



UNIVERSIDADE FEDERAL DE SANTA CATARINA
CENTRO TECNOLÓGICO
PROGRAMA DE PÓS-GRADUAÇÃO EM ENGENHARIA DE AUTOMAÇÃO E
SISTEMAS

Bernardo Barancelli Schwedersky

**Contribuições para o controle preditivo não linear prático com modelos identificados por
redes de estado de eco**

Florianópolis
2022

Bernardo Barancelli Schwedersky

Contribuições para o controle preditivo não linear prático com modelos identificados por redes de estado de eco

Tese submetida ao Programa de Pós-Graduação em Engenharia de Automação e Sistemas da Universidade Federal de Santa Catarina para a obtenção do título de doutor em Engenharia de Automação e Sistemas obtido pelo Programa de Pós-Graduação em Engenharia de Automação e Sistemas.

Orientador: Prof. Rodolfo César Costa Flesch, Dr. Eng.

Florianópolis

2022

Ficha de identificação da obra elaborada pelo autor,
através do Programa de Geração Automática da Biblioteca Universitária da UFSC.

Schwedersky, Bernardo Barancelli
Contribuições para o controle preditivo não linear
prático com modelos identificados por redes de estado de
eco / Bernardo Barancelli Schwedersky ; orientador,
Rodolfo César Costa Flesch, 2022.
148 p.

Tese (doutorado) - Universidade Federal de Santa
Catarina, Centro Tecnológico, Programa de Pós-Graduação em
Engenharia de Automação e Sistemas, Florianópolis, 2022.

Inclui referências.

1. Engenharia de Automação e Sistemas. 2. Controle
preditivo baseado em modelo. 3. Identificação de sistemas.
I. Flesch, Rodolfo César Costa. II. Universidade Federal
de Santa Catarina. Programa de Pós-Graduação em Engenharia
de Automação e Sistemas. III. Título.

Bernardo Barancelli Schwedersky

Contribuições para o controle preditivo não linear prático com modelos identificados por redes de estado de eco

O presente trabalho em nível de doutorado foi avaliado e aprovado por banca examinadora composta pelos seguintes membros:

Prof. Tito Luís Maia Santos, Dr. Eng.

Universidade Federal da Bahia

Prof. Ademar Gonçalves da Costa Junior, Dr. Eng.

Instituto Federal da Paraíba

Prof. Jomi Fred Hübner, Dr. Eng.

Universidade Federal de Santa Catarina

Certificamos que esta é a **versão original e final** do trabalho de conclusão que foi julgado adequado para obtenção do título de doutor em Engenharia de Automação e Sistemas obtido pelo Programa de Pós-Graduação em Engenharia de Automação e Sistemas.

Prof. Werner Kraus Junior, Dr. Eng.

Coordenação do Programa de Pós-Graduação

Prof. Rodolfo César Costa Flesch, Dr. Eng.

Orientador

Florianópolis, 2022.

AGRADECIMENTOS

Eu gostaria de agradecer a todos que contribuíram diretamente e indiretamente durante o desenvolvimento desta tese. Um agradecimento especial é dedicado ao meu orientador Rodolfo César Costa Flesch, cujo auxílio foi essencial em todas etapas de realização deste trabalho. Agradeço também a todos os colegas do LIAE, em especial ao Hiago Dangui, Ahryman Nascimento e Samuel Rovea, que contribuíram no desenvolvimento e foram essenciais para alcançar os objetivos da tese. Agradecimentos especiais são dedicados aos meus pais, Werner e Rosa, irmãos, Mateus e Maurício, pelo incentivo e apoio durante todo o curso de doutorado. Também é feito um agradecimento especial à minha companheira Patricia, que me acompanhou durante toda a jornada do curso de doutorado, fornecendo motivação e inspirando dedicação, sendo importante em todas etapas da realização deste trabalho. Por fim, agradeço à Universidade Federal de Santa Catarina e ao Programa de Pós-Graduação em Engenharia de Automação e Sistemas pela oportunidade de realizar este curso, às agências de fomento CNPq (140283/2018-8) e CAPES, e à empresa parceira Embraco (Nidec Global Appliance) pelo suporte financeiro.

RESUMO

Abordagens de controle preditivo não linear (NMPC) são uma alternativa de controle avançado cada vez mais aplicada na indústria, com abordagens NMPC práticas (PNMPC) sendo uma opção para a redução do seu custo computacional. Um modelo que apresenta potencial para aplicação como modelo de predição em algoritmos PNMPC é a rede de estado de eco (ESN), capaz de identificar processos não lineares com algoritmos computacionalmente eficientes. Considerando este contexto, nesta tese é apresentado um estudo acerca da aplicação de modelos baseados na ESN para identificação e utilização como modelo de predição em abordagens PNMPC. É proposta uma arquitetura de modelo baseada em uma rede ESN com parcela integradora (ESNI) e uma abordagem de identificação relevante para MPC (MRI), com formulações para obtenção dos parâmetros do modelo ESNI por meio de dados obtidos do processo e estimação on-line. Foram propostas abordagens PNMPC com modelos de predição baseados no modelo ESNI. Tais algoritmos utilizam o modelo ESNI completo para obtenção da resposta livre do processo e obtêm a resposta forçada a partir dos coeficientes da resposta ao degrau do modelo ESNI, que são obtidos a cada instante de amostragem por meio de uma simulação numérica. São propostas abordagens que utilizam modelos ESNI identificados a partir de dados do processo (PNMPC-ESNI) e abordagens adaptativas (APNMPC-ESNI), nas quais o modelo ESNI é identificado on-line. Adicionalmente, é proposta uma abordagem que busca aprimorar o desempenho do controlador PNMPC-ESNI utilizando um procedimento iterativo (PNMPCI-ESNI), com o qual a importância da porção linearizada no modelo de predição é reduzida. Para avaliar as abordagens de identificação e controle propostas, foram realizados estudos de caso com processos não lineares simulados e reais. Os resultados da abordagem de identificação MRI-ESNI indicam que a utilização da abordagem MRI resultou em modelos com desempenho melhor, para a tarefa de predição de múltiplos instantes à frente, quando comparado com modelos ESNI, treinados sem a abordagem MRI. Os resultados da abordagem de controle PNMPC-ESNI evidenciaram que o controlador proposto apresenta desempenho 16% superior aos controladores com modelos identificados. Por sua vez, a avaliação da abordagem adaptativa APNMPC-ESNI mostrou que o controlador APNMPC-ESNI é capaz de se adaptar a variações paramétricas do processo, alcançando desempenho em malha fechada superior a um NMPC, quando considerado um cenário de testes em que o processo é sujeito a variação paramétrica. Os resultados do controlador PNMPCI-ESNI indicaram que a abordagem iterativa é capaz de aprimorar o desempenho do controlador, com melhorias de até 17% para a métrica e o cenário testados. O custo computacional dos controladores propostos foi avaliado, com os tempos de cômputo associados ao tamanho do reservatório do modelo ESNI e também ao número de iterações do procedimento iterativo, no caso do PNMPCI-ESNI. Os tempos máximos foram compatíveis com os processos avaliados, sendo mais de uma ordem de grandeza menores que os verificados para os controladores NMPC usados como comparação.

Palavras-chave: Controle preditivo não linear prático. Identificação relevante para MPC. Rede de estado de eco. Controle preditivo adaptativo.

ABSTRACT

Nonlinear predictive control (NMPC) approaches are a type of advanced control strategy which has been increasingly applied in industry. The computational cost associated with solving the NMPC problem can be reduced by using practical NMPC (PNMPC) approaches. A model that has potential for application as a prediction model in PNMPC algorithms is the echo state network (ESN), which is capable of identifying nonlinear processes with computationally efficient algorithms. Considering this context, this thesis presents a study about the application of models based on the ESN for system identification and later use as a prediction model in PNMPC approaches. A model architecture based on an ESN network with an integrating portion (ESNI) is proposed and an MPC relevant identification approach (MRI), with formulations to obtain the parameters of the ESNI model through data obtained from the process and through on-line estimation, is considered. Additionally, PNMPC approaches with prediction models based on the ESNI model are proposed. Such algorithms use the complete ESNI model to obtain the free response of the process and obtain the forced response from the step response coefficients of the ESNI model, which are obtained at each sampling instant by means of a numerical simulation. Approaches that use ESNI models identified from process data (PNMPC-ESNI) and also adaptive approaches (APNMPC-ESNI), in which the ESNI model is identified online, are proposed. Additionally, an approach that seeks to improve the performance of the PNMPC-ESNI controller using an iterative procedure (PNMPCI-ESNI) is proposed. It relies on reducing the importance of the linearized portion in the prediction model, thus improving the prediction accuracy. To evaluate the proposed identification and control approaches, case studies were carried out with simulated and real non-linear processes. The results of the MRI-ESNI identification approach indicate that the use of the MRI strategy resulted in models that performed better for the multiple-step ahead prediction task when compared to a baseline of ESNI models trained without the MRI approach. The results of the PNMPC-ESNI control approach showed that the proposed controller presents a performance 16% better than controllers with identified models. In its way, the evaluation of the APNMPC-ESNI adaptive approach showed that the APNMPC-ESNI controller is capable to adapt to process parameter changes, achieving a better closed-loop performance than an NMPC, when a test scenario in which the process is subject to parametric variation is considered. The results of the PNMPCI-ESNI controller indicate that the iterative approach is able to improve the controller performance, with improvements of up to 17% for the tested metric and scenario. The computational cost is mainly associated with the ESNI model reservoir size and also with the number of iterations of the iterative procedure for the particular case of the PNMPCI-ESNI. The maximum computing times were compatible with the evaluated processes, being more than one order of magnitude smaller than those verified for the NMPC controllers used for comparison purposes.

Keywords: Practical nonlinear model predictive control. MPC relevant identification. Echo state network. Adaptive model predictive control.

LISTA DE FIGURAS

Figura 1 – Diagrama acerca dos tipos de controladores MPC, fidelidade do modelo e complexidade computacional.	28
Figura 2 – Diagrama de blocos de um sistema de controle adaptativo.	29
Figura 3 – Diagrama de blocos para a estratégia DMC estendida.	32
Figura 4 – Estratégia da dinâmica externa.	34
Figura 5 – Modelos nas configurações série-paralelo, usado para predição, e paralelo, usado para simulação.	34
Figura 6 – Arquiteturas recorrentes tradicionais.	38
Figura 7 – Arquitetura da rede NARX.	39
Figura 8 – Arquitetura da rede ESN.	40
Figura 9 – Diagrama de blocos simplificado do modelo ESNI.	49
Figura 10 – Diagrama de blocos para o procedimento de identificação de um processo utilizando a ESN.	50
Figura 11 – Comparação entre a curva característica estática de sintonias do modelo MRI-ESNI e modelo ESNI, com a curva do processo de neutralização de pH.	57
Figura 12 – Avaliação do efeito do tamanho do horizonte e parâmetro de regularização no MSE para a característica estática e para o ensaio de teste.	58
Figura 13 – Resultado da identificação do processo de neutralização de pH com ESN.	58
Figura 14 – Representação gráfica do impacto da sintonia da taxa de vazamento e do tamanho do reservatório no RMSE final do modelo para a tarefa de predição de múltiplos instantes à frente. O círculo vermelho representa a sintonia escolhida para o modelo final.	62
Figura 15 – Simulação do cenário para identificação on-line do tanque de perfil cônico. Resultados mostram o desempenho dos modelos para a tarefa de predição de um instante à frente.	63
Figura 16 – Simulação do cenário de predição de múltiplos instantes à frente.	64
Figura 17 – Evolução do desempenho (RMSE), para o cenário de estimação de múltiplos instantes à frente, dos modelos estimados a cada instante de tempo do cenário de estimação.	65
Figura 18 – Comparação da característica estática não linear do processo para os modelos MRI-ESNI, linear e ESN-RLS-FFF, considerando os parâmetros dos modelos estimados nos instantes de tempo $t = 10$ s, $t = 750$ s, $t = 1500$ s e $t = 2250$ s.	66
Figura 19 – Diagrama de blocos para a demonstração da garantia de erro nulo em regime permanente do controlador PNMPC-ESNI.	75
Figura 20 – Ilustração da estrutura de controle PNMPC-ESNI.	77

Figura 21 – Comparação do resultado obtido pelo controlador proposto utilizando o modelo ESN, ESNI com $D(z^{-1}) = 1$, ESNI com $D_{pb}(z^{-1})$ no qual $b = 0,5$ e ESNI com $D_{pa}(z^{-1})$ no qual $c = 5$	80
Figura 22 – Comparação dos resultados obtidos pelo controlador PNMPC-ESNI considerando sintonias distintas.	81
Figura 23 – Resultados do controle de um reator de neutralização de pH com algoritmo PNMPC-ESNI.	82
Figura 24 – Valores finais da métrica IAE em função da sintonia para o algoritmo PNMPC-ESNI comparado com PNMPC-HW, GPC e NMPC.	83
Figura 25 – Resultados do controle da bancada de ensaio de compressores de refrigeração.	85
Figura 26 – Evolução da MIAPE para o controle da bancada de ensaio de compressores.	86
Figura 27 – Diagrama de blocos da abordagem NMPC prática adaptativa. O modelo ESNI é adaptado utilizando a abordagem MRI, com os elementos de \mathbf{W}^{out} sendo estimados com o algoritmo RLS com fator de esquecimento direcional. O controlador utiliza, a cada período de amostragem, o modelo adaptado para obtenção da resposta livre e da matriz de dinâmica.	88
Figura 28 – Controle em malha fechada e parâmetros \mathbf{W}^{out} estimados.	91
Figura 29 – Avaliação da convergência do desempenho do controlador e da influência do tamanho do reservatório no desempenho.	92
Figura 30 – Resultados em malha fechada do tanque de perfil cônico.	93
Figura 31 – Avaliação da influência do tamanho do reservatório no desempenho do controlador para ω_m e i_d	97
Figura 32 – Resultados em malha fechada do estimador para o cenário de estimação.	98
Figura 33 – Resultados em malha fechada para o PMSM.	99
Figura 34 – Ilustração da estrutura de controle PNMPCI-ESNI.	106
Figura 35 – Desempenho em malha fechada do controlador PNMPCI-ESNI considerando cinco configurações para o número máximo de iterações (i_{max}).	108
Figura 36 – Impacto do número máximo de iterações no valor final da métrica IAE e no tempo máximo de cômputo.	109
Figura 37 – Impacto da sintonia de λ/δ e i_{max} no valor final do IAE.	110
Figura 38 – Resultados do controle de um reator de neutralização de pH com algoritmo PNMPCI-ESNI.	111
Figura 39 – Diagrama de processo e instrumentos de um processo de neutralização de pH.	139
Figura 40 – Resultados da identificação do processo de neutralização de pH para modelos linear (azul) e Hammerstein-Wiener (marrom).	142
Figura 41 – Comparação entre as curvas de característica estática dos modelos linear (azul), Hammerstein-Wiener (marrom) e a curva real do processo de neutralização de pH (preto).	142

Figura 42 – Diagrama de processo e instrumentos de uma bancada de controle de pressão de compressores de refrigeração.	143
Figura 43 – Resultado da identificação da bancada de ensaio de compressores herméticos com modelos linear e Hammerstein.	144
Figura 44 – Representação esquemática do sistema de controle de um tanque de perfil cônico.	145

LISTA DE TABELAS

Tabela 1	– Comparação das métricas de desempenho dos modelos linear, Hammerstein e ESN na tarefa de identificação do processo de neutralização de pH.	59
Tabela 2	– Resultados da identificação on-line do tanque de perfil cônico.	67
Tabela 3	– Comparação das métricas de desempenho para os controladores PNMPC-ESNI, GPC, PNMPC-HW e NMPC.	83
Tabela 4	– Comparação das métricas de desempenho para os controladores GPC, PNMPC-HW e PNMPC-ESNI.	86
Tabela 5	– Resultados do controle em malha fechada do tanque de perfil cônico.	95
Tabela 6	– Métricas de desempenho para os resultados em malha fechada dos controladores.	101
Tabela 7	– Comparação das métricas de desempenho para os controladores PNMPCI-ESNI e NMPC.	110
Tabela 8	– Condições nominais de operação do processo de neutralização de pH.	141
Tabela 9	– Condições de operação nominal do motor de indução com ímãs permanentes.	146

LISTA DE ABREVIATURAS E SIGLAS

AFF	fator de esquecimento adaptativo – do inglês <i>adaptive forgetting factor</i>
APNMPC	controle preditivo não linear prático baseado em modelo adaptativo – do inglês <i>adaptive practical nonlinear model predictive control</i>
APRBS	sinal binário pseudo-aleatório modulado em amplitude – do inglês <i>amplitude modulated pseudo-random binary signal</i>
BPDC	<i>backpropagation-decorrelation</i>
CE	esforço de controle – do inglês <i>control effort</i>
DMC	controle por matriz dinâmica – do inglês <i>dynamic matrix control</i>
EFK	filtro de Kalman estendido – do inglês <i>extended Kalman filter</i>
ELM	<i>extreme learning machine</i>
ESN	rede de estado de eco – do inglês <i>echo state network</i>
ESNI	rede de estado de eco com termo integrador – do inglês <i>echo state network with integral term</i>
FFF	fator de esquecimento fixo – do inglês <i>fixed forgetting factor</i>
GPC	controle preditivo generalizado – do inglês <i>generalized predictive control</i>
HW	<i>Hammerstein-Wiener</i>
IAE	integral dos erros absolutos – do inglês <i>integral of the average errors</i>
IAPE	erro de seguimento de referência percentual médio – do inglês <i>integral of the average percentual errors</i>
ITAE	integral dos erros absolutos ponderados no tempo – do inglês <i>integral of time absolute error</i>)
KF	filtro de Kalman – do inglês <i>Kalman filter</i>
LASSO	<i>least absolute shrinkage and selection operator</i>
LSM	máquina de estado líquido – do inglês <i>liquid state machine</i>
LSTM	memória de longo curto prazo – do inglês <i>long short-term memory</i>
MAPE	erro médio absoluto percentual – do inglês <i>mean absolute percentage error</i>
MIAPE	erro médio de seguimento de referência percentual médio – do inglês <i>mean integral of the average percentual errors</i>
MIMO	entradas múltiplas e saídas múltiplas – do inglês <i>multiple input and multiple output</i>
MLP	<i>perceptron</i> de múltiplas camadas – do inglês <i>multilayer perceptron</i>
MPC	controle preditivo baseado em modelo - do inglês <i>model predictive control</i>
MPC-NPL	controle preditivo baseado em modelo com linearização e predição não linear – do inglês <i>model predictive control with linearization and nonlinear prediction</i>
MRI	identificação relevante ao MPC – do inglês <i>MPC relevant identification</i>

MSAP	predição de múltiplos instantes à frente – do inglês <i>multi-step ahead prediction</i>
MSE	erro médio quadrático – do inglês <i>mean squared error</i>
NARX	modelo não linear, autorregressivo com entrada exógena – do inglês <i>nonlinear autoregressive model with exogenous input</i>
NEPSAC	<i>nonlinear extended prediction self-adaptive control</i>
NLP	programação não linear – do inglês <i>nonlinear programming</i>
NMPC	controle preditivo não linear baseado em modelo – do inglês <i>nonlinear model predictive control</i>
OSAP	predição de um instante à frente – do inglês <i>one-step ahead prediction</i>
PLS-PH	mínimos quadrados parciais com horizonte de predição – do inglês <i>partial least squares prediction horizon</i>
PMSM	motor síncrono de ímãs permanentes – do inglês <i>permanent magnet synchronous motor</i>
PNMPC	controle preditivo não linear prático baseado em modelo – do inglês <i>practical nonlinear model predictive control</i>
PRBS	sinal binário pseudo-aleatório modulado em amplitude – do inglês <i>pseudo-random binary signal</i>
QP	programação quadrática – do inglês <i>quadratic programming</i>
RBF	função de base radial – do inglês <i>radial basis function</i>
ReLU	<i>rectified linear unit</i>
RLS	método dos mínimos quadrados recursivo – do inglês <i>recursive least squares</i>
RMLP	<i>perceptron</i> de múltiplas camadas recorrente – do inglês <i>recurrent multi-layer perceptron</i>
RMSE	raiz do erro médio quadrático – do inglês <i>root mean squared error</i>
RNA	rede neural artificial
RNN	rede neural recorrente – do inglês <i>recurrent neural network</i>
RVFL	<i>random vector functional link</i>
SISO	entrada única e saída única – do inglês <i>single input and single output</i>
SMAPE	erro médio simétrico absoluto percentual – do inglês <i>Symmetric Mean Absolute Percentage Error</i>
SQP	programação quadrática sequencial – do inglês <i>sequential quadratic programming</i>
SRN	rede recorrente simples – do inglês <i>simple recurrent network</i>
TDL	linha de atrasos discretos – do inglês <i>tap delay line</i>
UFSC	Universidade Federal de Santa Catarina
ZOH	sustentador de ordem zero – do inglês <i>zero-order hold</i>

LISTA DE SÍMBOLOS

$a(k)$	realização de ruído branco
b	parâmetro de ajuste do filtro passa-baixa
b_k	bias do neurônio artificial k
c	parâmetro de ajuste do filtro passa-alta
c_{de}^{para}	taxa de conexão da matriz de pesos \mathbf{W}_{de}^{para}
d	perturbação
$D(z^{-1})$	função de transferência para a perturbação
e_p	erro de predição
$E[\cdot]$	operador esperança matemática
$f(\cdot)$	função de ativação dos neurônios da rede ESN
\hat{F}_j	filtro com coeficientes da resposta ao degrau para a j -ésima predição à frente
\mathbf{f}	vetor de resposta livre
$g(i)$	coeficientes da resposta ao impulso de \hat{H}
\mathbf{g}	vetor de coeficientes da resposta ao degrau
\mathbf{G}	matriz de dinâmica com coeficientes da resposta ao degrau
\mathbf{G}_{PNMPC}	matriz de dinâmica com coeficientes da resposta ao degrau para o algoritmo PNMPC
$h(t)$	nível do tanque cônico
H	altura do tanque cônico
$H(z^{-1})$	função de transferência para a perturbação com integrador
\mathbf{I}	matriz identidade
i_d	corrente do estator no referencial rotativo
i_q	corrente do estator no referencial rotativo
i_{max}	número máximo de iterações
J	função objetivo
J_{OSAP}	função objetivo baseada no erro de predição de um instante à frente
J_{MSAP}	função objetivo baseada no erro de predição de múltiplos instantes à frente
K_1	constante de dissociação primária
K_2	constante de dissociação secundária
m	número de entradas do processo
\mathbf{M}	matriz usada para verificação da propriedade de estado de eco em modelos com vazamento
n	número de saídas do processo
n_x	número de atrasos relacionados à entrada
n_y	número de atrasos relacionados à saída
N	tamanho do horizonte de predição
N_1	início do horizonte de predição

N_2	final do horizonte de predição
N_u	tamanho do horizonte de controle
$p(\cdot)$	função de ativação do neurônio artificial
P	matriz de covariância
P_p	número de par de polos
q^{-1}	filtro que representa um atraso simples
q_i	vazão da variável i
r	número de estados do reservatório
$r(\cdot)$	função que mapeia as entradas no estado interno do modelo
s	coeficiente da resposta ao degrau
$s(\cdot)$	função que mapeia o estado interno do modelo na saída
r	valor de referência para a saída do processo em tempo contínuo
R	raio do tanque cônico
R_s	resistência de enrolamento
T	número de amostras para treinamento
T_{el}	torque de eletromagnético
T_l	torque de carga
T_s	passo de discretização
u	sinal de controle em tempo contínuo
$u(k)$	sinal de controle em tempo discreto
u_{base}	vetor de entrada para obtenção da resposta base
u_{livre}	vetor de entrada para obtenção da resposta livre
$u_{forçada}$	vetor de entrada para obtenção da resposta forçada
$u_{ótima}$	vetor de entrada para obtenção da resposta ótima
u_d	tensão do estator no referencial rotativo
u_q	tensão do estator no referencial rotativo
u(k)	vetor de sinais de controle em tempo discreto
u_e	vetor de entradas do modelo ESN
$\Delta u(k)$	incremento do sinal de controle
$\Delta \mathbf{u}(k)$	vetor de incrementos do sinal de controle
v_{de}^{para}	escalamento da matriz de pesos \mathbf{W}_{de}^{para}
v_k	soma das entradas ponderadas e bias do neurônio artificial k
V	volume do reator de neutralização
w_k	peso sináptico do neurônio artificial k
\mathbf{W}_{de}^{para}	matriz de pesos sinápticos que conecta duas camadas do modelo ESN
\mathbf{W}^{out}	matriz de pesos sinápticos associados à camada de saída do modelo ESN
x	estado interno do modelo ESN em tempo contínuo
\dot{x}	derivada do estado interno do modelo ESN em tempo contínuo
x_k	entrada do neurônio artificial k

\mathbf{x}_e	vetor de estados do reservatório
\mathbf{x}_{base}	vetor de estados do reservatório usados para obter a resposta base
$\mathbf{x}_{forçada}$	vetor de estados do reservatório usados para obter a resposta forçada
\mathbf{x}_l	vetor de estados do reservatório filtrados
\mathbf{x}_{livre}	vetor de estados do reservatório usados para obter a resposta livre
$\mathbf{x}_{ótima}$	vetor de estados do reservatório usados para obter a resposta ótima
\mathbf{X}_e	matriz formada pelos vetores de estados do reservatório, em um período de T amostras
\mathbf{X}_l	matriz formada pelos vetores de estados do reservatório filtrados, em um período de T amostras
\mathbf{y}	resposta temporal do processo em tempo contínuo
$y(k)$	resposta temporal do processo em tempo discreto
$\hat{y}(k)$	predição do modelo para o valor da saída no instante k
$\hat{\mathbf{y}}$	vetor de predições do modelo para o valor da saída
\mathbf{Y}	matriz formada pelo vetor de saídas reais em um período de T amostras
\mathbf{y}_{base}	vetor de predição da resposta base
\mathbf{y}_e	vetor de saídas do modelo ESN
$\mathbf{y}_{forçada}$	vetor de predição da resposta forçada
\mathbf{y}_l	vetor de saídas filtradas do modelo ESN
\mathbf{y}_{livre}	vetor de predição da resposta livre
$\mathbf{y}_{ótima}$	vetor de predição da resposta ótima
\mathbf{Y}_e	matriz formada pelos vetores de saídas em um período de T amostras
\mathbf{Y}_l	matriz formada pelos vetores de saídas filtradas em um período de T amostras
$z(\cdot)$	função não linear estática
α	taxa de vazamento do reservatório
β	coeficiente de regularização da função objetivo
ϵ	erro de predição
δ	ponderação para o erro de seguimento de referência
Δ	termo que representa a primeira diferença
λ	ponderação para a variação do sinal de controle
$\rho(\cdot)$	raio espectral
$\Phi(\omega)$	espectro de potência
φ	fator de esquecimento
$\sigma(\cdot)$	função sigmoide
Ψ_{PM}	constante de força eletromotriz
ω_m	velocidade angular do rotor

SUMÁRIO

1	INTRODUÇÃO	20
1.1	Pesquisas relacionadas	22
1.2	Problema de pesquisa	24
1.2.1	Objetivo geral	25
1.2.2	Objetivos específicos	25
1.3	Estrutura do documento	25
2	REVISÃO DA LITERATURA	26
2.1	Introdução	26
2.2	Controle preditivo não linear prático	26
2.3	Controle preditivo não linear adaptativo	29
2.4	Controle preditivo não linear com redes neurais artificiais	31
2.5	Identificação de sistemas com redes neurais recorrentes	33
2.5.1	Redes neurais recorrentes convencionais	36
2.5.2	Rede de estado de eco	39
2.6	Identificação relevante para MPC	42
2.7	Comentários finais	44
3	IDENTIFICAÇÃO RELEVANTE PARA MPC COM ESN	46
3.1	Introdução	46
3.2	Modelo não linear baseado na rede de estado de eco	48
3.3	Formulação do problema de identificação relevante para MPC	50
3.4	Identificação do modelo ESNI offline	54
3.4.1	Treinamento da ESNI usando o método de regressão com regularização	54
3.4.2	Exemplo 1: Identificação de um processo de neutralização de pH com ESN	56
3.5	Identificação do modelo ESNI on-line	59
3.5.1	Treinamento do mecanismo de leitura do reservatório por método dos mínimos quadrados recursivo com fator de esquecimento direcional	60
3.5.2	Exemplo 2: Identificação de um tanque de perfil cônico	61
3.6	Comentários Finais	68
4	ALGORITMOS NMPC PRÁTICOS COM REDES DE ESTADO DE ECO	69
4.1	Introdução	69
4.2	Formulação do problema NMPC	70
4.3	Algoritmo NMPC prático para modelos ESNI	73
4.3.1	Predição não linear e linearização	73
4.3.2	Exemplo 3: Controle de um reator de neutralização de pH com PNMPC-ESNI	79

4.3.3	Exemplo 4: Controle de uma bancada de ensaio de compressores her- méticos com PNMPC-ESNI	84
4.4	Algoritmo NMPC prático adaptativo para modelos ESNI	86
4.4.1	Estimação on-line do modelo ESNI e integração com PNMPC	87
4.4.2	Exemplo 5: Controle adaptativo de um tanque cônico	88
4.4.2.1	Resultados da estimação em malha fechada	90
4.4.2.2	Resultados do controle em malha fechada	92
4.4.3	Exemplo 6: Controle adaptativo de um motor síncrono de ímãs perma- nentes	94
4.4.3.1	Resultados da estimação em malha fechada	97
4.4.3.2	Resultados do controle em malha fechada	98
4.5	Algoritmo NMPC prático com predição iterativa para modelos ESNI . . .	102
4.5.1	Predição iterativa com respostas base e ótima	103
4.5.2	Exemplo 7: Controle de um processo de neutralização de pH	108
4.6	Comentários Finais	112
5	CONSIDERAÇÕES FINAIS	114
5.1	Conclusões	114
5.2	Sugestões para trabalhos futuros	116
5.3	Trabalhos publicados	117
	REFERÊNCIAS	120
	APÊNDICE A – DESCRIÇÃO DOS PROCESSOS EM ESTUDO	138
A.1	Reator de neutralização de pH	138
A.1.1	Identificação do processo de neutralização de pH	140
A.2	Bancada de controle de pressões	142
A.2.1	Identificação da bancada de controle de pressões	143
A.3	Tanque de perfil cônico	144
A.4	Motor síncrono de ímãs permanentes	145
	APÊNDICE B – MÉTRICAS DE DESEMPENHO	147
B.1	Métricas de desempenho para modelos	147
B.2	Métricas de desempenho para controladores	147

1 INTRODUÇÃO

Um dos paradigmas de controle avançado que se encontra bem estabelecido e conta com boa aceitação na indústria é o controle preditivo baseado em modelo (MPC, do inglês *Model Predictive Control*). De forma geral, as abordagens mais aplicadas são as baseadas em representações com modelos de predição lineares, como o controle preditivo generalizado (GPC, do inglês *Generalized Predictive Control*), proposto em Clarke *et al.* (1987), o controle por matriz dinâmica (DMC, do inglês *Dynamic Matrix Control*), proposto em Cutler e Ramaker (1980), e as abordagens MPC no espaço de estados (LEE *et al.*, 1994). Esse paradigma possui grande aplicação em processos da indústria petroquímica desde sua proposição e vem sendo adotado, com o passar do tempo, em outros domínios de aplicação (QIN; BADGWELL, 2003; CAMACHO; BORDONS, 2007; FORBES *et al.*, 2015).

A utilização de modelos de predição não lineares, que resultam em um problema de controle preditivo não linear (NMPC, do inglês *Nonlinear Model Predictive Control*), ainda não é tão comum. Esse tipo de abordagem pode ser especialmente útil em certas ocasiões, como em processos altamente não lineares. Processos moderadamente não lineares que enfrentam amplas mudanças em sua região de operação consistem também em exemplos em que o NMPC apresenta grande potencial de aplicação. Por fim, processos em que o controlador opera próximo a condições singulares, por exemplo a mudança no sinal do ganho ou a multiplicidade na entrada, são casos em que o uso de abordagens NMPC pode trazer benefícios claros frente à aplicação de controladores lineares (HENSON, 1998; BIEGLER, 2000).

Vários aspectos fazem com que a utilização de abordagens NMPC seja desafiadora. A principal delas, a qual surge diretamente da não linearidade do processo, consiste na solução on-line de um problema de programação não linear (NLP, do inglês *NonLinear Programming*), ao menos em parte, o qual não é convexo e requer algoritmos complexos para resolução. A solução on-line de um NLP resulta em um custo computacional elevado sem que haja garantia de otimalidade global e tempo de convergência do algoritmo. Outro desafio que faz com que alternativas não lineares não sejam tão exploradas é dificuldade associada à identificação de modelos não lineares (HENSON, 1998; MAYNE, 2000; BIEGLER, 2000; CAMACHO; BORDONS, 2007).

Alguns métodos para evitar a necessidade de resolução de um NLP em uma abordagem NMPC são encontrados na literatura. Uma forma comum consiste em obter uma versão em que parte da formulação NMPC utiliza versões linearizadas do modelo não linear, fazendo com que o problema de otimização resultante seja de programação quadrática (QP, do inglês *Quadratic Programming*). Em geral, tais métodos apresentam um compromisso entre a fidelidade da aproximação do modelo de predição comparado ao modelo não linear e o custo computacional, associado à obtenção dessas predições e à resolução do problema de otimização. Esse tipo de abordagem é referenciado na literatura como abordagem MPC prática ou computacionalmente eficiente (PLUCENIO, 2010; ŁAWRYŃCZUK, 2014).

A estratégia geralmente utilizada para a identificação de sistemas dinâmicos consiste na

estimação dos parâmetros do modelo por meio da otimização de uma função custo do erro de predição de um instante à frente (LJUNG, 1999). Apesar de ser a abordagem mais comum e produzir resultados suficientemente bons na maior parte dos casos, ela não é a escolha ideal quando o modelo será utilizado para predição em um MPC. O modelo utilizado em um MPC deve apresentar um desempenho bom para a predição de múltiplos instantes à frente, pois o modelo é utilizado no MPC para prever a evolução do comportamento futuro do processo considerando um horizonte futuro (GOPALUNI *et al.*, 2003; GOPALUNI *et al.*, 2004). Na maior parte dos casos, o tamanho do horizonte de predição usado pelo MPC é maior que um, fazendo com que abordagens de identificação de sistemas que se baseiam na otimização de funções do erro de predição de múltiplos instantes à frente sejam escolhas mais apropriadas (GOPALUNI *et al.*, 2004; CAMACHO; BORDONS, 2007). A família de algoritmos que otimizam o modelo para a predição de múltiplos instantes à frente é conhecida como identificação relevante ao MPC (MRI, do inglês *MPC Relevant Identification*).

Além do desafio associado à solução do problema MRI, a obtenção de um modelo não linear para utilização em uma abordagem NMPC traz diversos outros desafios. Uma opção comum consiste na utilização de modelos fenomenológicos, baseados nas equações fundamentais do processo. Essa alternativa é interessante por resultar em um modelo globalmente válido, característica desejada para o controle. Porém, a dificuldade associada à dedução de tais modelos consiste em um grande problema. Os modelos empíricos são uma alternativa aos modelos fenomenológicos, transformando dados acerca da entrada e da saída de um processo em uma relação entre a entrada e a saída, que pode ser usada para prever comportamentos futuros (LEE, 2000; FOSS; SCHEI, 2007).

Uma forma frequentemente explorada na literatura para obtenção de modelos não lineares empíricos consiste no uso de redes neurais artificiais (RNAs). A identificação por meio de RNAs surgiu no início da década de 1990, após a apresentação de estratégias para criar e treinar arquiteturas capazes de realizar processamento temporal, por Werbos (1990). De forma geral, os trabalhos presentes na literatura exploram, em especial, uma classe de arquiteturas de redes neurais artificiais. Essa classe estende o modelo *perceptron* de múltiplas camadas (MLP, do inglês *MultiLayer Perceptron*), modelo mais popular de RNA para problemas estáticos, para que ele possa tratar problemas dinâmicos. Tais arquiteturas apresentam conexões recorrentes, podendo ser empregadas para modelar problemas com contexto temporal, inclusive sistemas dinâmicos não lineares. Essa classe de arquiteturas apresenta diversos modelos distintos, desde versões simplificadas como as apresentadas por Jordan (1986) e Elman (1990), até versões genéricas, que possibilitam o treinamento dos pesos de realimentação, caso da rede não linear autorregressiva com entradas exógenas (NARX, do inglês *Nonlinear AutoRegressive with eXogenous inputs*) e da rede *perceptron* de múltiplas camadas recorrente (RMLP, do inglês *Recurrent MultiLayer Perceptron*) (NARENDRA; PARTHASARATHY, 1990; PARLOS *et al.*, 1991; PUSKORIUS; FELDKAMP, 1994).

A aplicação dessa classe de RNAs apresentou resultados promissores, mas que foram,

de certa forma, limitados por algumas características indesejadas, como o alto tempo do treinamento e os fenômenos do desaparecimento e explosão do gradiente, os quais afetam redes MLP com diversas camadas escondidas e também redes recorrentes derivadas do modelo MLP. Tal problema decorre da utilização do algoritmo da retropropagação do erro (do inglês, *error backpropagation*) para o treinamento da rede, o qual é a alternativa comumente aplicada na literatura. Esse algoritmo propaga os sinais de erro da saída para a entrada, os quais decaem exponencialmente, fazendo com que o erro nas camadas iniciais possa desaparecer ou crescer indefinidamente, em casos que a rede é profunda (SCHMIDHUBER, 2015; HOCHREITER, 1998).

Pesquisas recentes exploraram estratégias para o treinamento eficiente de redes recorrentes, por meio de paradigmas alternativos para o aprendizado de problemas dinâmicos. Um desses paradigmas é chamado de computação de reservatório. A ideia compartilhada pelas arquiteturas de RNAs desse paradigma consiste em eliminar muitos dos problemas encontrados no treinamento de modelos MLP recorrentes, como o desaparecimento do gradiente, por meio de abordagens em que a parte dinâmica do modelo não é ajustada durante o treinamento. As principais arquiteturas que se aproveitam desse paradigma são a rede de estado de eco (ESN, do inglês *Echo-State Network*), proposta em Jaeger (2002), e a máquina de estado líquido (LSM, do inglês *Liquid State Machine*). No contexto da identificação de sistema, várias formulações de ESNs já foram consideradas para identificação utilizando dados adquiridos da operação do processo (DAI *et al.*, 2009; ANTONELLO *et al.*, 2017; WANG *et al.*, 2020). Existem também abordagens que estimam on-line os parâmetros do modelo ESN, utilizando, primariamente, algoritmos baseados no método dos mínimos quadrados recursivo (RLS, do inglês *Recursive Least Squares*) (JAEGER, 2003; SUSSILLO; ABBOTT, 2009; HUANG *et al.*, 2016; YANG *et al.*, 2019).

1.1 PESQUISAS RELACIONADAS

Existem trabalhos na literatura que abordam a obtenção de abordagens NMPC práticas de forma geral. Uma referência importante é o trabalho apresentado em Plucenio *et al.* (2007), complementado em Plucenio (2010), o qual apresenta uma abordagem para obtenção de controladores NMPC práticos (PNMPC, do inglês *Practical Nonlinear MPC*) com uma formulação geral, apesar de originalmente a abordagem ter sido apresentada considerando modelos fenomenológicos não lineares.

Quanto à aplicação de abordagens com arquiteturas de redes neurais em MPC, existem variadas fontes na literatura abordando tal tema. Existem abordagens que utilizam RNAs desempenhando o papel do otimizador, para a solução do problema de otimização resultante (YAN; WANG, 2012a; LI *et al.*, 2015; XU *et al.*, 2016; WANG *et al.*, 2020), e também abordagens que buscam sintetizar o comportamento geral de um controlador MPC utilizando uma RNA, a qual substitui o controlador após aprender seu funcionamento (PARISINI; ZOPPOLI, 1995; DRGOÑA *et al.*, 2018; HERTNECK *et al.*, 2018; CHEN *et al.*, 2022). Porém, grande parte dos

trabalhos explora estratégias em que a RNA desempenha o papel do modelo de predição, sempre apresentando grande similaridade com os trabalhos precursores de Narendra e Parthasarathy (1990) e Psychogios e Ungar (1990), os quais apresentaram abordagens para a identificação de sistemas não lineares e o MPC com arquiteturas de redes neurais, respectivamente.

