Rede de Ecos de Estados Robusta para Identificação Recursiva de Sistemas *

Renan Bessa e Guilherme de Alencar Barreto

Universidade Federal do Ceará, Departamento de Engenharia de
Teleinformática, Fortaleza, Ceará, Brasil,
(e-mail: renanbessa@alu.ufc.br e gbarreto@ufc.br).

Abstract: The use of recurrent neural networks in online system identification is very limited in real-world applications, mainly due to the propagation of errors caused by the iterative nature of the prediction task over multiple steps ahead. Bearing this in mind, we revisit design issues regarding the robustness of the *echo state network* (ESN) model in such online learning scenarios using a recursive estimation algorithm and an outlier robust-variant of it. By means of a comprehensive set of experiments, we show that the performance of the ESN is dependent on the adequate choice of the feedback pathways and that the prediction instability is amplified by the norm of the output weight vector, an often neglected issue in related studies.

Resumo: O emprego de redes neurais recorrentes na identificação *online* de sistemas é bastante limitada em aplicações reais, principalmente devido à propagação do erro causada pela natureza iterativa da tarefa de predição ao longo de vários passos à frente. Tendo isso em mente, revisita-se questões de projeto relacionadas à robustez do modelo da rede de ecos de estados (*echo state network*, ESN) em tais cenários de aprendizado, usando um algoritmo de estimação recursiva e uma de suas variantes robusta a *outlier*. Por meio de um conjunto abrangente de experimentos, mostra-se que o desempenho da ESN é dependente da escolha adequada das vias de realimentação e que a instabilidade da previsão é ampliada pela norma do vetor de peso de saída, uma questão frequentemente negligenciada em estudos relacionados.

Keywords: System identification; Recurrent neural networks; Echo state network; Robustness to outliers.

Palavras-chaves: Identificação de sistemas; Rede neural recorrente; Rede de ecos de estados; Robustez a outliers.

1. INTRODUÇÃO

A rede de ecos de estados (*echo state network*, ESN) (Jaeger, 2001) é uma rede neural recorrente (*recurrent neural network*, RNN) com uma estrutura especial chamada reservatório de dinâmicas, contendo um grande número de neurônios esparsamente auto- e interconectados. Os pesos da camada de entrada, dos neurônios do reservatório, os internos e os responsáveis pela conexão direta da entrada com a saída do sistema são fixos e aleatoriamente atribuídos. O treinamento da ESN consiste basicamente na estimação dos pesos dos neurônios da camada de saída, também chamada de camada de *readout*, por meio do método dos mínimos quadrados ordinários. Assim, a arquitetura da ESN e seu treinamento são conceitualmente mais simples se comparados aos das demais RNNs, geralmente extensões da rede MLP e do algoritmo de retropropagação dos erros.

A ESN tem sido aplicada com sucesso na predição de séries caóticas, reconstrução de atratores e identificação de sistemas não lineares (Jaeger, 2002). Com o objetivo de adaptar essa rede ao processamento de dados reais, conta-

minados com ruídos não gaussianos (e.g. ruído impulsivo) e/ou outliers, vários trabalhos têm desenvolvido técnicas e métodos robustos para a estimação dos pesos da camada de saída em cenários mais desafiadores. Um das estratégias utiliza a regularização de Tikhonov, que penaliza vetores de pesos com norma elevada, reduzindo a ocorrência de sobreajuste do modelo aos dados corrompidos (Lukoševičius and Jaeger, 2009).

Abordagens mais complexas na estimação dos pesos também têm sido propostas, incluindo inferência bayesiana, em que se substitui a usual função de verossimilhança gaussiana, que é bastante sensível a outliers, por uma distribuição laplaciana (Li et al., 2012) ou gaussiana mista (Han and Xu, 2018). Além disso, o emprego de critérios de aprendizagem baseados na teoria da informação, tais como critério da máxima correntropia (*maximum correntropy*) (Guo et al., 2017) e da máxima correntropia generalizada (*maximum generalized correntropy*) (Zhang et al., 2018), também tem alcançado bons resultados na diminuição da sensibilidade a outliers.

