,096835	0,072635	0,071053	0,009704	0,029793

- Resultados de SRG

Tabela 78 – Resultados de SRG dos submodelos em cada experimento no estudo de caso 4

Exp	Submodelo					
	1	2	3	4	5	6
1	3,429968	3,271855	3,583378	4,132125	3,977327	4,137063
2	2,381097	2,257894	2,371999	3,003214	2,805881	2,942597
3	3,422538	3,259826	3,434740	3,159818	3,057752	3,187063
4	3,216769	3,120924	3,040185	3,696112	3,483499	3,980994
5	3,145549	3,044906	3,048014	2,830366	2,783802	2,894717
6	3,023146	2,997465	3,132342	4,015377	3,291316	3,848429
7	3,138954	3,138954	3,461390	2,951807	2,951807	2,948279
8	2,375361	2,129843	2,085816	2,677229	2,458326	2,514722
9	4,067537	3,331420	3,800672	3,477025	2,876108	3,194577
10	2,985468	2,855025	2,854546	3,386135	3,328233	3,382802

e) Estudo de Caso 5

- Melhores parâmetros globais

Tabela 79 – Melhores parâmetros globais encontrados pelo AEIQ-R em cada experimento no estudo de caso 5

AEIQ-R						
Experimento	N	ρ_W	s_{in}	s_{back}	α	c_W
1	100	0,7747	0,9593	0,4756	0,8818	0,7742
2	61	0,7642	0,8341	0,8688	0,7109	0,1886
3	116	0,8494	0,8992	0,5911	0,9320	0,1861
4	135	0,9514	0,6645	0,4412	0,9538	0,7132
5	88	0,9135	0,8954	0,1530	0,4674	0,3422
6	112	0,6908	0,7327	0,6869	0,7954	0,4442
7	145	0,8059	0,5894	0,7076	0,9356	0,7926
8	190	0,9494	0,6769	0,8092	0,9830	0,3453
9	157	0,8012	0,8595	0,7543	0,6969	0,7519
10	186	0,8537	0,7098	0,6682	0,9001	0,6644

Tabela 80 – Melhores parâmetros globais encontrados pelo CMA-ES em cada experimento no estudo de caso 5

CMA-ES						
Experimento	N	ρ_W	s_{in}	s_{back}	α	c_W
1	94	0,8559	0,9529	0,0741	0,9827	0,4103
2	58	0,9012	0,9373	0,7584	0,9895	0,6251
3	127	0,9771	0,8274	0,5081	0,9559	0,3786
4	84	0,9787	0,8226	0,2026	0,9501	0,7053
5	125	0,9150	0,7020	0,5825	0,9792	0,8913
6	113	0,9900	0,8984	0,6068	0,9816	0,6653
7	141	0,9546	0,8179	0,0702	0,9568	0,6003
8	83	0,9608	0,7497	0,4237	0,9821	0,3156
9	66	0,9447	0,9982	0,6047	0,9330	0,4777
10	124	0,9900	0,8850	0,3545	0,9494	0,5844

- Resultados de RMSE

Tabela 81 – Resultados de RMSE da solução final de cada submodelo por experimento no estudo de caso 5 (parte 1)

Exp	Submodelo								
	AEIQ-R/Random/ESN			AEIQ-R/SRG/ESN			AEIQ-R/AEIQ-R/ESN		
	Trein.	Val.	Teste	Trein.	Val.	Teste	Trein.	Val.	Teste
1	0.009520	0.009268	0.016727	0.018511	0.012115	0.102382	17.590414	0.006032	0.013708
2	3.134058	0.012602	0.017197	50.650387	50.231037	50.393024	8.465890	0.006638	0.015047
3	0.008975	0.009986	0.019188	0.008452	0.011485	0.025981	11.335946	0.005944	0.015724
4	18.777651	0.010802	0.019430	18.065150	18.210762	18.022109	0.007338	0.006543	0.015029
5	156.999982	0.014242	0.032944	0.012419	0.015118	0.023013	0.008337	0.006813	0.016838
6	9.169607	0.009684	0.018163	9.169607	0.009684	0.018163	11.272064	0.005883	0.014279
7	0.007102	0.009138	0.018687	8.922917	0.009552	0.018392	0.531152	0.006080	0.013606
8	0.006685	0.009506	0.021894	0.006458	0.014421	0.027255	0.006362	0.005807	0.018362
9	0.006022	0.010215	0.022924	15.820886	15.757661	15.551645	1.780102	0.006060	0.018979
10	0.005994	0.009404	0.020504	0.005358	0.009769	0.025448	0.952328	0.005909	0.013060