Alguns trabalhos, como o de Ławryńczuk (2007), lidaram com a resolução do problema de otimização resultante do NMPC, propondo abordagens práticas. O trabalho apresentado em da Cruz *et al.* (2013) também explora uma abordagem NMPC prática, porém considera um modelo neural, para o controle de um sistema de quatro tanques. Existem também estratégias, como a apresentada em Jin e Su (2008), para a utilização de RNAs recorrentes em abordagens MPC adaptativas. Trabalhos recentes abordam ainda algumas arquiteturas clássicas de RNA, como em Wysocki e Ławryńczuk (2015), no qual a rede de Jordan, uma das primeiras arquiteturas recorrentes, é aplicada. São encontrados na literatura, também, trabalhos que abordam variadas aplicações, como em Yan e Wang (2012b), no qual um MPC com RNA é aplicado para o controle da trajetória de embarcações, e o trabalho de Demmers *et al.* (2018), no qual uma abordagem de MPC com RNA foi usada para controlar o crescimento de aves e suínos. Existem também trabalhos recentes que utilizam a rede com memória do longo curto prazo (LSTM, do inglês *Long Short-Term Memory*) como modelo de predição, com destaque para as abordagens apresentados em Schwedersky *et al.* (2019), que propõe um algoritmo NMPC prático, e a apresentada em Karimanzira e Rauschenbach (2020), que propõe uma estratégia NMPC para o controle de uma planta simulada de dessalinização por osmose reversa. Outros trabalhos que utilizam as LSTMs como modelo de predição são encontrados em Jeon e Kim (2021), Zarzycki e Ławryńczuk (2021) e Terzi *et al.* (2021). Com exceção dos trabalhos apresentados em da Cruz *et al.* (2013) e Schwedersky *et al.* (2019), que propõe abordagens NMPC práticas, todas as demais se baseiam na solução direta do problema NLP.

Por sua vez, a utilização de arquiteturas de RNA recorrentes baseadas em computação de reservatório em abordagens MPC ainda é incipiente. Uma das primeiras abordagens foi apresentada em Pan e Wang (2012), na qual um modelo baseado em ESN é usado para predição em um MPC, enquanto uma segunda ESN foi utilizada para solução do problema de otimização, considerando uma abordagem neurodinâmica. Em tal trabalho, a predição é obtida por meio de uma expansão em série de Taylor, específica para a arquitetura utilizada. Por sua vez, o trabalho de Jordanou *et al.* (2018) aplicou uma abordagem semelhante, na qual o modelo é identificado por meio de uma ESN, a qual é usada no MPC para predição da saída, aplicando, novamente, uma expansão em série de Taylor. Esta é uma formulação no espaço de estados, na qual o modelo ESN sofre uma linearização sucessiva, a cada período de amostragem, considerando os coeficientes da série de Taylor truncada para montar uma formulação NMPC prática. Por fim, em Armenio *et al.* (2019), uma formulação NMPC, no espaço de estados, foi proposta. Nesse trabalho, foi apresentada uma condição suficiente para garantia incremental da estabilidade entre entrada e estados do modelo ESN.

Quanto à obtenção de modelo em métodos de identificação MRI, as abordagens mais co-

munas são baseadas no método proposto em Shook *et al.* (1992), no qual os parâmetros do modelo são estimados utilizando um método baseado no erro de predição de um instante à frente associado à filtragem dos dados de entrada e de saída do processo. Tais abordagens são geralmente implementadas considerando modelos lineares em seus parâmetros (GOPALUNI *et al.*, 2003). Existem algumas abordagens MRI para estruturas de modelos não lineares, como por exemplo modelos NARX (QUACHIO; GARCIA, 2014), Hammerstein (QUACHIO; GARCIA, 2017), Wiener (QUACHIO; GARCIA, 2019), e modelos provenientes da modelagem fenomenológica (PČOLKA *et al.*, 2016).

1.2 PROBLEMA DE PESQUISA

A obtenção de modelos não lineares identificados e com alta fidelidade para uso como modelo de predição em controladores preditivos ainda é uma área com diversos avanços a serem alcançados. As técnicas exploradas até então apresentam, tipicamente, algumas limitações, podendo ser baseadas em modelos que requerem um alto conhecimento do processo ou classes de modelos específicas para alguns tipos de comportamentos não lineares. As abordagens que utilizam RNAs recorrentes para tal fim já são exploradas na literatura, porém geralmente são aplicados modelos baseados em arquiteturas recorrentes tradicionais, apesar da ascensão de arquiteturas recorrentes modernas, as quais já mostraram vantagens em outros domínios de aplicação (LI *et al.*, 2012; SUNDERMEYER *et al.*, 2012; GRAVES *et al.*, 2013). Quando utilizadas arquiteturas baseadas na computação de reservatório, são geralmente exploradas abordagens para problemas específicos, focadas na aplicação. Existem oportunidades associadas à utilização de ESNs para identificação de sistemas e posterior utilização como modelo de predição em abordagens NMPC. Principalmente, existe uma carência na literatura de formulação para identificação de sistemas relevante ao MPC, considerando modelos baseados na ESN, a qual pode trazer vantagens em relação às técnicas de identificação tradicionalmente aplicadas para a ESN.

A formulação de estratégias NMPC usando modelos baseado na ESN para predição também é um campo de estudos com desafios a serem suplantados. Existem na literatura formulações NMPC que aplicam esse tipo de modelo para obtenção de predições, porém tais algoritmos apresentam limitações, seja relacionada com a fidelidade entre o modelo e a predição usada no controle, seja com a necessidade da resolução de um problema de otimização não linear. De forma geral, as formulações apresentadas na literatura são baseadas em abordagens no espaço de estados, existindo uma oportunidade de pesquisa em abordagens de NMPC práticas baseadas em formulações clássicas, de forma semelhante à abordagem proposta em Plucenio (2010). Existem também oportunidades para o estudo de formas para redução do impacto da linearização do modelo ESN, de modo a melhorar a qualidade das predições não lineares do NMPC prático. Outra oportunidade ainda inexplorada está relacionada à maior vantagem dos modelos baseados na computação de reservatório, que é baixa complexidade computacional associada ao seu treinamento, o que pode ser uma vantagem para a síntese de algoritmos NMPC

práticos adaptativos.

1.2.1 Objetivo geral

O objetivo geral desta tese consiste em desenvolver abordagens para identificação de sistemas e controle preditivo não linear prático baseados em modelos obtidos por redes de estado de eco.

1.2.2 Objetivos específicos

Os objetivos específicos desta tese consistem em:

- pesquisar na literatura abordagens para identificação de sistemas dinâmicos não lineares baseadas em ESNs e técnicas de controle NMPC prático;
- propor abordagens para identificação de sistemas dinâmicos relevante ao MPC baseadas em ESN;
- propor algoritmos de controle NMPC práticos para ESNs;
- propor algoritmos de controle NMPC práticos adaptativos para ESNs;
- avaliar experimentalmente as abordagens de identificação e controle propostas.

1.3 ESTRUTURA DO DOCUMENTO

No Capítulo 2 é apresentada uma revisão da literatura acerca de técnicas para identificação de sistema utilizando ESN, técnicas de identificação relevante ao MPC, algoritmos de controle NMPC práticos e algoritmos NMPC adaptativos. Em seguida, no Capítulo 3 é proposta uma abordagem para identificação de sistemas relevante ao MPC, considerando modelos baseados na rede ESN. Em seguida, no Capítulo 4, são propostas abordagens para o controle NMPC prático considerando modelos ESN. Por fim, as considerações finais deste trabalho são apresentadas no Capítulo 5.

2 REVISÃO DA LITERATURA

Sumário

2.1	Introdução	26
2.2	Controle preditivo não linear prático	26
2.3	Controle preditivo não linear adaptativo	29
2.4	Controle preditivo não linear com redes neurais artificiais	31
2.5	Identificação de sistemas com redes neurais recorrentes	33
2.5.1	Redes neurais recorrentes convencionais	36
2.5.2	Rede de estado de eco	39
2.6	Identificação relevante para MPC	42
2.7	Comentários finais	44

2.1 INTRODUÇÃO

Este capítulo apresenta uma revisão da literatura, cobrindo os temas essenciais que são abordados nesta tese. O primeiro assunto abordado é o controle preditivo não linear prático, o qual é discutido na Seção 2.2. Em seguida, são apresentadas as principais referências da literatura acerca do controle preditivo baseado em modelo adaptativo. Na Seção 2.4 são exploradas as referências que abordam algoritmos MPC com RNAs em sua formulação. Em seguida, são detalhados os métodos para a identificação de sistemas dinâmicos não lineares com RNAs recorrentes, os quais são detalhados na Seção 2.5. As técnicas de identificação de sistemas relevante para controle preditivo são apresentadas na Seção 2.6. Uma discussão acerca das abordagens de identificação e controle abordadas neste capítulo é apresentada na Seção 3.6. Por fim, na Seção 2.7 são apresentados alguns comentários finais acerca da revisão da literatura realizada.

2.2 CONTROLE PREDITIVO NÃO LINEAR PRÁTICO

O controle de processos dinâmicos consiste em estratégias usadas para levar a saída de um processo, $y \in \mathbb{R}^m$, a qual é sujeita a perturbações, $d \in \mathbb{R}^o$, para um valor desejado de referência, $r \in \mathbb{R}^m$, por meio da variação de um sinal de controle, $u \in \mathbb{R}^n$. Na teoria de controle preditivo baseado em modelo, um modelo explícito do processo é utilizado para a formulação de um problema de otimização cujo objetivo consiste em encontrar uma sequência de incrementos de controle futuros que minimize uma dada função custo definida pelo projetista. Tipicamente, o problema de otimização é construído considerando uma função custo que considera uma soma ponderada de critérios: uma parcela que penaliza o erro de seguimento de referência; e outra que penaliza a variação do sinal de controle. Considerando a predição da saída para o instante

$k + N$, tomada em k , como $\hat{y}(k + N | k)$, para o caso com entrada única e saída única (SISO, do inglês *Single Input and Single Output*), o vetor de predições futuras é definido como

$$\hat{\mathbf{y}} = \left[\hat{y}(k + N_1 | k) \quad \dots \quad \hat{y}(k + N_2 | k) \right]^T, \quad (1)$$

ao longo de um horizonte de predição $N = N_2 - N_1$. Já o vetor de incrementos de controle futuros é definido como

$$\Delta \mathbf{u} = \left[\Delta u(k | k) \quad \dots \quad \Delta u(k + N_u - 1 | k) \right]^T, \quad (2)$$

ao longo de um horizonte de predição N_u . Considerando um vetor de referências futuras \mathbf{r} , definido com a mesma estrutura do vetor de predições, o problema de otimização pode ser definido como (CAMACHO; BORDONS, 2007)

$$\min_{\Delta \mathbf{u}} J = \sum_{i=N_1}^{N_2} \delta(i) [\hat{y}(k + i | k) - r(k + i)]^2 + \sum_{j=1}^{N_u} \lambda(j) [\Delta u(k + j - 1)]^2, \quad (3)$$

no qual δ e λ representam as ponderações para o erro e esforço de controle, respectivamente.

Considerando que o modelo usado para predição é não linear, o problema de otimização resultante é, geralmente, não convexo. Segundo Mayne (2000), esse é um dos grandes desafios para o desenvolvimento do controle preditivo não linear, pois a resolução de tal problema traz limitações, especialmente em exemplos multivariáveis com diversas variáveis controladas e de processo. Como o problema de otimização é não convexo, a solução desse problema resulta em uma demanda computacional que cresce exponencialmente com o número de variáveis de decisão, dificultando a resolução on-line do problema. O segundo grande desafio apontado por Mayne (2000) consiste na necessidade do desenvolvimento de versões robustas, as quais devem considerar as inevitáveis incertezas decorrentes dos erros de modelagem ou perturbações não medidas, área que têm sido bastante explorada na literatura (ZHENG, 2000).

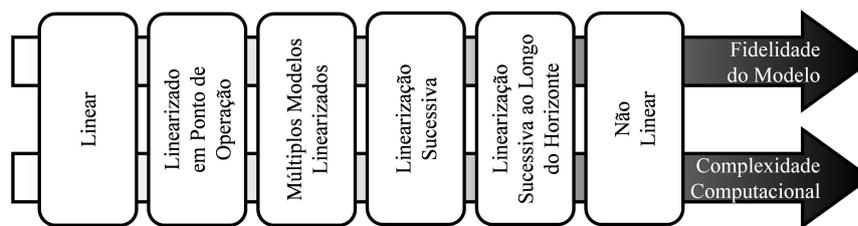
A literatura apresenta algumas alternativas para lidar com a não convexidade resultante da não linearidade do modelo. A alternativa comumente aplicada consiste na linearização do modelo não linear, a qual pode ser realizada por diversos tipos de métodos. A forma mais trivial consiste em métodos com linearização em volta de pontos de operação. Essa variante engloba diversos métodos, desde os que consideram apenas um modelo linearizado estático, até métodos que utilizam múltiplos modelos lineares. O trabalho de Townsend *et al.* (1998) é um exemplo do emprego de múltiplos modelos locais, pois apresenta uma implementação de DMC (CUTLER; RAMAKER, 1980) que utiliza um modelo global formado por um conjunto de submodelos localmente válidos, obtidos por meio da identificação do processo não linear.

Alternativas mais complexas consistem nos métodos com linearização sucessiva e ao longo da trajetória (GARCIA, 1984; GATTU; ZAFIRIOU, 1992; LEE; RICKER, 1993; GATTU; ZAFIRIOU, 1995; DE OLIVEIRA *et al.*, 1995; KURTZ; HENSON, 1997; ZHENG, 1997; SEKI *et al.*, 2002). Tais métodos consistem em abordagens que utilizam o modelo não linear para obter versões linearizadas considerando a trajetória predita do processo. Essa estratégia resulta, geralmente, em uma representação mais fiel do modelo não linear, especialmente em casos com

severa não linearidade, podendo considerar, inclusive, o comportamento não linear ao longo do horizonte de predição. Um dos primeiros exemplos do emprego desse tipo de estratégia é apresentado no trabalho de Garcia (1984), o qual apresenta um algoritmo em que o modelo não linear do processo é linearizado sucessivamente, à medida que o estado do processo muda, sendo usado para obter os coeficiente das resposta ao degrau, em uma abordagem que estende o algoritmo DMC.

Na Figura 1 é apresentada uma representação dessas classes de algoritmos MPC, evidenciando a proximidade dos métodos apresentados com as implementações puramente lineares ou não lineares. Com base nisso, é evidenciado quais métodos utilizam modelo de predição com maior fidelidade, e também é explicitada a variação da complexidade computacional dos vários métodos.

Figura 1 – Diagrama acerca dos tipos de controladores MPC, fidelidade do modelo e complexidade computacional.



Fonte: elaborado pelo autor.

Os métodos que aplicam linearização sucessiva e ao longo da trajetória resultam em algoritmos com modelo de predição mais fiéis ao modelo não linear. Um algoritmo que utiliza esse método de linearização foi apresentado por Plucenio *et al.* (2007), denominado MPC Prático para Sistemas Não Lineares (PNMPC).

No algoritmo PNMPC, para o caso SISO, o vetor de predições futuras é obtido de forma semelhante ao que ocorre no algoritmo DMC (CUTLER; RAMAKER, 1980), em função dos incrementos futuros de controle, sendo descrito por

$$\hat{\mathbf{y}} = \mathbf{G}\Delta\mathbf{u} + \mathbf{f}, \quad (4)$$

na qual \mathbf{f} representa a resposta livre do sistema e \mathbf{G} corresponde à matriz de dinâmica do sistema, obtida usando os coeficientes da resposta ao degrau do sistema. Quando o modelo é linear, essa representação é possível devido ao princípio da superposição, o qual é verificado em sistemas lineares. Tal princípio não é verificado para sistemas não lineares, porém o algoritmo PNMPC utiliza uma estrutura semelhante, com

$$\hat{\mathbf{y}} = \mathbf{G}_{\text{PNMPC}}\Delta\mathbf{u} + \mathbf{f}, \quad (5)$$

na qual \mathbf{f} corresponde à resposta livre obtida considerando os incrementos futuros de controle nulos, e a matriz $\mathbf{G}_{\text{PNMPC}} = \frac{\partial \hat{\mathbf{y}}}{\partial \Delta \mathbf{u}}$ representa o gradiente das saídas preditas em relação aos incrementos de controle futuros.

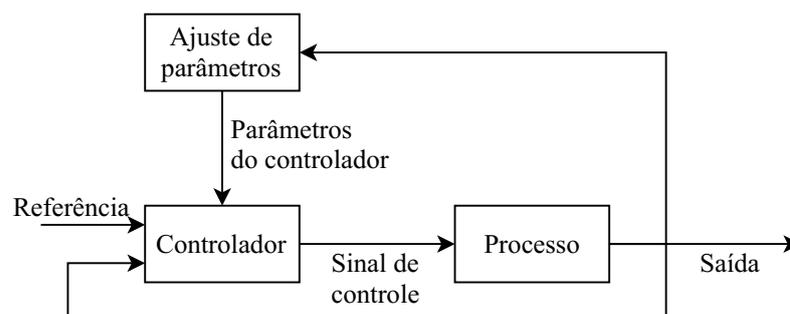
Essa representação afim em relação à variável de decisão $\Delta \mathbf{u}$ faz com que o problema de otimização possa ser resolvido com técnicas de programação quadrática, da mesma forma que no algoritmo DMC. Outros algoritmos presentes na literatura apresentam ideias semelhantes ao PNMPC, com destaque para o algoritmo MPC-NPL (do inglês, *MPC algorithm with Nonlinear Prediction and Linearization*), apresentado em Ławryńczuk (2007), o qual emprega o mesmo tipo de modelo de predição, fundamentado na obtenção do gradiente das saídas futuras em relação aos incrementos de controle futuros.

Existem outras formas para obtenção de um modelo de predição afim em relação aos sinais de controle futuros. Essas alternativas podem resultar em mudanças na complexidade computacional do algoritmo, ao mesmo tempo que afetam a qualidade da aproximação obtida pelo algoritmo do modelo não linear. Algumas dessas alternativas são apresentadas com detalhes em Ławryńczuk (2013) e em Ławryńczuk (2014).

2.3 CONTROLE PREDITIVO NÃO LINEAR ADAPTATIVO

Como foi definido em Åström e Wittenmark (2013, p. 1), “um controlador adaptativo é um controlador com parâmetros ajustáveis e um mecanismo para ajuste desses parâmetros”. Dessa forma, abordagens de controle adaptativo utilizam métodos de adaptação e aprendizado para modificar, on-line, os parâmetros do controlador, sendo utilizadas em problemas nos quais os parâmetros do processo variam no tempo ou são, a princípio, incertos. Uma representação geral de controladores adaptativos é apresentada na Figura 2. Nela, um sistema adaptativo pode ser interpretado como um sistema com duas malhas, sendo uma malha dedicada ao controle do processo e uma segunda para a adaptação dos parâmetros do controlador. No contexto do controle preditivo, a adaptação pode ser dos parâmetros do modelo de predição, de parâmetros de ajuste do controlador, como por exemplo os horizontes de predição e de controle e as ponderações da função custo, ou ambos.

Figura 2 – Diagrama de blocos de um sistema de controle adaptativo.



Fonte: adaptado de Åström e Wittenmark (2013).

A utilização de mecanismos adaptativos associados a controladores preditivos ainda é incipiente na literatura, visto que grande parte das abordagens MPC adaptativas se restringem a modelos de predição lineares (MAYNE, 2014). Dentre as várias abordagens para controladores preditivos adaptativos, as mais abundantes são integradas com técnicas de MPC robusto. De forma geral, essas estratégias usam mecanismos de adaptação para reduzir o conservadorismo das abordagens de MPC robusto (MAYNE; MICHALSKA, 1993; FUKUSHIMA *et al.*, 2007; MARAFIOTI *et al.*, 2014; VICENTE; TRODDEN, 2019). Outro tipo de abordagem abundante são as técnicas que aplicam métodos de identificação *Set Membership*, também combinadas com abordagens MPC robustas (ADETOLA *et al.*, 2009; ADETOLA; GUAY, 2011; TANASKOVIC *et al.*, 2014; BUJARBARUAH *et al.*, 2019; TANASKOVIC *et al.*, 2019).

Além desses algoritmos associados a técnicas de MPC robusto, existem abordagens de MPC dual. No controle dual, existem dois objetivos: i) o controle do processo cujas características são desconhecidas; ii) e a investigação das dinâmicas do processo, por meio de um sinal de sondagem (do inglês, *probing signal*) (WITTENMARK, 1995). Existem abordagens duais para MPC linear, nas quais o modelo é parametrizado por funções de base ortonormal, como nos trabalhos apresentados em Heirung *et al.* (2017) e em Kumar *et al.* (2019a), e também abordagens para NMPC, com modelos do tipo Wiener parametrizado por funções de base ortonormal, como o estudo apresentado em Kumar *et al.* (2019b).

Existem também estratégias que utilizam modelos adaptativos baseados no aprendizado de máquina. Um exemplo é o trabalho apresentado em Su e Wu (2009), que propõe a utilização de uma rede neural recorrente diagonal, estimada usando o filtro de Kalman estendido (EKF, do inglês *Extended Kalman Filter*), e com o problema NLP sendo resolvido por meio de um otimizador por enxame de partículas. Abordagem similar é apresentada em Reddi e Rani (2016), cujo modelo é parametrizado por meio de uma rede neural e também é estimada usando EKF. Por sua vez, em Akpan e Hassapis (2011), o modelo é parametrizado utilizando uma rede neural série-paralelo, treinada com um algoritmo RLS.

Quanto às abordagens em que parâmetros do controlador MPC são adaptadas, uma das que se destaca consiste nos algoritmos de *self-triggered* MPC. Este tipo de estratégia é utilizada para reduzir a complexidade computacional da resolução do NMPC, usando um mecanismo que escolhe os instantes de tempo no qual o controlador será executado, fazendo com que o problema NMPC seja resolvido aperiodicamente (HASHIMOTO *et al.*, 2017). Outros exemplos desse tipo de abordagem são as propostas em Sun *et al.* (2019) e Li *et al.* (2020), que utilizam mecanismos adaptativos para, a cada instante em que o controlador é executado, ser capaz de adaptar seu horizonte de predição, definindo a sequência de ações de controle para esse horizonte e também o próximo instante em que o controlador será executado, reduzindo ainda mais o custo computacional.

2.4 CONTROLE PREDITIVO NÃO LINEAR COM REDES NEURAIAS ARTIFICIAIS

O interesse pelo uso de redes neurais em aplicações de controle se deu naturalmente, seguindo a tendência iniciada após a publicação do algoritmo da retropropagação do erro, em Rumelhart *et al.* (1986), o qual impulsionou a pesquisa em redes neurais artificiais. Na área do controle de processos, o artigo de Narendra e Parthasarathy (1990) teve grande importância para o crescimento no interesse acerca do uso de RNAs. A grande relevância de tal artigo se deu por apresentar alguns caminhos que a pesquisa acerca de RNAs no controle poderia tomar. Uma dessas tendências era a utilização das redes neurais para identificação de sistemas dinâmicos não lineares.

O potencial apresentado pela possibilidade de construção de modelos não lineares utilizando técnicas neurais foi logo explorado na área do controle preditivo baseado em modelo. Os trabalhos precursores foram os de Hernandaz e Arkun (1990) e de Psychogios e Ungar (1990). O primeiro utilizou um modelo não linear identificado usando RNAs para a correção do erro de modelagem em um algoritmo de controle DMC. Por sua vez, o segundo utilizou uma RNA como modelo de predição em um controlador MPC.

A diversidade nas formas de utilização das RNAs em algoritmos MPC é uma característica que se revela perante uma análise aprofundada da literatura. Os trabalhos que envolvem a utilização de RNAs em algoritmos MPC podem ser sumarizados em três classes (TATJEWSKI; ŁAWRYŃCZUK, 2006):

- algoritmos MPC com otimização não linear e um modelo de processo ou modelo de predição baseado em RNA;
- algoritmos MPC com linearização on-line ou *offline* de um modelo de rede neural e otimização quadrática;
- modelos de RNA aplicados à redução de complexidade computacional de algoritmos MPC, executando a função do otimizador ou aproximando o comportamento do controlador.

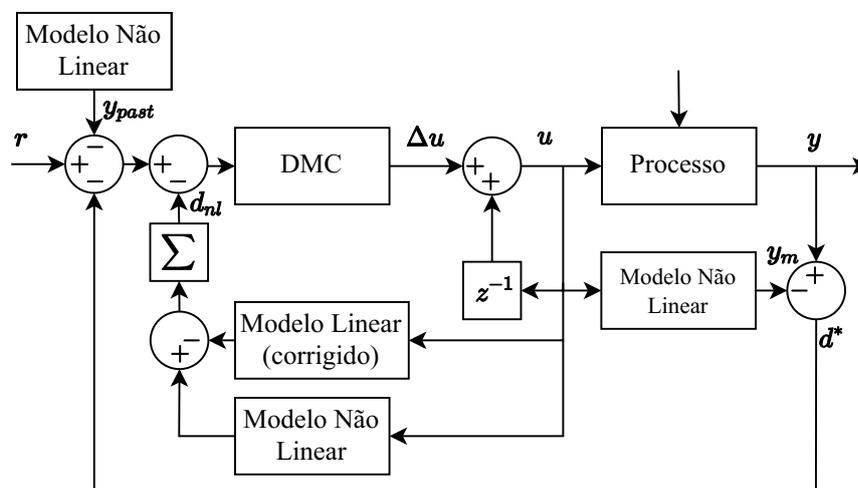
Apenas as duas primeiras classes se encontram no domínio de interesse deste trabalho. A terceira opção está estreitamente relacionada com a área da programação neurodinâmica, com mais detalhes acerca desse tipo de técnica podendo ser encontrados em Bertsekas (2009).

A primeira classe de algoritmos, baseados em otimização não linear, foi amplamente utilizada nos trabalhos precursores no uso de RNAs em abordagens MPC. Esses trabalhos exploravam o uso de redes neurais recorrentes (RNN, do inglês *Recurrent Neural Networks*) convencionais, principalmente modelos baseados nas redes MLP e redes com unidades baseadas em funções de base radial (RBF, do inglês *Radial Basis Functions*) aplicando realimentação entre a saída e entrada, redes essas que implementam modelos NARX. Os desafios enfrentados nesses estudos consistiam na obtenção dos modelos usando essas arquiteturas precursoras e também na resolução do problema de otimização não linear (SU; MCAVOY, 1993; TEMENG *et al.*,

1995; DECLERCQ; DE KEYSER, Robain, 1996; SOLOWAY; HALEY, 1996; TRAJANOSKI; WACH, 1998; YU; GOMM, 2003; AL SEYAB; CAO, 2008a, 2008b).

Um dos primeiros trabalhos que explorou a utilização de redes neurais em abordagens sem a necessidade da resolução de um problema NLP foi o de Draeger *et al.* (1995), o qual usou a RNA em uma abordagem denominada *Extended-DMC*, abordagem essa que considera a diferença entre o modelo não linear e a representação linear usada no DMC para correção do modelo de predição linear do DMC. O esquema de funcionamento desse algoritmo é apresentado no diagrama de blocos da Figura 3. Nela, é apresentada a função da rede neural, a qual não é usada diretamente pra obtenção da predição, e sim para obter uma estimativa do erro de predição, o qual é usado pra corrigir o modelo de predição do DMC.

Figura 3 – Diagrama de blocos para a estratégia DMC estendida.



Fonte: adaptado de Draeger *et al.* (1995).

Entre os trabalhos relevantes, destaca-se também o trabalho de Liu e Kadiramanathan (1998), o qual apresentou um algoritmo NMPC prático baseado na linearização da RNA. O uso de RNAs em abordagens NMPC práticas teve continuidade em diversos trabalhos, explorando arquiteturas baseadas em modelos não lineares com estruturas Wiener e Hammerstein. Nesses trabalhos, a RNA foi utilizada para obtenção da parte não linear estática, e várias alternativas para solução prática do problema NMPC foram exploradas. Alguns trabalhos exploraram, também, arquiteturas de RNN que implementam modelos NARX e arquiteturas clássicas, como a rede de Jordan (PICHE *et al.*, 2000; PENG *et al.*, 2006; ŁAWRYŃCZUK, 2007; ŁAWRYŃCZUK; TATJEWSKI, 2007; ŁAWRYŃCZUK, 2010; WYSOCKI; ŁAWRYŃCZUK, 2015).

Existem também alguns trabalhos que exploram abordagens não convencionais. Por exemplo, Wang e Wang (1999) utilizam um modelo de RNN pseudo-linear, o qual usa a função de ativação linear, fazendo com que o MPC resultante seja linear. Estratégias baseadas em otimização não linear usando redes RBF também são encontradas, incluindo versões adaptativas (ALEXANDRIDIS; SARIMVEIS, 2005; HAN *et al.*, 2015). Outras abordagens interessantes são o trabalho de Jin e Su (2008), o qual implementa um NMPC com RNN e otimização por

exame de partículas, e as abordagens de Najim *et al.* (1997) e Pan e Wang (2008), as quais usam RNN como preditor e também uma RNA como otimizador.

No domínio das aplicações, existem alguns trabalhos relevantes, como o controle de um motor a combustão, em Colin *et al.* (2007), e o controle de um processo de pasteurização, em Karla e Bakker (1995). O trabalho de Mu *et al.* (2005), por sua vez, apresenta a aplicação de dois controladores, sendo um NMPC completo e um NMPC prático, ambos aplicando um modelo baseado em RNA para o controle de uma turbina de aeronave. Outra aplicação que merece destaque consiste no controle da trajetória de embarcações, apresentada em Yan e Wang (2012b).

2.5 IDENTIFICAÇÃO DE SISTEMAS COM REDES NEURAIIS RECORRENTES

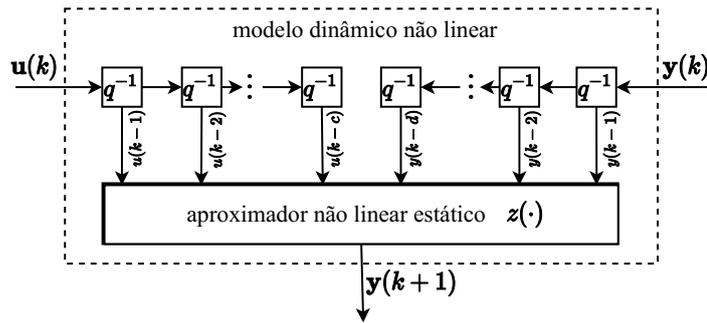
Conforme Ljung (1999, p. 1) definiu: “a identificação de sistemas lida com o problema da construção de modelos matemáticos de sistemas dinâmicos, baseados em dados observados do sistema”. Por meio de análise experimental, baseada em medidas ou observações do sistema, são obtidos modelos que representam a sua dinâmica, podendo ser utilizados para diversas finalidades, destacando-se predição e simulação. Outros exemplos típicos de aplicações são controle, análise, otimização e detecção de falhas.

Na tarefa de predição de um instante à frente, para sistemas dinâmicos de tempo discreto, as entradas passadas do sistema, $u(k-j)$, onde $j \in \mathbb{Z}^+$, e suas saídas passadas, $y(k-j)$, são usadas para a predição da saída do processo no próximo período de amostragem, $y(k+1)$. Neste trabalho é usada a notação para a predição de um instante à frente realizada no instante atual como $\hat{y}(k+1)$. Tipicamente, as predições são representadas por $\hat{y}(k+j|k+l)$, indicando a predição para o instante $k+j$ realizada no instante $k+l$. Porém, neste trabalho optou-se por simplificar a notação e omitir o instante em que é realizada a predição quando for o instante atual, k . No caso mais simples, apenas um instante à frente é predito. Se mais de um instante à frente é predito, a tarefa é denominada predição de múltiplos instantes à frente e o número de instantes N_p preditos é denominado horizonte de predição.

De forma geral, os modelos entrada-saída não lineares são divididos em duas classes, modelos com dinâmica externa ou interna. A estratégia da dinâmica externa é a mais frequentemente aplicada para a obtenção de sistemas dinâmicos não lineares. O termo, dinâmica externa decorre do fato de que esses modelos podem ser claramente separados em duas partes: um modelo não linear estático e um banco de filtros dinâmicos externos, como mostrado na Figura 4. Tipicamente, o modelo de filtro escolhido é um atraso simples q^{-1} no tempo, formando uma linha de atrasos. Em princípio, qualquer modelo de arquitetura pode ser escolhido para o aproximador $z(\cdot)$, como por exemplo, uma função não-linear estática, que pode gerar um sistema Hammerstein ou Wiener, ou uma rede neural estática, que gera um modelo neural dinâmico.

Os modelos dinâmicos não lineares podem ser utilizados em duas configurações gerais: para predição e para simulação. Essas configurações são tradicionalmente denominadas como configuração série-paralela, quando é realizada predição, e configuração paralela, quando é

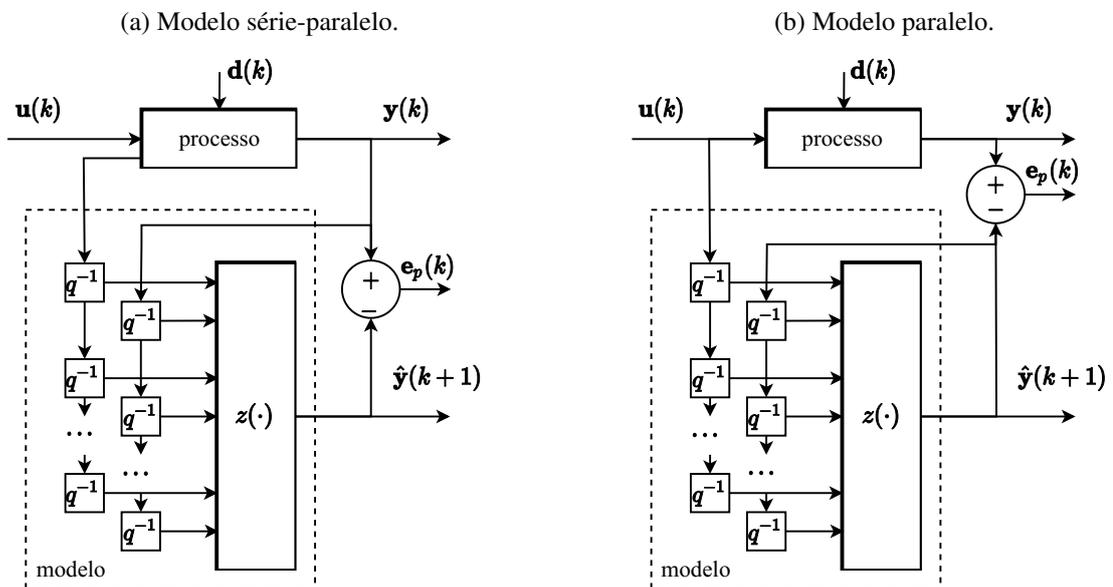
Figura 4 – Estratégia da dinâmica externa.



Fonte: adaptado de Nelles (2001).

realizada simulação. Tais configurações são ilustradas na Figura 5. No modelo série-paralelo a informação é puramente alimentada para frente, já que o modelo tem acesso à medição real do sistema. No caso do modelo paralelo, a alimentação acontece de forma recorrente, pois o modelo não possui informação da saída real do processo, que geralmente não está operando junto com a simulação.

Figura 5 – Modelos nas configurações série-paralelo, usado para previsão, e paralelo, usado para simulação.



Fonte: adaptado de Nelles (2001).

De forma geral, a classe de modelos com dinâmica externa, apesar de poder descrever uma grande classe de problemas não lineares, não é tão geral quanto os modelos com dinâmica interna. As limitações aparecem em problemas nos quais o processo apresenta mais de uma não linearidade, estados internos não-mensuráveis com uma grande não linearidade ou em casos nos quais as não linearidades são não-inversíveis (NELLES, 2001).

Os modelos não lineares com dinâmica interna são baseados na adição de um mecanismo de memória interna em estruturas estáticas. Essa classe de modelos pode ser descrita por meio

da seguinte representação no espaço de estados:

$$x(k + 1) = r(x(k), u(k)) \quad (6)$$

$$\hat{y}(k) = s(x(k)), \quad (7)$$

onde as variáveis de estados, $x(k)$, descrevem os estados internos do modelo, estados esses que normalmente não apresentam relação direta com os estados físicos do sistema; $r(\cdot)$ e $s(\cdot)$ representam funções não lineares. O número de estados internos do sistema está ligado, tipicamente, à estrutura do modelo, podendo ser escolhidos desde valores próximos à ordem esperada do processo até valores maiores que a ordem real do processo (NELLES, 2001).

Em contraste com os modelos de dinâmica externa, não é necessário o uso de entradas ou saídas passadas. Isso faz com que os modelos com dinâmica interna apresentem uma redução na dimensionalidade do espaço de entrada. Outra diferença é que, devido à inexistência de realimentação externa, o modelo pode operar apenas na configuração paralela. Consequentemente, esse modelo não é particularmente adequado à tarefa de predição de um instante à frente.

A tarefa de identificação de sistemas pode ser compreendida como uma tarefa de aprendizado supervisionado, o qual é um dos principais paradigmas do aprendizado de máquina. No aprendizado supervisionado, os modelos são obtidos através de exemplos contendo pares com entrada e saída, os quais são utilizados para o ajuste de parâmetros de um modelo. As redes neurais artificiais (RNAs) são uma classe de modelos amplamente utilizado em tarefas de aprendizado (HAYKIN, 2009). Elas fazem parte da subclasse dos modelos paramétricos, a qual é caracterizada por assumir uma estrutura para o modelo a ser identificado. Apesar de a aplicação de RNA exigir a definição de uma estrutura para o modelo neural, elas são capazes de modelar processos a partir de pouco ou nenhum conhecimento *a priori* sobre a física que rege o processo. Uma característica que decorre do uso de RNAs é o fato de que, na maioria das arquiteturas de redes neurais, os parâmetros das redes não podem ser interpretados e nenhum significado físico dos resultados do processo de identificação pode ser extraído (ISERMANN; MÜNCHHOF, 2011; LEVINE, 2010).

O campo de pesquisa no qual as redes neurais artificiais está inserida passou por diversas mudanças, as quais são sumarizadas por alguns autores como ondas de interesse (*waves*), ou gerações (GOODFELLOW *et al.*, 2016). As redes de primeira geração foram precursoras e, inspirando-se no comportamento de um neurônio biológico, buscaram criar unidades básicas de processamento e algoritmos de aprendizagem simples, capazes de solucionar problemas simples. Os principais exemplos são o modelo de neurônio apresentado em McCulloch e Pitts (1943) e o modelo *perceptron* apresentado em Rosenblatt (1958). A segunda geração é caracterizada pela adoção do conceito de conexionismo, o qual trouxe forte inspiração em conceitos da ciência cognitiva. A ideia fundamental explorada pelo conexionismo consiste no surgimento de um comportamento inteligente quando se utiliza um número grande de unidades simples de processamento conectadas em uma rede. O conexionismo foi impulsionado pelo trabalho de Rumelhart *et al.* (1986), o qual apresentou o método de retropropagação do erro, que ainda hoje

é aplicado para o treinamento de uma grande variedade de RNAs. A terceira geração de redes neurais, a qual ainda está em curso, tem como principal característica a utilização de arquiteturas com vários níveis de abstração, um paradigma denominado aprendizado profundo (do inglês, *deep learning*). O conceito fundamental explorado pelas abordagens dessa geração é a utilização eficiente de arquiteturas com várias camadas, tendência impulsionada pelo desenvolvimento de algoritmos para treinamento eficiente desse tipo de arquitetura, o que resultou em avanços também no desenvolvimento das redes recorrentes (GOODFELLOW *et al.*, 2016).

O desenvolvimento de arquiteturas de RNA recorrentes e a sua aplicação em tarefas de identificação de sistemas teve início durante a segunda geração de pesquisas em RNA. As primeiras arquiteturas de redes recorrentes eram versões modificadas das arquiteturas estáticas alimentadas adiante, obtendo modelos usando o conceito da dinâmica externa (PARLOS *et al.*, 1991; PUSKORIUS; FELDKAMP, 1994). Por sua vez, as abordagens que surgiram na terceira geração consideram modelos com níveis maiores de abstração, em arquiteturas profundas com elementos específicos para armazenamento de informação. A principal arquitetura de rede neural recorrente dessa geração é a rede com memória de longo-curto prazo (LSTM), proposta em Hochreiter e Schmidhuber (1997), a qual, assim como outras RNA recorrentes dessa geração, resultam em modelos com dinâmica interna.

Nas próximas subseções são detalhadas algumas arquiteturas de redes neurais recorrentes que apresentam potencial para aplicação em problemas de identificação de sistemas. Na Seção 2.5.1 são apresentadas as arquiteturas baseadas nas redes neurais alimentadas adiante estáticas. Por sua vez, a Seção 2.5.2 apresenta as abordagens que empregam arquiteturas de redes de estado de eco.

2.5.1 Redes neurais recorrentes convencionais

Nas RNAs, as unidades básicas de processamento são, de forma geral, denominadas neurônios artificiais, devido às primeiras formulações apresentarem morfologia e funcionamento inspirados no neurônio biológico. O modelo geral, o qual é usado em grande parte das arquiteturas de RNAs, incluindo algumas arquiteturas atuais, apresenta três elementos fundamentais: um conjunto de pesos sinápticos, um somador e uma função de ativação (HAYKIN, 2009).