Outras configurações robustas da ESN abrem mão da habitual camada de saída linear por uma não linear, buscando apreender o máximo de informação oriunda do reservatório de dinâmicas. Uma dessas abordagens é

* Os autores agradecem ao apoio financeiro a esse estudo por parte das seguintes agências de fomento: CAPES (finance code 001), FUNCAP (BMD-008-01413.01.02-17) e CNPq (309451 / 2015-9).

a formulação usando máquina de vetor suporte (*support vector machine*) com função custo robusta, tais como a função perda ϵ -insensível (ϵ -insensitive loss) ou a função de Huber (Shi and Han, 2007). É possível também usar regras de aprendizado não lineares, como o algoritmo dos mínimos quadrados recursivo kernelizado (*kernel recursive least squares*) (Zhou et al., 2018). Apesar de não terem sido avaliadas diretamente em cenários com outliers, existem trabalhos em que arquiteturas ESN treinadas com o algoritmo *Laplacian eigenmaps* (Han and Xu, 2018) e modelos de regressão através de processos gaussianos (Chatzis and Demiris, 2011) podem ser capazes de lidar com os efeitos adversos causados por outliers.

Apesar dos trabalhos supracitados, a aplicação da ESN em identificação online de sistemas dinâmicos não lineares é ainda limitada. Isto é particularmente verdadeiro em cenários envolvendo dados reais em previsões iterativas de múltiplos passos à frente. Como a chance de encontrar outliers em tais cenários é elevada, a propagação de erros devido à natureza iterativa da tarefa de predição causa divergência (ou seja, instabilidade) nos sinais previstos com o passar do tempo. Tal instabilidade é amplificada pela norma do vetor de peso da saída, um problema muitas vezes negligenciado em estudos anteriores.

Tendo em mente a estabilidade da previsão, revisita-se a ESN combinada com o algoritmo dos mínimos quadrados recursivo (*recursive least squares*, RLS), doravante referida pelo acrônimo RLS-ESN (Jaeger, 2002), para identificação online de sistemas na presença de outliers. Faz-se também uma avaliação da influência de diferentes vias de realimentação no desempenho da ESN. Por fim, propõem-se a substituição do algoritmo RLS por uma variante robusta a outliers: o algoritmo *recursive least M-estimate* (RLM) (Chan and Zou, 2004), dando origem ao modelo RLM-ESN. Um estudo comparativo abrangente é realizado usando dados reais e sintéticos contaminados por outliers em diferentes proporções, a fim de avaliar o desempenho das diferentes variantes da ESN discutidas neste artigo na tarefa de identificação online de sistemas dinâmicos não lineares.

2. FUNDAMENTOS DA ESN

O processo de identificação de sistemas em tempo discreto necessita da disponibilidade de um conjunto de entradas e de saídas do sistema de interesse: $(\mathbf{u}(n), \mathbf{t}(n))_{n=0}^{N_{all}}$, em que $\mathbf{u}(n) = [u_1(n), u_2(n), \dots, u_K(n)]^T$ é um vetor de entrada K -dimensional e $\mathbf{t}(n) = [t_1(n), t_2(n), \dots, t_L(n)]^T$ é um vetor de saída (*target*) L -dimensional em um dado instante n , $n = 1, \dots, N_{all}$. Esse conjunto de dados é dividido entre o subconjunto de treinamento e o de teste, $N_{all} = N_{train} + N_{test}$. Uma vez que o treinamento adequado esteja finalizado, a ESN deve ser capaz de estimar um vetor de saídas $\mathbf{y}(n) \in \mathbb{R}^{L \times 1}$ que produza o menor desvio possível de $\mathbf{t}(n)$. Para tanto, emprega-se o método *teacher forcing* que possibilita o treinamento da ESN ao substituir $\mathbf{y}(n-1)$ pelo valor desejado da iteração anterior $\mathbf{t}(n-1)$, como pode ser visualizado na Figura 1.

As saídas dos R neurônios do reservatório, também chamadas de variáveis de estados, são representadas pelo vetor $\mathbf{x}(n) \in \mathbb{R}^{R \times 1}$ cuja dinâmica é descrita por

Figura 1. Arquitetura da ESN durante o treinamento. As linhas tracejadas representam as vias de retroalimentação opcionais. $\mathbf{W}$, $\mathbf{W}_{in}$, $\mathbf{W}_{back}$ e $\mathbf{W}_{out}$ são matrizes de pesos.