Tabela 82 – Resultados de RMSE da solução final de cada submodelo por experimento no estudo de caso 5 (parte 2)

Exp	Submodelo								
	CMA-ES/Random/ESN			CMA-ES/SRG/ESN			CMA-ES/CMA-ES/ESN		
	Trein.	Val.	Teste	Trein.	Val.	Teste	Trein.	Val.	Teste
1	0.026805	0.017950	0.096019	0.064760	0.023753	0.044683	0.009953	0.008715	0.014900
2	0.013097	0.012550	0.017583	41.215815	42.503633	36.639480	0.010768	0.009248	0.017348
3	0.010144	0.010897	0.025751	0.012098	0.011879	0.029661	0.008767	0.009959	0.019220
4	0.030855	0.024270	0.045812	0.036670	0.029559	0.056516	0.008378	0.008663	0.015647
5	0.007649	0.011570	0.017773	27.391386	27.279702	27.292119	0.011145	0.009976	0.018535
6	0.009081	0.010607	0.016393	0.013737	0.014085	0.027907	0.011123	0.010497	0.017855
7	0.019546	0.015014	0.059466	3.854083	0.024762	3.738935	22.018653	0.008963	0.026651
8	0.013571	0.012768	0.019458	23.761778	24.203425	24.014938	0.009571	0.009096	0.015559
9	0.013407	0.013172	0.016695	0.026661	0.020804	0.029882	0.007806	0.006832	0.015830
10	0.015303	0.014237	0.029793	0.015246	0.014919	0.027354	0.007092	0.007461	0.019818

- Resultados de SRG

Tabela 83 – Resultados de SRG dos submodelos em cada experimento no estudo de caso 5

Exp	Submodelo					
	1	2	3	4	5	6
1	24.012255	21.030606	22.421510	20.185442	17.208731	19.688774
2	14.962679	14.239809	17.223413	23.632158	21.556909	27.448706
3	33.138734	30.147198	27.799616	30.072400	26.612723	30.892909
4	25.294072	24.833949	26.605856	11.325395	10.834704	10.928634
5	7.757851	7.488371	7.522546	36.357581	30.698605	35.125214
6	25.903187	25.903187	27.691217	29.944685	29.774088	32.327911
7	38.271912	37.085011	40.445763	19.172450	17.899582	20.385284
8	56.279877	52.154377	52.404578	17.002904	16.245347	17.180167
9	34.900569	29.357321	33.151210	19.404899	18.487383	20.396960
10	42.826428	40.507720	43.429100	21.528416	20.592471	19.924643

f) Estudo de Caso 6

- Melhores parâmetros globais

Tabela 84 – Melhores parâmetros globais encontrados pelo AEIQ-R em cada experimento no estudo de caso 6

AEIQ-R						
Experimento	N	ρ_W	s_{in}	s_{back}	α	c_W
1	84	0,8061	0,8218	0,1955	0,8856	0,3141
2	74	0,8967	0,6593	0,1436	0,9274	0,6681
3	69	0,8047	0,6593	0,2576	0,8471	0,3630
4	61	0,7751	0,7559	0,3236	0,8092	0,5206
5	70	0,6217	0,8516	0,1749	0,8917	0,7413
6	69	0,6736	0,9178	0,2701	0,8927	0,6738
7	77	0,7483	0,7685	0,2152	0,9063	0,5971
8	66	0,7856	0,7752	0,3633	0,9067	0,4427
9	75	0,7100	0,8162	0,2346	0,8868	0,5703
10	77	0,7188	0,8436	0,1760	0,8835	0,6689