Os pesos sinápticos armazenam grande parte do conhecimento no modelo de neurônio artificial. Esses pesos sinápticos são responsáveis por ponderar as informações que entram no neurônio, as quais são somadas no elemento somador. De forma geral, um parâmetro de *bias* é também considerado, podendo ser entendido como um peso sináptico conectado a uma entrada adicional, sempre ativada. A função de ativação, $p(\cdot)$, mapeia a soma dos dados de entrada, ponderados pelos pesos sinápticos, para o valor de saída do neurônio. O funcionamento de um

neurônio artificial k com j entradas, é descrito matematicamente pelas expressões

$$v_k = \sum_{j=1}^m w_{kj}x_{kj} + b_k \quad (8)$$

$$y_k = p(v_k) \quad (9)$$

em que x_{kj} representa sinais de entrada; w_{kj} representa pesos sinápticos do neurônio k ; b_k representa um *bias*; v_k representa a saída do elemento somador, contendo a soma das entradas ponderadas e o *bias* b_k ; $p(\cdot)$ representa a função de ativação; e y_k representa o sinal de saída do neurônio.

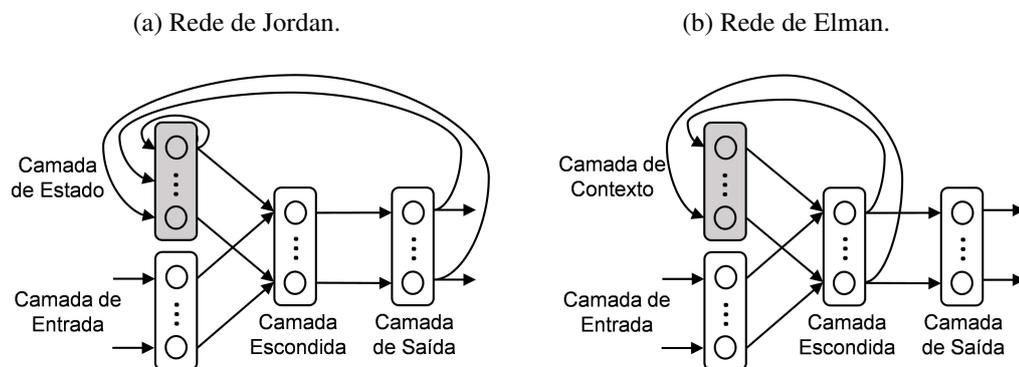
Diversas funções podem ser escolhidas para a ativação do neurônio. A saturação é uma função clássica, presente na formulação original do *perceptron*, porém é raramente utilizada em formulações atuais. As funções do tipo sigmoide comumente usadas são a logística e a tangente hiperbólica, as quais são deriváveis, característica importante para os algoritmos de treinamento, que de forma geral necessitam computar o gradiente do erro da rede (HAYKIN, 2001). Outra função que recebe destaque na literatura é a ReLU (do inglês, *Rectified Linear Unit*), usada em arquiteturas profundas para evitar os fenômenos de desaparecimento e explosão do gradiente, que limitam o treinamento de arquiteturas profundas (GLOROT *et al.*, 2011). No decorrer deste trabalho, por simplicidade, essas unidades serão nomeadas pela função de ativação usada, por exemplo, unidade *tanh* quando for um neurônio artificial com função de ativação for tangente hiperbólica, unidade σ quando a função de ativação for logística sigmoide e unidade ReLU.

As arquiteturas de RNAs tradicionais são as alimentadas adiante, as quais englobam os modelos de redes mais tradicionais e utilizados, as redes *perceptron* de múltiplas camadas (MLP) e as redes com função de ativação de base radial (RBF). Elas diferem entre si pelo tipo de modelo de neurônio artificial que utilizam, mas são empregadas no mesmo contexto, principalmente para a resolução de problemas de estáticos de aprendizado supervisionado. Esses modelos tradicionais de redes neurais foram desenvolvidos com capacidade para processar apenas padrões de entrada estáticos. Para lidar com estímulos variantes no tempo e padrões dinâmicos, algumas abordagens incorporaram nessas arquiteturas tradicionais estratégias capazes de contornar essa limitação, seja adicionando o tempo como uma dimensão adicional na entrada ou por meio de conexões recorrentes (HAYKIN, 2001).

Em Jordan (1986), foi definido que se uma rede apresenta um ou mais ciclos em que é possível seguir um caminho entre uma unidade e ela mesma, então essa rede pode ser denominada recorrente. Nesse trabalho foi apresentada a primeira arquitetura recorrente, que ficou conhecida como rede de Jordan, mostrada na Figura 6a. Essa rede apresenta laços de realimentação entre a saída da rede e a entrada, a qual é denominada camada de estado. Nessa arquitetura, a camada de estado apenas replica as saídas da rede e pode apresentar realimentações entre seus neurônios. De forma geral, os pesos de entrada da camada de estado não são treináveis, fazendo com que esses neurônios se comportem como uma entrada qualquer, não necessitando de métodos de treinamento alternativos aos tradicionalmente usados em redes MLP.

Outra arquitetura recorrente encontrada na literatura, denominada rede recorrente simples (SRN, do inglês *Simple Recurrent Network*) e comumente tratada como rede de Elman, foi apresentada por Elman (1990). Essa arquitetura expande a arquitetura MLP por meio da adição de uma camada de contexto, a qual consiste em uma realimentação entre a camada escondida e a entrada da rede. Essa camada de contexto faz com que a rede de Elman seja capaz de lidar com problemas temporais, assim como a rede de Jordan. De forma semelhante à rede de Jordan, a camada de contexto apenas replica os valores da saída dos neurônios da camada escondida, não sendo possível treinar esses pesos (HAYKIN, 2009).

Figura 6 – Arquiteturas recorrentes tradicionais (nem todas conexões são mostradas). A rede de Jordan utiliza uma *camada de estado*, enquanto a rede de Elman apresenta uma *camada de contexto*, as quais desempenham funções semelhantes.



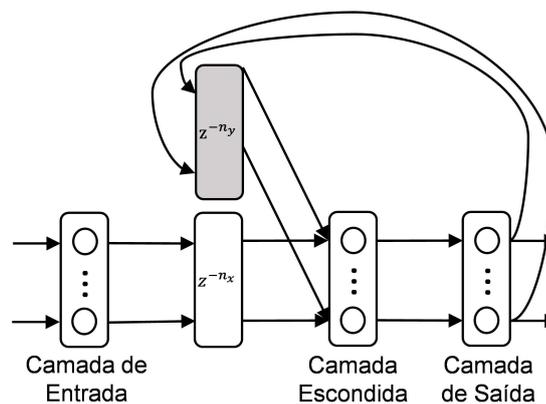
Fonte: elaborado pelo autor.

De maneira geral, as redes de Jordan e Elman podem ser utilizadas em problemas dinâmicos, sendo empregadas principalmente para o processamento de linguagem natural. Na prática, ambos os modelos ficaram de certa forma obsoletos, em detrimento de modelos mais gerais, que permitem o treinamento dos pesos das realimentações. Um tipo de abordagem também muito aplicado consiste nas redes com atraso temporal (tradução livre para *Time-Delay*), introduzidas em Lang *et al.* (1990). Nesse tipo de arquitetura, o tempo é tratado como uma dimensão adicional da entrada, fazendo com que a rede receba em sua camada de entrada não apenas as instâncias relacionadas ao tempo atual, mas também as relacionadas a instantes passados. Essa estratégia resulta em um alto número de parâmetros de entrada, pois, por exemplo, para uma janela temporal de 5 amostras, a dimensão da camada de entrada será $5n_i$, sendo n_i o número de padrões de entrada.

Além dessas arquiteturas clássicas, outras abordagens originárias do modelo MLP são apresentadas na literatura. As principais são a rede rede auto-regressiva não linear com entradas exógenas (NARX, do inglês *Nonlinear AutoRegressive with eXogenous inputs*) e a rede MLP recorrente (RMLP, do inglês *Recurrent MultiLayer Percetron*), que diferem das anteriores pelo fato de apresentarem pesos treináveis nas realimentações. A primeira delas, a NARX, utiliza a realimentação dos neurônios de saída para a camada de entrada, gerando um modelo NARX. Esse modelo apresenta sua entrada aplicada a uma linha de atrasos com tamanho n_x , sua saída

realimentada para a camada de entrada, aplicada a uma linha de atrasos com tamanho n_y , e utiliza um *perceptron* de múltiplas camadas como bloco construtivo básico. Esse modelo é mostrado na Figura 7 (HAYKIN, 2001).

Figura 7 – Arquitetura da rede NARX. São usadas como entrada da rede instâncias atuais e passadas da entrada, bem como instâncias passadas da saída da própria rede.



Fonte: elaborado pelo autor.

Essa arquitetura é caracterizada por incorporar partes do modelo *perceptron* de múltiplas camadas, associado com linhas de atrasos discretos (TDL, do inglês *Tap Delay Line*). Essas linhas de atrasos são conectadas a cada entrada, preservando a informação temporal contida nas entradas. Esse modelo incorpora, adicionalmente, a sua própria saída realimentada como uma entrada, associada também a uma linha de atrasos discretos. Além disso, o modelo conta com uma ou mais camadas escondidas, responsáveis pelo processamento da informação entre as camadas de entrada e de saída da rede (HAYKIN, 2009).

2.5.2 Rede de estado de eco

As redes de estado de eco (ESN) são arquiteturas de RNA recorrentes baseadas em um paradigma alternativo ao dos modelos tradicionais de redes neurais, o da computação de reservatório, proposto simultaneamente por Jaeger (2002) e Maass *et al.* (2002). Esse paradigma se baseia na utilização de um sistema dinâmico não linear com pesos fixos gerados aleatoriamente, por exemplo uma rede neural, para mapear as entradas em um espaço de alta dimensionalidade, no qual a tarefa de classificação ou de regressão é realizada eficientemente (LUKOŠEVIČIUS, 2012).

O conjunto de estados desse sistema dinâmico compõe a estrutura chamada de reservatório (tradução livre para *reservoir*). Esse reservatório¹ pode ser entendido como um *kernel*¹ temporal, projetando a entrada em um espaço dinâmico não linear. Durante a operação desse reservatório, os seus estados formam uma trajetória dependente dos estímulos externos e tam-

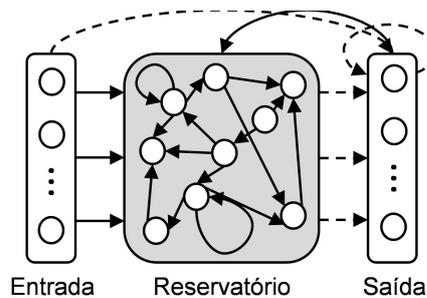
¹ *Kernels* são utilizados na teoria do aprendizado estatístico para a expansão dos algoritmos de vetor suporte para problemas não lineares. O *kernel* é empregado para mapear a entrada em um espaço de alta dimensionalidade, fazendo com que o problema possa ser resolvido mais facilmente (SMOLA; SCHÖLKOPF, 2004).

bém da memória de estímulos passados. A computação é feita pela camada de saída, que lê os estados instantâneos do reservatório (JAEGER *et al.*, 2007).

O modelo ESN, proposto em Jaeger e Haas (2004), é a principal arquitetura de redes neurais que utiliza esse paradigma, sendo a máquina de estado líquido (LSM, do inglês *Liquid State Machine*), proposta em Maass *et al.* (2002), e a rede *BackPropagation-DeCorrelation* (BPDC), proposta em Steil (2004), outros tipos de RNAs que compartilham o conceito da computação de reservatório. Os modelos ESN e BPDC utilizam um reservatório composto por neurônios analógicos, idênticos aos utilizados em redes MLP, e são utilizados principalmente em aplicações de engenharia. Já o modelo LSM utiliza neurônios pulsantes (tradução livre para *spiking neurons*), sendo usado principalmente em aplicações de neurociência computacional por ser um dos modelos mais fiéis ao comportamento de algumas áreas do cérebro humano (MAASS; BISHOP, 2001).

A estrutura básica de uma ESN é ilustrada na Figura 8. O reservatório é constituído de unidades com ativação não linear, geralmente *tanh*, conectadas de forma recorrente, característica responsável por seu comportamento dinâmico. As conexões entre os neurônios do reservatório são fixas e geradas de forma aleatória. Existem diversas versões de ESN, as quais utilizam conexões opcionais que podem ser utilizadas dependendo da aplicação desejada. Na prática, as únicas conexões obrigatórias são as internas ao reservatório e o mecanismo de leitura do reservatório, o qual liga o reservatório com a camada de saída (LUKOŠEVIČIUS, 2012).

Figura 8 – Arquitetura da rede ESN. Linhas sólidas representam conexões fixas, enquanto tracejadas representam pesos treináveis.



Fonte: adaptado de (ANTONELO, 2012).

A formulação original, proposta em Jaeger (2001), em tempo discreto, pode ser interpretada como um sistema no espaço de estados, sendo definida pelas equações

$$\begin{aligned} \mathbf{x}_e(k) &= f(\mathbf{W}_{\text{res}}^{\text{res}} \mathbf{x}_e(k-1) + \mathbf{W}_{\text{in}}^{\text{res}} \mathbf{u}_e(k) + \mathbf{W}_{\text{bias}}^{\text{res}}), \\ \mathbf{y}_e(k) &= g(\mathbf{W}_{\text{res}}^{\text{out}} \mathbf{x}_e(k) + \mathbf{W}_{\text{in}}^{\text{out}} \mathbf{u}_e(k) + \mathbf{W}_{\text{out}}^{\text{out}} \mathbf{y}_e(k-1) + \mathbf{W}_{\text{bias}}^{\text{out}}), \end{aligned} \quad (10)$$

nas quais $\mathbf{u}_e(k)$ e $\mathbf{y}_e(k)$ representam a entrada e a saída da rede no instante k , respectivamente; $\mathbf{x}_e(k)$ representa o vetor de estados do reservatório no instante k ; $\mathbf{W}_{\text{res}}^{\text{res}}$, $\mathbf{W}_{\text{in}}^{\text{res}}$ e $\mathbf{W}_{\text{bias}}^{\text{res}}$ representam as matrizes que conectam o reservatório, a entrada e o termo de *bias*, com o reservatório,

respectivamente; $\mathbf{W}_{\text{res}}^{\text{out}}$, $\mathbf{W}_{\text{in}}^{\text{out}}$, $\mathbf{W}_{\text{out}}^{\text{out}}$ e $\mathbf{W}_{\text{bias}}^{\text{out}}$ representam as matrizes que conectam o reservatório, a entrada, a saída e o termo de *bias*, com a saída do modelo, respectivamente; $f(\cdot)$ representa a função de ativação dos estados do reservatório; e $g(\cdot)$ representa a função de ativação da saída da rede.

A formulação original da ESN apresenta uma derivação, denominada rede ESN com unidades integradoras com vazamento (*leaky*-ESN). Tal formulação tem como objetivo aprimorar a capacidade de modelagem para dinâmicas lentas, sendo a variante mais comumente aplicada na literatura. A formulação original da rede ESN com unidades integradoras com vazamento foi definida em tempo contínuo, sendo posteriormente discretizada, já que sua utilização é, geralmente, em tempo discreto. A formulação original foi apresentada em Jaeger (2001), tendo sido posteriormente detalhada em Jaeger *et al.* (2007). A equação de atualização dos estados, em tempo contínuo, é definida por meio da expressão

$$\dot{\mathbf{x}}_e = -\alpha \mathbf{x}_e + f(\mathbf{W}_{\text{res}}^{\text{res}} \mathbf{x}_e + \mathbf{W}_{\text{in}}^{\text{res}} \mathbf{u}_e + \mathbf{W}_{\text{bias}}^{\text{res}}), \quad (11)$$

na qual \mathbf{x}_e e \mathbf{u}_e representam, em tempo contínuo, o vetor de estados do reservatório e o vetor de entradas da rede, respectivamente; e $\alpha \in (0, 1]$ representa a taxa de vazamento. Se essa equação for discretizada utilizando o método de Euler, com passo de discretização T_s , a equação (11) pode ser reescrita resultando na expressão

$$\mathbf{x}_e(k) = (1 - T_s \alpha) \mathbf{x}_e(k-1) + T_s f(\mathbf{W}_{\text{res}}^{\text{res}} \mathbf{x}_e(k-1) + \mathbf{W}_{\text{in}}^{\text{res}} \mathbf{u}_e(k-1) + \mathbf{W}_{\text{bias}}^{\text{res}}). \quad (12)$$

As redes integradoras com vazamento são provenientes da modelagem da cognição e de redes neurais biológicas, sendo que a formulação mais comum, apresentada em Graben *et al.* (2008), apresenta algumas diferenças em relação à formulação utilizada para obtenção de (11). Usando a formulação de rede integradora com vazamento proveniente de Graben *et al.* (2008), a equação de atualização de estados da rede *leaky*-ESN, em tempo contínuo, é descrita pela expressão

$$\dot{\mathbf{x}}_e = -\alpha \mathbf{x}_e + \alpha f(\mathbf{W}_{\text{res}}^{\text{res}} \mathbf{x}_e + \mathbf{W}_{\text{in}}^{\text{res}} \mathbf{u}_e + \mathbf{W}_{\text{bias}}^{\text{res}}), \quad (13)$$

a qual, após o mesmo processo de discretização utilizando o método de Euler, com passo de discretização T_s , resulta na equação de estados, em tempo discreto, descrita por

$$\mathbf{x}_e(k) = (1 - T_s \alpha) \mathbf{x}_e(k-1) + T_s \alpha f(\mathbf{W}_{\text{res}}^{\text{res}} \mathbf{x}_e(k-1) + \mathbf{W}_{\text{in}}^{\text{res}} \mathbf{u}_e(k-1) + \mathbf{W}_{\text{bias}}^{\text{res}}). \quad (14)$$

Considerando o modelo em tempo discreto, com $T_s = 1$, é obtida a formulação de reservatório mais comumente aplicada na literatura, definida por

$$\mathbf{x}_e(k) = (1 - \alpha) \mathbf{x}_e(k-1) + \alpha f(\mathbf{W}_{\text{res}}^{\text{res}} \mathbf{x}_e(k-1) + \mathbf{W}_{\text{in}}^{\text{res}} \mathbf{u}_e(k-1) + \mathbf{W}_{\text{bias}}^{\text{res}}), \quad (15)$$

sendo que, para $\alpha = 1$, a formulação *leaky*-ESN é idêntica à formulação ESN original, descrita em (11).

As arquiteturas de ESN já foram exploradas de distintas formas para a tarefa de identificação de sistemas, a partir do método proposto em Jaeger (2003), o qual focava na identificação de sistemas não lineares adaptativa. Essas abordagens podem ser divididas em duas classes distintas: modelos identificados a partir de séries de dados adquiridas do processo e modelos identificados on-line.

Quando se trata do problema de identificação baseado em séries de dados adquiridas previamente do processo, a ESN pode ser treinada utilizando um método da regressão linear ordinária (JAEGER, 2001), ou métodos com regularização, como a regularização de Tikhonov (SHI; HAN, 2007) e o LASSO (do inglês, *Least Absolute Shrinkage and Selection Operator*) (QIAO *et al.*, 2018). Existem, também, outras abordagens, como a apresentada em Zhang *et al.* (2018), que propõe o uso da função custo correntropia generalizada no treinamento da ESN, e o trabalho apresentado em Sun *et al.* (2017), que propõe uma arquitetura ESN *Wavelet-Sigmoid*.

No campo das aplicações, destaca-se o trabalho apresentado em Yang *et al.* (2009), que aplicou a ESN para identificar a dinâmica da capacidade de corrente de linhas de transmissão e identificação de sistemas de potência (DAI *et al.*, 2009). Outras aplicações são o uso da ESN para a obtenção de um modelo para estimação da pressão de fundo poços de petróleo (ANTONELO *et al.*, 2017) e a identificação de um sistema de geração eólica (CHEN *et al.*, 2019).

Quando se trata de abordagens para identificação on-line, a ESN é geralmente aplicada considerando o algoritmo dos mínimos quadrados recursivo para estimação on-line dos pesos da camada de saída da rede. Esse tipo de abordagem foi proposta inicialmente em Jaeger (2003), sendo que alguns trabalhos apresentados na literatura expandem essa estratégia. A formulação descrita em Song *et al.* (2011a) também considera um algoritmo de estimação recursiva, porém aplica um mecanismo de fator de esquecimento adaptativo, o qual se baseia em um modelo de detecção de mudanças. No trabalho apresentado em Yang *et al.* (2019), a formulação se baseia em um algoritmo dos mínimos quadrados recursivo, com estimação recursiva esparsa.

2.6 IDENTIFICAÇÃO RELEVANTE PARA MPC

Os controladores MPC obtêm o sinal de controle por meio da minimização de uma função objetivo, a qual, usualmente, apresenta uma estrutura baseada no erro de predição ao longo de um horizonte. É necessário, então, um modelo capaz de produzir predições da saída do processo para todo o horizonte considerado no problema de otimização. Dessa forma, tal modelo deve ser capaz de produzir boas predições para múltiplos instantes à frente (GOPALUNI *et al.*, 2003).

Em uma tarefa de identificação de sistemas dinâmicos convencional, o modelo é obtido por meio de uma abordagem baseada no erro de predição de um instante à frente, o qual minimiza a função objetivo definida como

$$J_{\text{OSAP}} = \min \left(\frac{1}{Tm} \sum_{i=1}^m \sum_{j=1}^T \|y_i(j) - \hat{y}_i(j)\|^2 \right), \quad (16)$$

na qual $y_i(j)$ representa os valores observados para a i -ésima saída; $\hat{y}_i(j)$ representa a saída do modelo para a i -ésima saída; m representa o número de saídas; e T representa o número de observações.

Como a aplicação do modelo em um MPC implica a necessidade de um modelo capaz de produzir boas previsões para N instantes à frente, o ideal é que o problema de identificação considere como objetivo, a minimização do erro de previsão de múltiplos instantes à frente. Essa função objetivo pode ser descrita por meio da equação

$$J_{\text{MSAP}} = \min \left(\frac{1}{Tm} \sum_{i=1}^m \sum_{j=1}^T \left(\frac{1}{N_i} \sum_{k=0}^{N_i-1} \|y_i(j-k) - \hat{y}_i(j-k)\|^2 \right) \right), \quad (17)$$

na qual N_i representa o horizonte de previsão para a i -ésima saída.

As abordagens de identificação focadas na obtenção de modelos para MPC, as quais utilizam a equação (17) são denominadas como estratégias de identificação relevante para MPC (MRI). A equação (16), dependendo da estrutura do modelo, pode ser uma função quadrática e, em algumas condições, apresentar uma solução fechada. De forma distinta, a equação (17) é inerentemente não linear, sendo necessário o emprego de métodos de programação não linear (NLP) para sua solução.

Algumas alternativas para a solução do problema MRI e obtenção de modelos paramétricos por meio da solução do problema descrito na equação (17) são propostas na literatura. As abordagens MRI podem ser divididas em dois tipos de estratégias, baseadas na utilização de um pré-filtro (SHOOK *et al.*, 1992; HUANG; WANG, 1999; GOPALUNI *et al.*, 2004; POTTS *et al.*, 2014) ou na solução direta do problema de otimização não linear (LAURÍ *et al.*, 2010; ŽÁČEKOVÁ *et al.*, 2011; QUACHIO; GARCIA, 2011; CHI *et al.*, 2014; ŽÁČEKOVÁ *et al.*, 2014; BALTAR *et al.*, 2015).

As abordagens baseadas em pré-filtragem, inicialmente propostas em Shook *et al.* (1992), são capazes de evitar a solução direta da equação (17), filtrando os dados de entrada e saída do processo utilizando um filtro obtido a partir do modelo de perturbação do processo. No caso do método proposto em Shook *et al.* (1992), o modelo de perturbação deve ser conhecido *a priori*, enquanto que na abordagem proposta em Huang e Wang (1999), um processo de identificação preliminar é utilizado para obtenção do modelo de perturbação.

Por sua vez, as estratégias que se baseiam na solução direta do problema de otimização não linear obtêm um modelo único e global sem a necessidade de um pré-filtro, utilizando abordagens baseadas em um método dos mínimos quadrados parciais com horizonte de previsão (PLS-PH, do inglês *Partial Least Squares Prediction Horizon*) (LAURÍ *et al.*, 2010). Outros estudos que se destacam são a integração deste tipo de abordagem no algoritmo GPC (CHI *et al.*, 2014), e sua aplicação na identificação de processos em malha fechada (QUACHIO; GARCIA, 2011) e na identificação e controle preditivo de sistemas de aquecimento, ventilação, e ar condicionado (ŽÁČEKOVÁ *et al.*, 2011, 2014).

O desenvolvimento de abordagens para estruturas não lineares vem sendo tema de estudo em trabalhos recentes da literatura, com destaque para abordagens baseadas em modelos não

lineares caixa cinza (PČOLKA *et al.*, 2016), modelos do tipo NARX (QUACHIO; GARCIA, 2014) e em modelos lineares do tipo Hammerstein-Wiener (QUACHIO; GARCIA, 2017, 2019). Tais abordagens MRI para modelos não lineares levam em conta a estrutura do modelo, em formulações nas quais a estratégia MRI é baseada no uso do pré-filtro obtido do modelo de perturbação. Por exemplo, a abordagem MRI para modelos não lineares do tipo Hammerstein-Wiener, proposta em Quachio e Garcia (2019), se utiliza da estrutura orientada a blocos do modelo para propor uma abordagem na qual o pré-filtro é aplicado aos sinais de entrada ou sinais intermediários do modelo, dependendo da estrutura.

2.7 COMENTÁRIOS FINAIS

Neste capítulo foi apresentada uma revisão da literatura explorando as alternativas que já foram aplicadas para obtenção de modelos não lineares com ESNs e as abordagens exploradas na literatura para implementação de controladores NMPC práticos e NMPC adaptativos, em especial os trabalhos que exploraram a utilização de RNNs em controladores NMPC dessa classe. As abordagens que implementam controladores NMPC práticos com modelos obtidos por meio de RNNs seguem uma linha semelhante, aplicando arquiteturas de RNNs tradicionais, de forma geral, baseadas em estruturas recorrentes oriundas do modelo MLP, com alguns trabalhos recentes explorando a utilização da ESN como modelo de predição.

Os controladores NMPC práticos compartilham de várias similaridades, porém, são frequentemente desenvolvidos considerando modelos de predição específicos, o que dificulta a aplicação direta para novas estruturas de modelos. Essa é uma limitação importante, já que avanços nessa área são comuns, com novas arquiteturas sendo frequentemente apresentadas. A rede de estado de eco foi identificada na literatura como uma arquitetura de RNN com potencial para aplicação em abordagens NMPC, porém, seu uso ainda incipiente neste contexto. Existe uma abordagem de NMPC prático baseada em ESN, apresentada em Jordanou *et al.* (2018), porém, essa é uma formulação de MPC no espaço de estados, resultando em um problema de otimização com uma complexidade alta, devido ao tamanho geralmente elevado do reservatório da ESN. Existem diversas oportunidades para formulação de NMPCs práticos, em abordagens com linearização numérica, semelhante à formulação original apresentada em Plucenio *et al.* (2007). Em uma abordagem com linearização numérica, é possível obter um problema de otimização cujas matrizes apresentam dimensões associadas ao número de entradas e saídas do processo, e não ao tamanho do reservatório da ESN, resultando em um algoritmo com complexidade computacional, geralmente, bem menor que a do algoritmo apresentado em Jordanou *et al.* (2018).

Sobre a identificação de sistemas com modelos baseados na ESN, foram encontradas na literatura várias abordagens, não só para identificação de modelos utilizando dados previamente adquiridos, mas também algoritmos para identificação on-line, baseados no algoritmo dos mínimos quadrados recursivo. Apesar da existência desses métodos para identificação on-line com a ESN, a utilização da ESN em algoritmos NMPC em abordagens adaptativas ainda não foi

explorada na literatura. Adicionalmente, foram identificadas abordagens para identificação de sistemas relevante ao MPC para modelos não lineares, porém, esse tipo de abordagem ainda não foi explorado no contexto dos modelos baseados na ESN.

Na sequência deste trabalho são apresentadas algumas propostas que almejam explorar essas deficiências identificadas na literatura. No Capítulo 3 é apresentada uma estratégia MRI para identificação de processos não lineares com modelos ESN, sendo detalhado o algoritmo para identificação usando dados coletados do processo e um algoritmo para estimação on-line. No Capítulo 4, é apresentada a proposta de formulação para um algoritmo de controle NN MPC prático para modelos obtidos por meio de ESNs.

3 IDENTIFICAÇÃO RELEVANTE PARA MPC COM ESN

Sumário

3.1	Introdução	46
3.2	Modelo não linear baseado na rede de estado de eco	48
3.3	Formulação do problema de identificação relevante para MPC	50
3.4	Identificação do modelo ESNI offline	54
3.4.1	Treinamento da ESNI usando o método de regressão com regularização	54
3.4.2	Exemplo 1: Identificação de um processo de neutralização de pH com ESN	56
3.5	Identificação do modelo ESNI on-line	59
3.5.1	Treinamento do mecanismo de leitura do reservatório por método dos mínimos quadrados recursivo com fator de esquecimento direcional	60
3.5.2	Exemplo 2: Identificação de um tanque de perfil cônico	61
3.6	Comentários Finais	68

3.1 INTRODUÇÃO

Quando se almeja utilizar um modelo dinâmico para predição em controladores preditivos, é desejado que o modelo seja capaz de retornar predições para múltiplos instantes à frente. Como foi detalhado na Seção 2.6, existem técnicas para obtenção de modelos adequados à aplicação em MPC, as quais são denominadas abordagens MRI, e se baseiam na obtenção de modelos otimizados considerando como critério, o erro de predição de múltiplos instantes à frente, em detrimento do erro de predição de um instante à frente, o qual é tipicamente utilizado. Existem abordagens na literatura para identificação de sistemas dinâmicos não lineares utilizando a rede de estado de eco (ESN), os quais foram detalhados na Subseção 2.5.2, porém, tais abordagens consideram em suas formulações, o problema de predição de um instante à frente, sendo que as abordagens MRI para identificação de modelos baseados em ESN ainda não foram devidamente explorados na literatura.

Este capítulo explora essa lacuna da literatura, propondo um método para identificação de sistemas dinâmicos com modelos baseados em redes de estado de eco, em uma abordagem MRI. Uma arquitetura de ESN para identificação de sistemas dinâmicos não lineares é proposta, considerando uma estrutura adequada para utilização em um MPC, seguida pela demonstração de uma abordagem MRI para identificação desse modelo baseado em ESN. São apresentados alguns métodos para o treinamento da arquitetura ESN, considerando a abordagem MRI proposta, baseados em séries de dados adquiridas do processo (off-line) e também para estimação recursiva (on-line) do modelo proposto. A abordagem MRI, bem como as técnicas de identificação

propostas para o modelo baseado em ESN, são avaliadas por meio da identificação de processo de neutralização de pH e de um tanque de perfil cônico.

As contribuições à literatura detalhadas neste capítulo consistem na:

- proposição de uma arquitetura de ESN adequada ao uso em algoritmos MPC. Essa arquitetura apresenta uma estrutura que considera um modelo para a perturbação e uma parcela integradora, as quais são importantes para o desenvolvimento do algoritmo MPC, que é apresentado nos Capítulos 4;
- formulação de uma abordagem MRI para modelos baseados em ESN. Essa abordagem é apresentada para a arquitetura de ESN proposta, porém, a demonstração é geral, e pode ser facilmente estendida para outras arquiteturas de ESN;
- adaptação dos algoritmos de treinamento da ESN, baseados no método de regressão com regularização e no método dos mínimos quadrados recursivo, para treinamento considerando a abordagem MRI;
- utilização de um fator de esquecimento direcional em conjunto com o método dos mínimos quadrados recursivo para o treinamento on-line da ESN.

O desenvolvimento apresentado neste capítulo foi publicado nos seguintes artigos:

- SCHWEDERSKY, B. B.; FLESCHE, R. C. C.; DANGUI, H. A. C. Echo State Networks for Online, Multi-step MPC Relevant Identification. **Engineering Applications of Artificial Intelligence**, v. 108, p. 104596, fev. 2022. DOI: 10.1016/j.engappai.2021.104596
- SCHWEDERSKY, B. B.; FLESCHE, R. C. C.; DANGUI, H. A. C. Nonlinear MIMO System Identification with Echo State Networks, **Journal of Control, Automation and Electrical Systems**, jan. 2022. DOI: 10.1007/s40313-021-00874-y
- SCHWEDERSKY, B. B.; FLESCHE, R. C. C.; DANGUI, H. A. C. Identificação de Sistemas Dinâmicos Não Lineares Multivariáveis com Redes de Estado de Eco. In: XXIII CONGRESSO BRASILEIRO DE AUTOMÁTICA. Porto Alegre, Brasil. **Anais [...]**. Campinas: SBA, 2020. p. 1-8. DOI: 10.48011/asba.v2i1.1713

Neste capítulo, na Seção 3.2, um modelo não linear baseado na rede de estado de eco é apresentado. Em seguida, na Seção 3.3 é apresentada uma formulação relevante para controle preditivo para o problema de identificação do modelo ESN proposto. Em seguida, na Seção 3.4, um método de treinamento para o modelo ESN proposto, considerando a abordagem de identificação relevante para MPC, é apresentado, sendo complementado por exemplos de sua utilização. Adicionalmente, na Seção 3.5 é detalhada uma abordagem para estimação on-line do modelo ESN proposto, sendo apresentado um método de estimação baseado no algoritmo dos mínimos quadrados recursivo com fator de esquecimento adaptativo direcional e também um exemplo de sua aplicação. Por fim, na Seção 3.6 são apresentados alguns comentários finais acerca do método proposto e da comparação com as formulações semelhantes apresentadas na

literatura, discutindo sobre a qualidade dos modelos resultantes e o custo computacional, para treinamento e para obtenção das predições.

3.2 MODELO NÃO LINEAR BASEADO NA REDE DE ESTADO DE ECO

A rede de estado de eco foi explorada na literatura em diversas abordagens para identificação de sistemas dinâmicos, sendo que arquiteturas e métodos de treinamento distintos foram propostos, adequando-se a tipos distintos de aplicações. Dentre as arquiteturas de ESN tipicamente exploradas nesse tipo de problema, as arquiteturas baseadas no modelo *leaky*-ESN, detalhada na Seção 2.5.2, são as mais comuns, sendo ela a arquitetura em que o modelo proposto neste trabalho se baseia. Nessa estrutura, a rede é composta por uma camada de entrada e uma de saída, conectadas a um reservatório com vazamento, o qual é constituído por neurônios artificiais conectados recorrentemente.

O modelo de neurônio considerado nessa ESN usa funções de ativação com formato sigmoide, sendo empregada neste caso a função tangente hiperbólica, definida como

$$\tanh(r) = \frac{1 - e^{-2r}}{1 + e^{2r}}. \quad (18)$$

Cada neurônio do reservatório é alimentado com o sinal de entrada da ESN e com os valores passados da saída dos neurônios do reservatório por meio de conexões recorrentes, geralmente esparsas. O conjunto formado pela saída de todos neurônios do reservatório forma o estado do reservatório, $\mathbf{x}_e \in \mathbb{R}^r$, o qual é determinado pela equação de transição de estados

$$\mathbf{x}_e(k) = (\mathbf{1} - \alpha)\mathbf{x}_e(k-1) + \alpha f(\mathbf{W}_{\text{in}}^{\text{res}} \mathbf{u}_e(k-1) + \mathbf{W}_{\text{res}}^{\text{res}} \mathbf{x}_e(k-1) + \mathbf{W}_{\text{bias}}^{\text{res}} \mathbf{1}). \quad (19)$$

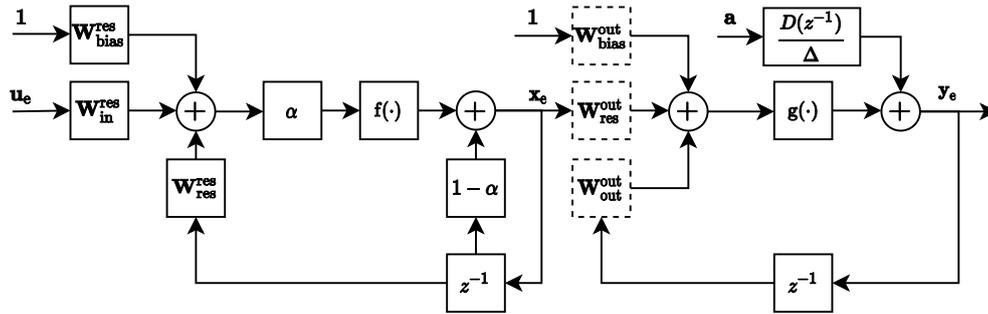
A saída da arquitetura proposta é formada pela combinação linear dos estados do reservatório e, em casos específicos, por saídas passadas e entradas atuais da ESN. Adicionalmente, devido ao objetivo deste trabalho consistir na utilização da ESN em algoritmos MPC, a arquitetura proposta considera um termo adicional somado à saída, composto pela integral do ruído branco. A equação de saída do modelo proposto é definida pela expressão

$$\mathbf{y}_e(k) = \mathbf{W}_{\text{res}}^{\text{out}} \mathbf{x}_e(k) + \mathbf{W}_{\text{bias}}^{\text{out}} \mathbf{1} + \mathbf{W}_{\text{out}}^{\text{out}} \mathbf{y}_e(k-1) + \frac{D(z^{-1})a(k)}{\Delta} \quad (20)$$

na qual $a(k)$ representa o ruído branco; $D(z^{-1})$ representa uma função de transferência para a perturbação, em z^{-1} ; e $\Delta = 1 - z^{-1}$ representa um termo integrador. O termo de perturbação modela o erro entre a predição e a saída real do processo, sendo o ruído branco integrado uma opção usada na literatura quando a característica estocástica da perturbação é desconhecida (HUANG *et al.*, 2003). O uso deste termo é importante, já que muitas aplicações industriais apresentam perturbações não estacionárias. Esse tipo de modelo de perturbação também é usado em algumas formulações de controle preditivo, como por exemplo o GPC (CLARKE *et al.*, 1987), e sua adoção é importante para fazer com que o controlador resultante possua ação integral. Essa arquitetura de rede ESN, considerando o modelo de perturbação e a parcela

integradora, é denominada no decorrer do trabalho como rede de estado de eco com termo integrador (ESNI, do inglês *Echo State Network with Integral term*). Um diagrama de blocos dessa estrutura proposta é apresentado na Figura 9.

Figura 9 – Diagrama de blocos simplificado do modelo ESNI. Blocos com linhas sólidas representam matrizes com pesos fixos e blocos com linhas tracejadas representam matrizes ajustáveis do modelo.



Fonte: elaborado pelo autor

A particularidade que as redes baseadas em computação de reservatório apresentam, em contraste com outros modelos de redes neurais recorrentes, é a presença de parâmetros ajustáveis apenas na camada de saída, sendo que as conexões da camada de entrada e do reservatório são fixas e definidas em uma etapa prévia ao treinamento. Devido à isso, a criação do reservatório de neurônios é uma tarefa fundamental para o sucesso do modelo, pois esses pesos não serão ajustados. As conexões não treináveis, W_{res}^{res} , W_{in}^{res} , W_{out}^{res} e W_{bias}^{res} , são geradas usando uma distribuição específica. No caso deste trabalho é usada a distribuição uniforme. Os principais parâmetros definidos pelo projetista são a taxa de conexão (c_{de}^{para}) e o escalamento (v_{de}^{para}), os quais definem a esparsidade e a força das conexões, respectivamente. De forma geral, as matrizes W_{res}^{res} e W_{in}^{res} são criadas com alta esparsidade (LUKOŠEVIČIUS, 2012).

A matriz de pesos recorrente W_{res}^{res} , que define as conexões entre os neurônios dentro do reservatório, recebe atenção especial. O principal parâmetro projetado é denominado raio espectral, $\rho(W_{res}^{res})$, o qual corresponde ao maior autovalor de W_{res}^{res} . Geralmente, a matriz W_{res}^{res} é criada usando uma esparsidade pré-definida e então é escalada, fazendo com que o maior autovalor seja igual ao raio espectral desejado.

A escala de tempo do reservatório pode ser ajustada usando o parâmetro α , denominado taxa de vazamento. Esse parâmetro define a quantidade do estado passado que será preservada pelo reservatório com unidades integradoras com vazamento (do inglês, *leaky integrator units*). Para $\alpha \in (0, 1]$, a memória do reservatório pode ser ajustada, sendo que valores pequenos da taxa de vazamento levam a um reservatório com maior memória, enquanto valores próximos de 1 resultam em pouca memória.