$$\mathbf{x}(n) = \mathbf{f}(\mathbf{W}\mathbf{x}(n-1) + \mathbf{W}_{in}\mathbf{u}(n) + \mathbf{W}_{back}(\mathbf{t}(n-1) + \mathbf{v}_1(n)) + \mathbf{v}_2(n) - \mathbf{b}), \quad (1)$$

onde $\mathbf{f}(\cdot)$ é uma função de ativação não linear com operação elemento a elemento, normalmente a função sigmoide logística ou a tangente hiperbólica, $\mathbf{W} \in \mathbb{R}^{R \times R}$ é a matriz de pesos do reservatório, $\mathbf{W}_{in} \in \mathbb{R}^{R \times K}$ e $\mathbf{W}_{back} \in \mathbb{R}^{R \times L}$ são, respectivamente, as matrizes de pesos que conectam a entrada e a saída do modelo ao reservatório. Os termos $\mathbf{v}_1(n) \in \mathbb{R}^{L \times 1}$ e $\mathbf{v}_2(n) \in \mathbb{R}^{R \times 1}$ são vetores de ruído branco opcionais, enquanto $\mathbf{b} \in \mathbb{R}^{R \times 1}$ é o vetor de limiares.

A saída do modelo neural é calculada como

$$\mathbf{y}(n) = \mathbf{f}_{out}(\mathbf{W}_{out}\mathbf{h}(n)), \quad (2)$$

em que $\mathbf{f}_{out}(\cdot)$ é a função de ativação dos neurônios de saída (usualmente, a função identidade), $\mathbf{W}_{out} \in \mathbb{R}^{L \times (1+K+R+L)}$ é a matriz de peso de saída, $\mathbf{h}(n) = [-1, \mathbf{u}(n), \mathbf{x}(n), (\mathbf{t}(n-1) + \mathbf{v}_1(n))]^T \in \mathbb{R}^{H \times 1}$ é um vetor concatenado, tal que $H = 1 + K + R + L$.

Os elementos das matrizes $\mathbf{W}$, $\mathbf{W}_{in}$ e $\mathbf{W}_{back}$ são atribuídos aleatoriamente e se mantêm fixos durante o processo de treinamento e teste do modelo. Entretanto, $\mathbf{W}$ deve ser esparsa, isto é, deve conter apenas uma pequena porcentagem de elementos não nulos. A fim de garantir a propriedade de “ecos dos estados”¹, o raio espectral de $\mathbf{W}$ deve ser menor que 1, sendo esta uma condição suficiente para maioria dos casos. As magnitudes dos vetores de ruídos $\mathbf{v}_1(n)$ e $\mathbf{v}_2(n)$ devem ser escolhidos de forma a equilibrar estabilidade e acurácia. Estes vetores também ajudam na regularização da solução. Ao fim do treino, as realimentações das saídas previstas pela ESN são realizadas a fim de avaliar o desempenho do modelo em simulação livre, conforme ilustrado na Figura 2. Mais detalhes da configuração da ESN podem ser encontrados em Jaeger (2001).

2.1 Algoritmos recursivos para estimação de parâmetros

O presente artigo lida apenas com sistemas SISO (uma entrada, uma saída) e promove a estimação recursiva do vetor de pesos $\mathbf{w}_{out} \in \mathbb{R}^{H \times 1}$ da camada de *readout* em que apenas uma saída é considerada. Além disso, comparam-se dois algoritmos para estimação dos parâmetros, a saber: (i) o clássico algoritmo RLS, cuja função custo é $J_{LS}(n) = \sum_{i=1}^n \gamma^{n-i} e^2(i)$, em que $0 < \gamma \leq 1$ é o fator

¹ Propriedade esta que relacionada às propriedades assintóticas da dinâmica do reservatório quando excitado por um sinal exógeno.

Figura 2. Arquitetura padrão da ESN em simulação livre (as linhas tracejadas são vias opcionais).

de esquecimento e $e(n) = t(n) - \mathbf{w}_{out}^T(n-1)\mathbf{h}(n)$ é o erro de predição no instante n ; e (2) o algoritmo RLM, que possui a função custo

$$J_\rho(n) = \sum_{i=1}^n \gamma^{n-i} \rho(e(i)), \quad (3)$$

onde $\rho(e)$ é uma função que possui o propósito de limitar os efeitos negativos oriundos de erros de estimação muito grandes (causados, em geral, por ruído não gaussiano ou por outliers). O algoritmo RLM se reduz ao RLS quando $\rho(e(i)) = e^2(i)$.