Tabela 85 – Melhores parâmetros globais encontrados pelo CMA-ES em cada experimento no estudo de caso 6

CMA-ES						
Experimento	N	ρ_W	s_{in}	s_{back}	α	c_W
1	74	0,7202	0,9478	0,2270	0,9512	0,6015
2	79	0,6385	0,9633	0,1812	0,9458	0,3405
3	84	0,6320	0,9028	0,1685	0,9842	0,3288
4	80	0,6195	0,9875	0,1700	0,9641	0,6317
5	80	0,7255	0,9671	0,2330	0,9464	0,6663
6	68	0,7222	0,9216	0,2696	0,9606	0,2445
7	81	0,7692	0,9570	0,2167	0,9300	0,3767
8	80	0,7099	0,9664	0,2195	0,9838	0,3805
9	76	0,7061	0,9624	0,2069	0,9677	0,4912
10	84	0,6258	0,9802	0,1650	0,9773	0,8233

- Resultados de RMSE

Tabela 86 – Resultados de RMSE da solução final de cada submodelo por experimento no estudo de caso 6 (parte 1)

Exp	Submodelo								
	AEIQ-R/Random/ESN			AEIQ-R/SRG/ESN			AEIQ-R/AEIQ-R/ESN		
	Trein.	Val.	Teste	Trein.	Val.	Teste	Trein.	Val.	Teste
1	0,073333	0,120280	0,231581	0,124347	0,633336	8,885537	0,093082	0,058562	0,254985
2	0,120801	0,139820	0,233225	0,129651	0,220895	0,254181	0,108431	0,056332	0,208492
3	0,099147	0,137123	0,359254	0,081654	0,142255	0,191194	0,095419	0,073673	0,242985
4	0,085505	0,134215	0,216823	0,097322	0,136679	0,235222	0,101698	0,059205	0,274263
5	0,084701	0,141785	0,318650	0,084701	0,141785	0,318650	0,089121	0,055422	0,275359
6	0,077249	0,136581	73,096574	0,083827	0,161989	0,178916	0,085040	0,062617	0,345844
7	0,092452	0,130771	0,325987	0,090495	0,203398	0,260584	0,094752	0,056699	0,221960
8	0,100094	0,130576	0,271351	0,100094	0,130576	0,271351	0,086778	0,064738	0,176050
9	0,078535	0,130845	0,387828	0,078535	0,130845	0,387828	0,093905	0,050152	0,308943
10	0,082783	0,132177	0,212962	0,074038	0,177866	0,242913	0,097549	0,055972	0,256968

Tabela 87 – Resultados de RMSE da solução final de cada submodelo por experimento no estudo de caso 6 (parte 2)

Exp	Submodelo								
	CMA-ES/Random/ESN			CMA-ES/SRG/ESN			CMA-ES/CMA-ES/ESN		
	Trein.	Val.	Teste	Trein.	Val.	Teste	Trein.	Val.	Teste
1	0,094684	0,126231	0,332055	0,094684	0,126231	0,332055	0,091376	0,078241	0,213702
2	0,082393	0,134276	0,237154	0,110860	0,240649	0,250642	0,086085	0,077597	0,261731
3	0,095217	0,140200	0,402754	0,076157	0,150367	0,869377	0,088754	0,077513	0,183289
4	0,086346	0,139250	0,208571	0,096414	0,197570	0,272588	0,096889	0,079892	0,251649
5	0,086810	0,135460	0,319659	0,083701	0,167216	0,244167	0,082285	0,077607	0,214769
6	0,123988	0,150593	0,403794	0,100379	0,198051	0,305865	0,096095	0,059589	0,211893
7	0,089652	0,135866	0,265539	0,070647	0,158991	0,210808	0,076596	0,078905	0,202189
8	0,086175	0,142144	0,214683	0,069810	0,182890	0,295249	0,083107	0,074591	0,167296
9	0,085534	0,143254	0,235196	0,085534	0,143254	0,235196	0,095607	0,077564	0,225611
10	0,073521	0,136794	0,250499	0,084842	0,167157	0,240073	0,087938	0,083270	0,210090