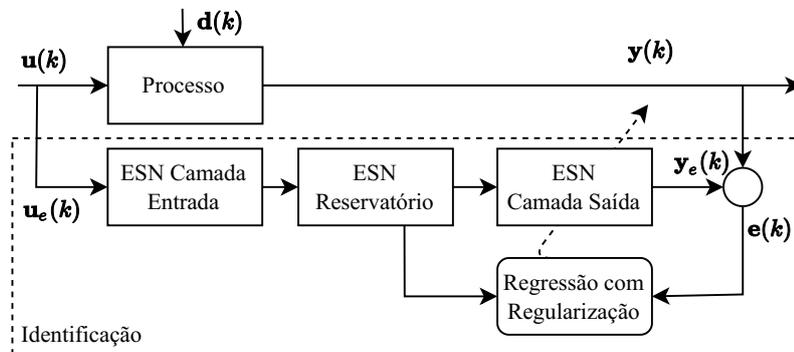
Uma condição importante para o sucesso da ESN é a propriedade de estado de eco (do inglês, *echo state property*). Um reservatório possui a propriedade de estado de eco se os estados do mesmo convergirem assintoticamente, sob a influência de uma entrada constante.

Para redes ESN que empregam neurônios sem vazamento, a condição suficiente para obtenção da propriedade de estado de eco é $\rho(|\mathbf{W}_{\text{res}}^{\text{res}}|) < 1$, em que $|\mathbf{W}_{\text{res}}^{\text{res}}|$ representa a matriz formada pelo valor absoluto de cada elemento da matriz original. Para o caso em que são usadas unidades com vazamento, a condição suficiente é $\rho(\mathbf{M}) < 1$, sendo $\mathbf{M} = |\mathbf{W}_{\text{res}}^{\text{res}}| + (1 - \alpha)\mathbf{I}$, com \mathbf{I} representando a matriz identidade de mesma dimensão de $\mathbf{W}_{\text{res}}^{\text{res}}$. A verificação das condições é detalhada em Yildiz *et al.* (2012). Mais detalhes acerca da inicialização dos modelos são encontrados em Lukoševičius (2012).

3.3 FORMULAÇÃO DO PROBLEMA DE IDENTIFICAÇÃO RELEVANTE PARA MPC

Nesta seção é apresentada uma formulação do problema de identificação relevante para MPC considerando o modelo ESNI. A estrutura para identificação de um processo utilizando a ESNI é apresentado por meio de um diagrama de blocos na Figura 10. O modelo baseado na ESNI é alimentado com exemplos que consistem em pares entrada e saída do processo, $\mathbf{u}(k)$ e $\mathbf{y}(k)$, respectivamente. Os exemplos são processados pela camada de entrada, a qual pode ser usada para normalização dos dados de entrada caso não haja normalização prévia, e excitam o reservatório. As saídas ativadas do reservatório, em conjunto com as saídas do processo, são usadas para ajustar os pesos da camada de saída.

Figura 10 – Diagrama de blocos para o procedimento de identificação de um processo utilizando a ESN.



Fonte: elaborado pelo autor.

Como apresentado em (20), a saída do modelo ESNI é formada pela combinação linear dos estados ativados do reservatório, aumentados por um termo de *bias*, e por saídas passadas do processo. Para simplificar a notação, a saída do modelo ESNI será descrita pela expressão

$$\mathbf{y}_e(k) = \mathbf{W}^{\text{out}}(k)\mathbf{x}(k) + H(z^{-1})a(k), \quad (21)$$

na qual

$$\mathbf{x}(k) = \begin{bmatrix} 1 \\ \mathbf{x}_e(k) \\ \mathbf{y}_e(k-1) \end{bmatrix}, \quad (22)$$

$$\mathbf{W}^{\text{out}}(k) = \begin{bmatrix} \mathbf{W}_{\text{bias}}^{\text{out}}(k) & \mathbf{W}_{\text{res}}^{\text{out}}(k) & \mathbf{W}_{\text{out}}^{\text{out}}(k) \end{bmatrix}, \quad (23)$$

$$H(z^{-1}) = \frac{D(z^{-1})}{\Delta}. \quad (24)$$

e $a(k)$ representa uma realização de ruído branco com média nula, assim como definido em Ljung (1999).

O processo de treinamento do modelo ESNI consiste em encontrar o valor ótimo de $\mathbf{W}^{\text{out}} \in \mathbb{R}^{m \times (1+s+m)}$, o qual minimize o erro médio quadrático entre a saída medida, $y(k)$, e a saída estimada, $\hat{y}(k)$. De forma geral, o problema de otimização é definido considerando como critério a ser minimizado o erro de predição de um instante à frente (OSAP, do inglês *One-Step Ahead Prediction*), o qual é definido como

$$J_{\text{OSAP}} = \min \left(\frac{1}{Tm} \sum_{i=1}^m \sum_{j=1}^T \|y_i(j) - \hat{y}_i(j)\|^2 \right). \quad (25)$$

Para simplificar a demonstração, o desenvolvimento da formulação é apresentado para o caso SISO, sendo que a generalização para o caso com entradas múltiplas e saídas múltiplas (MIMO, do inglês *Multiple Input and Multiple Output*) é apresentada no final da seção. Para o caso SISO, a saída real do processo pode ser definida por um termo oriundo da saída do modelo ESNI (predição), \mathbf{W}^{out} , o qual é complementado por um termo que representa a perturbação, $H(z^{-1})$, por meio da expressão

$$y(k) = \mathbf{W}^{\text{out}}(k)\mathbf{x}(k) + H(z^{-1})a(k). \quad (26)$$

O preditor estimado ótimo pode ser descrito como uma função dos parâmetros estimados do modelo $\hat{\mathbf{W}}^{\text{out}}$ e \hat{H} , por meio de

$$\hat{y}(k | k-1) = \hat{H}^{-1} \hat{\mathbf{W}}^{\text{out}} \mathbf{x}(k) + (1 - \hat{H}^{-1})y(k). \quad (27)$$

O erro de predição de um instante à frente pode ser obtido pela subtração entre (27) e (26), sendo descrito pela expressão

$$\epsilon(k | k-1) = y(k) - \hat{y}(k | k-1) = \hat{H}^{-1} \begin{bmatrix} \tilde{\mathbf{W}}^{\text{out}} & H \end{bmatrix} \begin{bmatrix} \mathbf{x}(k) \\ a(k) \end{bmatrix}, \quad (28)$$

na qual $\tilde{\mathbf{W}}^{\text{out}} = \mathbf{W}^{\text{out}} - \hat{\mathbf{W}}^{\text{out}}$. Reescrevendo (25), a função objetivo para minimização do erro de predição de um instante à frente pode ser escrita como

$$J = \frac{1}{T} \sum_{k=1}^T \epsilon(k | k-1)^2. \quad (29)$$

Para um número de instantes T suficientemente grande, a função objetivo descrita em (29) pode ser aproximada pela expressão

$$J = E[\epsilon(k | k-1)^2], \quad (30)$$

na qual $E[\cdot]$ representa o operador esperança matemática. A aplicação do teorema de Parseval, resulta na expressão

$$E[\epsilon(k | k-1)^2] = \frac{1}{2\pi} \int_{-\pi}^{\pi} \frac{1}{|\hat{H}(e^{j\omega})|^2} \begin{bmatrix} \tilde{\mathbf{W}}^{\text{out}}(e^{j\omega}) & H(e^{j\omega}) \end{bmatrix} \\ \times \begin{bmatrix} \Phi_x(\omega) & \Phi_{xa}(\omega) \\ \Phi_{ax}(\omega) & \sigma_a^2 \end{bmatrix} \begin{bmatrix} \tilde{\mathbf{W}}^{\text{out}}(e^{-j\omega}) \\ H(e^{-j\omega}) \end{bmatrix} d\omega, \quad (31)$$

na qual $\Phi_x(\omega)$ representa o espectro de potência de x ; $\Phi_{ax}(\omega)$ representa o espectro cruzado entre a e x ; $\Phi_{xa}(\omega)$ representa o espectro cruzado entre x e a ; e σ_a^2 representa a variância de a .

As estimativas ótimas para os parâmetros do modelo, $\hat{\mathbf{W}}^{\text{out}}$ e \hat{H} , os quais minimizam o erro de predição de um instante à frente, podem ser obtidos ao minimizar (30), ou seja

$$\begin{bmatrix} \hat{\mathbf{W}}^{\text{out}} & \hat{H} \end{bmatrix} = \operatorname{argmin} \frac{1}{2\pi} \int_{-\pi}^{\pi} \frac{1}{|\hat{H}(e^{j\omega})|^2} \begin{bmatrix} \tilde{\mathbf{W}}^{\text{out}}(e^{j\omega}) & H(e^{j\omega}) \end{bmatrix} \\ \times \begin{bmatrix} \Phi_x(\omega) & \Phi_{xa}(\omega) \\ \Phi_{ax}(\omega) & \sigma_a^2 \end{bmatrix} \begin{bmatrix} \tilde{\mathbf{W}}^{\text{out}}(e^{-j\omega}) \\ H(e^{-j\omega}) \end{bmatrix} d\omega. \quad (32)$$

Ao invés de otimizar o modelo considerando o erro de predição de um instante à frente, no problema de identificação relevante para MPC, os parâmetros do modelo, $\mathbf{W}^{\text{out}}(k)$, devem ser obtidos por meio da minimização de um problema de otimização cuja função objetivo minimize o erro de predição de múltiplos instantes à frente (MSAP, do inglês *Multi-Step Ahead Prediction*), ao longo de um horizonte de predição N , por meio da expressão

$$J_{\text{MSAP}} = \min \left(\frac{1}{T} \sum_{j=1}^T \left(\frac{1}{N} \sum_{k=0}^{N-1} \|y(j-k) - \hat{y}(j-k)\|^2 \right) \right). \quad (33)$$

Reescrevendo (33) e considerando T suficientemente grande, o erro de predição de múltiplos instantes à frente pode ser definido pela expressão

$$J_{\text{MSAP}} = \frac{1}{N} \sum_{j=1}^N E[y(k+j) - \hat{y}(k+j | k)]^2. \quad (34)$$

A j -ésima predição à frente pode ser escrita como (LJUNG, 1999)

$$\hat{y}(k+j | k) = \hat{F}_j \hat{H}^{-1} \hat{\mathbf{W}}^{\text{out}} \mathbf{x}(k+j) + (1 - \hat{F}_j \hat{H}^{-1})y(k+j), \quad (35)$$

na qual

$$\hat{F}_j = \sum_{i=0}^{j-1} g(i)z^{-i}, \quad (36)$$

$g(i)$ representa a resposta ao impulso dos coeficientes de \hat{H} , e

$$y(k+j) = \hat{\mathbf{W}}^{\text{out}} \mathbf{x}(k+j) + Ha(k+j). \quad (37)$$

Subtraindo (35) e (37), é possível obter a expressão

$$\begin{aligned} y(k+j) - \hat{y}(k+j|k) &= \hat{F}_j \hat{H}^{-1} \tilde{\mathbf{W}}^{\text{out}} \mathbf{x}(k+j) - \hat{F}_j \hat{H}^{-1} y(k+j) \\ &= \hat{F}_j \hat{H}^{-1} \begin{bmatrix} \tilde{\mathbf{W}}^{\text{out}} & H \end{bmatrix} \begin{bmatrix} \mathbf{x}(k+j) \\ a(k+j) \end{bmatrix}. \end{aligned} \quad (38)$$

Ao aplicar o teorema de Parseval, $E[(y(k+j) - \hat{y}(k+j|k))^2]$ pode ser reescrito como

$$\begin{aligned} E[(y(k+j) - \hat{y}(k+j|k))^2] &= \frac{1}{2\pi} \int_{-\pi}^{\pi} \frac{|\hat{F}_j(e^{j\omega})|^2}{|\hat{H}(e^{j\omega})|^2} \begin{bmatrix} \tilde{\mathbf{W}}^{\text{out}}(e^{j\omega}) & H(e^{j\omega}) \end{bmatrix} \\ &\quad \times \begin{bmatrix} \Phi_x(\omega) & \Phi_{xa}(\omega) \\ \Phi_{ax}(\omega) & \sigma_a^2 \end{bmatrix} \begin{bmatrix} \tilde{\mathbf{W}}^{\text{out}}(e^{-j\omega}) \\ H(e^{-j\omega}) \end{bmatrix} d\omega. \end{aligned} \quad (39)$$

A substituição de (39) em (34) resulta em

$$\begin{aligned} J_{\text{MSAP}} &= \frac{1}{N} \sum_{j=N_1}^{N_2} \frac{1}{2\pi} \int_{-\pi}^{\pi} \frac{|\hat{F}_j(e^{j\omega})|^2}{|\hat{H}(e^{j\omega})|^2} \begin{bmatrix} \tilde{\mathbf{W}}^{\text{out}}(e^{j\omega}) & H(e^{j\omega}) \end{bmatrix} \\ &\quad \times \begin{bmatrix} \Phi_x(\omega) & \Phi_{xa}(\omega) \\ \Phi_{ax}(\omega) & \sigma_a^2 \end{bmatrix} \begin{bmatrix} \tilde{\mathbf{W}}^{\text{out}}(e^{-j\omega}) \\ H(e^{-j\omega}) \end{bmatrix} d\omega. \end{aligned} \quad (40)$$

O operador soma pode ser movido para dentro da integral em (40), fazendo com que as estimativas ótimas para a tarefa de estimação de j -instantes à frente, $\tilde{\mathbf{W}}^{\text{out}}$ e \hat{H} , possam ser definidas pela expressão

$$\begin{aligned} \begin{bmatrix} \hat{\mathbf{W}}^{\text{out}} & \hat{H} \end{bmatrix} &= \underset{\mathbf{W}}{\text{argmin}} \frac{1}{N} \frac{1}{2\pi} \int_{-\pi}^{\pi} \frac{\sum_{j=1}^N |\hat{F}_j(e^{j\omega})|^2}{|\hat{H}(e^{j\omega})|^2} \begin{bmatrix} \tilde{\mathbf{W}}^{\text{out}}(e^{j\omega}) & H(e^{j\omega}) \end{bmatrix} \\ &\quad \times \begin{bmatrix} \Phi_x(\omega) & \Phi_{xa}(\omega) \\ \Phi_{ax}(\omega) & \sigma_a^2 \end{bmatrix} \begin{bmatrix} \tilde{\mathbf{W}}^{\text{out}}(e^{-j\omega}) \\ H(e^{-j\omega}) \end{bmatrix} d\omega. \end{aligned} \quad (41)$$

Comparando as expressões (41) e (32), com o termo constante $1/N$ sendo movido para dentro da integral, a diferença é apenas o termo

$$\hat{L} = \frac{1}{N} \sum_{j=1}^N |\hat{F}_j(e^{j\omega})|^2. \quad (42)$$

Portanto, as estimativas que minimizam o erro de múltiplos instantes à frente podem ser obtidas com um método de minimização do erro de um instante à frente, considerando \mathbf{y}_e e \mathbf{x}_e filtrados pelo filtro \hat{L} , o qual é considerado fixo. Se $H(z^{-1})$ for conhecido, o método resulta em filtrar $\mathbf{y}_l(k) = \hat{L}(\mathbf{y}(k))$ e $\mathbf{x}_l(k) = \hat{L}(\mathbf{x}(k))$. Quando $H(z^{-1})$ é desconhecido, um processo de identificação preliminar para obter $\hat{H}(z^{-1})$ é necessário. Neste caso, os parâmetros do modelo devem ser estimados com o método do erro de predição de um instante à frente, considerando o filtro $\hat{H}(z^{-1}) = 1$. O modelo estimado é então usado para estimar $\hat{H}(z^{-1})$, o qual pode ser usado para filtrar os dados e estimar o modelo ESNI para predições de múltiplos instantes à frente. O método MRI para modelos ESNI é sumarizado no Algoritmo 1.

Algoritmo 1 Método MRI para modelos ESNI.

- 1: **Se** Modelo para perturbação $H(z^{-1})$ é desconhecido **Faça**
- 2: Obter os estados do reservatório $\mathbf{x}_e(k)$ usando a equação (19);
- 3: Estimar \mathbf{W}^{out} utilizando um método do erro de predição de um instante à frente;
- 4: Estimar a função de transferência $\hat{H}(z^{-1})$;
- 5: **Fim Se**
- 6: Obter o filtro $\hat{L}(z^{-1})$, usando as equações (42) e (36);
- 7: Obter as saídas filtradas $\mathbf{y}_f(k) = \hat{L}(z^{-1})\mathbf{y}(k)$;
- 8: Obter os estados do reservatório $\mathbf{x}_e(k)$ usando a equação (19);
- 9: Obter o estado aumentados do reservatório $\mathbf{x}(k)$ usando a equação (22);
- 10: Obter os estados filtrados do reservatório $\mathbf{x}_f(k) = \hat{L}(z^{-1})\mathbf{x}(k)$;
- 11: Estimar \mathbf{W}^{out} usando as medições filtradas da saída e os estados aumentados filtrados, por meio de um método do erro de predição de um instante à frente.

No caso MIMO, como cada saída do modelo é associada a um horizonte de predição N_i , o filtro $\hat{L}(z^{-1})$ deve ser parametrizado para cada horizonte N_i . Então, existe um filtro $\hat{L}(z^{-1})$ associado com cada saída do processo e vetor de estados aumentado, $\mathbf{x}(k)$.

No restante deste trabalho, quando o modelo utilizar a abordagem MRI no seu algoritmo de treinamento, será utilizado o prefixo MRI (por exemplo, MRI-ESNI), de forma a diferenciar os modelos treinados com algoritmos que utilizam a abordagem MRI dos que não utilizam tal abordagem.

3.4 IDENTIFICAÇÃO DO MODELO ESNI OFFLINE

Esta seção apresenta uma abordagem para identificação de sistemas dinâmicos não lineares a partir de séries de dados obtidas do sistema a ser identificado (*offline*), para modelos de redes com arquitetura ESNI. Na Seção 3.4.1 é apresentado a abordagem de treinamento baseada no método de regressão com regularização. Na seção 3.4.2 é apresentado um exemplo de aplicação da abordagem proposta, identificando um processo de neutralização de pH simulado.

3.4.1 Treinamento da ESNI usando o método de regressão com regularização

Como definido em (21), a saída da ESNI é formada pela combinação linear dos estados do reservatório, saídas passadas, e um termo de *bias*, sendo representada por uma matriz \mathbf{W}^{out} multiplicando o vetor de estados do reservatório, o qual é aumentado por um termo de *bias* e saídas passadas. A equação (21) pode ser reescrita utilizando os exemplos de entrada e saída do processo pela expressão

$$\mathbf{Y}_e = \mathbf{W}^{\text{out}}\mathbf{X}_e, \quad (43)$$

sendo que a matriz $\mathbf{Y}_e \in \mathbb{R}^{m \times T}$ é definida como

$$\mathbf{Y}_e = \begin{bmatrix} \mathbf{y}_e(k) & \mathbf{y}_e(k+1) & \cdots & \mathbf{y}_e(k+T-1) \end{bmatrix}, \quad (44)$$

e $\mathbf{X}_e \in \mathbb{R}^{(r+2m) \times T}$ é uma matriz formada como

$$\mathbf{X}_e = \begin{bmatrix} \mathbf{1} & \mathbf{1} & \cdots & \mathbf{1} \\ \mathbf{x}_e(k) & \mathbf{x}_e(k+1) & \cdots & \mathbf{x}_e(k+T-1) \\ \mathbf{y}_e(k-1) & \mathbf{y}_e(k) & \cdots & \mathbf{y}_e(k+T-2) \end{bmatrix}, \quad (45)$$

ambas geradas por um reservatório excitado por $\mathbf{u}_e(k)$ durante um tempo de treinamento de T amostras (LUKOŠEVIČIUS, 2012).

O processo de aprendizado consiste em encontrar um valor ótimo para $\mathbf{W}^{\text{out}} \in \mathbb{R}^{m \times (1+r+m)}$ o qual minimize o erro médio quadrático entre $\mathbf{y}_e(k)$ e $\mathbf{y}(k)$, para os instantes $k, k+1, \dots, k+T-1$. A matriz \mathbf{Y} é a que contém os valores desejados para as saídas do treinamento, tendo a mesma forma de \mathbf{Y}_e . O processo de treinamento consiste, então, em encontrar a solução do sistema de equações lineares

$$\mathbf{Y} = \mathbf{W}^{\text{out}} \mathbf{X}_e, \quad (46)$$

o qual é geralmente sobredeterminado, quando $T > 1 + n + r$.

A matriz \mathbf{W}^{out} pode ser obtida usando métodos de regressão linear, sendo que para redes ESN, o método de regressão com regularização (do inglês, *ridge regression*) é comumente aplicado, sendo definido como (SHI; HAN, 2007):

$$\begin{aligned} \mathbf{W}^{\text{out}} = \underset{\mathbf{W}^{\text{out}}}{\operatorname{argmin}} \quad & \frac{1}{N_y} \sum_{i=1}^{N_y} \left(\sum_{n=1}^T (y_{e_i}(n) - y_i(n))^2 + \beta \|\mathbf{w}_i^{\text{out}}\|^2 \right) \\ \text{sujeito a} \quad & y_{e_i} = \mathbf{w}_i^{\text{out}} \begin{bmatrix} 1 \\ \mathbf{x}_e \\ y_e \end{bmatrix} \quad \forall i \in 1 \dots m \end{aligned} \quad (47)$$

A função objetivo encontra a matriz \mathbf{W}^{out} que minimiza o erro e, ao mesmo tempo, considera um termo de regularização β , o qual penaliza pesos cuja norma é elevada e é usado para evitar o superajuste do modelo. A solução desse problema é descrita como:

$$\mathbf{W}^{\text{out}} = (\mathbf{X}_e^T \mathbf{X}_e + \beta \mathbf{I})^{-1} \mathbf{X}_e^T \mathbf{Y}. \quad (48)$$

Se considerado o algoritmo MRI, proposto na Seção 3.3, é possível treinar o modelo ESNI para uma tarefa de identificação de múltiplos instantes à frente na qual \mathbf{y}_l e \mathbf{x}_l , que representam as saídas do processo e estados do reservatório filtrados, usando $\hat{L}(z^{-1})$. Dessa forma, é usada a matriz $\mathbf{Y}_l \in \mathbb{R}^{m \times T}$, a qual é definida pela expressão

$$\mathbf{Y}_l = \begin{bmatrix} \mathbf{y}_l(k) & \mathbf{y}_l(k+1) & \cdots & \mathbf{y}_l(k+T-1) \end{bmatrix}, \quad (49)$$

e a matriz $\mathbf{X}_l \in \mathbb{R}^{(1+r+m) \times T}$, descrita por

$$\mathbf{X}_l = \begin{bmatrix} \mathbf{1} & \mathbf{1} & \cdots & \mathbf{1} \\ \mathbf{x}_l(k) & \mathbf{x}_l(k+1) & \cdots & \mathbf{x}_l(k+T-1) \\ \mathbf{y}_l(k-1) & \mathbf{y}_l(k) & \cdots & \mathbf{y}_l(k+T-2) \end{bmatrix}. \quad (50)$$

Com isso, é possível obter \mathbf{W}^{out} em uma abordagem MRI, reescrevendo (48), por meio de

$$\mathbf{W}^{\text{out}} = (\mathbf{X}_l^T \mathbf{X}_l + \beta \mathbf{I})^{-1} \mathbf{X}_l^T \mathbf{Y}_l. \quad (51)$$

3.4.2 Exemplo 1: Identificação de um processo de neutralização de pH com ESN

Para demonstrar a aplicação da estratégia de identificação de sistemas não lineares baseada no modelo MRI-ESNI, foi realizada a identificação do processo de neutralização de pH, o qual é descrito no Apêndice A.1. O modelo fenomenológico foi utilizado como sistema dinâmico real, sendo que as equações diferenciais do sistema dinâmico foram aproximadas utilizando o método de integração numérica Runge-Kutta 45. Por meio de uma simulação do sistema dinâmico foi obtido um conjunto de dados para a identificação do seu comportamento dinâmico. O período de amostragem de 10 segundos foi selecionado para o modelo. Como sinal de excitação foi utilizada uma sequência projetada de forma a conduzir o ponto de operação do processo em diferentes valores dentro da faixa de pH entre 5 e 10.

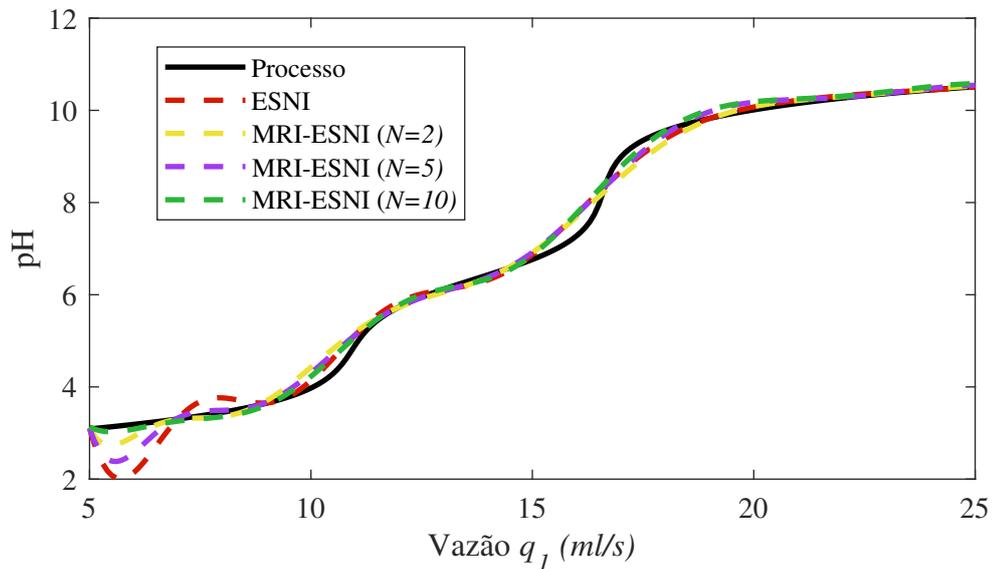
O conjunto de dados foi dividido em três porções distintas, usadas para treinar, validar e testar os modelos. A primeira foi usada diretamente para o treinamento dos modelos. A segunda porção foi utilizada para validação dos modelos treinados, auxiliando na seleção de arquiteturas, hiperparâmetros e do modelo final. Por fim, o conjunto de testes foi usado para comparar a qualidade da predição de todos modelos, sendo usado apenas quando os modelos finais já haviam sido selecionados.

A matriz de conexões do reservatório $\mathbf{W}_{\text{res}}^{\text{res}}$ foi projetada considerando 1000 unidades com taxa de vazamento $\alpha = 0,4$, raio espectral $\rho = 0,99$ e taxa de conexão $c = 0,01$. A matriz de entrada $\mathbf{W}_{\text{in}}^{\text{res}}$ foi inicializada considerando todos pesos iguais a +1 ou -1 (50% para cada caso), e teve seus valores definidos considerando a razão das escalas entre a saída e a entrada do processo, neste caso 1. Para as demais matrizes, a inicialização considerou uma taxa de conexão $c = 1$, e escalamento $v = 1$. O treinamento utilizou regularização com o parâmetro $\beta = 0,001$. Tais valores foram selecionados por meio de um procedimento de busca em grade no qual foram ajustados os principais hiperparâmetros, avaliando como métrica o MSE do modelo para o conjunto de validação.

De forma a avaliar a qualidade do modelo obtido considerando a estratégia de identificação proposta, é utilizada a curva característica estática dos modelos não lineares. Esta curva apresenta o comportamento não linear estático do processo, representando o mapeamento estático entre as entradas e as saídas do processo, sendo interessante para avaliar parte do comportamento do modelo para todos os possíveis pontos de operação, os quais podem não ser cobertos pela série de dados de teste. A forma com que essa curva característica estática é obtida, bem como os resultados para o processo não linear e demais modelos, são detalhados no Apêndice A. Na Figura 11 é ilustrada a comparação entre as curvas características estáticas do processo não linear do modelo ESNI treinado pelo método de regressão com regularização, e do modelo MRI-ESNI, o qual é treinado pelo método de regressão com regularização, considerando o pré-filtro $L(z^{-1})$.

A análise da curva obtida para os modelos MRI-ESNI e ESNI identificados evidencia que a abordagem proposta é capaz de identificar o comportamento geral da variação do ganho estático. Porém, para vazões menores que 10 a curva apresenta uma diferença substancial em

Figura 11 – Comparação entre a curva característica estática de sintonias do modelo MRI-ESNI e modelo ESNI, com a curva do processo de neutralização de pH.



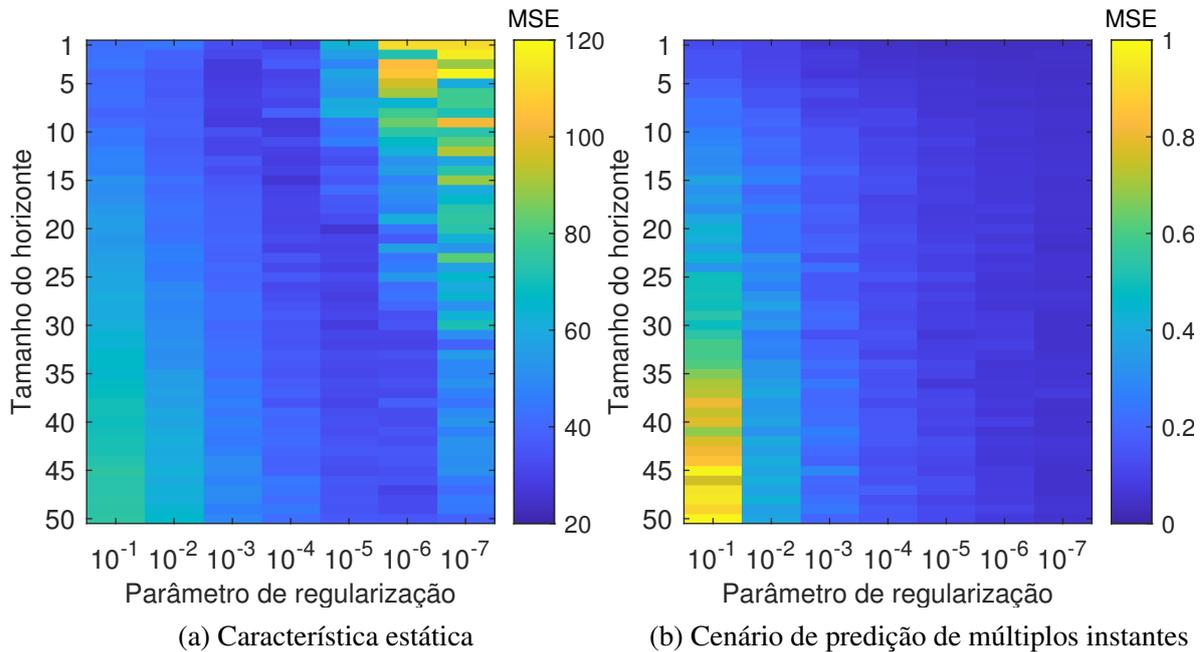
Fonte: dados da pesquisa.

relação à curva ideal. Quando o filtro MRI é utilizado, a discrepância nesta região é amenizada, sendo que a diferença observada, decorrente do número de amostras N do filtro, foi pequena. A abordagem MRI proposta pode ser interpretada como uma forma de regularização do modelo, já que a utilização do filtro $L(z^{-1})$ faz com que a solução encontrada durante o processo de otimização seja ajustada para o problema de predição de múltiplos instantes à frente, resultando, geralmente, em um modelo com menor variância. Para avaliar a influência do uso da abordagem MRI, considerando o ajuste do modelo para diferentes horizontes, foram realizadas simulações variando o parâmetro de regularização e o tamanho do horizonte. Para cada combinação, foram treinados cinco modelos, sendo avaliados para cada modelo dois parâmetros, o MSE para o cenário de predição de múltiplos instantes à frente e o MSE entre a característica estática do modelo e a real do processo. O resultado dessas simulações é apresentado na Figura 12, na qual o desempenho de cada configuração é apresentado em um mapa de calor.

Considerando o ensaio de validação, o melhor desempenho é verificado quando usados os menores valores para o parâmetro de regularização e para o horizonte. Por sua vez, para a característica estática, existe uma região intermediária, a qual resulta nos melhores resultados, sendo que modelos que combinam alto parâmetro de regularização e alto horizonte resultam nas piores métricas. Para este caso, é possível verificar que o uso de um maior horizonte auxilia na regularização do modelo, já que maiores horizontes permitem a utilização de menores parâmetros de regularização. Para todos os parâmetros de regularização, o melhor desempenho para a característica estática foi alcançado com horizontes maiores que 4, evidenciando que a abordagem MRI auxilia no ajuste do modelo para predição de longos horizontes.

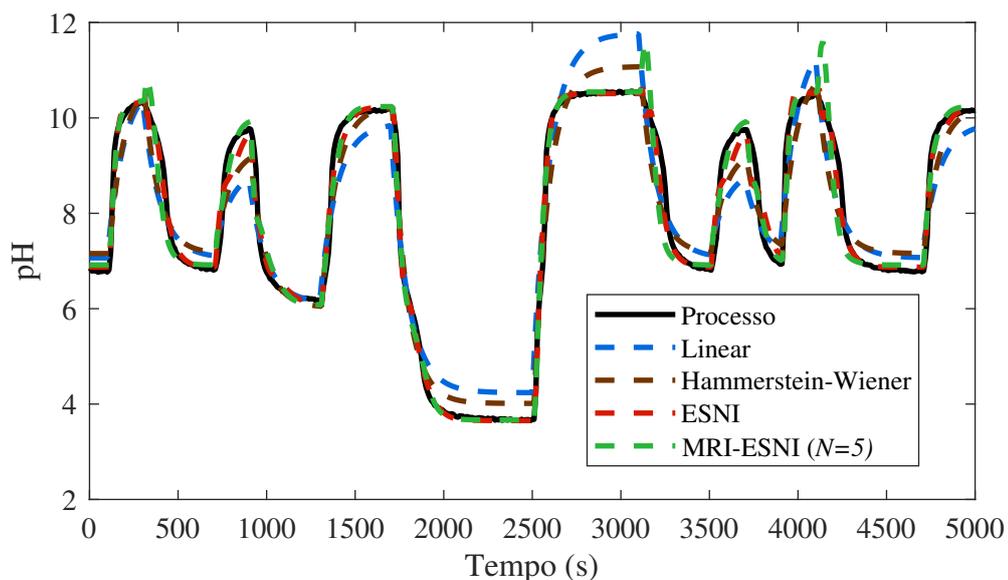
Os resultados obtidos pelo modelo proposto foram também comparados com os obtidos por um modelo linear em seus parâmetros e para o modelo não linear da classe Hammerstein-

Figura 12 – Avaliação do efeito do tamanho do horizonte e parâmetro de regularização no MSE para a característica estática e para o ensaio de teste.



Wiener. A obtenção desses modelos é apresentada em detalhes no Apêndice A. Uma representação gráfica dos resultados obtidos pelo modelo proposto e os utilizados para comparação, no conjunto de testes, é ilustrada na Figura 13, sendo que apenas 5000 s desse conjunto são mostrados.

Figura 13 – Resultado da identificação do processo de neutralização de pH para modelos linear (azul), Hammerstein (marrom), ESNI (vermelho) e MRI-ESNI (verde).



Devido à alta não linearidade do processo de neutralização de pH, o modelo linear apresentou um desempenho ruim, sendo incapaz de representar com precisão a dinâmica do

processo. O modelo ESNI, considerando a abordagem MRI ou não, apresentou boa precisão, modelando bem as não linearidades do sistema, o que resulta em uma representação mais próxima do sistema real. O modelo não linear do tipo Hammerstein apresentou resultados bons em algumas porções do ensaio, porém, erros elevados foram verificados em outras porções do ensaio, como por exemplo entre 2500 s e 3200 s.

O resultado dos modelos é comparado na Tabela 1 por meio do erro quadrático médio (MSE, do inglês *Mean-Squared Error*), coeficiente de determinação R^2 e erro absoluto percentual médio (MAPE, do inglês *Mean Absolute Percentage Error*), métrica baseada em erros relativos.

Tabela 1 – Comparação das métricas de desempenho dos modelos linear, Hammerstein e ESN na tarefa de identificação do processo de neutralização de pH.

Tarefa	Modelo	MSE	R^2	MAPE
Predição de múltiplos instantes à frente	Linear	0,3737	0,9294	3,24%
	Hammerstein-Wiener	0,2410	0,9545	2,57%
	ESNI	0,0623	0,9882	2,17%
	MRI-ESNI ($N = 2$)	0,0714	0,9865	2,41%
	MRI-ESNI ($N = 5$)	0,0871	0,9835	2,80%
	MRI-ESNI ($N = 10$)	0,1084	0,9795	3,19%
Característica estática	Linear	66,16	0,9572	7,22%
	Hammerstein-Wiener	0,1567	0,9796	5,36%
	ESNI	0,0896	0,9883	4,16%
	MRI-ESNI ($N = 2$)	0,0423	0,9945	2,66%
	MRI-ESNI ($N = 5$)	0,0235	0,9933	1,74%
	MRI-ESNI ($N = 10$)	0,0317	0,9968	2,33%

Fonte: dados da pesquisa.

Os modelos baseados na arquitetura ESNI apresentaram melhores resultados, considerando as três métricas de desempenho avaliadas, quando comparados com os modelos linear e Hammerstein-Wiener. Avaliando o coeficiente de determinação R^2 , é possível concluir que os modelos ESNI foram capazes de modelar com melhor fidelidade o processo. Essa conclusão é suportada pelo MAPE, o qual demonstra que o erro médio percentual apresentado pelos modelos linear e Hammerstein-Wiener é maior que o obtido pelo modelo ESN. Quando é feita a comparação entre o modelo ESNI e os modelos MRI-ESNI, fica evidente que a abordagem MRI resulta em uma pequena piora no desempenho, quando é avaliado o cenário de predição, porém, apresenta uma melhora significativa quando é avaliada a característica estática do modelo.

3.5 IDENTIFICAÇÃO DO MODELO ESNI ON-LINE

Esta seção apresenta uma abordagem para identificação on-line de sistemas dinâmicos não lineares, considerando modelos baseados em redes com arquitetura ESNI. Como detalhado na Seção 3.2, o treinamento da matriz \mathbf{W}^{out} , em uma abordagem MRI, pode ser realizado estimando \mathbf{W}^{out} com um método baseado no erro de um instante à frente considerando os dados

de entrada e saída filtrados. Os métodos on-line com estimação recursiva para o treinamento de \mathbf{W}^{out} , baseados na otimização do erro de um instante à frente, já foram propostos na literatura, como detalhado na Seção 2.5.2. Neste estudo, é proposto o uso de um método dos mínimos quadrados recursivo, semelhante aos métodos encontrados na literatura, porém, é utilizado um fator de esquecimento adaptativo direcional.

Na Subseção 3.5.1 é apresentado um método para o treinamento da ESNI, utilizando o método dos mínimos quadrados recursivo com fator de esquecimento direcional. Na Subseção 3.5.2 é apresentado um exemplo de aplicação da abordagem proposta, identificando um tanque de perfil cônico.

3.5.1 Treinamento do mecanismo de leitura do reservatório por método dos mínimos quadrados recursivo com fator de esquecimento direcional

O objetivo da tarefa de identificação é estimar a matriz de saída do modelo ESNI, $\mathbf{W}^{\text{out}}(k)$, à medida que dados novos são adquiridos. Por meio do método dos mínimos quadrados recursivo, esse objetivo pode ser alcançado atualizando $\mathbf{W}^{\text{out}}(k)$ por meio da expressão

$$\mathbf{W}^{\text{out}}(k) = \mathbf{W}^{\text{out}}(k-1) + \frac{\mathbf{P}(k)\mathbf{x}_I(k-1)}{1 + \xi(k)} \hat{e}(k), \quad (52)$$

na qual

$$\xi(k) = \mathbf{x}_I(k-1)^T \mathbf{P}(k) \mathbf{x}_I(k-1), \quad (53)$$

$$\hat{e}(k) = \mathbf{y}_I(k) - \mathbf{W}^{\text{out}}(k-1) \mathbf{x}_I(k-1). \quad (54)$$

A matriz de covariância \mathbf{P} é atualizada, durante a estimação on-line da ESNI, por meio da equação (BOBÁL *et al.*, 2006)

$$\mathbf{P}(k) = \mathbf{P}(k-1) - \frac{\mathbf{P}(k-1)\mathbf{x}_I(k)\mathbf{x}_I^T(k)\mathbf{P}(k-1)}{\epsilon^{-1}(k) + \xi(k)}, \quad (55)$$

$$\epsilon(k) = \varphi(k) - \frac{1 - \varphi(k)}{\xi(k-1)}, \quad (56)$$

quando $\xi(k) > 0$, ou mantida constante nos demais casos, ou seja

$$\mathbf{P}(k) = \mathbf{P}(k-1) \quad \text{caso} \quad \xi(k) \leq 0. \quad (57)$$

A variável $\varphi(k) \in (0, 1]$ representa o fator de esquecimento, o qual pode assumir valores diferentes durante o processo de estimação. O fator de esquecimento é utilizado para fazer com que as estimativas sejam mais representativas das medições atuais, permitindo que o algoritmo de estimação dê menos importância para medições passadas. Para atualizar o fator de esquecimento durante o processo de estimação é utilizado um algoritmo direcional, o qual permite que o algoritmo de estimação atualize os parâmetros do processo na direção em que novos dados são adquiridos. O fator de esquecimento é atualizado com o algoritmo direcional por meio da

equação (KULHAVÝ, 1987)

$$\begin{aligned} \varphi(k)^{-1} &= 1 + (1 + \rho)[\ln(1 + \xi(k))] \\ &+ \left(\frac{(\nu(k) + 1)\eta(k)}{1 + \xi(k) + \eta(k)} - 1 \right) \frac{\xi(k)}{1 + \xi(k)}, \end{aligned} \quad (58)$$

na qual

$$\eta(k) = \frac{\hat{e}(k)^2}{\zeta(k)}, \quad (59)$$

$$\nu(k) = \varphi(k)[(\nu(k-1) + 1)], \quad (60)$$

$$\zeta(k) = \varphi(k) \left(\lambda(k-1) + \frac{\hat{e}(k)^2}{1 + \xi(k)} \right). \quad (61)$$

O parâmetro $\rho \in [0, 1]$ é usado para definir a quantidade de informação passada que será preservada em cada iteração, com valores pequenos representando menos informação sendo descartada (BOBÁL *et al.*, 2006). O algoritmo de estimação on-line para o modelo ESNI considerando o método dos mínimos quadrados recursivo com fator de esquecimento direcional é sumarizado no Algoritmo 2.