Os pesos de saída ótimos podem ser determinados pela diferenciação de $J_\rho(n)$ em respeito a $\mathbf{w}_{out}$ e igualando sua derivada a zero. Depois de alguma manipulação algébrica, o algoritmo RLM resultante envolve as seguintes equações:

$$\mathbf{w}_{out}(n) = \mathbf{w}_{out}(n-1) + e(n)\mathbf{k}(n), \quad (4)$$

$$\mathbf{k}(n) = \frac{q(e(n))\mathbf{S}(n-1)\mathbf{h}(n)}{q(e(n))\mathbf{h}^T(n)\mathbf{S}(n-1)\mathbf{h}(n) + \gamma} \quad (5)$$

$$\mathbf{S}(n) = \gamma^{-1} (\mathbf{I} - \mathbf{k}(n)\mathbf{h}^T(n)) \mathbf{S}(n-1) \quad (6)$$

em que $\mathbf{k}(n)$ é o vetor de ganho e $q(e) = \frac{1}{e} \frac{\partial \rho(e)}{\partial e}$. A matriz $\mathbf{S}(n)$ é a estimativa recursiva da inversa da matriz de correlação, i.e. $\mathbf{S}(n) = \mathbf{R}^{-1}(n)$. Neste artigo, utiliza-se a função de Huber clássica, definida como

$$\rho(e) = \begin{cases} \frac{e^2}{2}, & |e| < \xi \\ \xi|e| - \frac{\xi^2}{2}, & |e| > \xi, \end{cases} \Rightarrow q(e) = \begin{cases} 1, & |e| < \xi \\ \frac{\xi}{|e|}, & |e| > \xi, \end{cases} \quad (7)$$

onde ξ é um limiar de normalidade, sendo considerados outliers aquelas observações cujos valores absolutos dos erros associados forem maiores que esse limiar. É importante notar que $q(e) = 1$ para valores menores que ξ ; caso contrário, $q(e) \rightarrow 0$ conforme $|e| \rightarrow \infty$.

É possível estimar ξ continuamente para cada nova entrada. Para este fim, pode-se usar a seguinte expressão:

$$\xi(n) = 2,576\hat{\sigma}(n), \quad (8)$$

onde $\hat{\sigma}(n)$ é um estimador robusto para o desvio padrão dos resíduos:

$$\hat{\sigma}^2(n) = c_1 \text{med}\{e^2(n), \dots, e^2(n - N_w + 1)\}, \quad (9)$$

tal que $c_1 = 1.483(1 + 5/(N_w - 1))$, $\text{med}\{\cdot\}$ é a mediana amostral e N_w é o comprimento da janela amostral. A robustez e a complexidade desse método são diretamente proporcionais ao valor de N_w .

3. METODOLOGIA DE AVALIAÇÃO E SIMULAÇÃO

Com base no diagrama da Figura 2, propõem-se avaliar as seguintes variações da arquitetura da ESN: (i) **Modelo 1** (M1) - vias de realimentação 1 e 2 desabilitadas; (ii) **Modelo 2** (M2) - via de realimentação 1 habilitada e via de realimentação 2 desabilitada; e (iii) **Modelo 3** (M3) - vias de realimentação 1 e 2 habilitadas.

Além disso, três abordagens para estimação do vetor de pesos de saída $\mathbf{w}_{out}$ são avaliadas. O algoritmo RLS padrão é usado como referência de comparação com o algoritmo RLM. Uma terceira abordagem é específica para a tarefa de predição iterada, visando proporcionar maior estabilidade na predição em horizontes de tempo mais longos, quando os outliers estão presentes nos dados.

RLM com Detecção de Outliers (*RLM with Outlier Detection*, RLM-OD): Na iteração n , a saída predita é calculada como $y(n) = \mathbf{w}_{out}^T(n-1)\mathbf{h}(n)$. O erro de predição é calculado como $e(n) = t(n) - \mathbf{w}_{out}^T(n-1)\mathbf{h}(n)$ e utilizado pelo algoritmo RLM tal como apresentado nas Eqs. (4) e (5). Contudo, se $|e(n)| > \xi$, a saída desejada $t(n)$ provavelmente é um *outlier*. Então, na próxima iteração do treinamento, substitui-se a observação real $t(n)$ pelo seu valor predito $y(n)$ para evitar o preenchimento do vetor de regressores da entrada da ESN com um *outlier*.