- Resultados de SRG

Tabela 88 – Resultados de SRG dos submodelos em cada experimento no estudo de caso 6

Exp	Submodelo					
	1	2	3	4	5	6
1	1,494988	0,966940	0,891707	1,094138	1,094138	1,444896
2	2,998575	2,119943	2,665895	1,734378	1,474467	2,016188
3	0,799263	0,448276	0,639418	2,272649	1,407154	1,699657
4	1,627072	0,672690	1,260474	2,618820	1,841742	2,226580
5	2,040358	2,040358	2,201541	1,377077	1,243865	1,263172
6	1,406815	1,153290	0,955588	1,514931	1,423989	1,682908
7	0,991813	0,811574	0,732634	1,417061	1,007046	1,331889
8	2,301386	2,301386	3,139188	1,551201	1,177279	1,487636
9	0,780819	0,780819	1,100474	0,985478	0,985478	1,170160
10	2,087297	1,864966	2,674661	1,973100	1,499316	2,030238

g) Estudo de Caso 7

- Melhores parâmetros globais

Tabela 89 – Melhores parâmetros globais encontrados pelo AEIQ-R em cada experimento no estudo de caso 7

AEIQ-R						
Experimento	N	ρ_W	s_{in}	s_{back}	α	c_W
1	280	0,3315	0,2575	0,1705	0,6282	0,5547
2	260	0,3719	0,2953	0,1642	0,6004	0,7049
3	313	0,2621	0,3165	0,1340	0,6569	0,4031
4	261	0,3117	0,3334	0,1625	0,6763	0,4943
5	264	0,3595	0,2985	0,1755	0,6504	0,3470
6	318	0,3239	0,2621	0,1395	0,7330	0,3896
7	286	0,2816	0,3035	0,1083	0,6322	0,3479
8	323	0,3506	0,2935	0,1957	0,7424	0,6473
9	314	0,2882	0,2571	0,0988	0,6646	0,4091
10	257	0,3988	0,3504	0,1808	0,6507	0,4917

Tabela 90 – Melhores parâmetros globais encontrados pelo CMA-ES em cada experimento no estudo de caso 7

CMA-ES						
Experimento	N	ρ_W	s_{in}	s_{back}	α	c_W
1	296	0,3116	0,2795	0,1231	0,6688	0,5960
2	276	0,3512	0,2809	0,1500	0,6870	0,4206
3	319	0,3513	0,2928	0,1207	0,7100	0,7025
4	353	0,3277	0,3008	0,1263	0,7100	0,5849
5	329	0,3009	0,2512	0,0902	0,7068	0,4974
6	348	0,3879	0,3185	0,1333	0,7766	0,3761
7	293	0,3127	0,2584	0,1170	0,6794	0,8463
8	303	0,3164	0,2904	0,1348	0,7043	0,6796
9	282	0,3251	0,2549	0,1356	0,7332	0,6909
10	295	0,3068	0,2844	0,1334	0,6923	0,5237

- Resultados de RMSE

Tabela 91 – Resultados de RMSE da solução final de cada submodelo por experimento no estudo de caso 7 (parte 1)