Algoritmo 2 Estimação on-line da ESNI com o método dos mínimos quadrados recursivo usando fator de esquecimento direcional.

- 1: Medir a saída atual do processo;
 - 2: Obter os estados filtrados do reservatório $\mathbf{x}_l(k)$ por meio do método MRI para ESNI;
 - 3: Obter o fator de esquecimento direcional $\varphi(k)^{-1}(k)$ usando a equação definida em (58);
 - 4: Obter a matriz de covariância $\mathbf{P}(k)$ usando a equação (55);
 - 5: Obter $\mathbf{W}^{\text{out}}(k)$ com a equação (52);
 - 6: Retornar ao passo 1.
-

3.5.2 Exemplo 2: Identificação de um tanque de perfil cônico

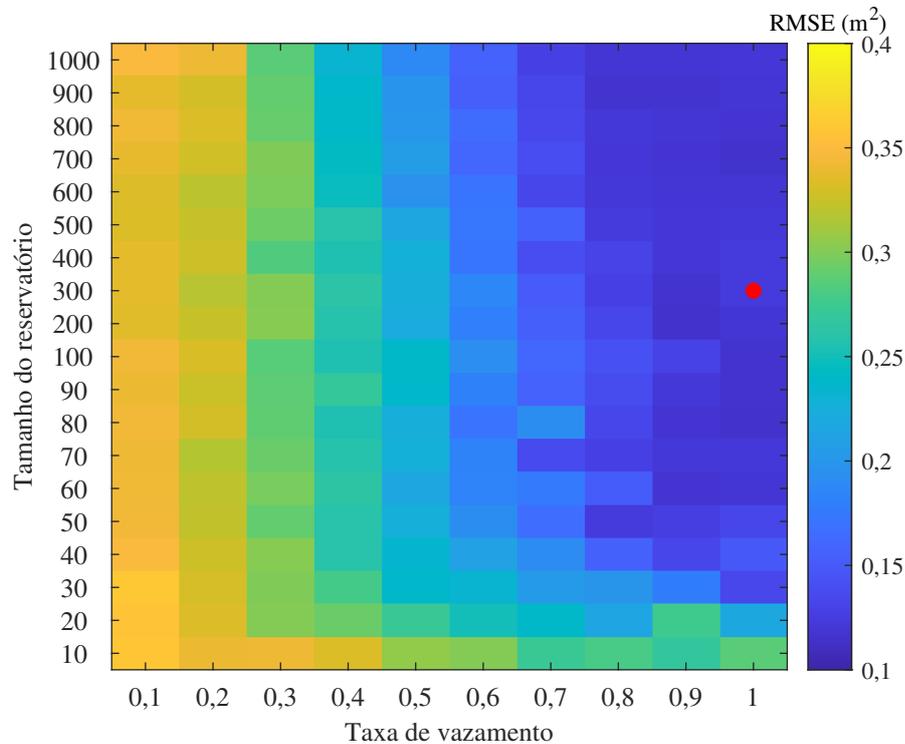
Este exemplo apresenta a aplicação da técnica de identificação on-line proposta para o modelo ESNI, avaliando o resultado da abordagem on-line e também da técnica de identificação MRI proposta. O processo identificado consiste em um tanque de perfil cônico, no qual nível de fluido do tanque é a saída do processo, sendo manipulada pela vazão de saída do tanque. A descrição geral do processo, de sua simulação e da identificação de alguns modelos de comparação é apresentada no Apêndice A.3.

O modelo ESNI foi criado considerando a estrutura apresentada na Seção 3.3. Para a criação do reservatório, foi considerado uma quantidade de neurônio $r = 300$, sendo inicializado considerando uma matriz de pesos esparsos $\mathbf{W}_{\text{res}}^{\text{res}}$, com 1% de conexões não nulas, novamente obtidas utilizando uma distribuição uniforme, considerando o intervalo $[-1, +1]$. Após a inicialização, a matriz $\mathbf{W}_{\text{res}}^{\text{res}}$ foi multiplicada por um escalar para garantir um raio espectral $\rho(\mathbf{W}_{\text{res}}^{\text{res}}) = 0,99$. A matriz de entrada, $\mathbf{W}_{\text{in}}^{\text{res}}$ foi criada considerando todos pesos iguais a +1 ou -1 (50% para cada caso), sendo posteriormente multiplicada por um escalar considerando

a razão das escalas entre a saída e a entrada do processo, neste caso 20. O estimador recursivo teve seus parâmetros inicializados considerando: $N = 10$; elementos da diagonal principal da matriz de covariância $\mathbf{P}_{ii}(0) = 100$; $\varphi(0) = 1$; $\lambda(0) = 0,001$; e $\nu(0) = 10^{-6}$.

A sintonia dos principais hiperparâmetros do modelo foi realizada por meio de uma busca em grade, a qual considerou três parâmetros: o tamanho do reservatório; a taxa de vazamento; e o raio espectral da matriz $\mathbf{W}_{\text{res}}^{\text{res}}$. Uma representação gráfica desse processo de ajuste dos hiperparâmetros é apresentada na Figura 14, a qual apresenta o impacto no desempenho do modelo considerando variações na taxa de vazamento e tamanho do reservatório, considerando o raio espectral $\rho(\mathbf{W}_{\text{res}}^{\text{res}}) = 0,99$.

Figura 14 – Representação gráfica do impacto da sintonia da taxa de vazamento e do tamanho do reservatório no RMSE final do modelo para a tarefa de predição de múltiplos instantes à frente. O círculo vermelho representa a sintonia escolhida para o modelo final.



Fonte: dados da pesquisa.

Para avaliar o desempenho do modelo ESNI estimado pelo método proposto, considerando a tarefa de predição de múltiplos instantes à frente, alguns modelos de comparação foram considerados. O primeiro deles é um modelo, em tempo discreto, linear em seus parâmetros, descrito pela expressão

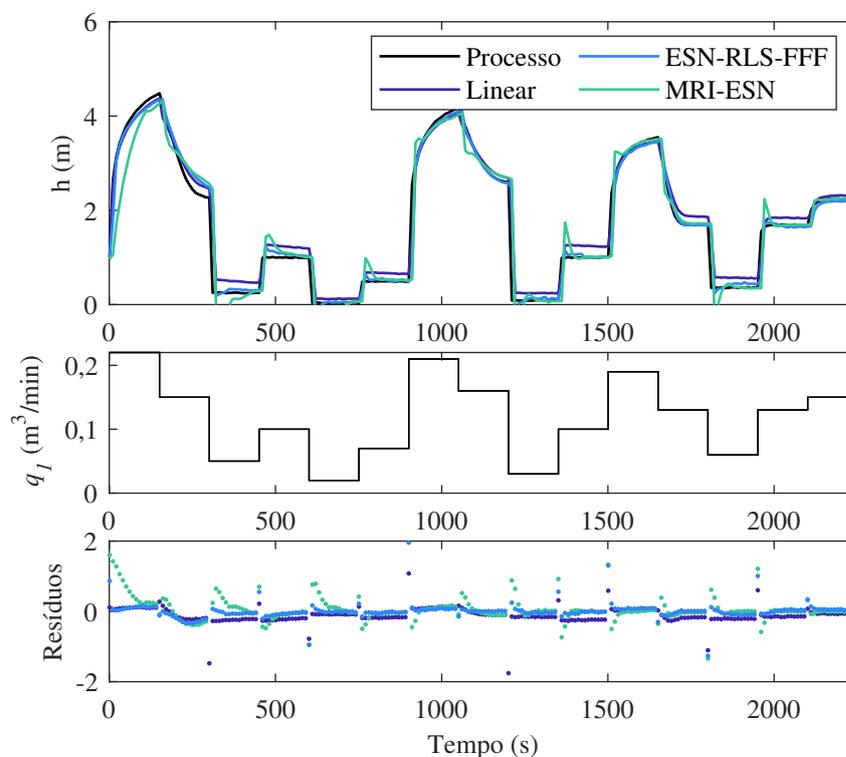
$$y_l(k) = -a_1 y_l(k-1) + b_1 u(k-1), \quad (62)$$

no qual a_1 e b_1 são estimados utilizando o método RLS com fator de esquecimento direcional. O segundo modelo usado como comparação é uma ESN com seus parâmetros estimados pelo algoritmo RLS usando um fator de esquecimento fixo (ESN-RLS-FFF), como proposto em Jaeger (2003). O terceiro consiste em uma ESN com seus parâmetros estimado pelo algoritmo

RLS com um fator de esquecimento adaptativo (ESN-RLS-AFF) (SONG *et al.*, 2011b). O quarto e o quinto modelos são uma ESN estimada com um algoritmo RLS esparsado, sendo que o quarto usa como penalidade a norma ℓ_0 (ESN-RLS- ℓ_0), enquanto o quinto usa ℓ_1 (ESN-RLS- ℓ_1), ambos seguindo a formulação proposta em Yang *et al.* (2019). O sexto modelo usado para comparação consiste em uma ESN estimada utilizado um filtro de Kalman (ESN-KF), modelo que é parte da abordagem de ESN dual, proposta em Sheng *et al.* (2012). Todos modelos utilizaram a mesma arquitetura de reservatório que o modelo proposto e consideraram a mesma inicialização do algoritmo de estimação, quando possível, ou uma inicialização que buscava replicar a inicialização utilizada pelo algoritmo proposto.

Para o treinamento dos modelos foi criado um cenário de estimação on-line que consiste na simulação do processo considerando mudanças na entrada do processo, na qual a faixa de variação é $q_1 = 0, 25\text{m}^3/\text{min}$, com 15 mudanças nos valores de vazão de entrada. O resultado dessa simulação é apresentado na Figura 15, em conjunto com os resíduos dos modelos para a tarefa de predição de um instante à frente. Por simplicidade, os resultados de apenas dois modelos de comparação são apresentados, porém, os desempenhos de todos modelos de comparação são detalhados na análise dos resultados.

Figura 15 – Simulação do cenário para identificação on-line do tanque de perfil cônico. Resultados mostram o desempenho dos modelos para a tarefa de predição de um instante à frente.



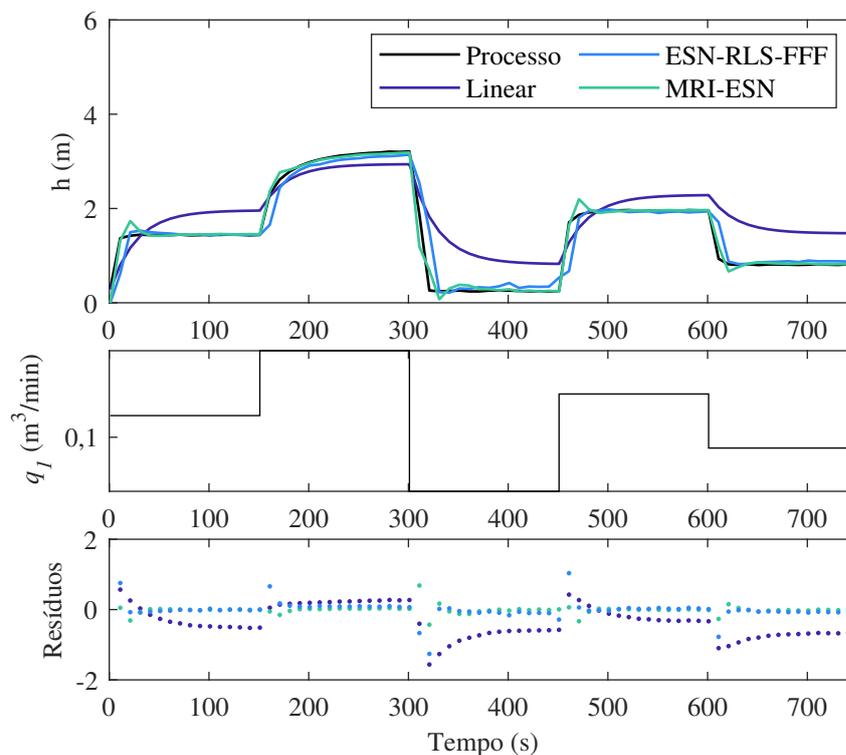
Fonte: dados da pesquisa.

A avaliação dos resultados considera o desempenho dos modelos para a tarefa de predição de um instante à frente, obtido utilizando o resultado do cenário usado para a estimação dos modelos, e também considerando o desempenho para a tarefa de predição de múltiplos instantes

à frente, por meio de um cenário de predição de múltiplos instantes à frente. Esse cenário avalia a efetividade da abordagem MRI, já que a tarefa de predição de múltiplos instantes à frente é mais próxima da que será encontrada quando o modelo for usado em uma implementação MPC. A curva da característica estática dos modelos também é utilizada nessa etapa de avaliação, para demonstrar o quão bem o modelo foi capaz de estimar a característica não linear estática do modelo.

Devido ao fato de o processo de estimação atualizar o modelo à medida que novas informações são adquiridas, é possível verificar como o desempenho do modelo evoluiu durante a simulação usada para estimação. Para isso, o modelo estimado em cada instante da simulação foi armazenado, sendo então avaliado usando o cenário de predição de múltiplos instantes à frente. Esse cenário consiste na simulação do processo considerando 5 variações na entrada do processo, no qual o modelo, após a devida inicialização, é testado na configuração paralela, ou seja, sem o conhecimento das saídas medidas do processo. Esse teste emula o comportamento esperado em uma aplicação MPC, já que, nesse tipo de tarefa, as predições futuras do comportamento do processo são necessárias. Dessa forma, a configuração paralela é necessária pois as saídas futuras do processo não são conhecidas no instante em que a predição é realizada. Os resultados obtidos pelos modelos, considerando os modelos estimados ao fim do processo de estimação, são apresentados na Figura 16. Novamente, por simplicidade, apenas os resultados obtidos para dois modelos de comparação são apresentados, porém, os resultados de todos os modelos são apresentados na discussão.

Figura 16 – Simulação do cenário de predição de múltiplos instantes à frente.

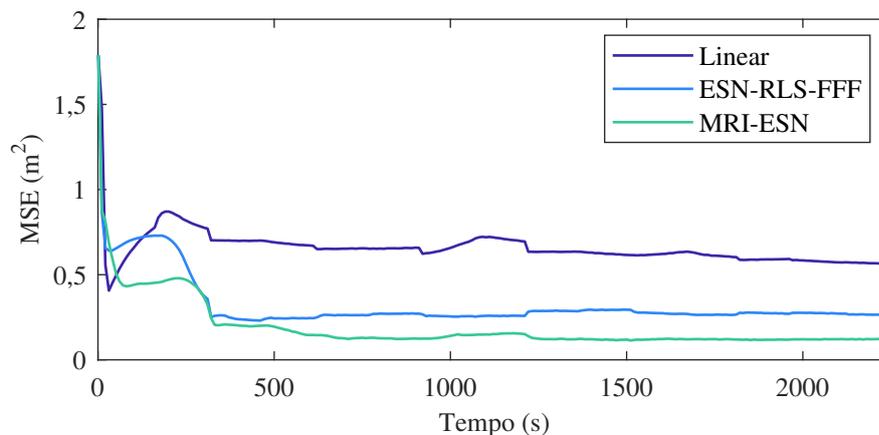


Fonte: dados da pesquisa.

Como verificado na Figura 15, os modelos de comparação, os quais consideram em seu algoritmo de estimação o critério de otimização baseado no erro de predição de um instante à frente, apresentaram melhores resultados que o modelo MRI-ESNI. Porém, quando é considerada a tarefa de predição de múltiplos instantes à frente, a qual é apresentada na Figura 16, o modelo MRI-ESNI apresentou os melhores resultados.

Para avaliar o processo de estimação do modelo proposto, o desempenho dos modelos estimados a cada instante de tempo do cenário de estimação pode ser avaliado. Para isso, é obtido a raiz do erro médio quadrático (RMSE do inglês, *Root Mean Squared Error*), correspondendo a cada um destes modelos testado usando o cenário obtido ao testar cada um desses modelos com o cenário de predição de múltiplos instantes à frente. Uma representação gráfica da evolução do desempenho dos modelos é apresentada na Figura 17. Cada um dos valores de RMSE corresponde ao desempenho de um modelo obtido a cada período de amostragem do ensaio apresentado na Figura 15. O resultado para o modelo obtido no último instante de tempo ($t = 2250$ s) é apresentado na Figura 16.

Figura 17 – Evolução do desempenho (RMSE), para o cenário de estimação de múltiplos instantes à frente, dos modelos estimados a cada instante de tempo do cenário de estimação.



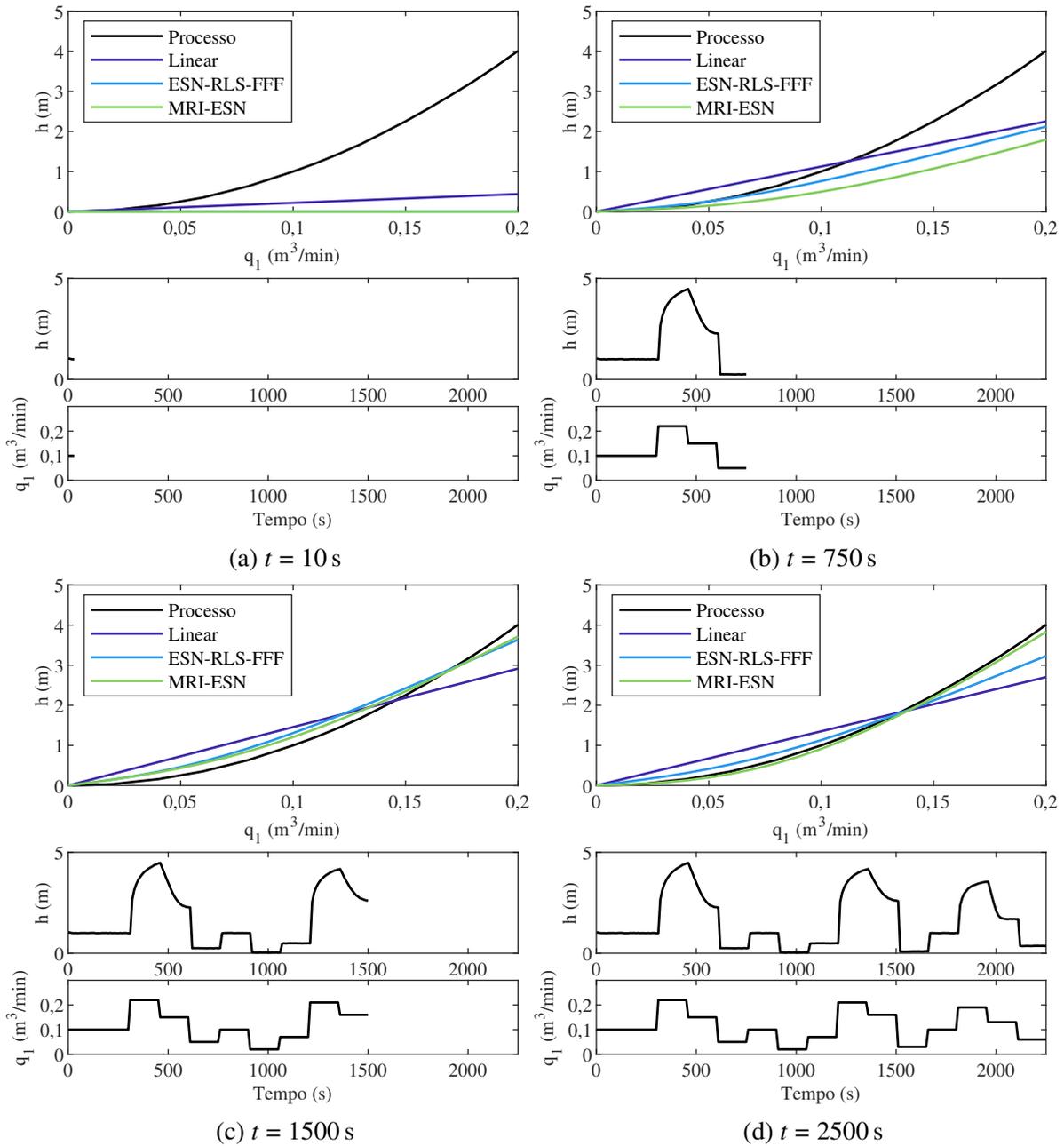
Fonte: dados da pesquisa.

O índice de desempenho RMSE de todos modelos inicia ruim, com valores elevados, já que todos iniciam com nenhum conhecimento acerca da dinâmica do processo. A melhora do indicador durante o processo de estimação indica que os parâmetros de todos modelos apresentados na Figura 17 apresentam uma taxa de convergência semelhante. O modelo proposto apresenta indicadores melhores para praticamente todo o processo de estimação, apresentando um RMSE menor que todos os modelos de comparação considerados neste estudo.

Outra forma de avaliar a qualidade dos modelos consiste em avaliar a curva de característica estática. Foram avaliados os modelos obtidos em quatro instantes de tempo da simulação de estimação, sendo que a curva de característica estática para cada um desses instantes é apresentada na Figura 18.

Por inspeção visual, é possível observar que o modelo ESNI evolui durante o processo de

Figura 18 – Comparação da característica estática não linear do processo para os modelos MRI-ESNI, linear e ESN-RLS-FFF, considerando os parâmetros dos modelos estimados nos instantes de tempo $t = 10\text{ s}$, $t = 750\text{ s}$, $t = 1500\text{ s}$ e $t = 2250\text{ s}$.



Fonte: dados da pesquisa.

identificação on-line, coletando mais informação das diversas condições de operação à medida que nova informação está disponível. Por sua vez, o modelo ESN-RLS-FFF, o qual representa os modelos de comparação não lineares nessa análise, apresentou comportamento semelhantes, porém, o modelo proposto foi capaz de se aproximar melhor do resultado ideal. O modelo linear apresenta uma capacidade de modelagem menor que os demais modelos, dado que ele é capaz de identificar apenas uma aproximação média do ganho estático do processo.

Os resultados finais para o modelo proposto e para as alternativas testadas para fins de comparação, são sumarizados na Tabela 2. O RMSE e o coeficiente de determinação (R^2) foram usados como métricas de desempenho, sendo avaliados os cenários de predição de um instante à frente e de predição de múltiplos instantes à frente. O cenário de um instante à frente considera as predições obtidas a cada instante da simulação usada para estimação dos modelos. O cenário de múltiplos instantes à frente, por sua vez, considera o modelo obtido no último instante da simulação usada para estimação dos modelos, o qual é testado usando o cenário apresentado na Figura 16.

Tabela 2 – Resultados da identificação on-line do tanque de perfil cônico.

Tarefa	Modelo	RMSE (m ²)	R^2	Tempo de cômputo (ms)
Predição de um instante à frente	Linear	0,39	0,92	0,067
	ESN-RLS-FFF	0,45	0,90	6,0
	ESN-RLS-AFF	0,43	0,91	6,8
	ESN-RLS- ℓ_0	0,45	0,90	6,3
	ESN-RLS- ℓ_1	0,44	0,91	6,2
	ESN-RLS-KF	0,44	0,90	6,2
	ESN-RLS-DFF	0,44	0,91	6,6
	MRI-ESNI ($N = 1$)	0,49	0,87	6,9
	MRI-ESNI ($N = 5$)	0,50	0,88	7,1
	MRI-ESNI ($N = 10$)	0,53	0,86	7,3
Predição de múltiplos instantes à frente	Linear	0,45	0,90	0,067
	ESN-RLS-FFF	0,27	0,92	6,0
	ESN-RLS-AFF	0,22	0,94	6,8
	ESN-RLS- ℓ_0	0,28	0,91	6,3
	ESN-RLS- ℓ_1	0,26	0,92	6,2
	ESN-RLS-KF	0,29	0,91	6,2
	ESN-RLS-DFF	0,23	0,94	6,6
	MRI-ESNI ($N = 1$)	0,24	0,94	6,9
	MRI-ESNI ($N = 5$)	0,13	0,98	7,1
	MRI-ESNI ($N = 10$)	0,12	0,98	7,3

Fonte: dados da pesquisa.

As métricas de desempenho indicam que o modelo MRI-ESNI proposto não é a melhor escolha quando a tarefa consiste na predição de um instante à frente. Porém, quando a tarefa de predição é de múltiplos instantes à frente, o modelo proposto apresenta um desempenho melhor que todos os modelos de comparação, com RMSE menor e um coeficiente de determinação R^2

maior. Os melhores resultados são alcançados com valores de N maiores, já que os resultados do modelo MRI-ESNI para a tarefa de predição de múltiplos instantes à frente melhora, enquanto os resultados para a tarefa de predição de um instante à frente não sofrem modificação significativa. Esse comportamento ocorre devido ao fato de o modelo MRI-ESNI utilizar um algoritmo de estimação cujo critério de otimização considera o erro de múltiplos instantes à frente. Quanto à complexidade computacional, os tempos de cômputo máximos requeridos pelo modelo MRI-ESNI foram 7,3 ms, enquanto o tempo demandado pelos modelos ESN de comparação foram entre 6,0 ms e 6,8 ms. O uso do filtro MRI apresenta um impacto pequeno no tempo de cômputo, e de forma geral, o modelo proposto apresenta tempos compatíveis com aplicações de controle.

3.6 COMENTÁRIOS FINAIS

Neste capítulo foi explorada a utilização de uma formulação para identificação de sistemas dinâmicos não lineares, relevante ao MPC. Essa formulação se baseou em um modelo com estrutura ESNI, a qual considera uma rede ESN acrescida de um modelo de perturbação. Considerando a formulação MRI-ESNI proposta, foram descritos métodos para o treinamento do modelo considerando abordagens baseadas em dados previamente obtidos do processo e treinamento on-line do modelo. A avaliação da abordagem MRI e do modelo ESNI foi realizada utilizando simulações de um processo de neutralização de pH em de um tanque de perfil cônico.

Quanto à utilização da abordagem MRI associada ao modelo ESNI, foi observado que o desempenho do modelo, quando avaliado em um cenário de predição de múltiplos instantes à frente, foi superior ao obtido pelo mesmo modelo, sem a utilização da abordagem MRI. Em contrapartida, o desempenho para a tarefa de predição é afetado, com o modelo MRI-ESNI sendo superado por uma abordagem ESNI. Esse é um comportamento esperado, já que a abordagem MRI é alcançada utilizando um pré-filtro nas séries de dados, o qual atenua informações de alta frequência. Essa atenuação faz com que o problema de identificação priorize as dinâmicas de média e baixa frequências, resultando em um modelo mais adequado à predição de múltiplos instantes à frente. Esse comportamento ficou evidente quando verificada a característica estática não linear dos modelos identificados, visto que o modelo MRI-ESNI se adequou melhor à característica estática real do processo.

A utilização do modelo ESNI, identificado utilizando a abordagem MRI, é explorada no decorrer do trabalho. Uma formulação NMPC prática, utilizando o modelo MRI-ESNI, é apresentada na Seção 4.3, no qual são detalhados o procedimento para obtenção das predições e a formulação do problema de otimização para obtenção do sinal de controle. A abordagem para identificação on-line do modelo ESNI também é explorada em uma abordagem NMPC prática, sendo essa formulação apresentada na Seção 4.4.

4 ALGORITMOS NMPC PRÁTICOS COM REDES DE ESTADO DE ECO

Sumário

4.1	Introdução	69
4.2	Formulação do problema NMPC	70
4.3	Algoritmo NMPC prático para modelos ESNI	73
4.3.1	Predição não linear e linearização	73
4.3.2	Exemplo 3: Controle de um reator de neutralização de pH com PNMPC-ESNI	79
4.3.3	Exemplo 4: Controle de uma bancada de ensaio de compressores herméticos com PNMPC-ESNI	84
4.4	Algoritmo NMPC prático adaptativo para modelos ESNI	86
4.4.1	Estimação on-line do modelo ESNI e integração com PNMPC	87
4.4.2	Exemplo 5: Controle adaptativo de um tanque cônico	88
4.4.3	Exemplo 6: Controle adaptativo de um motor síncrono de ímãs permanentes	94
4.5	Algoritmo NMPC prático com predição iterativa para modelos ESNI	102
4.5.1	Predição iterativa com respostas base e ótima	103
4.5.2	Exemplo 7: Controle de um processo de neutralização de pH	108
4.6	Comentários Finais	112

4.1 INTRODUÇÃO

A rede de estado de eco é uma arquitetura de rede neural recorrente que apresenta bons resultados para a identificação de sistemas dinâmicos não lineares desconhecidos, seja em abordagens em que o modelo é obtido a partir de dados obtidos da operação do processo ou em abordagens em que o modelo é obtido on-line, como apresentado no Capítulo 3. O uso de modelo ESN para predição em controladores MPC já foi explorado em alguns trabalhos na literatura, como detalhado na Seção 2.4, porém, a utilização de abordagens NMPC práticas nas quais o modelo ESN é linearizado numericamente ainda não foi explorada.

Este capítulo explora a utilização de modelos baseados na ESN para predição em controladores NMPC práticos, baseados na linearização numérica do modelo não linear. São propostos algoritmos NMPC práticos baseados em modelos com arquitetura ESNI, a qual é detalhada na Seção 3.2, sendo apresentados os procedimentos para obtenção da resposta livre não linear e da resposta forçada, por meio da qual é construída a matriz de dinâmica do modelo. Adicionalmente, são detalhadas a construção e a solução do problema de otimização associado ao

controlador PNMPC. A utilização da abordagem proposta é exemplificada por meio do controle de processos simulados e processos reais.

As contribuições à literatura detalhadas neste capítulo consistem na proposição de:

- um algoritmo NMPC prático com predição não linear e linearização baseado no modelo ESNI;
- um algoritmo NMPC prático adaptativo com modelo ESNI estimado on-line pelo método dos mínimos quadrados recursivo;
- um algoritmo NMPC prático com predição não linear iterativa e linearização baseado no modelo ESNI;
- diferentes estratégias para obtenção da matriz de dinâmicas, permitindo a obtenção de versões mais fiéis ao modelo não linear e versões mais computacionalmente eficiente.

Os principais desenvolvimentos apresentados neste capítulo foram publicados nos seguintes artigos:

- SCHWEDERSKY, B. B.; FLESCHE, R. C. C.; ROVEA, S. B. Adaptive Practical Nonlinear Model Predictive Control for Echo State Network Models. **IEEE Transactions on Neural Networks and Learning Systems**. Institute of Electrical and Electronics Engineers (IEEE), *No prelo*. DOI: 10.1109/TNNLS.2021.3109821.
- SCHWEDERSKY, B. B.; FLESCHE, R. C. C.; DANGUI, H. A. C.; IERVOLINO, L. A. Practical Nonlinear Model Predictive Control Using an Echo State Network Model. In: 2018 INTERNATIONAL JOINT CONFERENCE ON NEURAL NETWORKS (IJCNN). Rio de Janeiro, Brasil. **Proceedings [...]**. New York: IEEE, 2018. DOI: 10.1109/IJCNN.2018.8489446.

Este capítulo é estruturado da seguinte forma. Na Seção 4.2 é apresentado o problema de controle NMPC. Por sua vez, na Seção 4.3 é proposto um algoritmo de controle NMPC prático com predição não linear e linearização cujo modelo de predição é baseado em um modelo ESNI, considerando a estrutura proposta na Seção 3.2. Na Seção 4.4 é apresentado um algoritmo de controle NMPC prático adaptativo, considerando um modelo ESNI estimado pelo algoritmo dos mínimos quadrados recursivo. Já na Seção 4.5 é proposto um algoritmo NMPC prático com predição não linear iterativa e linearização baseado no modelo ESNI. Por fim, na Seção 4.6 são apresentados alguns comentários finais acerca da abordagem de controle proposta.

4.2 FORMULAÇÃO DO PROBLEMA NMPC

Neste trabalho, é considerado que o objetivo da estratégia de controle é levar a trajetória da saída de um processo, $\mathbf{y} \in \mathbb{R}^m$, sujeita a perturbações não mensuráveis, $\mathbf{d} \in \mathbb{R}^m$, para um valor desejado de referência, $\mathbf{r} \in \mathbb{R}^m$, por meio da variação de um sinal de controle, $\mathbf{u} \in \mathbb{R}^n$. As

abordagens de controle preditivo alcançam esse objetivo utilizando um modelo do processo para prever o comportamento futuro do mesmo. Com base no modelo de predição e nas restrições operacionais, elas obtêm a ação de controle a ser aplicada no processo por meio da solução de um problema de otimização. Dessa forma, os controladores MPC podem ter sua operação dividida em duas porções com funções distintas: um bloco responsável por obter predições do comportamento futuro do processo; e um bloco responsável por resolver um problema de otimização, o qual definirá uma sequência de ações de controle futuras que minimize uma função custo, que tipicamente conta com uma soma ponderada do erro de seguimento de referência e dos esforços de controle futuros.

A sequência de predições para uma dada saída do processo $i \in [1, m]$, obtida no instante de tempo k , em um horizonte de predição é representada como $\hat{y}_i(k+q | k)$, para $q = N_{1,i} \dots N_{2,i}$. Nas formulações consideradas neste trabalho, $k+N_{1,i}$ é o primeiro instante de tempo do horizonte de predição para a i -ésima saída do processo e $N_{2,i}$ é o último. A sequência de ações de controle ótimas, para uma entrada $j \in [1, n]$, $u_j(k+s)$, é obtida pelo controlador MPC, para um dado horizonte de controle $N_{u,j}$ no qual ($s = 0 \dots N_{u,j} - 1$), ao minimizar um funcional do erro de seguimento de referência e do esforço de controle. Apenas a primeira ação de controle, obtida por meio da minimização do função custo, é aplicada como sinal de controle do processo, sendo que na próxima iteração do algoritmo todo o processo é repetido, para obtenção do próximo sinal de controle (CAMACHO; BORDONS, 2007).

Definindo $N_i = N_{2,i} - N_{1,i} + 1$ como horizonte de predição associado à saída i , o vetor de predições da saída, tomadas no instante de tempo k , para uma dada saída i , é escrito como

$$\hat{\mathbf{y}}_i = \left[\hat{y}_i(k + N_{1,i} | k) \quad \dots \quad \hat{y}_i(k + N_{2,i} | k) \right]^T; \quad (63)$$

o vetor contendo a sequência de referências futuras para uma dada saída i , conhecidas no instante de tempo k , é definido como

$$\hat{\mathbf{r}}_i = \left[\hat{r}_i(k + N_{1,i} | k) \quad \dots \quad \hat{r}_i(k + N_{2,i} | k) \right]^T; \quad (64)$$

e o vetor de incrementos de controle, obtidos no instante de tempo k , para uma dada entrada j , é definido como

$$\Delta \mathbf{u}_j = \left[\Delta u_j(k | k) \quad \dots \quad \Delta u_j(k + N_{u,j} - 1 | k) \right]^T. \quad (65)$$

Uma expressão geral para a função custo pode ser definida como

$$J = \sum_{i=1}^m [\hat{\mathbf{y}}_i - \mathbf{r}_i]^T \delta_i [\hat{\mathbf{y}}_i - \mathbf{r}_i] + \sum_{j=1}^n \Delta \mathbf{u}_j^T \lambda_j \Delta \mathbf{u}_j, \quad (66)$$

sendo que os coeficientes δ_i e λ_j representam as ponderações para o erro de seguimento de referência e para o esforço de controle, respectivamente. Para o caso SISO, um aumento na relação $\delta(j)/\lambda(j)$ prioriza a resposta transitória, enquanto uma diminuição penaliza a variação do sinal de controle, resultando em sinais de controle mais suaves. Para o caso MIMO, a ideia é a

mesma, porém também são consideradas ponderações relativas entre os erros de seguimento de referência e sinais de controle (CAMACHO; BORDONS, 2007).

O vetor de incrementos de controle futuros é definido como

$$\Delta \mathbf{u} = \left[\Delta \mathbf{u}_1^T \quad \dots \quad \Delta \mathbf{u}_n^T \right]^T, \quad (67)$$

sendo que esses incrementos de controle são obtidos por meio da minimização da função custo descrita em (66). O problema de otimização associado a este problema MPC é formulado como

$$\begin{aligned} \min_{\Delta \mathbf{u}} \quad & J \\ \text{sujeito a} \quad & \mathbf{A} \Delta \mathbf{u} \leq \mathbf{b}, \end{aligned} \quad (68)$$

na qual a matriz \mathbf{A} e o vetor \mathbf{b} definem as restrições do problema de otimização. Detalhes acerca das restrições tipicamente utilizadas e da estrutura de \mathbf{A} e \mathbf{b} são encontrados em Camacho e Bordons (2007).

Quando o modelo usado para obtenção das predições $\hat{\mathbf{y}}_i$ é não linear, o problema em (68) necessita de métodos de programação não linear para sua solução. Para evitar a necessidade da solução de um NLP, os algoritmos propostos nesta tese se baseiam em abordagens de controle NMPC práticas, nas quais um modelo não linear é empregado para predição do comportamento futuro do processo, sendo que esse modelo é utilizado de duas formas distintas: o modelo não linear original é usado para prever a porção do comportamento futuro do processo que dependa apenas das ações de controle conhecidas até o instante de tempo atual; e uma versão linearizada do modelo não linear, obtida a cada período de amostragem, é usada para prever a porção do comportamento futuro do processo que depende das ações de controle futuras. Dois exemplos de formulações que exploram esse tipo de abordagem são o algoritmo PNMPC apresentado em Plucenio *et al.* (2007), formulação da qual deriva o algoritmo proposto neste trabalho, e o algoritmo MPC-NPL, apresentado em Ławryńczuk (2007). Essa divisão do modelo de predição é realizada de forma a tornar a função custo em (66) afim em relação às ações de controle. Dessa forma, é possível empregar técnicas de programação quadrática para solução do problema em (68). A principal vantagem dessa proposta em relação à resolução do problema NLP é que técnicas de programação quadrática apresentam uma solução fechada, quando o problema é irrestrito, e apresentam algoritmos computacionalmente eficientes para solução do caso restrito.

São apresentados no decorrer deste capítulo algoritmos NMPC práticos que utilizam modelos de predição com estrutura ESNI. Na Seção 4.3 é proposto um algoritmo NMPC prático com predição não linear e linearização, no qual uma versão linearizada do modelo ESNI é obtida por meio de um método numérico. Essa formulação é expandida em uma abordagem adaptativa, apresentada na Seção 4.4. Por fim, na Seção 4.5 é proposto um algoritmo NMPC prático iterativo com predição não linear e linearização, o qual utiliza um procedimento iterativo para reduzir a influência da porção da predição que depende da representação linearizada.

4.3 ALGORITMO NMPC PRÁTICO PARA MODELOS ESNI

Na abordagem NMPC prática, proposta neste trabalho, o modelo ESNI é utilizado para a predição do comportamento futuro do processo, sendo que, no decorrer deste estudo, este algoritmo proposto é denominado PNMPC-ESNI. Nesta abordagem, o modelo ESNI é utilizado para obtenção da resposta livre do processo, porção da predição que depende apenas das informações passadas do processo (estado do sistema), enquanto uma versão linearizada do modelo ESNI é obtida, por meio de um procedimento de linearização numérica, para obtenção da resposta forçada, a qual consiste na porção da predição que depende apenas das ações de controle atuais e futuras.