3.1 Avaliação e Simulação

Todos os conjuntos de dados usados nos experimentos são referentes a sistemas SISO. Estes estão elencados na Tabela 1 com alguns detalhes adicionais. Com a finalidade de projetar e avaliar os modelos neurais, os conjuntos de dados são divididos das seguintes formas: a primeira metade dos pares de amostras é para o treinamento, de 1 a N_{train} , e a segunda metade é para a validação, de $N_{train}+1$ a N_{all} . Vale ressaltar que, uma vez que os sistemas de interesse são SISO, o vetor de entrada $\mathbf{u}(n) \in \mathbb{R}^L$, o vetor da saída desejada $\mathbf{t}(n) \in \mathbb{R}^L$ e o vetor de ruído aleatório $\mathbf{v}_1(n) \in \mathbb{R}^K$ tornam-se escalares (i.e. $K = 1$ and $L = 1$). O vetor de ruído aleatório $\mathbf{v}_2 \in \mathbb{R}^R$ possui a mesma dimensão do número de neurônios do reservatório.

Dois critérios são empregados na avaliação das variantes da ESN. A raiz do erro quadrático médio (*root mean square error*, RMSE) das previsões em simulação livre, definido por

$$RMSE = \sqrt{\frac{\sum_{n=N_{train}+1}^{N_{all}} (t(n) - y(n))^2}{N_{test}}} \quad (10)$$

e a norma euclidiana (l_2) do vetor de pesos da camada de *readout*, $\|\mathbf{w}_{out}\|$, reportada como "norm" nos gráficos. A inclusão da norma do vetor de pesos como figura de mérito não é usual, mas para a presente proposta esta torna-se fundamental para avaliar a estabilidade da capacidade preditiva de longo prazo da ESN.

As configurações da ESN são padronizadas para todas as variantes avaliadas, exceto o número de neurônios R , e estão mosradas na Tabela 2. O valor de R foi otimizado para cada sistema, sem a presença de outliers e com o algoritmo RLS, buscando equilíbrio entre RMSE, sobreajuste e estabilidade, para $R \in \{10, 25, 50, 100, 150, \dots, 500\}$. O algoritmo RLM possui um parâmetro extra, N_w , que define

Tabela 1. Lista dos conjuntos de dados utilizados no experimentos computacionais.

Sistema	Descrição	N_{all}	Ref.
TOP	Um problema de décima ordem sintético (Exemplo 5)	2000	Atiya and Parlos (2000)
Tank	Tanques em cascatas (Tank2.mat; tanque inferior: y_0)	7500	Wigren (2010)
Dryer	Uma bancada de teste agindo como um secador de cabelo (96-006)	1000	De Moor et al. (1997)
Exchanger	Um trocador de calor a vapor de líquido saturado (97-002)	4000	De Moor et al. (1997)
Robot Arm	Um braço robótico flexível (96-009)	1024	De Moor et al. (1997)

Tabela 2. Configuração padrão de vários hiperparâmetros para as simulações.

Configurações da ESN	
Raio espectral ($\mathbf{W}$)	0.98
Esparsidade ($\mathbf{W}$)	2%
Pesos de entrada	$\mathbf{W}_{in} \sim U(-0.1, 0.1)$
Pesos de realimentação	$\mathbf{W}_{back} \sim U(-0.1, 0.1)$
Pesos do bias	$\mathbf{b} \sim U(-0.1, 0.1)$
Vetor de ruído 1	$\mathbf{v}_1 \sim U(-0.0001, 0.0001)$
Vetor de ruído 2	$\mathbf{v}_2 \sim U(-0.0001, 0.0001)$
Configurações do RLS/RLM	
Fator de esquecimento (γ)	0.99998
$\mathbf{S}(0)$	$10^4 \mathbf{I}$

o tamanho da janela amostral para o cálculo do limiar ξ . Decide-se avaliar todas as amostras até a interação n , ou seja, N_w é crescente e varia de 1 a N_{train} .