Exp	Submodelo								
	AEIQ-R/Random/ESN			AEIQ-R/SRG/ESN			AEIQ-R/AEIQ-R/ESN		
	Trein.	Val.	Teste	Trein.	Val.	Teste	Trein.	Val.	Teste
1	1,161208	1,720863	2,752954	1,158965	1,846033	3,186629	1,101852	1,095967	2,701125
2	1,194023	1,663456	2,868020	1,091968	1,880513	2,967105	1,158056	1,209690	2,759793
3	1,036601	1,610779	2,799033	1,114010	1,831720	2,970639	1,114678	1,099845	2,747262
4	1,205684	1,548225	2,772804	1,147799	1,786913	3,236174	1,068958	0,997114	2,753570
5	1,135134	1,490163	2,760942	1,078608	1,923573	2,771291	1,155720	1,040321	2,654622
6	1,117984	1,587707	2,960452	1,167150	2,585248	3,154206	1,164326	1,021197	2,779402
7	1,109097	1,597041	3,138800	1,145604	1,985444	3,268385	1,140333	1,043715	2,682183
8	1,139098	1,434774	2,876004	1,141827	1,547206	1505,356564	1,142388	1,118994	2,723421
9	1,114539	1,580570	2,700473	1,098891	1,981963	3,051776	1,093132	1,130026	2,699410
10	1,153152	1,587759	2,716539	1,180252	1,605595	2,855328	1,180748	1,054358	2,694884

Tabela 92 – Resultados de RMSE da solução final de cada submodelo por experimento no estudo de caso 7 (parte 2)

Exp	Submodelo								
	CMA-ES/Random/ESN			CMA-ES/SRG/ESN			CMA-ES/CMA-ES/ESN		
	Trein.	Val.	Teste	Trein.	Val.	Teste	Trein.	Val.	Teste
1	1,137937	1,577672	2,898795	1,169347	1,677565	2,901589	1,129608	1,104002	2,772102
2	1,148189	1,729460	3,065028	1,148189	1,729460	3,065028	1,083747	1,088376	2,771488
3	1,153120	1,643344	2,861400	1,111495	2,712754	170,161509	1,067032	1,122372	2,654336
4	1,088756	1,513774	2,902844	1,070841	1,644514	2,793645	1,027495	1,074095	2,720620
5	1,106578	1,670473	2,865718	1,180883	1,723743	3,107992	1,119515	1,108861	2,552125
6	1,118735	1,605841	2,919477	1,152297	1,832389	3,260316	1,132675	1,122400	2,895655
7	1,172376	1,587139	2,534678	1,186203	1,614962	2,799396	1,107932	1,145103	2,646257
8	1,167903	1,544636	2,727647	1,188673	1,593274	2,920231	1,062439	1,068705	2,699191
9	1,188762	1,534766	2,868619	1,174086	2,291538	3,312877	1,121968	1,090131	2,650962
10	1,135267	1,557188	3,004230	1,167631	1,689119	2,926369	1,104093	1,154514	2,747343

- Resultados de SRG

Tabela 93 – Resultados de SRG dos submodelos em cada experimento no estudo de caso 7

Exp	Submodelo					
	1	2	3	4	5	6
1	229,252745	209,387511	239,945794	359,645451	329,500052	351,029222
2	240,715401	222,131999	252,806033	340,826477	340,826477	363,348653
3	472,452566	444,862030	452,167804	474,521293	469,058785	504,443450
4	482,504610	447,105228	512,938498	587,719143	563,496952	587,112759
5	371,400927	343,799799	366,509107	359,134457	322,098348	330,971135
6	447,826818	414,227775	423,713790	760,143305	733,200676	778,941603
7	335,378708	325,390763	344,680989	283,810687	278,585163	283,414158
8	610,378514	596,731283	629,955419	463,696135	440,649285	458,232809
9	289,423137	271,487883	276,225186	343,854038	320,080860	341,930703
10	485,635650	448,670279	471,060486	401,354711	383,420127	403,880071

h) Estudo de Caso 8

- Melhores parâmetros globais

Tabela 94 – Melhores parâmetros globais encontrados pelo AEIQ-R em cada experimento no estudo de caso 8