No decorrer desta seção, é detalhado como o modelo ESNI é utilizado para obtenção da predição, com detalhes acerca da obtenção das respostas livre e forçada. Em seguida, são apresentados um pseudo-código do algoritmo proposto e dois exemplos da aplicação do mesmo, considerando um processo de neutralização de pH simulado e o controle de uma bancada real de ensaio de compressores de refrigeração.

4.3.1 Predição não linear e linearização

Na arquitetura de controle proposta, o modelo é utilizado para a predição do comportamento futuro do sistema, sendo que a predição é dividida em duas porções, denominadas resposta livre e forçada. O modelo completo não linear é usado para obter a resposta livre não linear do processo, enquanto uma versão linearizada obtida a cada período de amostragem é usada para obter a resposta forçada. Essa divisão é a característica principal dos algoritmos NMPC práticos, resultando em uma equação de predição afim em relação ao incremento de controle, $\Delta \mathbf{u}$, a qual, para o caso MIMO de um sistema com m saídas e n entradas, é definida pela equação

$$\hat{\mathbf{y}}(k) = \mathbf{y}_{\text{forçada}}(k) + \mathbf{y}_{\text{livre}}(k) = \mathbf{G}_{\text{PNMPC}}(k)\Delta \mathbf{u}(k) + \mathbf{y}_{\text{livre}}(k), \quad (69)$$

na qual

$$\hat{\mathbf{y}}(k) = \begin{bmatrix} \hat{\mathbf{y}}_1(k)^T & \dots & \hat{\mathbf{y}}_m(k)^T \end{bmatrix}^T, \quad (70)$$

$$\mathbf{y}_{\text{forçada}}(k) = \begin{bmatrix} \mathbf{y}_{\text{forçada},1}(k)^T & \dots & \mathbf{y}_{\text{forçada},m}(k)^T \end{bmatrix}^T, \quad (71)$$

$$\mathbf{y}_{\text{livre}}(k) = \begin{bmatrix} \mathbf{y}_{\text{livre},1}(k)^T & \dots & \mathbf{y}_{\text{livre},m}(k)^T \end{bmatrix}^T, \quad (72)$$

$$\mathbf{G}_{\text{PNMPC}}(k) = \begin{bmatrix} \mathbf{G}_{\text{PNMPC},11}(k) & \dots & \mathbf{G}_{\text{PNMPC},1n}(k) \\ \vdots & \ddots & \vdots \\ \mathbf{G}_{\text{PNMPC},m1}(k) & \dots & \mathbf{G}_{\text{PNMPC},mn}(k) \end{bmatrix}, \quad (73)$$

$$\Delta \mathbf{u}(k) = \begin{bmatrix} \Delta \mathbf{u}_1^T(k) & \dots & \Delta \mathbf{u}_n^T(k) \end{bmatrix}^T. \quad (74)$$

Os vetores de predição das respostas livre e forçada, para a i -ésima saída, são estruturados da mesma forma que o vetor de predições da saída, sendo definidos como

$$\mathbf{y}_{\text{livre},i}(k) = \begin{bmatrix} y_{\text{livre},i}(k + N_{1,i}) & \dots & y_{\text{livre},i}(k + N_{2,i}) \end{bmatrix}^T, \quad (75)$$

$$\mathbf{y}_{\text{forçada},i}(k) = \begin{bmatrix} y_{\text{forçada},i}(k + N_{1,i}) & \dots & y_{\text{forçada},i}(k + N_{2,i}) \end{bmatrix}^T. \quad (76)$$

O vetor de resposta livre, $\mathbf{y}_{\text{livre}}(k)$, é obtido da trajetória não linear do modelo ESNI. Considerando o conjunto de horizontes de predição $N = \{N_i\}_{i=1}^m$, o modelo é simulado até o fim do maior horizonte de predição, (i.e. $\max(N)$), considerando todos os incrementos de controle futuros nulos, com o vetor de entradas descrito como

$$\begin{aligned} \mathbf{u}_{\text{livre}}(k) &= \begin{bmatrix} \mathbf{u}_{\text{livre}}(k | k)^T & \dots & \mathbf{u}_{\text{livre}}(k + \max(N) - 1 | k)^T \end{bmatrix}^T \\ &= \begin{bmatrix} \mathbf{u}(k - 1 | k - 1)^T & \dots & \mathbf{u}(k - 1 | k - 1)^T \end{bmatrix}^T. \end{aligned} \quad (77)$$

Usando as equações (19) e (20), o modelo ESNI é simulado e o vetor de resposta livre é obtido como

$$\mathbf{y}_{\text{livre}}(k + i | k) = \mathbf{W}^{\text{out}} \begin{bmatrix} 1 \\ \mathbf{x}_{\text{livre}}(k + i | k) \\ \mathbf{y}_{\text{livre}}(k + i - 1 | k) \end{bmatrix} + \frac{D(z^{-1})a(k)}{\Delta}. \quad (78)$$

na qual $\mathbf{y}_{\text{livre}}(k | k) = \mathbf{y}(k)$. Neste caso, os estados do modelo ESNI durante a simulação para obtenção da resposta livre são obtidos por meio da equação

$$\mathbf{x}_{\text{livre}}(k + i | k) = \tanh \left(\mathbf{W}_{\text{in}}^{\text{res}} \begin{bmatrix} 1 \\ \mathbf{u}_{\text{livre}}(k + i - 1 | k) \end{bmatrix} + \mathbf{W}_{\text{res}}^{\text{res}} \mathbf{x}_{\text{livre}}(k + i - 1 | k) \right), \quad (79)$$

na qual $i \in [1, \max(N)]$. Os estados iniciais, $\mathbf{x}_{\text{livre}}(k | k)$, são obtidos considerando o sinal de entrada, $\mathbf{u}(k - 1)$, o qual foi aplicado ao processo no instante de tempo passado, por meio de

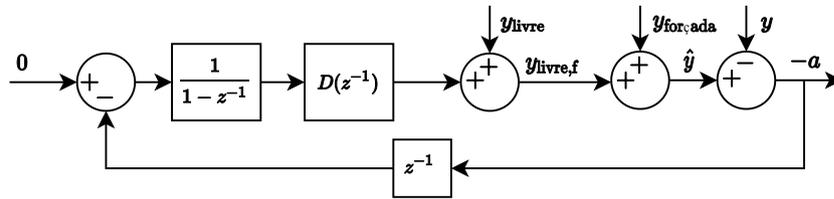
$$\mathbf{x}_{\text{livre}}(k | k) = \tanh \left(\mathbf{W}_{\text{in}}^{\text{res}} \begin{bmatrix} 1 \\ \mathbf{u}(k - 1) \end{bmatrix} + \mathbf{W}_{\text{res}}^{\text{res}} \mathbf{x}_{\text{livre}}(k - 1 | k - 1) \right). \quad (80)$$

Os vetores de resposta livre, (75), para cada saída i , são extraídos do resultado de (78), considerando os respectivos horizontes de predição N_i , já que a simulação considera o maior horizonte de predição do conjunto N . Considerando cada vetor de resposta livre para cada saída do processo, o vetor de resposta livre MIMO pode ser construído, considerando a estrutura apresentada em (72).

Devido ao modelo ESNI ter em sua estrutura um termo que modela a perturbação, o qual considera a integral do erro de predição filtrado, o controlador é capaz de apresentar erro nulo em regime permanente para seguimento de referências do tipo degrau e rejeição de perturbações do tipo degrau. Uma forma de demonstrar isto é representando esse problema como uma malha de controle, demonstração semelhante à apresentada em Plucenio (2010). Se esse problema for representado pela forma apresentada na Figura 19, tem-se como objetivo fazer com que $\hat{\mathbf{y}}$

convirja para y em regime permanente, pois neste caso, se o controlador resultante tiver ação integral haverá erro nulo em regime permanente para referências do tipo degrau e perturbações do tipo degrau. Dessa forma, o objetivo é fazer com que a perturbação $a = y - \hat{y}$, convirja para a referência 0 em regime permanente. Devido à porção correspondente à resposta forçada convergir à 0 em regime permanente, e ao integrador, $\frac{1}{1-z^{-1}}$, a perturbação a apresenta erro nulo em relação à referência 0, considerando o filtro $D = 1$, se o sistema em malha fechada for estável. Com isso, é possível garantir que, em regime permanente, o modelo de predição apresentará erro nulo em relação à saída considerando perturbações do tipo degrau, fazendo com que o controlador seja capaz de apresentar erro nulo em regime permanente no caso irrestrito, para esse tipo de perturbação.

Figura 19 – Diagrama de blocos para a demonstração da garantia de erro nulo em regime permanente do controlador PNMPC-ESNI.



Fonte: elaborado pelo autor.

O modelo da perturbação contém, adicionalmente, um filtro definido pela função de transferência $D(z^{-1})$. Esse filtro, o qual deve possuir ganho estático unitário, pode ser usado para modificar o comportamento dinâmico do modelo de perturbação. Uma alternativa para o projeto desse filtro consiste em utilizar uma arquitetura passa-baixa, a qual pode ser definida como

$$D_{pb}(z^{-1}) = \frac{(1-b)}{(1-bz^{-1})}, \quad (81)$$

onde o parâmetro $b \in [0, 1]$ está associado com a frequência de corte do filtro. A adição de um filtro passa-baixa tem como efeito a atenuação do sinal utilizado para correção da resposta livre, o que resulta, em malha fechada, em um sinal de controle menos suscetível ao ruído de medição. Em contrapartida, o desempenho do controlador para a rejeição de perturbações é afetado.

Outra opção consiste na utilização de um filtro passa-alta, definido como

$$D_{pa}(z^{-1}) = \frac{c - (c-1)z^{-1}}{1}, \quad (82)$$

no qual o parâmetro $c \in [1, \infty[$ está associado com a frequência de corte do filtro. O comportamento do filtro $D_{pa}(z^{-1})$ amplifica o erro entre a saída predita e a medida, resultando em um desempenho melhor para rejeição de perturbação. Por outro lado, a adição desse filtro resulta em um sinal de controle mais suscetível ao ruído de medição e tipicamente tem impactos negativos na robustez do controlador. Para o GPC, o qual considera um modelo da perturbação similar, o uso de um filtro passa-baixa tipicamente está associado a uma melhora na robustez do sistema

em malha fechada (CAMACHO; BORDONS, 2007). O efeito da correção da resposta livre e a influência da adição do filtro $D(z^{-1})$ em variantes passa-baixa e passa-alta são avaliados no exemplo detalhado na Seção 4.3.2.

A matriz de dinâmica $\mathbf{G}_{\text{PNMPC}}(k) \in \mathfrak{R}^{\sum_{i=1}^m N_i \times \sum_{j=1}^n N_{u,j}}$ é constituída por matrizes $\mathbf{G}_{\text{PNMPC},ij}(k)$, as quais são formadas pelos coeficientes da resposta ao degrau do modelo ESNI, para um par saída j e entrada i . Cada matriz $\mathbf{G}_{\text{PNMPC},ij}(k)$ pode ser definida como

$$\mathbf{G}_{\text{PNMPC},ij}(k) = \begin{bmatrix} s_{1,1}(k) & 0 & \cdots & 0 \\ s_{2,1}(k) & s_{1,2}(k) & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ s_{N_i,1}(k) & s_{(N_i-1),2}(k) & \cdots & s_{(N_i-N_{u,j}+1),(N_{u,j})}(k) \end{bmatrix}, \quad (83)$$

na qual o vetor $\mathbf{g}_{ijs}(k) = [s_{1,s}(k) \dots s_{N_i,s}(k)]^T$ corresponde aos coeficientes entre $N_{1,i}$ e $N_{2,i}$, com $N_{1,i} = d_i + 1$ e $d_i = \min(d_{ij}) \forall j$. A obtenção desses coeficientes para o caso de modelos não lineares é mais complexa que para modelos lineares, já que o critério da superposição não é verificado. Para modelos que se baseiam em RNN, esses coeficiente podem ser obtidos por meio da simulação da resposta do modelo ao aplicar como entrada um degrau na entrada j em $k + s$. A amplitude do degrau pode ser definida, para cada entrada j , como $w = \Delta u_j(k-1)/10$, a qual representa um décimo do último sinal de controle aplicado no processo, resultando em uma linearização local da dinâmica do modelo.

A obtenção da coluna s de cada vetor de resposta forçada, associada à entrada j modelo ESNI, é alcançada simulando o modelo considerando como entrada

$$\mathbf{u}_{\text{forçada},js}(k) = \mathbf{u}_{\text{livre}}(k) + w \begin{bmatrix} \mathbf{0}_{(j-1) \max(N)} \\ \mathbf{0}_{s-1} \\ \mathbf{1}_{\max(N)-s+1} \\ \mathbf{0}_{(n-j) \max(N)} \end{bmatrix}, \quad (84)$$

na qual $\mathbf{0}_a \in \mathfrak{R}^a$ e $\mathbf{1}_a \in \mathfrak{R}^a$ representam vetores de zeros e uns, respectivamente.

O vetor $\mathbf{g}_{ijs}(k)$ é extraído de cada coluna de

$$\mathbf{g}_{js}(k) = w^{-1}(\mathbf{y}_{\text{forçada},js}(k) - \mathbf{y}_{\text{livre}}(k)), \quad (85)$$

considerando

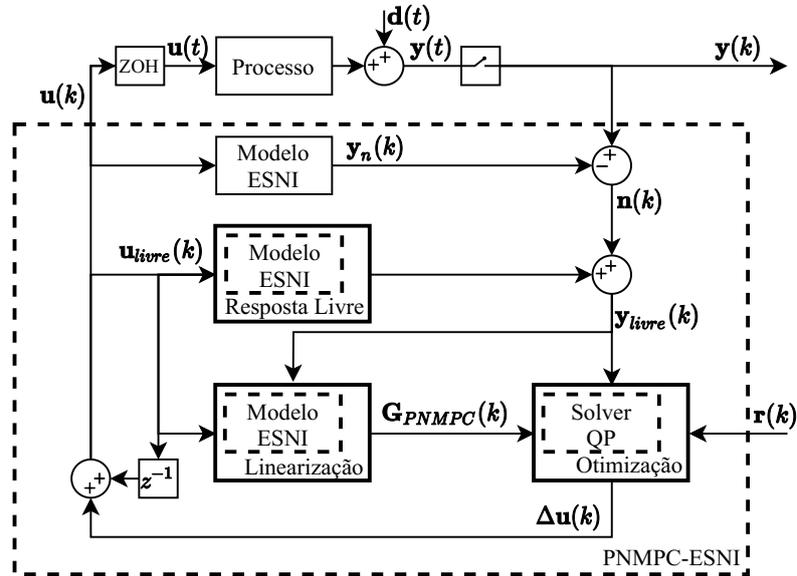
$$\mathbf{y}_{\text{forçada},js}(k+i|k) = \mathbf{W}^{\text{out}} \begin{bmatrix} 1 \\ \mathbf{x}_{\text{forçada},js}(k+i|k) \\ \mathbf{y}_{\text{forçada},js}(k+i-1|k) \end{bmatrix} + \frac{D(z^{-1})a(k)}{\Delta}, \quad (86)$$

e $\mathbf{y}_{\text{forçada},js}(k|k) = \mathbf{y}(k)$. Os estados do modelo ESNI são obtidos usando a equação

$$\mathbf{x}_{\text{forçada},js}(k+i|k) = \tanh(\mathbf{W}_{\text{in}}^{\text{res}} \begin{bmatrix} 1 \\ \mathbf{u}_{\text{forçada},js}(k+i-1|k) \end{bmatrix} + \mathbf{W}_{\text{res}}^{\text{res}} \mathbf{x}_{\text{forçada},js}(k+i-1|k)), \quad (87)$$

sendo que os estados iniciais são obtidos da mesma forma descrita em (80). Uma representação geral do funcionamento do algoritmo proposto é apresentada por meio de um diagrama de blocos na Figura 20.

Figura 20 – Ilustração da estrutura de controle PNMPC-ESNI.



Fonte: elaborado pelo autor.

O algoritmo PNMPC-ESNI é sumarizado no Algoritmo 3 como:

Algoritmo 3 Algoritmo PNMPC para modelos ESNI

-
- 1: Medir as saídas atuais do processo.
 - 2: **Para** $i = 1 : m$ **Faça**
 - 3: Obter os estados \mathbf{x}_e do reservatório, considerando como entrada $\mathbf{u}_{\text{livre},i}(k)$, definido em (77).
 - 4: Obter a resposta livre não linear $\mathbf{y}_{\text{livre},i}(k)$ da saída da ESNI, conforme (75),
 - 5: **Fim Para**
 - 6: Construir o vetor concatenado de resposta livre não linear MIMO $\mathbf{y}_{\text{livre}}(k)$, como definido em (72)
 - 7: **Para** $i = 1 : m$ **Faça**
 - 8: **Para** $j = 1 : n$ **Faça**
 - 9: **Para** $s = 1 : N_{u,j}$ **Faça**
 - 10: Linearizar o modelo ESNI por meio da simulação da resposta ao degrau em volta de $\mathbf{u}_{\text{livre}}(k)$, obtendo a coluna s da matriz de dinâmica SISO $\mathbf{G}_{ij}(k)$ utilizando (85).
 - 11: **Fim Para**
 - 12: **Fim Para**
 - 13: **Fim Para**
 - 14: Construir a matriz de dinâmica MIMO $\mathbf{G}_{\text{PNMPC}}(k)$ conforme a estrutura em (73).
 - 15: Resolver o problema QP associado ao PNMPC para obtenção dos incrementos de controle futuro $\Delta\mathbf{u}(k)$.
 - 16: **Para** $j = 1 : n$ **Faça**
 - 17: Aplicar o primeiro elemento do vetor de incrementos de controle no processo, i.e. $u_j(k) = u_j(k-1) + \Delta u_j(k)$.
 - 18: **Fim Para**
 - 19: Aguardar o término do período de amostragem.
 - 20: Retornar ao passo 1 com $k = k + 1$.
-

4.3.2 Exemplo 3: Controle de um reator de neutralização de pH com PNMPC-ESNI

O primeiro exemplo usado para demonstrar o funcionamento do algoritmo PNMPC-ESNI é o controle de um reator de neutralização de pH, descrito no Apêndice A.1. Este exemplo é utilizado para demonstrar algumas características importantes do algoritmo proposto, como a influência do ajuste dos horizontes e da sintonia das ponderações da função custo. São demonstrados o efeito da utilização do termo que modela a perturbação na ESNI, resultando em um controlador com erro nulo em regime permanente, da escolha do filtro para o erro de predição $D(z^{-1})$ e é demonstrado o impacto da sintonia nos resultados em malha fechada do controlador.

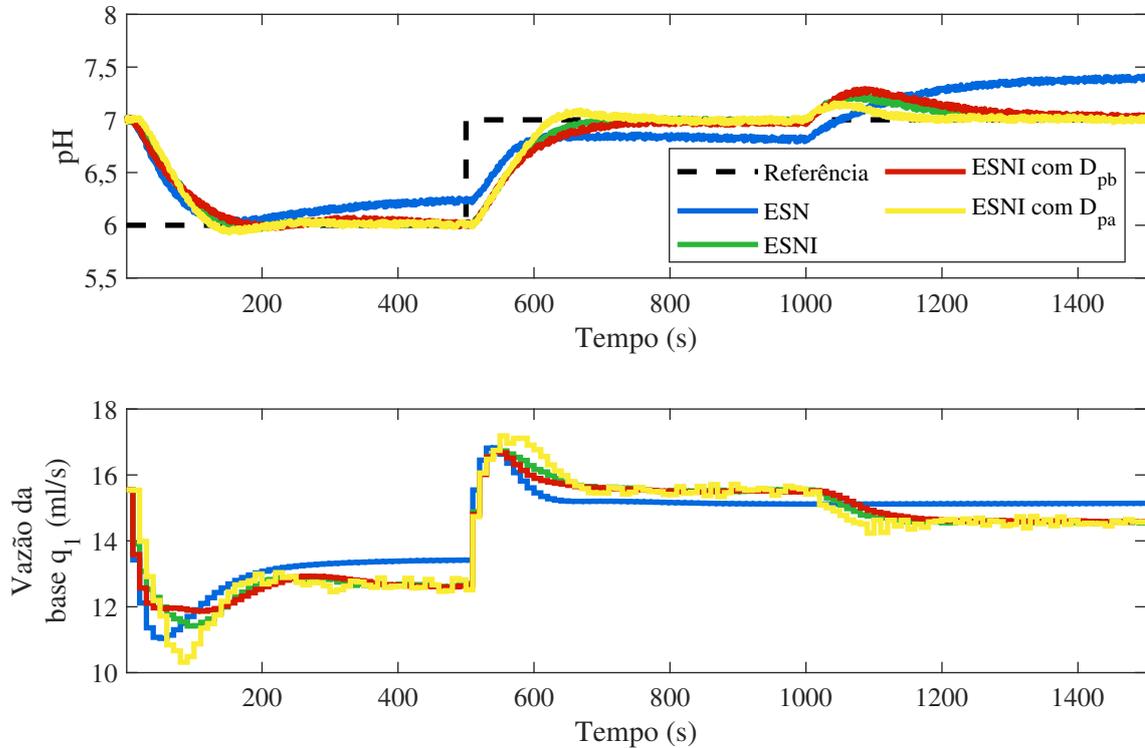
O modelo ESNI, identificado conforme apresentado na Seção 3.4.2, foi o utilizado para o desenvolvimento do controlador. Essa análise inicial considera um modelo com reservatório composto por 3000 unidades. O modelo fenomenológico foi utilizado como processo real, e o controlador foi testado em cenários nos quais o problema de controle é mais desafiador, ou seja, em regiões de pH neutro. O período de amostragem considerado foi de 10 s. A simulação desse estudo de caso foi realizada utilizando um computador com processador Intel Core i7-9750H, 16 gb de memória e foi programada na linguagem MATLAB.

O primeiro experimento realizado consiste na avaliação da importância da utilização do modelo de perturbação, para garantia de erro nulo em regime permanente, e o efeito da utilização do filtro $D(z^{-1})$ na resposta do controlador. Para tal, foram realizadas simulações considerando casos em que a porção que modela a perturbação é desconsiderada, sendo que esse modelo é denominado ESN, e simulações considerando a utilização do modelo de perturbação com três configurações distintas, $D(z^{-1}) = 1$, $D_{pb}(z^{-1})$ e $D_{pa}(z^{-1})$. Para essa análise foi utilizada uma sintonia $\lambda/\alpha = 0,2$, sendo que a influência da sintonia é discutida nos parágrafos seguintes. Os resultados dessa simulação são apresentados na Figura 21.

A análise do resultado em malha fechada indica que o controlador que não utilizou a parcela que modela o erro de predição (ESN) é incapaz de garantir erro nulo em regime permanente, o que é esperado, devido ao modelo empregado não utilizar como entrada as saídas medidas do processo. As versões que utilizaram o modelo ESNI apresentaram erro nulo em regime permanente, sendo capazes de rejeitar perturbações na saída. Foram testadas as duas formulações distintas para o filtro $D(z^{-1})$, com a resposta $D_{pb}(z^{-1})$ representando a versão com filtro passa baixa ajustado com $b = 0,5$ e a resposta D_{pa} correspondendo ao filtro passa-alta com ajustado com $c = 5$. A versão que utiliza um filtro passa-baixa na correção da resposta livre atenuou o efeito do ruído de medição na correção da resposta livre, resultado em um sinal de controle mais suave, efeito alcançado em detrimento da rapidez na rejeição a perturbação. Por sua vez, o algoritmo PNMPC-ESNI com o filtro passa-alta foi capaz de acelerar a rejeição a perturbação, ao mesmo tempo que o sinal de controle acabou sendo mais afetado por ruídos na medição. Para o restante dos resultados apresentados nesta tese, foi usado o modelo ESNI considerando $D(z^{-1}) = 1$.

O segundo cenário de simulação almejou demonstrar o efeito da sintonia do controlador proposto na resposta em malha fechada do sistema. Devido ao sistema ser SISO, foi variada a

Figura 21 – Comparação do resultado obtido pelo controlador proposto utilizando o modelo ESN, ESNI com $D(z^{-1}) = 1$, ESNI com $D_{pb}(z^{-1})$ no qual $b = 0,5$ e ESNI com $D_{pa}(z^{-1})$ no qual $c = 5$.)



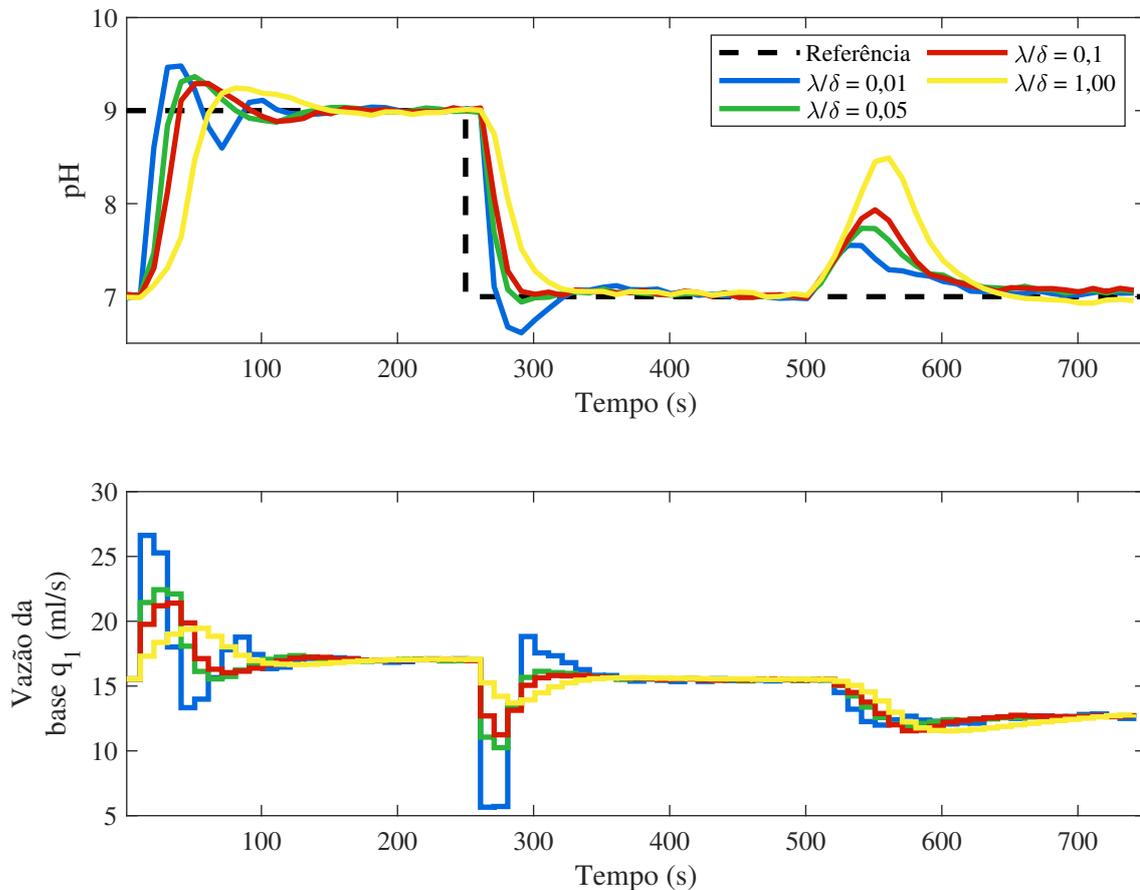
Fonte: dados da pesquisa.

razão λ/δ . Foi selecionada uma mudança de referência na região mais desafiadora do processo, entre os valores de pH 7 e 9. O horizonte de predição escolhido foi $N = 10$, com $N_1 = 1$ e $N_2 = 10$, de forma a garantir que toda a dinâmica em malha aberta seja considerada no cálculo do controlador. Para o horizonte de controle, o valor $N_u = 3$ foi usado. Os resultados estão representados graficamente na Figura 22.

Os resultados indicam que o aumento do valor de λ/δ faz com que o controlador dê maior importância para a minimização do esforço de controle, fazendo com que a resposta transitória seja mais lenta, o que é o comportamento esperado. Valores muito pequenos de λ/δ fazem com que o processo apresente ações de controle mais abruptas, enquanto sintonias com valores muito elevados resultam em comportamento lento de malha fechada.

Os resultados do controlador PNMPC-ESNI, considerando o modelo ESNI com reservatório composto por 3000 unidades, foi comparado com os resultados obtidos por um controlador GPC, um controlador PNMPC com modelo não linear do tipo Hammerstein-Wiener (PNMPC-HW) e um controlador NMPC que considerou como modelo de predição o modelo fenomenológico do processo. Para a solução do problema de otimização não linear associado ao NMPC, foi utilizado um algoritmo de programação quadrática sequencial (SQP, do inglês *Sequential Quadratic Programming*) (BIEGLER, 2000). A obtenção desses modelos é detalhada no Apêndice A. Foi considerado um cenário de testes no qual foram realizadas 5 variações na re-

Figura 22 – Comparação dos resultados obtidos pelo controlador PNMPC-ESNI considerando sintonias distintas.



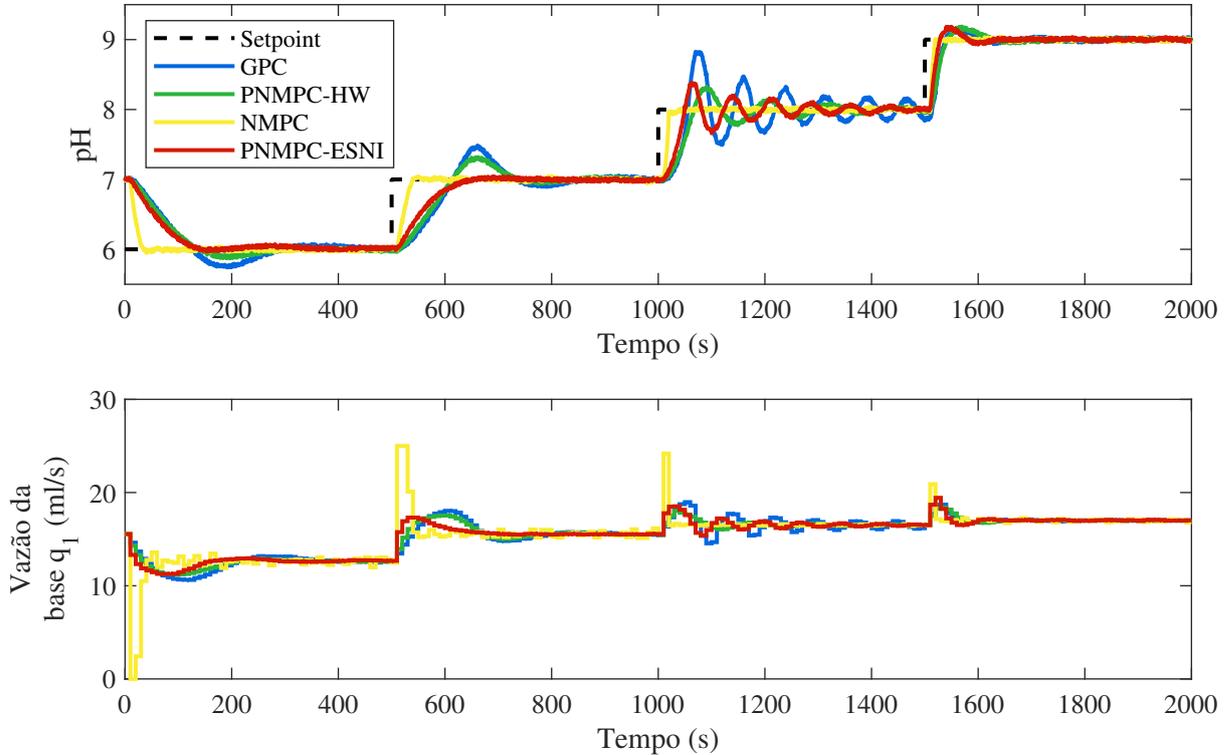
Fonte: dados da pesquisa.

gião de operação do controlador, avaliando a capacidade dos controladores operarem em regiões distintas de operação do processo não linear. A sintonia final de cada controlador consistiu na que resultou no melhor valor para a métrica IAE neste cenário. Para o controlador proposto foi utilizado $\lambda/\delta = 0,09$, para o GPC $\lambda/\delta = 1$, para o PNMPC-HW foi usado $\lambda/\delta = 0,5$ e para o NMPC $\lambda/\delta = 0,01$. Os resultados considerando tal sintonia são apresentados na Figura 23. Uma análise detalhada do impacto da sintonia no IAE de cada controlador é apresentada no decorrer da seção.

O comportamento em malha fechada no cenário de controle proposto evidencia uma clara vantagem das implementações não lineares, em especial nos pontos de operação próximos à região não linear mais desafiadora. Entre as alternativas com modelo não linear, o controlador proposto apresentou resultados melhores que os obtidos pelo controlador PNMPC com modelo Hammerstein, apresentando menores sobressinais e tempos de acomodação mais rápidos.

Devido ao fato de o modelo ESNI apresentar maior fidelidade que os modelos linear e Hammerstein-Wiener, utilizados nos controladores para comparação, é possível utilizar uma sintonia menos conservadora no controlador proposto. Dessa forma, são verificadas transições de referência mais rápidas para todas as referências. Devido ao grande erro de modelagem apresentado pelo modelo linear, o controlador GPC utilizou uma sintonia bastante conservadora,

Figura 23 – Resultados do controle de um reator de neutralização de pH com algoritmo PNMPC-ESNI, PNMPC com modelo Hammerstein, GPC e NMPC. O gráfico superior apresenta o sinal de referência (linha tracejada) e os valores de pH (linhas sólidas), enquanto no gráfico inferior são mostrados os sinais de controle.

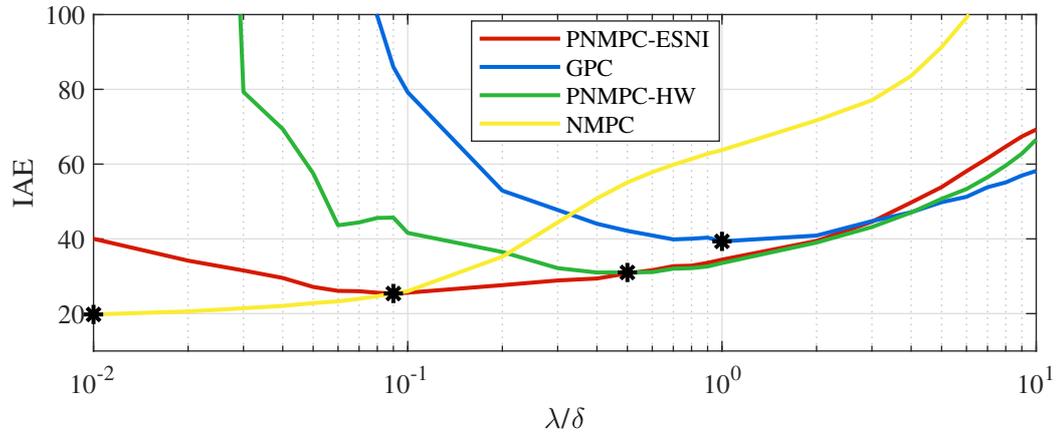


Fonte: dados da pesquisa.

de forma a evitar um comportamento oscilatório em determinadas condições de operação, o que faz com que nas demais regiões de operação o controlar seja demasiadamente lento. Por sua vez, a comparação do controlador proposto com o NMPC evidencia a existência de erros de modelagem no modelo ESNI, e também a existência de um impacto resultante da utilização de uma versão linearizada do modelo não linear, pois o desempenho do controlador NMPC foi superior ao alcançado pelo PNMPC-ESNI.

A comparação quantitativa dos resultados em malha fechada dos controladores foi realizada utilizando o IAE como métrica. Na Figura 24 são apresentados os valores finais dessa métrica, considerando sintonias distintas para todos controladores. Cada ponto do gráfico corresponde ao IAE do respectivo controlador, testado utilizando o cenário apresentado na Figura 22. O valor correspondente à melhor sintonia, a qual foi utilizada para obtenção dos resultados apresentados na Figura 23 e na Tabela 3 é destacado com um asterisco. Os resultados do algoritmo proposto são melhores que os apresentados pelos controladores usados como referência, sendo que o PNMPC-ESNI apresentou valor final da IAE de 25,64, enquanto o PNMPC-HW apresentou valor final de 30,85. Os valores finais para o GPC e o NMPC foram 40,60 e 12,31, respectivamente. Para a métrica IAE e para o cenário de teste proposto, o PNMPC-ESNI apresentou uma melhoria de 16% em relação ao PNMPC-HW e de 36% em relação ao GPC, porém, um desempenho 52% pior que o NMPC. Esse resultado evidencia que a abordagem proposta traz

Figura 24 – Valores finais da métrica IAE em função da sintonia para o algoritmo PNMPC-ESNI comparado com PNMPC-HW, GPC e NMPC.



Fonte: dados da pesquisa.

melhorias em relação a controladores MPC baseados em representações lineares e em modelo não lineares identificados, porém, apresenta resultados muito inferiores ao caso ideal, que é o controlador NMPC com modelo perfeito e solução do problema NLP.

Na Tabela 3 são sumarizados os valores finais das métricas de desempenho dos controladores, sendo apresentados os resultados do controlador proposto considerando reservatórios com 500 e 3000 unidades. Outra métrica apresentada é o tempo de cômputo requerido para obtenção do sinal de controle. A complexidade computacional do algoritmo proposto foi comparada com a dos demais algoritmos considerados, sendo que os valores médios e máximos são apresentados na Tabela 3.

Tabela 3 – Comparação das métricas de desempenho para os controladores PNMPC-ESNI, GPC, PNMPC-HW e NMPC.

Controlador	IAE	ITAE	Tempo de cômputo médio (s)	Tempo de cômputo máximo (s)
PNMPC-ESNI (r=500)	28,02	236,25	0,13	0,16
PNMPC-ESNI (r=3000)	25,64	212,47	7,18	8,99
GPC	40,60	470,15	$2,15 \times 10^{-3}$	$5,90 \times 10^{-3}$
PNMPC-HW	30,85	269,41	$9,51 \times 10^{-2}$	$12,46 \times 10^{-2}$
NMPC	12,31	95,86	22,71	29,10

Fonte: dados da pesquisa.

Os tempos de cômputo requeridos pelo controlador proposto demonstram uma das principais vantagens da utilização de uma abordagem NMPC prática, que é a redução dos tempos de cômputo máximos e médios. O controlador proposto apresentou tempos médios superiores ao GPC, o que é esperado, já que o controlador proposto apresenta todo um processo para obtenção, a partir do modelo ESNI, da matriz de dinâmica e do vetor de resposta forçada. Porém, o tempo requerido pelo PNMPC-ESI foi duas ordens de grandeza menor que o requerido pelo contro-

lador NMPC, no caso em que o modelo ESNI utilizou reservatório com 500 unidades. Para o caso com modelo ESN utilizando reservatório com 3000 unidades, os tempos foram maiores, porém, ainda menores que o período de amostragem do processo, o qual é 10 s. É importante notar que o NMPC não é capaz de entregar o sinal de controle dentro do período de amostragem do processo.