A robustez das variantes da ESN é avaliada em cenários de contaminação por outliers de 5%, 10% e 15% das amostras de treinamento. Esses são gerados por $\sigma_{train} \mathcal{T}(0, 2)$, em que σ_{train} é o desvio padrão dos dados originais de treinamento e $\mathcal{T}(0, 2)$ é uma distribuição t -Student com média zero e dois graus de liberdade. Os valores retornados dessa distribuição são saturados em ± 20 . As entradas e as saídas dos sistemas, com e sem outliers, são normalizadas, subtraindo-se a média e dividindo-as por três vezes o valor do desvio padrão, estatísticas das amostras de treinamento. O RMSE é calculado com os valores não normalizados dos dados. São realizadas 20 rodadas independentes de treinamento/teste, para os quais todos os pesos e os outliers são iniciados aleatoriamente. Em cada execução, os mesmos pesos e a mesma contaminação dos dados são aplicados a cada modelo investigado para que se possa realizar uma comparação de desempenho justa para os diferentes arquiteturas da ESN (M1, M2 e M3) e os diferentes métodos de estimação de parâmetros (RLS, RLM e RLM-OD).

4. RESULTADOS

Antes de executar as simulações finais, o número de neurônios no reservatório foi escolhido pela experimentação com dados livres de outliers. Observou-se a convergência para um valor mínimo da RMSE com o aumento do número de neurônios para todos os conjuntos de dados. Na Tabela 3, apresentam-se os números escolhidos e os resultados obtidos pela RLS-ESN para as três arquiteturas avaliadas (M1, M2 e M3). Os resultados são expressos no seguinte formato: $mean \pm std(median)$, sendo, respectivamente, a média, o desvio padrão e a mediana dos valores da RMSE e da norma de $\mathbf{w}_{out}$ calculados ao longo das 20 execuções independentes.

Mantendo o mesmo número de neurônios no reservatório, os desempenhos dos modelos neurais são melhor ilustrados

Tabela 3. Desempenho dos modelos utilizando o algoritmo RLS com dados livres de outliers.

TOP $R = 500$	M1	RMSE	$0.0312 \pm 0.0105(0.0279)$
		Norm	$98.04 \pm 20.34(101.73)$
	M2	RMSE	$0.0314 \pm 0.0110(0.0270)$
		Norm	$75.22 \pm 15.10(76.94)$
	M3	RMSE	$0.0344 \pm 0.0079(0.0316)$
		Norm	$62.27 \pm 16.72(67.88)$
Tank $R = 500$	M1	RMSE	$0.3227 \pm 0.0916(0.3202)$
		Norm	$152.64 \pm 47.61(146.76)$
	M2	RMSE	$0.1568 \pm 0.0244(0.1579)$
		Norm	$5.09 \pm 0.82(5.33)$
	M3	RMSE	$0.1056 \pm 0.0122(0.1033)$
		Norm	$2.36 \pm 0.08(2.36)$
Dryer $R = 250$	M1	RMSE	$0.1356 \pm 0.0377(0.1229)$
		Norm	$18.79 \pm 2.33(19.01)$
	M2	RMSE	$0.1238 \pm 0.0212(0.1165)$
		Norm	$8.59 \pm 0.85(8.43)$
	M3	RMSE	$0.1299 \pm 0.0170(0.1237)$
		Norm	$8.34 \pm 1.24(8.17)$
Exchanger $R = 450$	M1	RMSE	$0.2432 \pm 0.0305(0.2339)$
		Norm	$37.05 \pm 4.28(37.24)$
	M2	RMSE	$0.2507 \pm 0.0327(0.2400)$
		Norm	$33.449 \pm 2.80(33.23)$
	M3	RMSE	$0.2312 \pm 0.0348(0.2227)$
		Norm	$24.03 \pm 2.86(24.74)$
Robot Arm $R = 500$	M1	RMSE	$0.2840 \pm 0.0459(0.2785)$
		Norm	$165.89 \pm 16.35(172.85)$
	M2	RMSE	$0.2991 \pm 0.0552(0.2907)$
		Norm	$118.25 \pm 11.39(121.78)$
	M3	RMSE	$0.0238 \pm 0.0136(0.0198)$
		Norm	$2.33 \pm 0.30(2.25)$

nas Figuras 3 e 4, onde pode-se compará-los em diferentes cenários de contaminação e com os demais estimadores (RLM e RLM-OD). Deve ser notado que, para a variante M1, o algoritmo RLM-OD comporta-se exatamente como o RLM padrão, uma vez que a saída não é realimentada, pois as vias de conexões 1 e 2 são desabilitadas.