AEIQ-R						
Experimento	N	ρ_W	s_{in}	s_{back}	α	c_W
1	17	0,3069	0,8170	0,3621	0,7041	0,5062
2	17	0,4048	0,8460	0,5995	0,7538	0,6850
3	18	0,2917	0,9092	0,5706	0,7412	0,6667
4	18	0,5168	0,7955	0,4141	0,6371	0,4597
5	16	0,3320	0,8668	0,5305	0,5995	0,6187
6	17	0,2666	0,8302	0,4814	0,6819	0,5717
7	21	0,6844	0,8243	0,4566	0,8684	0,5432
8	17	0,2515	0,7866	0,6150	0,7042	0,5884
9	17	0,1925	0,9354	0,6253	0,6451	0,6479
10	18	0,3002	0,8257	0,4081	0,6392	0,7817

Tabela 95 – Melhores parâmetros globais encontrados pelo CMA-ES em cada experimento no estudo de caso 8

CMA-ES						
Experimento	N	ρ_W	s_{in}	s_{back}	α	c_W
1	17	0,4954	0,9632	0,2794	0,6770	0,7223
2	17	0,2792	0,9620	0,4526	0,6961	0,6560
3	18	0,5606	0,9731	0,2424	0,6075	0,7250
4	17	0,5071	0,9845	0,5797	0,7591	0,3901
5	17	0,6564	0,9065	0,3568	0,5708	0,8734
6	16	0,1408	0,9475	0,3774	0,7214	0,3628
7	17	0,5737	0,9151	0,4002	0,6958	0,4470
8	17	0,4162	0,9754	0,3410	0,6345	0,8951
9	16	0,3910	0,8800	0,3175	0,6856	0,6731
10	18	0,6415	0,8594	0,3010	0,5105	0,7764

- Resultados de RMSE

Tabela 96 – Resultados de RMSE da solução final de cada submodelo por experimento no estudo de caso 8 (parte 1)

Exp	Submodelo								
	AEIQ-R/Random/ESN			AEIQ-R/SRG/ESN			AEIQ-R/AEIQ-R/ESN		
	Trein.	Val.	Teste	Trein.	Val.	Teste	Trein.	Val.	Teste
1	0,176660	0,123811	0,152534	0,184482	0,493936	10,906988	0,595179	0,095448	0,134154
2	0,204241	0,119933	0,148890	0,542704	0,131793	0,164899	1,330722	0,088943	0,137112
3	0,289931	0,133489	0,163356	0,152261	0,140921	0,176874	0,273179	0,096790	0,133115
4	0,095925	0,122974	0,148541	0,095925	0,122974	0,148541	0,429023	0,091659	0,129919
5	0,095490	0,118435	0,144348	0,309132	0,180614	0,195387	1,329163	0,095353	0,142428
6	0,256893	0,116464	0,143869	0,221318	43,109019	55,414491	0,101546	0,093785	0,131366
7	0,099596	0,118079	0,133926	0,249464	0,148817	0,157435	0,116588	0,085841	0,126708
8	0,184740	0,118811	0,140180	0,245585	0,138264	0,159593	0,241557	0,095620	0,150908
9	0,227286	0,123972	0,152673	3,750375	2,394618	2,858014	0,442786	0,096697	0,146362
10	0,097507	0,124777	0,150540	0,097507	0,124777	0,150540	1,475325	0,089632	0,133050

Tabela 97 – Resultados de RMSE da solução final de cada submodelo por experimento no estudo de caso 8 (parte 2)