4.3.3 Exemplo 4: Controle de uma bancada de ensaio de compressores herméticos com PNMPC-ESNI

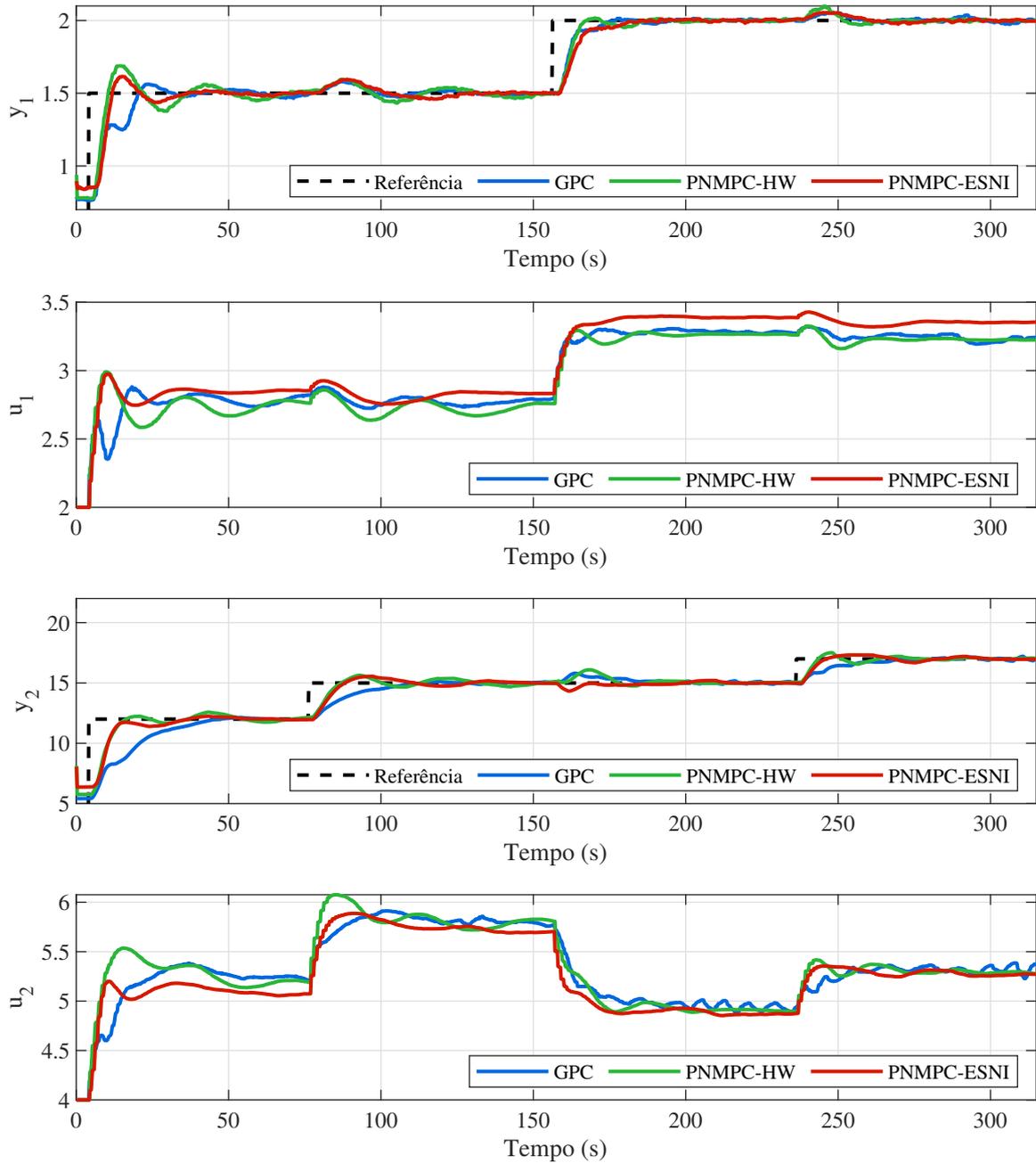
Este exemplo tem como objetivo demonstrar a aplicação do algoritmo PNMPC-ESNI no controle de um processo real e multivariável. O problema em estudo consiste no controle de uma bancada de ensaio de compressores herméticos de refrigeração, a qual é descrita no apêndice A.2. A bancada de ensaio de compressores apresenta em sua operação típica o foco no seguimento de referências, sendo que diversas mudanças de referência ocorrem em um ensaio típico. Esse processo apresenta como desafios o forte acoplamento entre as pressões de sucção (y_s) e de descarga (y_d), e o comportamento não linear da pressão de descarga (y_d). O período de amostragem do processo é 0,2 s.

Neste exemplo, o controlador PNMPC-ESNI foi implementado no processo real e os parâmetros de ajuste foram definidos considerando o comportamento dinâmico das variáveis de interesse. Um detalhamento do procedimento de sintonia de MPC para esse tipo de processo é encontrado em Dangu *et al.* (2018), no qual o método de sintonia foi derivado para o algoritmo GPC, o qual compartilha a mesma função custo usada no PNMPC-ESNI. Para os horizontes de predição de ambas as variáveis de processo foram escolhidos os valores $N_s = 30$ e $N_d = 100$, enquanto os horizontes de controle foram escolhidos como $N_{u,s} = 4$ e $N_{u,d} = 15$. Levando em conta os horizontes escolhidos, foram selecionados os parâmetros de ajuste $\lambda(j)$ e $\delta(i)$, de forma a manter um balanço entre seguimento de referência rápido e baixo esforço de controle. Por simplicidade, os parâmetros foram mantidos constantes para todo o horizonte e os valores selecionados foram $\lambda_s = 1500$, $\lambda_d = 450$, $\delta_s = 1$, e $\delta_d = 30$. Foram consideradas restrições no sinal de controle, que deve ser mantido no intervalo de 4 V até 8 V para u_s e de 2 V até 6 V para u_d . Esses valores foram definidos considerando as restrições operacionais e de segurança da bancada de ensaio real.

Foram implementados dois algoritmos de controle para fins de comparação, o algoritmo GPC e o algoritmo PNMPC com modelo não linear do tipo Hammerstein-Wiener. Ambos os controladores utilizaram os mesmos parâmetros de ajuste e os modelos usados são os apresentados no Apêndice A. Os resultados obtidos pelo algoritmo proposto e pelos algoritmos usados em cada caso para comparação são apresentados na Figura 25.

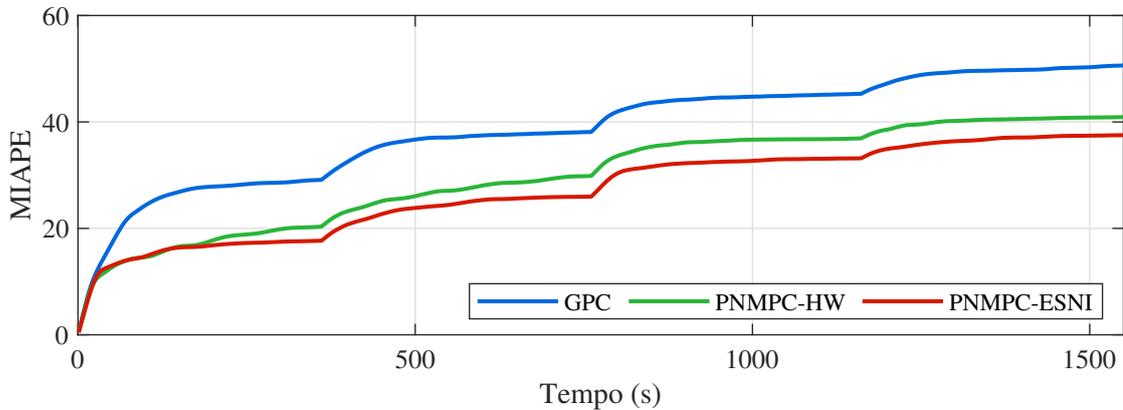
Por inspeção visual é possível indicar que as implementações não lineares apresentaram resultados notavelmente melhores que a abordagem baseada em um modelo de predição linear. Por sua vez, a diferença entre as duas implementações de PNMPC são mais sutis, porém, o algoritmo proposto apresenta tempos de acomodação mais rápidos e maior amortecimento,

Figura 25 – Resultados do controle da bancada de ensaio de compressores de refrigeração obtidos pelos algoritmos PNMPC-ESNI (linha vermelha), PNMPC com modelo Hammerstein-Wiener (linha verde) e GPC (linha azul). No primeiro e no terceiro gráficos são mostradas a referência (linha tracejada) e as variáveis controladas (linhas sólidas).



Fonte: dados da pesquisa.

Figura 26 – Evolução da MIAPE para o controle da bancada de ensaio de compressores obtidos pelos algoritmos PNMPC-ESN, PNMPC-HW e GPC.



Fonte: dados da pesquisa.

Tabela 4 – Comparação das métricas de desempenho para os controladores GPC, PNMPC-HW e PNMPC-ESNI.

Controlador	y_1 IAPE	y_2 IAPE	MIAPE
GPC	38,52%	62,80%	50,66%
PNMPC-H	41,92%	39,96%	40,94%
PNMPC-ESN	36,93%	38,13%	37,53%

Fonte: dados da pesquisa.

resultando em sinais de controle mais comportados. A utilização de um modelo mais preciso possibilitou ao algoritmo PNMPC-ESNI uma melhor modelagem do acoplamento entre as variáveis, levando a uma menor perturbação devido a tal efeito. Isso fica evidente ao observar y_2 durante a variação de referência entre 1,5 bar e 2 bar realizada em y_1 , o que resultou em uma perturbação que foi mais bem rejeitada pelo algoritmo proposto.

Para a avaliação do desempenho do algoritmos proposto foi utilizada como métrica IAPE, descrita no Apêndice B, a qual considera a integral do erro de forma percentual, permitindo que os erros das saídas controladas em um processo MIMO possam ser comparados e sintetizados em uma métrica única. A Figura 26 ilustra a evolução da média da IAPE para ambas as variáveis controladas (MIAPE), e a Tabela 4 apresenta os valores finais. Para a métrica IAPE, o controlador PNMPC-ESNI apresentou resultados superiores considerando ambas as variáveis. Para MIAPE, a qual representa a média entre a IAPE de cada saída, considerando o cenário de controle testado, o PNMPC-ESNI apresentou desempenho 8,3% e 25,9% melhor que os controladores de referência, PNMPC com modelo Hammerstein-Wiener e GPC, respectivamente.

4.4 ALGORITMO NMPC PRÁTICO ADAPTATIVO PARA MODELOS ESNI

A principal vantagem dos modelos baseados em ESNs consiste em seu treinamento eficiente, o que possibilita a sua realização on-line, como demonstrado na Subseção 3.5.1. O algoritmo NMPC prático baseado na ESNI, apresentado na Seção 4.3, utiliza modelos obtidos

a partir de dados previamente adquiridos do processo. Quando o modelo é atualizado on-line, o controlador é capaz de se adaptar a mudanças paramétricas que acontecem na operação do processo, resultando em uma abordagem adaptativa.

Nesta seção, a formulação de controlador PNMPC para modelos ESNI é expandida, de forma a considerar uma abordagem em que o modelo é adaptado on-line, resultando em um algoritmo PNMPC adaptativo. Para essa formulação, é utilizado o método para identificação on-line de modelos ESNI proposto na Seção 3.5.1, o qual se baseia em uma abordagem MRI com um estimador RLS que considera um fator de esquecimento adaptativo. No restante desta seção é apresentada a integração entre o algoritmo de treinamento on-line do modelo ESNI e o controlador PNMPC-ESNI, proposto na Seção 4.3, sendo que o algoritmo completo, considerando o método para estimação on-line do modelo e o controlador PNMPC, é apresentado por meio de um pseudo-código. A operação do controlador NMPC prático adaptativo proposto é demonstrada utilizando dois estudos de caso simulados, o primeiro considerando um processo de controle de um tanque com perfil cônico e o segundo considerando o controle de um motor síncrono de ímãs permanentes.

4.4.1 Estimação on-line do modelo ESNI e integração com PNMPC

A abordagem de controle adaptativa proposta nesta tese é sintetizada por dois blocos principais: um bloco responsável pela estimação on-line do modelo ESNI; e o outro que representa o controlador NMPC prático. O bloco de estimação on-line utiliza o algoritmo MRI proposto na Seção 3.3, considerando o método dos mínimos quadrados recursivo com fator de esquecimento direcional, para obtenção on-line dos parâmetros da matriz $\mathbf{W}^{\text{out}}(k)$. Esses parâmetros são atualizados a cada período de amostragem, sendo utilizados pelo algoritmo PNMPC na obtenção da resposta livre e da matriz dinâmica. Essa estratégia é denominada no restante desta tese como APNMPC-ESNI (do inglês, *Adaptive PNMPC-ESNI*).

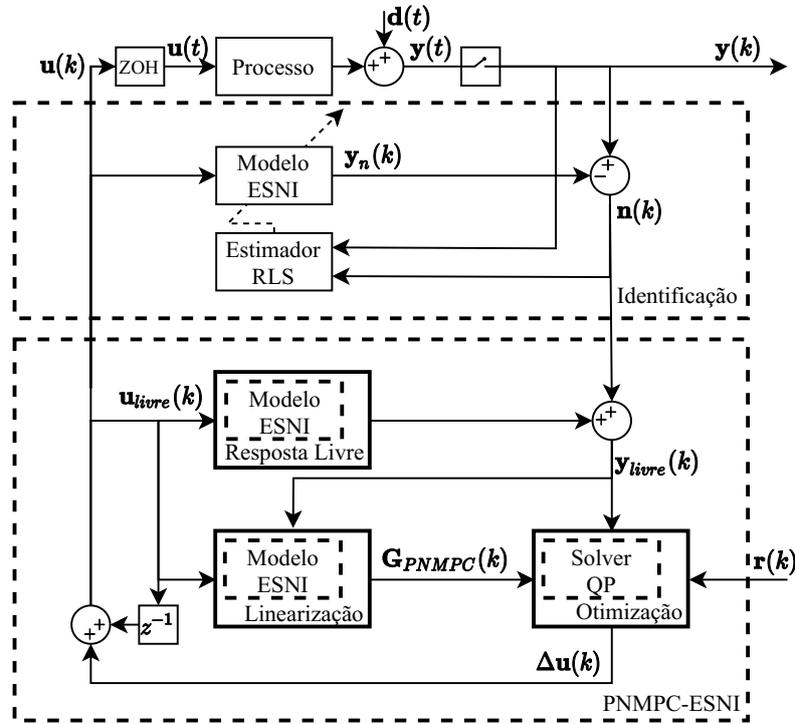
Um diagrama de blocos que ilustra o funcionamento geral do algoritmo APNMPC-ESNI proposto é apresentado na Figura 27, detalhando os mecanismos de adaptação do modelo e o controlador PNMPC-ESNI. O bloco de estimação on-line do modelo é integrado ao controlador, estimando os parâmetros do modelo ESNI a cada período de amostragem. Os parâmetros estimados são compartilhados com o controlador, sendo usados para a obtenção das respostas livre e forçada.

Devido ao fato de o modelo ESNI ser estimado on-line, os elementos de $\mathbf{W}^{\text{out}}(k)$ podem variar no tempo. Portanto, o vetor de resposta livre é obtido pela equação

$$\mathbf{y}_{\text{livre}}(k+i|k) = \mathbf{W}^{\text{out}}(k) \begin{bmatrix} 1 \\ \mathbf{x}_{\text{livre}}(k+i|k) \\ \mathbf{y}_{\text{livre}}(k+i-1|k) \end{bmatrix} + \frac{D(z^{-1})a(k)}{\Delta}, \quad (88)$$

considerando os estados do modelo obtidos por (79) e (80). De forma semelhante, para obter a matriz de dinâmica $\mathbf{G}_{\text{APNMPC},ij}(k)$ é utilizado um procedimento similar ao da matriz

Figura 27 – Diagrama de blocos da abordagem NMPC prática adaptativa. O modelo ESNI é adaptado utilizando a abordagem MRI, com os elementos de \mathbf{W}^{out} sendo estimados com o algoritmo RLS com fator de esquecimento direcional. O controlador utiliza, a cada período de amostragem, o modelo adaptado para obtenção da resposta livre e da matriz de dinâmica.



Fonte: elaborado pelo autor.

$G_{PNMPC,ij}(k)$, utilizando a equação

$$y_{forçada,js}(k+i|k) = \mathbf{W}^{\text{out}}(k) \begin{bmatrix} 1 \\ \mathbf{x}_{forçada,js}(k+i|k) \\ y_{forçada,js}(k+i-1|k) \end{bmatrix} + \frac{D(z^{-1})a(k)}{\Delta}. \quad (89)$$

A estratégia de controle APNMPC-ESNI é sumarizada no Algoritmo 4.

4.4.2 Exemplo 5: Controle adaptativo de um tanque cônico

Para exemplificar a utilização do controlador APNMPC-ESNI, o controle em malha fechada de um tanque de perfil cônico foi utilizado. O período de amostragem considerado é 5 s. A simulação desse estudo de caso foi realizada utilizando um computador com processador Intel Core i7-9750H, 16 gb de memória e foi programada na linguagem MATLAB. A descrição completa do processo, da sua simulação e da identificação de modelos para comparação é detalhada no Apêndice A.3.

Os controladores preditivos foram ajustados utilizando o conhecimento do funcionamento do processo e de suas condições de operação típicas. Os horizontes de previsão e de controle foram ajustados utilizando os valores $N = 50$ e $N_u = 3$, respectivamente. Por sua

Algoritmo 4 Algoritmo APNMPC para modelos ESNI

- 1: Medição das saídas atuais do processo.
- 2: Obtenção dos estados filtrados do reservatório $\mathbf{x}_l(k)$ por meio do método MRI para ESNI;
- 3: Obtenção do fator de esquecimento direcional $\varphi(k)^{-1}(k)$ usando a equação definida em (58);
- 4: Obtenção da matriz de covariância $\mathbf{P}(k)$ usando a equação (55);
- 5: Obter $\mathbf{W}^{\text{out}}(k)$ com (52);
- 6: **Para** $i = 1 : m$ **Faça**
- 7: Obter os estados \mathbf{x}_e do reservatório, considerando como entrada $\mathbf{u}_{\text{livre},i}(k)$, definido em (77).
- 8: Obter a resposta livre não linear $\mathbf{y}_{\text{livre},i}(k)$ da saída da ESNI, conforme (75),
- 9: **Fim Para**
- 10: Construir o vetor concatenado de resposta livre não linear MIMO $\mathbf{y}_{\text{livre}}(k)$, como definido em (72)
- 11: **Para** $i = 1 : m$ **Faça**
- 12: **Para** $j = 1 : n$ **Faça**
- 13: **Para** $s = 1 : N_{u,j}$ **Faça**
- 14: Linearizar o modelo ESNI por meio da simulação da resposta ao degrau em volta de $\mathbf{u}_{\text{livre}}(k)$, obtendo a coluna s da matriz de dinâmica SISO $\mathbf{G}_{ij}(k)$ utilizando (85).
- 15: **Fim Para**
- 16: **Fim Para**
- 17: **Fim Para**
- 18: Construir a matriz de dinâmica MIMO $\mathbf{G}_{\text{PNMPC}}(k)$ conforme a estrutura (73).
- 19: Resolver o problema QP associado ao PNMPC para obtenção dos incrementos de controle futuro $\Delta\mathbf{u}(k)$.
- 20: **Para** $j = 1 : n$ **Faça**
- 21: Aplicar o primeiro elemento do vetor de incrementos de controle no processo, i.e. $u_j(k) = u_j(k-1) + \Delta u_j(k)$.
- 22: **Fim Para**
- 23: Aguardar o término do período de amostragem.
- 24: Retornar ao passo 1 com $k = k + 1$.

vez, foram considerados para as ponderações do erro de seguimento de referência e esforço de controle valores constantes durante seus horizontes, com valores $\lambda = 200$ e $\delta = 1$.

O modelo ESNI foi inicializado considerando uma matriz $\mathbf{W}_{\text{res}}^{\text{res}}$ esparsa e com pesos aleatórios, considerando 10% de conexões não nulas, selecionadas conforme uma distribuição uniforme, no intervalo $[-1, +1]$. Após a inicialização, $\mathbf{W}_{\text{res}}^{\text{res}}$ foi reescalada, fazendo com que seu raio espectral seja $\rho(\mathbf{W}_{\text{res}}^{\text{res}}) = 0,99$. A matriz de entrada $\mathbf{W}_{\text{in}}^{\text{res}}$ foi inicializada considerando todos pesos iguais a $+1$ ou -1 (50% para cada caso), e reescalada considerando uma estimativa do ganho estático máximo do processo, a qual foi 10. O estimador RLS foi inicializado considerando os elementos da diagonal principal da matriz de covariância como $\mathbf{P}_{ii}(0) = 100$; $\varphi(0) = 1$; $\lambda(0) = 0,1$; e $\mathbf{v}(0) = 10^{-6}$.

O controlador adaptativo APNMPC-ESNI foi comparado com três controladores de referência. O primeiro consiste em um controlador preditivo generalizado adaptativo (aGPC, do inglês *adaptive GPC*) (MCINTOSH *et al.*, 1991), o qual considerou como modelo de predição um modelo linear de primeira ordem, e empregou um algoritmo RLS com fator de esqueci-

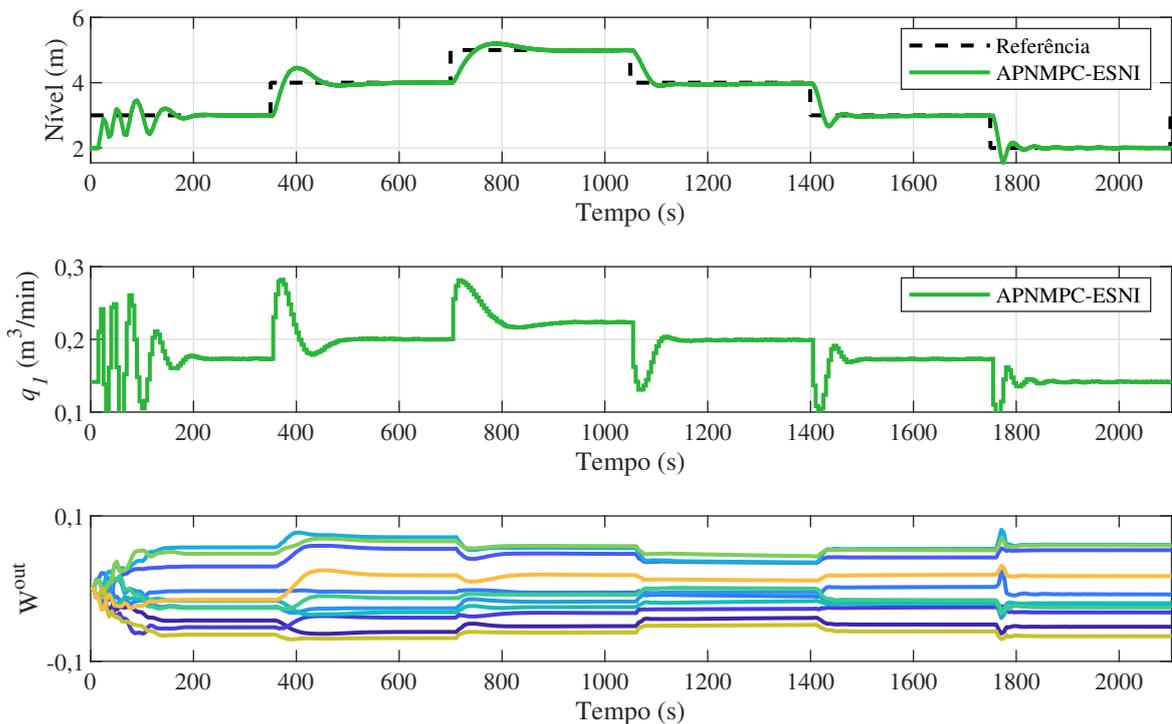
mento adaptativo para estimar on-line os coeficientes do modelo. O estimador RLS foi ajustado com os mesmos parâmetros considerados para o modelo ESNI, e o controlador utilizou os mesmos horizontes e ponderações que o controlador APNMPC-ESNI. O segundo controlador para comparação consiste na versão não adaptativa do controlador proposto, PNMPC-ESNI, o qual utilizou um modelo ESNI obtido em uma etapa de identificação prévia. O processo para identificação foi similar ao apresentado na Seção 3.4, com o processo sendo excitado em malha aberta com um sinal binário pseudo-aleatório modulado em amplitude (APRBS, do inglês *Amplitude modulated Pseudo-Random Binary Sequence*), e o modelo ESNI considerando inicialização igual àquela utilizada para o modelo adaptativo, e tamanho de reservatório 100. O terceiro controlador consiste em um MPC não linear (NMPC), com modelo de predição considerando as equações fenomenológicas em tempo contínuo do processo, aumentadas com um termo integrador e discretizadas utilizando o método trapezoidal. Foi utilizado um filtro de Kalman estendido como observador de estados, considerando os pesos associados aos ruídos de observação e processo inicializados considerando $r = 1 \times 10^{-3}$ e $q = 1 \times 10^{-3}$, respectivamente. Este controlador serve como melhor caso, já que é desenvolvido considerando um modelo perfeito do processo. Para a solução do problema de otimização não linear associado ao NMPC, foi utilizado um algoritmo SQP. Este controlador também utilizou a mesma sintonia do controlador APNMPC-ESNI.

4.4.2.1 Resultados da estimação em malha fechada

Devido ao fato de o modelo ESNI ser estimado on-line, permitindo com que o controlador se adapte às mudanças paramétricas do processo, os parâmetros da camada de saída da ESNI, \mathbf{W}^{out} , podem ser pré-inicializados, usando dados adquiridos previamente, estimados em malha aberta, ou estimado diretamente em malha fechada. Para o terceiro caso, quando o modelo não é pré-inicializado, o processo de estimação do modelo ESNI pode ser desafiador, pois a excitação do processo é geralmente limitada.

Para avaliar o processo de estimação on-line em malha fechada, sem considerar um procedimento para pré-inicialização do modelo, um cenário de teste foi desenvolvido, considerando diversas trocas de referências levando o processo para pontos de operação distintos. Este cenário simulou o controlador proposto, considerando o modelo ESNI com um reservatório com apenas 10 neurônios, para possibilitar a inspeção visual da evolução de cada parâmetro estimado, sendo que os resultados para distintos tamanhos de reservatório são apresentados em seguida. A primeira parte da simulação em malha fechada é apresentada na Figura 28, na qual é apresentado o comportamento temporal da variável de processo, da variável manipulada, e os elementos estimados de \mathbf{W}^{out} .

Os parâmetros de \mathbf{W}^{out} são inicializados com 0 e convergem inicialmente após as primeiras amostras do ensaio (entre 0 s e 200 s). Após as trocas de referência subsequentes, os parâmetros modificam pouco, principalmente quando novas informações se tornam disponíveis, que são os instantes em que a informação acerca de regiões de operação novas se tornam

Figura 28 – Controle em malha fechada e parâmetros W^{out} estimados.

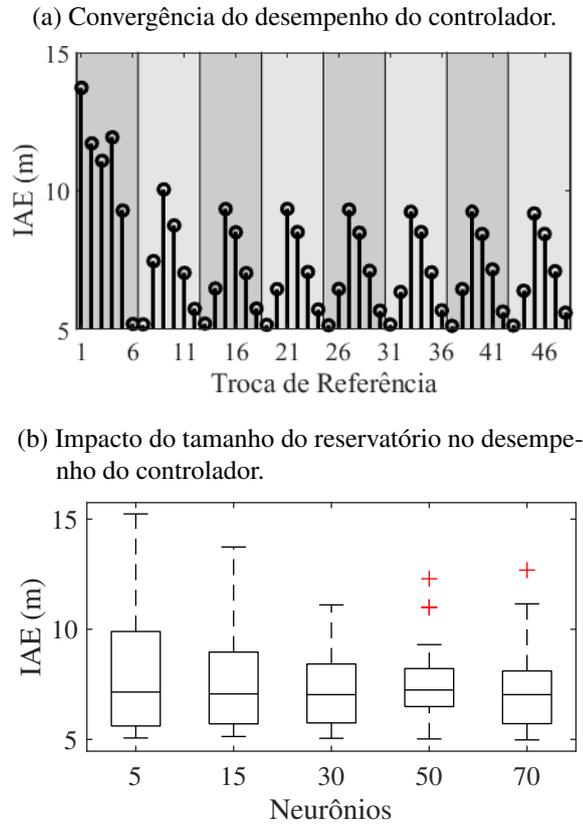
Fonte: dados da pesquisa.

disponíveis, por exemplo entre 1750 s e 2100 s).

Devido ao modelo não ser pré-inicializado, o controlador opera, nos primeiros instantes, com um modelo que representa mal a dinâmica do processo. À medida que o modelo ESNI evolui, durante o processo de estimação, adquirindo informação acerca do comportamento do processo em pontos de operação distintos, o desempenho em malha fechada melhora. Para avaliar de que forma o desempenho do controlador evolui, a simulação apresentada na Figura 28 foi repetida 8 vezes, considerando as mesmas trocas de referência, resultando em uma simulação com 48 trocas de referência. A integral do erro absoluto (IAE) para cada troca de referência é apresentada na Figura 29a, a qual apresenta a forma com que essa métrica de desempenho evolui durante o processo de estimação do modelo. Durante as primeiras trocas de referência, o desempenho evolui rapidamente, ao passo que o modelo aprende a dinâmica do processo, sendo que o desempenho converge após as primeiras 10 trocas de referência, o que é um indicativo de que os parâmetros do modelo ESNI convergiram.

Utilizando o mesmo cenário de teste, e considerando tamanhos diferentes de reservatório, o impacto da sintonia deste parâmetro foi avaliado. Considerando 6 tamanhos de reservatório (5, 15, 30, 50 e 70), a métrica IAE para cada transição de referência foi calculada, e o resultado é apresentado por meio de um diagrama de caixas na Figura 29b. O desempenho do controlador melhora à medida que o tamanho do reservatório aumenta, porém, para valores superiores a 30 os resultados passam a ser semelhantes.

Figura 29 – Avaliação da convergência do desempenho do controlador e da influência do tamanho do reservatório no desempenho.



Fonte: dados da pesquisa.

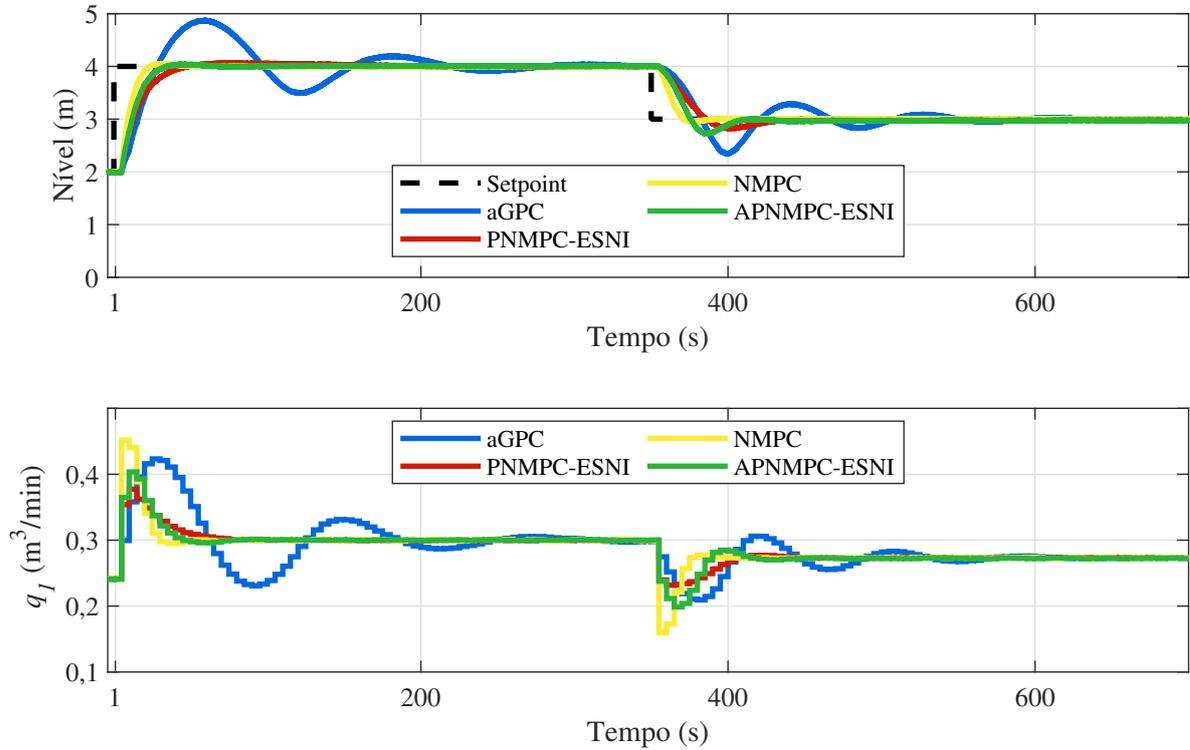
4.4.2.2 Resultados do controle em malha fechada

O desempenho do controlador em malha fechada foi avaliado com três tamanhos de reservatório (5, 15 e 30). O teste em malha fechada considerou dois cenários distintos. O primeiro cenário consiste em um cenário com variações de referências, iniciando com uma referência de 2,0 m, a qual é modificada para 4,0 m e posteriormente para 3,0 m. O segundo cenário considera as mesmas trocas de referência do primeiro cenário, porém considera uma mudança paramétrica no processo, com a diminuição de $0,01 \frac{\text{m}^3}{\text{min}\sqrt{\text{m}}}$ na vazão da válvula de efluente q_2 . Para ambos os cenários, o modelo ESNI foi pré-inicializado usando 6 trocas de referência, similarmente ao cenário apresentado na Figura 28. Os resultados para o modelo com reservatório com 30 unidades são apresentados na Figura 30a para o cenário de troca de referências e na Figura 30b para o cenário com variação paramétrica.

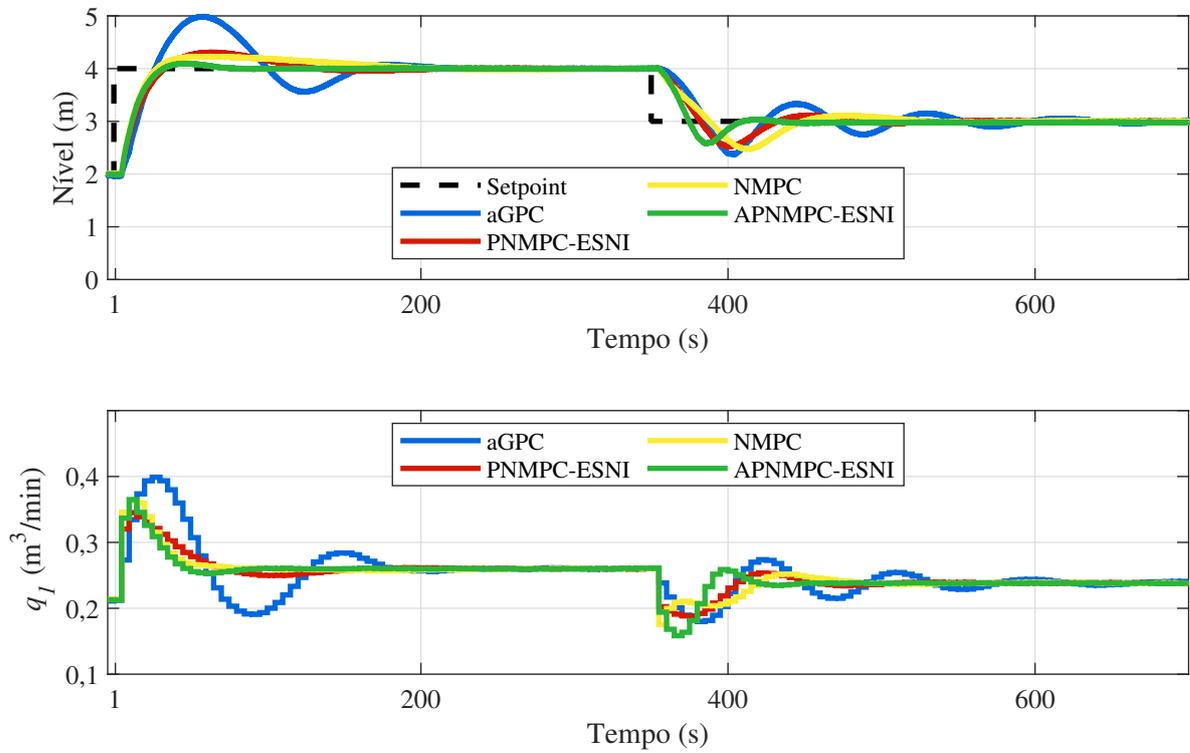
A simulação apresentada na Figura 30a avalia o controlador em uma tarefa em que os parâmetros do processo não variam e o processo opera em pontos de operação distintos. Neste cenário, o controlador NMPC apresenta o melhor desempenho, com rápidas trocas de referência a pouco sobressinal. O controlador APNMPC-ESNI apresentou resultados semelhantes aos alcançados pela versão não adaptativa, PNMPC-ESNI, indicando que o controlador adaptativo foi capaz de estimar um modelo não linear adequado. Por sua vez, o controlador aGPC não foi

Figura 30 – Resultados em malha fechada do tanque de perfil cônico.

(a) Cenário com troca de referências.



(b) Cenário com variação paramétrica do processo.



Fonte: dados da pesquisa.

capaz de alcançar bons resultados em malha fechada, já que seu desempenho sofreu grandes mudanças decorrentes das variações de referência, que levaram o processo a outros pontos de operação. Isso acontece pois o seu modelo linear, apesar de ser adaptativo, é uma aproximação linear da dinâmica do processo, impactando o desempenho do controlador nas mudanças de referência que levam o processo a pontos de operação distantes do atual.

Por sua vez, a simulação apresentada na Figura 30b testa os controladores em um cenário em que o processo apresenta variações paramétricas. Neste caso, controladores adaptativos apresentam potencial para superar controladores não adaptativos, já que eles são capazes de adaptar seus modelos de predição às variações paramétricas e representar com melhor precisão a dinâmica do processo. Para este cenário, o controlador APNMPC-ESNI alcança um desempenho similar ao verificado no primeiro cenário, o que não foi observado para os demais controladores, já que todos eles apresentaram alguma piora no desempenho, com tempos de acomodação mais demorados e maiores sobressinais.

Os resultados finais, para ambos cenários, são comparados utilizando algumas métricas de desempenho na Tabela 5. As métricas avaliadas são a IAE, o esforço de controle (CE), e os tempos de cômputo médios e máximos, para avaliar a eficiência computacional.

Os resultados apresentados na Tabela 5 demonstram que o tamanho do reservatório afeta o desempenho em malha fechada, já que as melhores métricas IAE foram alcançadas para $r = 30$. Todas as configurações do controlador APNMPC-ESNI apresentaram melhores resultados que o controlador aGPC, e a configuração com o maior número de unidades no reservatório apresentou IAE similar ao alcançado pelo PNMPC-ESNI, para o cenário de troca de referências. O maior benefício da utilização de uma abordagem adaptativa é observado quando os controladores são avaliados para o cenário com variação paramétrica do processo, cenário esse em que o IAE do controlador APNMPC-ESNI foi próximo ao observado no cenário de troca de referências, o que não foi verificado para os controladores não adaptativos, já que eles não foram capazes de se adaptar à variação paramétrica. Neste cenário, os controladores não adaptativos apresentaram uma clara piora no desempenho, o que fez com que o controlador aPNMPC-ESNI apresentasse as melhores métricas de desempenho.

Outro benefício do controlador proposto é o seu baixo custo computacional requerido para estimação do modelo e obtenção do sinal de controle. Como apresentado na Tabela 5, os tempos de cômputo máximos, considerando o maior reservatório testado ($r = 30$), foram 34,66 ms, o que é maior que o verificado para o algoritmo aGPC, porém, aproximadamente 9 vezes menor que o requerido pelo controlador NMPC.

4.4.3 Exemplo 6: Controle adaptativo de um motor síncrono de ímãs permanentes

O segundo exemplo da aplicação do algoritmo APNMPC-ESNI consiste no controle de um motor síncrono de ímãs permanentes (PMSM, do inglês *Permanent Magnet Synchronous Motor*) simulado. A descrição do PMSM, da sua simulação é detalhada no Apêndice A.4. O controle do PMSM considera um cenário em que a velocidade angular ω_m deve seguir um dado

Tabela 5 – Resultados do controle em malha fechada do tanque de perfil cônico.

Controlador	IAE cenário troca de referência (m)	IAE cenário variação paramétrica (m)	CE cenário troca de referência (m ³ /min) × 10 ⁻⁴	CE cenário variação paramétrica (m ³ /min) × 10 ⁻⁴	Tempo de cômputo médio (ms)	Tempo de cômputo máximo (ms)
aGPC	31,74	36,73	2,28	1,98	9,17	13,66
PNMPC-ESNI	14,49	19,45	2,31	1,76	34,56	53,12
NMPC	9,04	20,90	2,29	1,81	152,98	330,89
APNMPC-ESNI ($r = 5$)	16,17	15,51	2,28	1,96	15,43	24,41
APNMPC-ESNI ($r = 15$)	14,43	14,70	2,16	1,86	17,80	27,51
APNMPC-ESNI ($r = 30$)	13,37	13,77	2,21	1,71	21,95	34,66

Fonte: dados da pesquisa.

valor de referência e a corrente i_d é mantida próxima de 0 A. As variáveis manipuladas são u_d e u_q . Na simulação, o torque da carga é considerado constante, com $T_l = 0,05$ N m, e o valor da resistência elétrica do enrolamento é inicialmente mantido constante e posteriormente variado, quando um cenário adaptativo é considerado. Foi utilizado um período de amostragem de 0,1 ms. A simulação desse estudo de caso foi realizada utilizando um computador com processador Intel Core i7-9750H, 16 gb de memória e foi programada na linguagem MATLAB.

O controle do PMSM foi sintonizado utilizando conhecimento acerca das regiões típicas de aplicação do processo, com um procedimento similar ao proposto em Dangui *et al.* (2018). Os horizontes de predição, para ambas as variáveis controladas, foi selecionado como $N_\omega = N_i = 10$, e os horizontes de controle, para ambas as variáveis manipuladas, foi definido como $N_{u,\omega} = N_{u,i} = 3$. As sequências de ponderações do erro de seguimento de referência e esforço de controle foram definidas como constantes para todo horizonte, com $\delta_\omega = 1$, $\delta_i = 10$, $\lambda_\omega = 100$, e $\lambda_i = 5000$.