Como pode ser visto na Tabela 3, a presença de vias de realimentação da saída (variantes M2 e M3) reduz a norma da camada de saída da ESN, o que leva a uma melhoria na generalização e evita sobreajuste. Em outras palavras, essas vias de realimentação atuam como um regularizador do modelo. A variante M3, que realimenta a saída tanto para a camada de *readout* como para o reservatório, apresenta as menores normas e os valores das RMSEs são inferiores ou bastante próximos quando comparados com as respostas das demais variantes.

Na Figura 3, reportam-se os resultados para diferentes cenários de contaminação, nos quais os três modelos apresentaram valores semelhantes de RMSE ao usar o algoritmo de estimação do tipo RLM. Em outros termos, para esses conjuntos de dados, as variantes M2 e M3 apresentaram desempenhos similares, independentemente do algoritmo de estimação do tipo RLM utilizado. Sobre a norma do

Figura 3. Desempenho dos modelos dos sistemas (a) TOP, (b) Dryer e (c) Exchanger.

vetor de saída, todas as variantes (M1, M2 e M3) que usam algoritmos de estimativa do tipo RLM são praticamente insensíveis ao aumento da quantidade de outliers. Como esperado, nas estimações realizadas com o algoritmo RLS padrão, a norma do vetor de peso de saída aumenta consideravelmente com o aumento na quantidade de outliers.

Para os conjuntos de dados *Tank* e *Robot Arm*, o desempenho superior da variante M3 sobre as outras duas é evidente, como mostra a Figura 4. Para esses conjuntos de dados, o desempenho da variante M3 usando o algoritmo de estimativa RLM-OD proposto é consistentemente superior a mesma arquitetura usando o algoritmo RLM padrão. Isso é verdade não só em termos de valores menores de RMSE, mas também em termos de normas menores para o vetor de peso de saída.

Com o objetivo de ilustrar a robustez dos estimadores e a importância de uma norma reduzida na qualidade do modelo do sistema, apresenta-se as respostas de uma rodada de treinamento/teste para um cenário de 15% de contaminação com a variante M3 na Figura 5. As linhas pontilhadas verticais indicam o final da fase de treinamento e o início da tarefa de previsão de múltiplos passos à frente. O desempenho superior da variante M3, que possui todas as vias de realimentação, usando os

Figura 4. Desempenho dos modelos dos sistemas (a) Tank and (b) Robot arm.

Figura 5. Predições dos modelos (a) Tank e (b) Robot Arm para o cenário de contaminação de 15%.

algoritmos de estimativa robustos (RLM e RLM-OD) é bastante evidente.

Essas variantes (M3-RLM e M3-RLM-OD) foram capazes de aprender a dinâmica subjacente do sistema de interesse mesmo na presença de outliers durante a fase de treinamento e capazes de diminuir o efeito da propagação do erro durante a fase de teste devido às normas menores do vetor de peso de saída. Os resultados numéricos correspondentes para esse exemplo são relatados na Tabela 4.

Tabela 4. Resultados numéricos dos modelos (a) *Tank* e (b) *Robot Arm* para o cenário de contaminação de 15%.

		RLS	RLM	RLM-OD
<i>Tank</i>	RMSE	0.3553	0.1796	0.1063
	<i>Norm</i>	108.46	25.53	8.40
<i>Robot Arm</i>	RMSE	0.2509	0.0997	0.0482
	<i>Norm</i>	108.75	46.09	13.14

5. CONCLUSÕES E TRABALHOS FUTUROS

Neste trabalho, avaliou-se o desempenho do modelo ESN para tarefa de identificação recursiva em cenários contaminados por *outliers*. Duas variantes robustas a outlier do algoritmo de estimação recursiva RLS foram utilizadas para estimar o vetor de peso de saída, a saber o algoritmo RLM e uma variante denominada RLM-OD proposta neste artigo.

A partir da análise de um conjunto abrangente de experimentos computacionais empregando conjuntos de dados de *benchmarking*, verificou-se que a combinação de vias realimentação e algoritmos de estimação robustos levou a modelos de redes neurais recorrentes regularizadas. Dentre os modelos avaliados, a avariante M3-RLM-OD apresentou resultados bem consistentes e bastante promissores, sendo capaz de manter a norma do vetor de peso de saída pequena e, assim, reduzir a influência negativa dos outliers no desempenho de previsão iterativo de longo prazo da ESN.

Atualmente, está sendo ampliado os experimentos realizados nesse trabalho para a identificação *online* de sistemas dinâmicos de múltiplas entradas e múltiplas saídas.