Exp	Submodelo								
	CMA-ES/Random/ESN			CMA-ES/SRG/ESN			CMA-ES/CMA-ES/ESN		
	Trein.	Val.	Teste	Trein.	Val.	Teste	Trein.	Val.	Teste
1	0,433533	0,129196	0,177009	5,128502	5,217768	4,745321	1,212172	0,067180	0,120457
2	0,218356	0,124115	0,178979	0,287190	0,124158	0,160338	1,720629	0,077597	0,143561
3	0,167365	0,124440	0,164705	0,167365	0,124440	0,164705	0,473532	0,063533	0,114600
4	0,135219	0,122291	1,023578	4,205075	4,174520	4,222534	3,056767	0,073675	0,153759
5	0,128021	0,122158	0,153794	0,128021	0,122158	0,153794	6,040970	0,077321	0,122585
6	0,402161	0,124287	0,186191	0,402161	0,124287	0,186191	0,313099	0,089688	0,134200
7	0,086653	0,125877	0,145730	0,190872	0,163627	0,162943	3,237833	0,067312	0,129668
8	0,143967	0,132137	0,152585	0,143967	0,132137	0,152585	0,505183	0,080122	0,145399
9	0,096256	0,130435	0,150208	0,104430	0,267586	0,177618	0,783994	0,067127	0,149104
10	0,180694	0,137131	0,186113	0,257169	0,284927	0,208877	0,708126	0,076687	0,132570

- Resultados de SRG

Tabela 98 – Resultados de SRG dos submodelos em cada experimento no estudo de caso 8

Exp	Submodelo					
	1	2	3	4	5	6
1	2,596837	1,739645	1,928828	2,714177	2,200686	2,309289
2	1,424329	0,754586	2,188459	2,686595	1,867688	2,291176
3	2,214884	1,292026	2,648060	2,457557	2,457557	2,434852
4	1,736267	1,736267	2,536890	2,306995	1,611392	2,283728
5	2,016280	1,545009	2,155650	2,335233	2,335233	2,830488
6	1,711083	1,213235	1,643212	1,757172	1,757172	2,234108
7	5,332137	1,498826	3,397725	2,479910	2,229519	2,224332
8	1,382553	0,910947	1,946533	2,256277	2,256277	2,502750
9	2,489350	1,569035	1,765015	2,114119	2,088479	2,300708
10	1,750770	1,750770	2,395882	3,904493	2,850795	2,577490

i) Estudo de Caso 9

- Melhores parâmetros globais

Tabela 99 – Melhores parâmetros globais encontrados pelo AEIQ-R em cada experimento no estudo de caso 9

AEIQ-R						
Experimento	N	ρ_W	s_{in}	s_{back}	α	c_W
1	383	0,5048	0,4134	0,9219	0,9792	0,3905
2	420	0,2193	0,0428	0,5502	0,9850	0,5604
3	426	0,4878	0,2292	0,9000	0,9824	0,8492
4	364	0,4214	0,0865	0,6066	0,9229	0,5135
5	385	0,4795	0,2962	0,7941	0,9201	0,7244
6	430	0,3137	0,1137	0,5592	0,9748	0,7794
7	397	0,3176	0,0709	0,6773	0,9662	0,7157
8	400	0,3959	0,0992	0,5416	0,9131	0,7581
9	365	0,4441	0,2819	0,7363	0,9660	0,5984
10	433	0,4523	0,2458	0,7122	0,9614	0,5277

Tabela 100 – Melhores parâmetros globais encontrados pelo CMA-ES em cada experimento no estudo de caso 9

CMA-ES						
Experimento	N	ρ_W	s_{in}	s_{back}	α	c_W
1	422	0,3581	0,1220	0,7295	0,9832	0,3768
2	448	0,4419	0,1797	0,7980	0,9520	0,5556
3	427	0,2968	0,0393	0,5492	1,0000	0,4989
4	451	0,4158	0,0143	1,0000	0,9914	0,6817
5	423	0,4728	0,1443	0,7702	1,0000	0,1616
6	416	0,4846	0,1785	1,0000	1,0000	0,5320
7	399	0,4624	0,2296	0,9119	0,9918	0,4233
8	450	0,3730	0,1468	0,8605	1,0000	0,8078
9	444	0,3637	0,2579	0,8458	1,0000	0,5529
10	430	0,3450	0,0817	0,8026	0,9989	0,3893

- Resultados de RMSE

Tabela 101 – Resultados de RMSE da solução final de cada submodelo por experimento no estudo de caso 9 (parte 1)

Exp	Submodelo								
	AEIQ-R/Random/ESN			AEIQ-R/SRG/ESN			AEIQ-R/AEIQ-R/ESN		
	Trein.	Val.	Teste	Trein.	Val.	Teste	Trein.	Val.	Teste
1	0,203253	0,212837	0,345683	0,204305	0,215345	0,362993	0,203010	0,209414	0,341469
2	0,200774	0,207659	0,602864	0,201788	0,209737	0,413303	0,200422	0,206440	0,404311
3	0,202960	0,210411	0,372291	0,201472	0,211791	0,327616	0,201519	0,207762	0,335533
4	0,204881	0,210195	0,363737	0,204557	0,212593	0,343195	0,204653	0,208755	0,339644
5	0,203904	0,210636	0,347206	0,205194	0,213936	0,386961	0,204918	0,210201	0,328428
6	0,201882	0,208973	0,331630	0,202054	0,212559	0,471601	0,201144	0,206428	0,363950
7	0,201633	0,208973	0,393323	0,204570	0,214469	0,383360	0,201668	0,207583	0,394154
8	0,203439	0,211037	0,380428	0,203439	0,211037	0,380428	0,203709	0,208074	0,377302
9	0,204907	0,211984	0,523120	0,203642	0,213157	0,344946	0,202593	0,209412	0,345669
10	0,201311	0,209017	0,431829	0,202164	0,210819	0,377721	0,201994	0,207820	0,401864

Tabela 102 – Resultados de RMSE da solução final de cada submodelo por experimento no estudo de caso 9 (parte 2)

Exp	Submodelo								
	CMA-ES/Random/ESN			CMA-ES/SRG/ESN			CMA-ES/CMA-ES/ESN		
	Trein.	Val.	Teste	Trein.	Val.	Teste	Trein.	Val.	Teste
1	0,200290	0,208491	0,377351	0,200776	0,209558	0,387707	0,201049	0,207627	0,346632
2	0,202345	0,210538	0,344291	0,201822	0,212816	0,349370	0,200887	0,208393	0,377176
3	0,200471	0,208648	0,482597	0,200539	0,209129	0,346856	0,199738	0,206611	0,436414
4	0,201723	0,208915	0,553672	0,200629	0,209050	0,399120	0,202036	0,208233	0,409620
5	0,201995	0,210242	0,324921	0,204110	0,213254	0,319524	0,201207	0,207868	0,354887
6	0,199595	0,209571	0,331236	0,203053	0,210886	0,338940	0,201280	0,207824	0,343025
7	0,201168	0,209934	0,464948	0,201279	0,212030	0,307495	0,201098	0,209122	0,370164
8	0,200138	0,207931	0,362233	0,200103	0,210203	0,375530	0,200738	0,206037	0,349926
9	0,200250	0,208868	0,334087	0,199961	0,211123	0,339120	0,199686	0,207054	0,461988
10	0,199629	0,207636	0,353493	0,200450	0,209088	0,384384	0,199838	0,207620	0,403887

- Resultados de SRG

Tabela 103 – Resultados de SRG dos submodelos em cada experimento no estudo de caso 9

Exp	Submodelo					
	1	2	3	4	5	6
1	348,529453	341,099267	339,212106	244,403900	242,815378	242,076827
2	175,992167	158,698514	175,594093	279,602191	273,577607	277,082143
3	321,934082	309,075647	323,328091	181,204281	167,959657	175,788819
4	169,230660	161,042430	167,644349	346,845724	337,389404	340,912514
5	261,664770	256,916369	265,872815	272,879876	262,474618	282,802995
6	192,048170	180,727873	184,632267	343,476178	330,693144	370,507252
7	209,506780	196,798299	206,976737	320,651341	310,902024	323,729818
8	149,592659	149,592659	157,513358	334,997534	310,297222	313,798115
9	246,569337	243,553989	255,875307	335,828926	313,100348	335,575123
10	269,043492	256,200627	260,417680	275,556070	270,335624	261,163932