REFERÊNCIAS

Atiya, A. and Parlos, A. (2000). New results on recurrent network training: unifying the algorithms and accelerating convergence. *IEEE Transactions on Neural Networks*, 11(3), 697–709. doi:10.1109/72.846741. URL <https://doi.org/10.1109/72.846741>.

Chan, S.C. and Zou, Y.X. (2004). A recursive least m-estimate algorithm for robust adaptive filtering in impulsive noise: Fast algorithm and convergence performance analysis. *IEEE Transactions on Signal Processing*, 52(4), 975–991. doi:10.1109/tsp.2004.823496. URL <https://doi.org/10.1109/tsp.2004.823496>.

Chatzis, S.P. and Demiris, Y. (2011). Echo state gaussian process. *IEEE Transactions on Neural Networks*, 22(9), 1435–1445. doi:10.1109/tnn.2011.2162109. URL <https://doi.org/10.1109/tnn.2011.2162109>.

De Moor, B., De Gersem, P., De Schutter, B., and Favoreel, W. (1997). Daisy: A database for identification of systems. *JOURNAL A*, 38, 4–5. URL <http://www.esat.kuleuven.ac.be/sista/daisy/>. [Used dataset: Dryer: (96-006); Exchanger (97-002); Robot arm (96-009).].

Guo, Y., Wang, F., Chen, B., and Xin, J. (2017). Robust echo state networks based on correntropy induced loss function. *Neurocomputing*, 267, 295–303. doi:10.1016/j.neucom.2017.05.087. URL <https://doi.org/10.1016/j.neucom.2017.05.087>.

Han, M. and Xu, M. (2018). Laplacian echo state network for multivariate time series prediction. *IEEE Tran-*

sactions on Neural Networks and Learning Systems, 29(1), 238–244. doi:10.1109/tnnls.2016.2574963. URL <https://doi.org/10.1109/tnnls.2016.2574963>.

Jaeger, H. (2001). The “echo state” approach to analysing and training recurrent neural networks-with an erratum note. *Bonn, Germany: German National Research Center for Information Technology GMD Technical Report*, 148(34), 13.

Jaeger, H. (2002). Adaptive nonlinear system identification with echo state networks. In *Proceedings of the 15th International Conference on Neural Information Processing Systems*, NIPS’02, 609–616. MIT Press, Cambridge, MA, USA. URL <http://dl.acm.org/citation.cfm?id=2968618.2968694>.

Li, D., Han, M., and Wang, J. (2012). Chaotic time series prediction based on a novel robust echo state network. *IEEE Transactions on Neural Networks and Learning Systems*, 23(5), 787–799. doi:10.1109/tnnls.2012.2188414. URL <https://doi.org/10.1109/tnnls.2012.2188414>.

Lukoševičius, M. and Jaeger, H. (2009). Reservoir computing approaches to recurrent neural network training. *Computer Science Review*, 3(3), 127–149. doi:10.1016/j.cosrev.2009.03.005. URL <https://doi.org/10.1016/j.cosrev.2009.03.005>.

Shi, Z. and Han, M. (2007). Support vector echo-state machine for chaotic time-series prediction. *IEEE Transactions on Neural Networks*, 18(2), 359–372. doi:10.1109/tnn.2006.885113. URL <https://doi.org/10.1109/tnn.2006.885113>.

Wigren, T. (2010). Input-output data sets for development and benchmarking in nonlinear identification. *Technical Reports from the department of Information Technology*, 20, 2010–020. URL <http://www.it.uu.se/research/publications/reports/2010-020/2010-020-nc.pdf>.

Zhang, C., Guo, Y., Wang, F., and Chen, B. (2018). Generalized maximum correntropy-based echo state network for robust nonlinear system identification. In *2018 International Joint Conference on Neural Networks (IJCNN)*. IEEE. doi:10.1109/ijcnn.2018.8489249. URL <https://doi.org/10.1109/ijcnn.2018.8489249>.

Zhou, H., Huang, J., Lu, F., Thiyaagalingam, J., and Kirubarajan, T. (2018). Echo state kernel recursive least squares algorithm for machine condition prediction. *Mechanical Systems and Signal Processing*, 111, 68–86. doi:10.1016/j.ymssp.2018.03.047. URL <https://doi.org/10.1016/j.ymssp.2018.03.047>.