O estimador on-line do modelo ESNI foi inicializado com uma matriz $\mathbf{W}_{\text{res}}^{\text{res}}$ esparsa e aleatória, considerando 10% de conexões não nulas, amostradas de uma distribuição uniforme no intervalo $[-1, +1]$. Após a inicialização de $\mathbf{W}_{\text{res}}^{\text{res}}$, a matriz foi reescalada, fazendo com que o raio espectral seja $\rho(\mathbf{W}_{\text{res}}^{\text{res}}) = 0,99$, de forma a garantir a propriedade de estado de eco e, ao mesmo tempo, garantir que o modelo ESNI seja capaz de modelar dinâmicas lentas. Esse valor pode ser ajustado para melhor o desempenho do modelo, dependendo da tarefa de identificação, porém, o valor proposto se mostrou como um bom ajuste inicial para dinâmicas variadas. A matriz de entrada $\mathbf{W}_{\text{in}}^{\text{res}}$ foi inicializada com todos os pesos sendo +1 ou -1 (50% para cada caso), e reescalada usando o maior ganho estático estimado para o processo, o qual foi 20. Por sua vez, o estimador RLS foi inicializado considerando os elementos da diagonal principal da matriz de covariância como $\mathbf{P}_{ii}(0) = 1$; $\varphi(0) = 1$; $\lambda(0) = 0,1$; e $\nu(0) = 10^{-6}$.

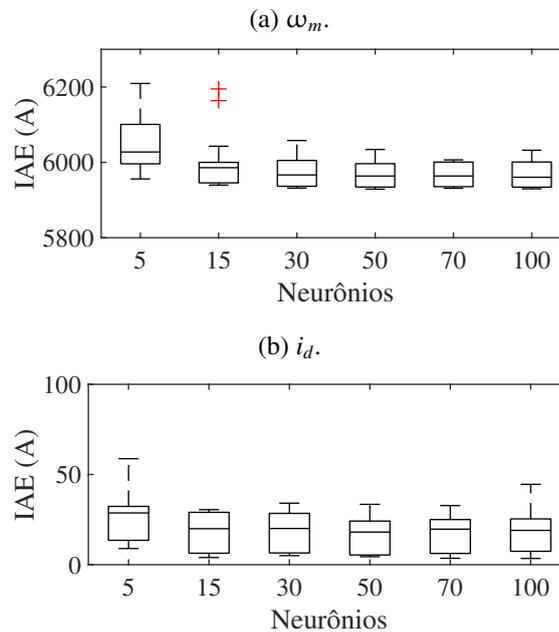
O controlador proposto foi comparado com três controladores. O primeiro foi o aGPC (MCINTOSH *et al.*, 1991), o qual utiliza um modelo linear de primeira ordem, com modelo estimado por um algoritmo RLS com fator de esquecimento adaptativo. O estimador RLS foi sintonizado com os mesmos parâmetros utilizados pelo modelo ESNI, o controlador utilizou os mesmos horizontes de predição e controle, e as mesmas ponderações que o controlador APNMPC-ESNI. O segundo controlador usado para comparação é a versão não adaptativa PNMPC-ESNI, sendo que o modelo ESN foi obtido previamente, a partir de dados de um ensaio de identificação. O terceiro controlador consiste em um NMPC, projetado utilizando o modelo fenomenológico do PMSM, aumentado com um integrador e discretizado utilizando o método trapezoidal. Um filtro de Kalman estendido foi utilizado como observador de estados, com a covariância das observações e ruído inicializados como $r = 1 \times 10^{-3}$ e $q = 1 \times 10^{-3}$, respectivamente. Esta implementação serve como o melhor caso, já que o controlador é projetado com um modelo perfeito. A solução do problema NLP é realizada com um algoritmo SQP (BIEGLER, 2000). A sintonia do controlador foi a mesma usada para o controlador APNMPC-ESNI.

4.4.3.1 Resultados da estimação em malha fechada

O controlador APNMPC-ESNI estima on-line os parâmetros do processo, para se adaptar às variações no comportamento dinâmico do processo. Os parâmetros ajustados do modelo ESNI, \mathbf{W}^{out} , podem ser pré-inicializados, usando dados adquiridos previamente do processo, ou estimados diretamente em malha fechada. Neste segundo caso, a estimação pode ser desafiadora, já que a excitação do processo é, geralmente, limitada.

Para avaliar o processo de estimação em malha fechada, sem a pré-inicialização do modelo, um cenário de estimação considerando diversas trocas de referência foi projetado. O impacto do tamanho do reservatório do modelo ESNI foi avaliado com esse cenário, considerando distintos tamanhos de reservatórios (5, 15, 30, 50, 70 e 100). Foi avaliado o IAE associado a cada troca de referência, para a velocidade angular e para a corrente i_d , com os resultados sendo apresentados por meio de diagramas de caixas nas Figuras 31a e 31b. Para ambas as variáveis controladas o aumento do tamanho do reservatório resultou em melhores resultados em malha fechada, porém, reservatórios com mais de 30 unidades apresentaram desempenhos similares.

Figura 31 – Avaliação da influência do tamanho do reservatório no desempenho do controlador para ω_m e i_d .

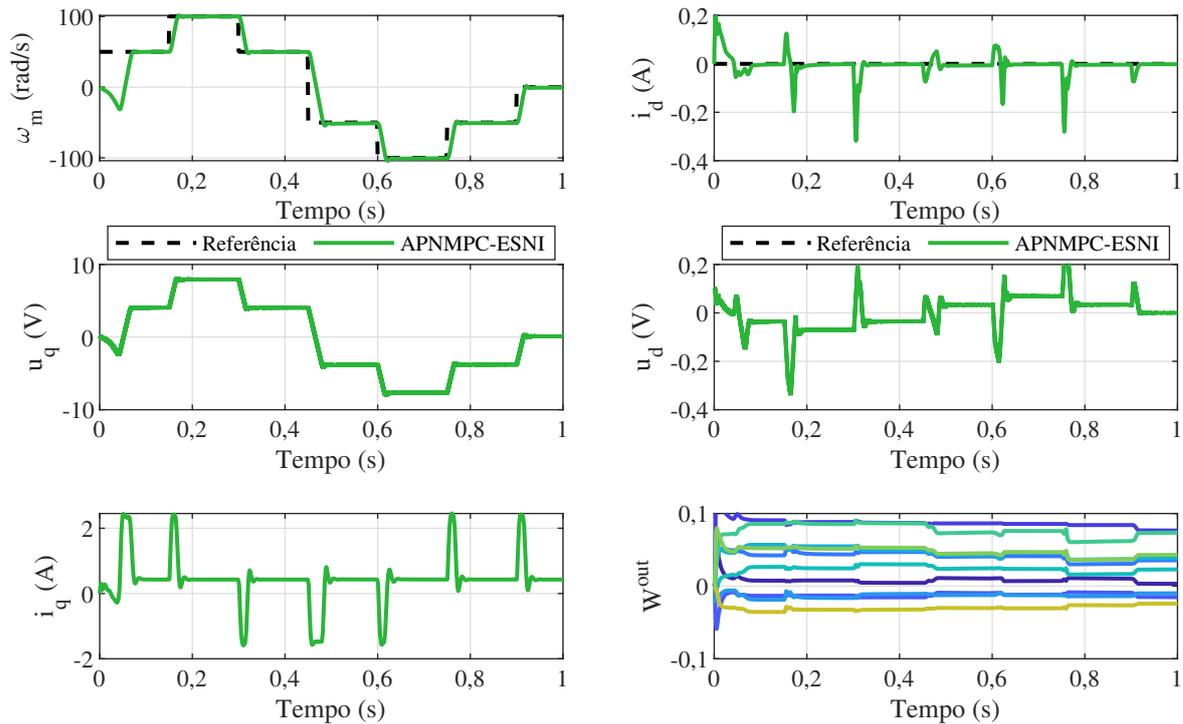


Fonte: dados da pesquisa.

Uma porção do cenário de estimação, considerando o modelo ESNI com um reservatório com 30 unidades, é apresentada na Figura 32, a qual contém o comportamento temporal das variáveis controladas e manipuladas e, adicionalmente, alguns dos parâmetros estimados da matriz \mathbf{W}_{out} .

Devido aos parâmetros do modelo ESNI não terem sido pré-inicializados, o controlador opera, nos primeiros instantes de tempo, com um modelo que não representa bem a dinâmica do processo. À medida que o modelo ESNI evolui durante o processo de estimação, adquirindo

Figura 32 – Resultados em malha fechada do estimador para o cenário de estimação.



Fonte: dados da pesquisa.

informações acerca do comportamento do processo em regiões de operação distintas, o desempenho do controlador melhora, com os parâmetros do modelo convergindo para valores constantes. Os instantes em que os parâmetros do modelo são mais atualizados são os instantes em que o processo apresenta excitação mais rica, o que ocorre durante as transições de referência.

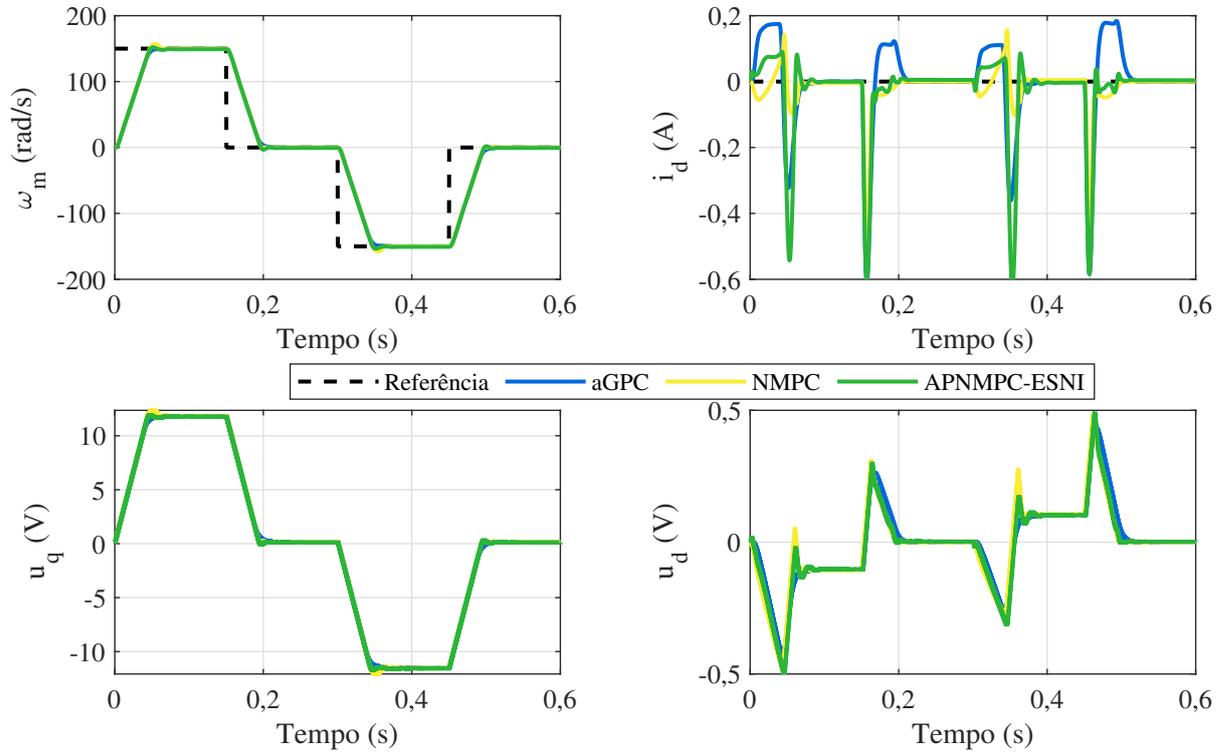
4.4.3.2 Resultados do controle em malha fechada

O desempenho em malha fechada foi avaliado considerando três tamanhos de reservatório (5, 30 e 50). Dois cenários de teste foram desenvolvidos, o primeiro consistindo em um cenário de trocas de referência, no qual a primeira referência de velocidade angular é 0 RPM, e são aplicadas quatro trocas de referência, entre +150 RPM, 0 RPM, -150 RPM e 0 RPM. O segundo cenário é equivalente ao primeiro, com as mesmas trocas de referência, porém, considera uma variação paramétrica, com um aumento de 20% na resistência de enrolamento, R_s , simulando o efeito de variações na temperatura de operação do motor. Para ambos os cenários, os modelos adaptativos foram pré-inicializados utilizando seis trocas de referência, de forma semelhante ao mostrado na Figura 32. Os resultados para o modelo ESNI cujo reservatório é composto por 30 unidades são apresentados nas Figuras 33a e 33b, para os cenários de troca de referência e com variação paramétrica, respectivamente.

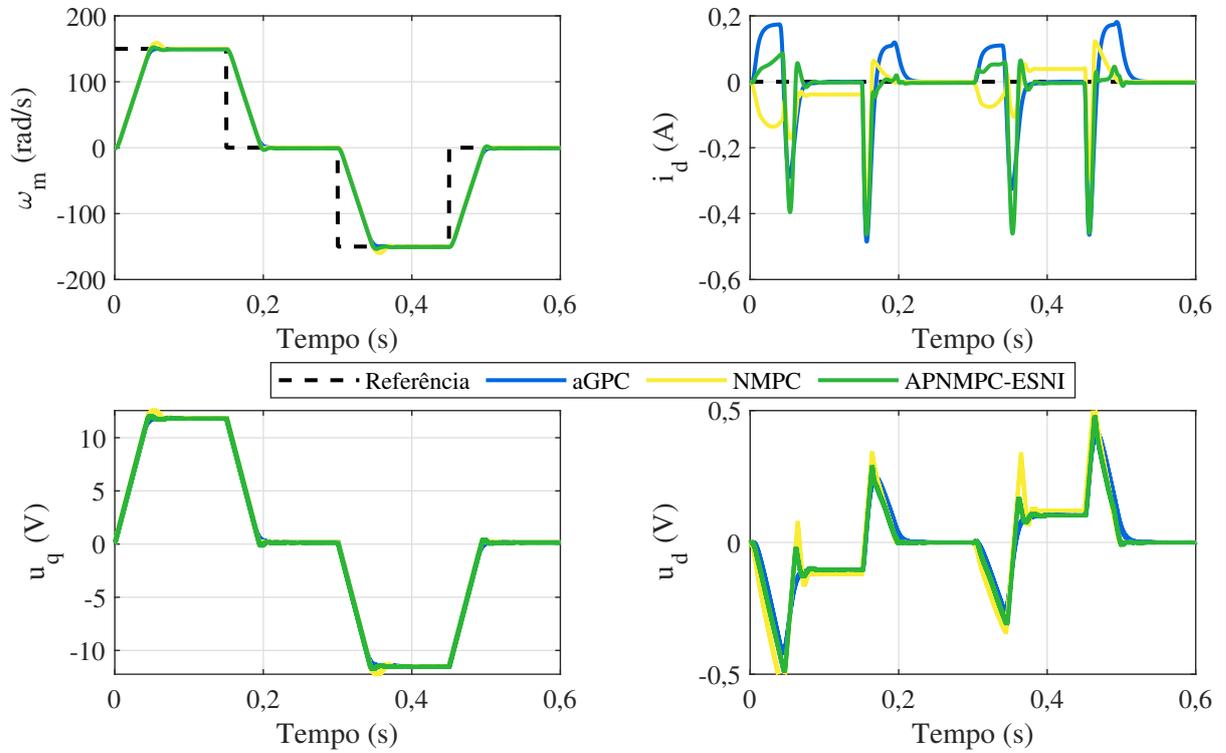
A simulação apresentada na Figura 33a avalia o controlador APNMPC-ESNI em uma tarefa na qual o comportamento do processo varia devido a mudanças na condição de operação do processo, à medida que o processo não linear é levado a diferentes regiões de operação. Nessa simulação, o controlador NMPC representa o limite superior de desempenho alcançável

Figura 33 – Resultados em malha fechada para o PMSM.

(a) Cenário com troca de referências.



(b) Cenário com variação paramétrica.



Fonte: dados da pesquisa.

por uma implementação MPC não linear, já que ele utiliza um modelo de predição com uma representação perfeita do processo. Neste caso, transições de referência rápidas e sobressinais quase inexistentes são verificados. O algoritmo proposto, o qual usa um modelo identificado, apresenta resultados similares aos verificados pelo NMPC, o que demonstra que a abordagem proposta é capaz de estimar um modelo não linear adequado, fazendo com que o resultado em malha fechada apresente trocas de referência rápidas. O controlador aGPC foi capaz de produzir bons resultados em malha fechada, porém, o desempenho em relação a i_d foi bastante afetado pela mudanças na região de operação do processo. Isso acontece devido ao modelo de predição ser uma representação linear, que apesar de ser adaptativa, é capaz, apenas, de aproximar linearmente a dinâmica do processo para o ponto de operação atual.

Por sua vez, a simulação apresentada na Figura 33b avaliou o desempenho do controlador em um cenário em que o processo sofre uma variação paramétrica. Nesse caso, controladores adaptativos apresentam um potencial para apresentar melhores resultados que abordagens não adaptativas, já que eles são capazes de adaptar seu modelo de predição para melhor representar o comportamento dinâmico atual do processo, o que resulta em uma sintonia mais apropriada do controlador aos novos parâmetros de operação do processo. Nesse cenário, foi verificado que as abordagens adaptativas apresentaram um desempenho similar ao verificado no primeiro cenário, porém, ambos os controladores não adaptativos (PNMPC-ESNI e NMPC) sofreram uma piora de desempenho, com tempos de acomodação mais demorados e maiores sobressinais.

Os resultados finais, para ambos os cenários, são sintetizados por meio de métricas de desempenho, apresentadas na Tabela 6. Neste exemplo, o desempenho do controlador é avaliado utilizando a IAE e também são apresentados os custos computacionais máximos e médios para cada controlador.

Os resultados demonstram que o tamanho do reservatório apresenta influência na qualidade das estimativas, já que, em geral, os menores valores para a IAE foram obtidos com um número de unidades no reservatório $r = 50$. Para a variável ω_m , os resultados foram similares para todos controladores, com uma variação máxima de 2% entre todas as implementações, para ambos os cenários de teste. Já para o controle de i_d , todas as configurações do controlador proposto, APNMPC-ESNI, apresentaram resultados melhores que os do aGPC. Considerando o cenário de troca de referências, a configuração com maior reservatório ($r = 50$) apresentou IAE similar ao verificado para o controlador não adaptativo PNMPC-ESN, o que indica que a abordagem de identificação on-line do modelo foi capaz de estimar um modelo adequado.

Quando o cenário com variação paramétrica é considerado, o maior benefício da abordagem adaptativa é verificado, já que o IAE do controlador APNMPC-ESNI é similar ao verificado para o cenário sem variação paramétrica. Isso não é verificado para os controladores PNMPC-ESN e NMPC, já que eles não incorporam um mecanismo capaz de compensar os efeitos da variação paramétrica do processo. O controlador APNMPC-ESNI é capaz de adaptar seu modelo de predição já que integra em seu algoritmo um procedimento de identificação on-line, permitindo que o modelo de predição se adapte a variações paramétricas. Adicionalmente, além de

Tabela 6 – Métricas de desempenho para os resultados em malha fechada dos controladores.

Controlador	ω_m IAE (RPM) (cenário de troca de referências)	i_d IAE (A) (cenário de troca de referências)	ω_m IAE (RPM) (cenário de variação paramétrica)	i_d IAE (A) (cenário de variação paramétrica)	Tempo de cômputo médio (ms)	Tempo de cômputo máximo (ms)
aGPC	$5,00 \times 10^4$	130,54	$5,08 \times 10^4$	121,96	7,29	11,33
PNMPC-ESN	$5,06 \times 10^4$	91,07	$5,23 \times 10^4$	138,20	4,47	6,57
NMPC	$4,99 \times 10^4$	54,51	$5,11 \times 10^4$	84,19	88,62	114,53
APNMPS-ESNI ($r = 5$)	$5,00 \times 10^4$	100,75	$5,16 \times 10^4$	113,39	4,88	11,25
APNMPS-ESNI ($r = 30$)	$5,05 \times 10^4$	98,62	$5,17 \times 10^4$	76,84	6,51	11,37
APNMPS-ESNI ($r = 50$)	$5,05 \times 10^4$	91,57	$5,15 \times 10^4$	74,12	6,53	12,48

Fonte: dados da pesquisa.

apresentar resultados similares ao caso sem variação paramétrica, o controlador APNMPC-ESNI apresentou melhor IAE que o controlador NMPC no cenário com variação paramétrica.

Quanto à complexidade computacional, como apresentado na Tabela 6, o tempo máximo de cômputo, considerando o maior reservatório ($r = 50$), foi 12,48 ms, o que é próximo ao requerido pelo aGPC e aproximadamente 10 vezes menor que o requerido pelo NMPC.

4.5 ALGORITMO NMPC PRÁTICO COM PREDIÇÃO ITERATIVA PARA MODELOS ESNI

O algoritmo PNMPC-ESNI e sua variante adaptativa APNMPC-ESNI se baseiam em uma abordagem na qual a predição é dividida em dois termos: a resposta livre, que depende apenas das ações de controle passadas; e a resposta forçada, que depende das ações de controle futuras. Devido ao fato de o modelo ESNI ser não linear, tal divisão é uma representação imprecisa do comportamento não linear real do processo, devido ao princípio da superposição não ser verificado para sistemas não lineares. Uma alternativa para diminuir o impacto dessa representação imprecisa é utilizar uma divisão diferente para o modelo de predição, a qual se baseie em uma interpretação alternativa para os termos resultantes dessa divisão. Uma interpretação alternativa foi proposta no algoritmo NEPSAC (do inglês, *Nonlinear Extended Prediction Self-Adaptive Control*) (DE KEYSER, Robin, 2009), o qual divide a predição não linear em dois termos: a resposta base, a qual é composta pela resposta do modelo não linear às ações de controle conhecidas, incluindo as ações de controle passadas e ações de controle futuras, quando conhecidas; e a resposta ótima, que é composta pela resposta do modelo não linear às ações de controle que serão obtidas por meio da resolução do problema de otimização. Com base nesta interpretação, o algoritmo NEPSAC utiliza um procedimento iterativo, no qual são obtidas estimativas do esforço de controle futuro, ao resolver o problemas de otimização, as quais são integradas à resposta base, na próxima iteração. Tal procedimento iterativo é conduzido até que a resposta base seja suficientemente maior que a resposta ótima, o que é um indicativo de que a predição é suficientemente precisa, já que a resposta base é obtida usando o modelo não linear original, fazendo com que a aproximação resultante da divisão do modelo de predição seja insignificante.

Esse tipo de abordagem apresenta um potencial para aprimorar o algoritmo PNMPC-ESNI, já que o impacto da divisão da predição em respostas livre e forçada pode ser reduzido ao utilizar um procedimento iterativo que se baseie nas respostas base e ótima. Devido a isso, nesta seção é proposta uma abordagem PNMPC iterativa para modelos ESNI (PNMPCI-ESNI). No decorrer da seção é formalizada a divisão do modelo de predição em resposta base e ótima, sendo detalhados os procedimentos para obtenção de cada termo. Em seguida, é apresentado o procedimento iterativo e são detalhados os critérios de parada do mesmo. Para exemplificar a utilização da estratégia de controle proposta, é utilizado o processo de neutralização de pH, com os resultados sendo comparados com os obtidos na Subseção 4.3.2

4.5.1 Predição iterativa com respostas base e ótima

Na arquitetura de controle PNMPCI-ESNI, o modelo ESNI é utilizado para a predição do comportamento futuro do sistema de forma semelhante ao PNMPC-ESNI, com a predição sendo dividida em duas porções, as respostas base e ótima. De forma similar à resposta livre, o modelo completo não linear é usado para obter a resposta base não linear do processo. Já a resposta ótima é obtida, a cada iteração do algoritmo, usando uma versão linearizada do modelo. De forma semelhante à divisão realizada no algoritmo PNMPC, a equação de predição resultante é afim em relação ao incremento de controle, $\Delta \mathbf{u}$, a qual, para o caso MIMO para um sistema com m saídas e n entradas, é definida pela expressão

$$\hat{\mathbf{y}}(k) = \mathbf{y}_{\text{ótima}}^p(k) + \mathbf{y}_{\text{base}}^p(k) = \mathbf{G}_{\text{PNMPCI}}^p(k) \Delta \mathbf{u}^p(k) + \mathbf{y}_{\text{base}}^p(k), \quad (90)$$

na qual p representa o número da iteração,

$$\mathbf{y}_{\text{ótima}}^p(k) = \begin{bmatrix} \mathbf{y}_{\text{ótima},1}^p(k)^T & \dots & \mathbf{y}_{\text{ótima},m}^p(k)^T \end{bmatrix}^T, \quad (91)$$

$$\mathbf{y}_{\text{base}}^p(k) = \begin{bmatrix} \mathbf{y}_{\text{base},1}^p(k)^T & \dots & \mathbf{y}_{\text{base},m}^p(k)^T \end{bmatrix}^T, \quad (92)$$

$$\mathbf{G}_{\text{PNMPCI}}^p(k) = \begin{bmatrix} \mathbf{G}_{\text{PNMPCI},11}^p(k) & \dots & \mathbf{G}_{\text{PNMPCI},1n}^p(k) \\ \vdots & \ddots & \vdots \\ \mathbf{G}_{\text{PNMPCI},m1}^p(k) & \dots & \mathbf{G}_{\text{PNMPCI},mn}^p(k) \end{bmatrix}, \quad (93)$$

$$\Delta \mathbf{u}^p(k) = \begin{bmatrix} \Delta \mathbf{u}_1^p(k)^T & \dots & \Delta \mathbf{u}_n^p(k)^T \end{bmatrix}^T. \quad (94)$$

As estruturas dos vetores de predição das respostas base e ótima, para a i -ésima saída, são semelhantes à do vetor de predições da saída, sendo definidas como

$$\mathbf{y}_{\text{base},i}^p(k) = \begin{bmatrix} y_{\text{base},i}^p(k + N_{1,i}) & \dots & y_{\text{base},i}^p(k + N_{2,i}) \end{bmatrix}^T, \quad (95)$$

$$\mathbf{y}_{\text{ótima},i}^p(k) = \begin{bmatrix} y_{\text{ótima},i}^p(k + N_{1,i}) & \dots & y_{\text{ótima},i}^p(k + N_{2,i}) \end{bmatrix}^T. \quad (96)$$

Devido ao modelo ESNI ser não linear, a equação de predição (90) é uma aproximação da resposta não linear do modelo, já que o princípio da superposição não é verificado para esse caso. Essa aproximação é mais precisa quando o termo $\mathbf{y}_{\text{ótima}}^p(k)$ é suficientemente pequeno quando comparado com $\mathbf{y}_{\text{base}}^p(k)$. A estratégia usada pra alcançar essa característica consiste em obter $\Delta \mathbf{u}^p(k)$ iterativamente, atualizando $\mathbf{u}_{\text{base}}^p(k)$ até que $\|\Delta \mathbf{u}^p(k)\| \approx 0$, o que resulta em um termo $\|\mathbf{y}_{\text{ótima}}^p(k)\| \approx 0$. Dessa forma, o algoritmo iterativo é repetido enquanto $\|\Delta \mathbf{u}^p(k)\| \geq \epsilon$ e o número de iterações $p \leq i_{\text{max}}$. Quando $i_{\text{max}} = 1$, o algoritmo PNMPCI-ESNI se torna idêntico ao algoritmo PNMPC-ESNI, apresentado na Seção 4.3.

O vetor de resposta base, $\mathbf{y}_{\text{base}}^p(k)$, é obtido da trajetória não linear do modelo ESNI. Considerando o conjunto de horizontes de predição $N = \{N_i\}_{i=1}^m$, o modelo é simulado até o fim do maior horizonte de predição (i.e. $\max(N)$), considerando todos os incrementos de controle futuros iguais a

$$\Delta \mathbf{u}_{\text{base},j}^p(k) = \Delta \mathbf{u}_{\text{base},j}^{p-1}(k) + \Delta \mathbf{u}_j^{p-1}(k), \quad (97)$$

e $\Delta \mathbf{u}_{\text{base},j}^1 = 0$, para formar o vetor de entrada

$$\begin{aligned} \mathbf{u}_{\text{base},j}^p(k) &= \left[\mathbf{u}_{\text{base}}^p(k|k)^T \quad \dots \quad \mathbf{u}_{\text{base}}^p(k + \max(N) - 1|k)^T \right]^T \\ &= \left[\mathbf{u}^p(k-1|k-1)^T + \Delta \mathbf{u}_{\text{base}}^p(k) \quad \dots \quad \mathbf{u}^p(k-1|k-1)^T + \Delta \mathbf{u}_{\text{base}}^p(k) \right]^T. \end{aligned} \quad (98)$$

Se forem usadas as equações (19) e (20), o modelo ESNI é simulado e o vetor de resposta base é obtido como

$$\mathbf{y}_{\text{base}}^p(k+i|k) = \mathbf{W}^{\text{out}} \begin{bmatrix} 1 \\ \mathbf{x}_{\text{base}}^p(k+i|k) \\ \mathbf{y}_{\text{base}}^p(k+i-1|k) \end{bmatrix} + \frac{D(z^{-1})a(k)}{\Delta}. \quad (99)$$

na qual $\mathbf{y}_{\text{base}}(k|k) = \mathbf{y}(k)$. Nesse caso, os estados do modelo ESNI, durante a simulação para obtenção da resposta base, são obtidos por meio da equação

$$\mathbf{x}_{\text{base}}^p(k+i|k) = \tanh(\mathbf{W}_{\text{in}}^{\text{res}} \begin{bmatrix} 1 \\ \mathbf{u}_{\text{base}}^p(k+i-1|k) \end{bmatrix}) + \mathbf{W}_{\text{res}}^{\text{res}} \mathbf{x}_{\text{base}}^p(k+i-1|k), \quad (100)$$

na qual $i \in [1, \max(N)]$. Os estados iniciais, $\mathbf{x}_{\text{base}}^p(k|k)$, são obtidos considerando o sinal de entrada, $\mathbf{u}(k-1)$, o qual foi aplicado ao processo no instante atual, por meio de

$$\mathbf{x}_{\text{base}}^p(k|k) = \tanh(\mathbf{W}_{\text{in}}^{\text{res}} \begin{bmatrix} 1 \\ \mathbf{u}(k-1) \end{bmatrix}) + \mathbf{W}_{\text{res}}^{\text{res}} \mathbf{x}_{\text{base}}^p(k-1|k-1). \quad (101)$$

Os vetores de resposta base (95), para cada saída i , são extraídos do resultado de (99), considerando os respectivos horizontes de predição N_i , já que a simulação considera o maior horizonte de predição do conjunto N . Considerando cada vetor de resposta base, o vetor de resposta base MIMO pode ser construído considerando a estrutura apresentada em (92).

De forma semelhante ao algoritmo PNMPC, a matriz de dinâmica $\mathbf{G}_{\text{PNMPCI}}^p(k) \in \mathbb{R}^{\sum_{i=1}^m N_i \times \sum_{j=1}^n N_{u,j}}$ é constituída por matrizes $\mathbf{G}_{\text{PNMPCI},ij}^p(k)$, as quais são formadas pelos coeficientes da resposta ao degrau do modelo ESNI, para um par de saída j e entrada i . Cada matriz $\mathbf{G}_{\text{PNMPCI},ij}^p(k)$ é definida pela expressão

$$\mathbf{G}_{\text{PNMPCI},ij}^p(k) = \begin{bmatrix} s_{1,1}^p(k) & 0 & \dots & 0 \\ s_{2,1}^p(k) & s_{1,2}^p(k) & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ s_{N_i,1}^p(k) & s_{(N_i-1),2}^p(k) & \dots & s_{(N_i-N_{u,j}+1),(N_{u,j})}^p(k) \end{bmatrix}, \quad (102)$$

na qual o vetor $\mathbf{g}_{ij,s}^p(k) = [s_{1,1}^p(k) \dots s_{N_i,1}^p(k)]^T$ corresponde aos coeficientes entre $N_{1,i}$ e $N_{2,i}$, com $N_{1,i} = d + 1$. Esses coeficientes são obtidos da mesma forma que para o algoritmo PNMPC, como descrito na Seção 4.3, simulando a resposta do modelo ao aplicar um degrau na entrada j em $k + s$. A amplitude do degrau é definida, para cada entrada j , como $w = \Delta u_j(k-1)/10$.

Para obter a coluna s de cada vetor de resposta ótima, associada à entrada j , o modelo ESNI é simulado considerando como entrada

$$\mathbf{u}_{\text{ótima},js}^p(k) = \mathbf{u}_{\text{base}}^p(k) + w \begin{bmatrix} \mathbf{0}_{(j-1) \max(N)} \\ \mathbf{0}_{s-1} \\ \mathbf{1}_{\max(N)-s+1} \\ \mathbf{0}_{(n-j) \max(N)} \end{bmatrix}, \quad (103)$$

na qual $\mathbf{0}_a \in \mathbb{R}^a$ e $\mathbf{1}_a \in \mathbb{R}^a$ representam vetores de zeros e uns, respectivamente.

O vetor $\mathbf{g}_{js}^p(k)$ é extraído de cada coluna de

$$\mathbf{g}_{js}^p(k) = w^{-1}(\mathbf{y}_{\text{ótima},js}^p(k) - \mathbf{y}_{\text{base}}^p(k)), \quad (104)$$

considerando

$$\mathbf{y}_{\text{ótima},js}^p(k+i|k) = \mathbf{W}^{\text{out}} \begin{bmatrix} 1 \\ \mathbf{x}_{\text{ótima},js}^p(k+i|k) \\ \mathbf{y}_{\text{ótima},js}^p(k+i-1|k) \end{bmatrix} + \frac{D(z^{-1})a(k)}{\Delta}, \quad (105)$$

e $\mathbf{y}_{\text{ótima},js}^p(k|k) = \mathbf{y}(k)$. Os estados do modelo ESNI são obtidos pela equação

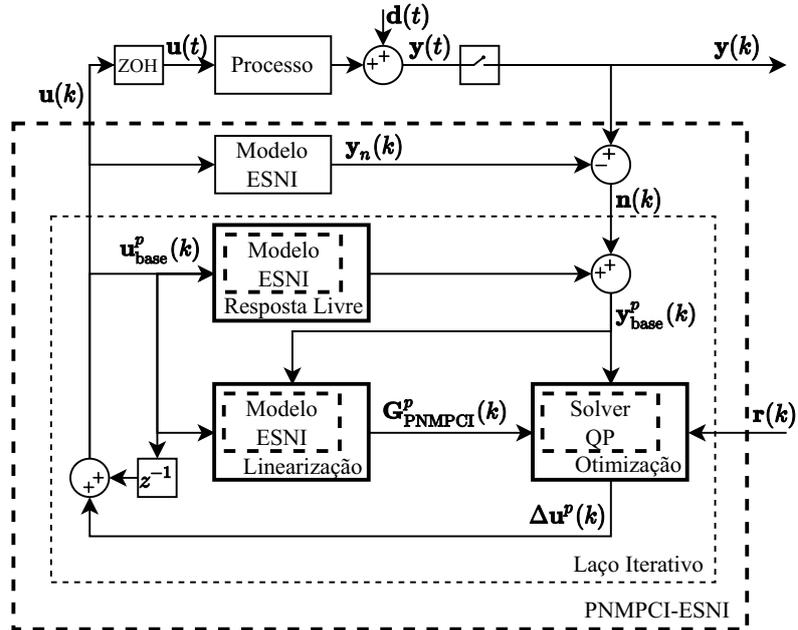
$$\mathbf{x}_{\text{ótima},js}^p(k+i|k) = \tanh(\mathbf{W}_{\text{in}}^{\text{res}} \begin{bmatrix} 1 \\ \mathbf{u}_{\text{ótima},js}^p(k+i-1|k) \end{bmatrix}) + \mathbf{W}_{\text{res}}^{\text{res}} \mathbf{x}_{\text{ótima},js}^p(k+i-1|k), \quad (106)$$

sendo que os estados iniciais são obtidos da mesma forma descrita em (101).

Uma representação geral do funcionamento deste algoritmo é apresentada por meio de um diagrama de blocos na Figura 34.

O algoritmo PNMPCE-ESNI é sumarizado no Algoritmo 5.

Figura 34 – Ilustração da estrutura de controle PNMPCI-ESNI.



Fonte: elaborado pelo autor.

Algoritmo 5 Algoritmo PNMPCI para modelos ESNI

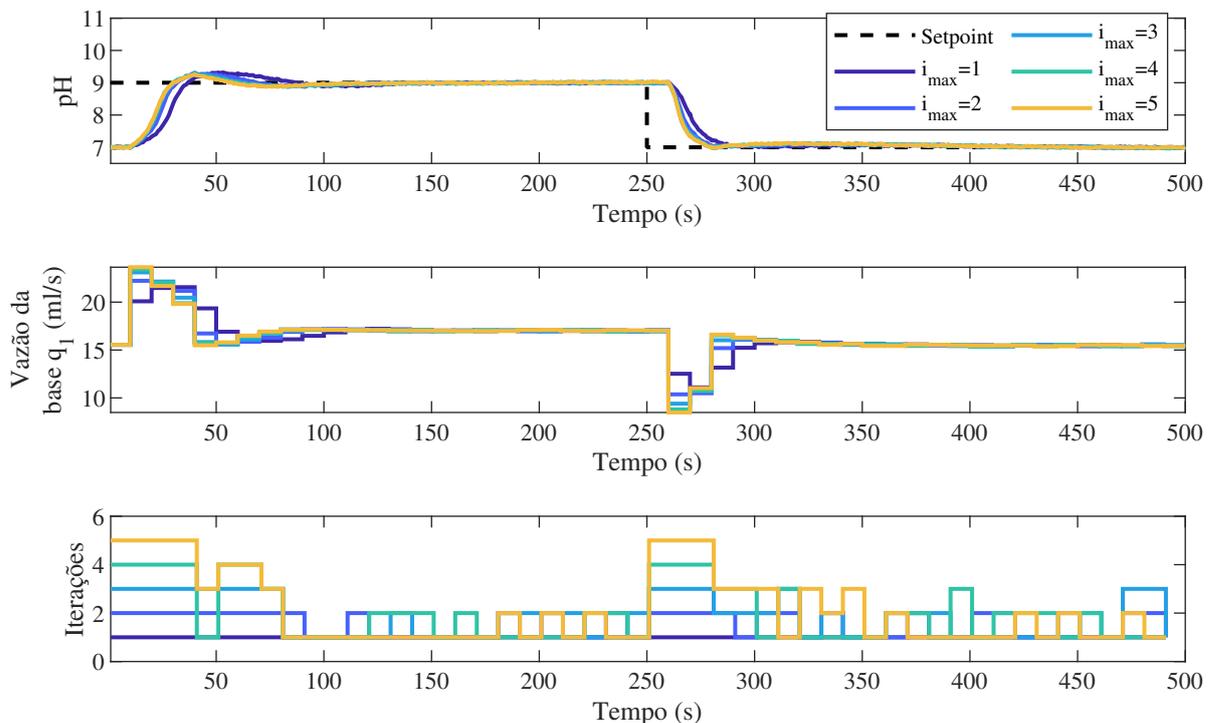
-
- 1: Medir as saídas atuais do processo.
 - 2: Definir $\mathbf{u}_{\text{base}}^1(k) = \mathbf{u}(k-1)$ e $p = 1$.
 - 3: **Enquanto** $\|\Delta\mathbf{u}^p\| \geq \epsilon$ e $p \leq i_{\text{max}}$ **Faça**.
 - 4: **Para** $i = 1 : m$ **Faça**
 - 5: Obter os estados \mathbf{x}_e^p do reservatório, considerando como entrada $\mathbf{u}_{\text{base},i}^p(k)$, definido em (98).
 - 6: Obter a resposta livre não linear $\mathbf{y}_{\text{base},i}^p(k)$ da saída da ESNI, conforme (95).
 - 7: **Fim Para**
 - 8: Construir o vetor concatenado de resposta livre não linear MIMO $\mathbf{y}_{\text{base}}^p(k)$, como definido em (92).
 - 9: **Para** $i = 1 : m$ **Faça**
 - 10: **Para** $j = 1 : n$ **Faça**
 - 11: **Para** $s = 1 : N_{u,j}$ **Faça**
 - 12: Linearizar o modelo ESNI por meio da simulação da resposta ao degrau em volta de $\mathbf{u}_{\text{base}}^p(k)$, obtendo a coluna s da matriz de dinâmica SISO $\mathbf{G}_{ij}^p(k)$ utilizando (104).
 - 13: **Fim Para**
 - 14: **Fim Para**
 - 15: **Fim Para**
 - 16: Construir a matriz de dinâmica MIMO $\mathbf{G}_{\text{PNMPCI}}^p(k)$ conforme a estrutura em (93).
 - 17: Resolver o problema QP associado ao PNMPC para obtenção dos incrementos de controle futuro $\Delta\mathbf{u}^p(k)$.
 - 18: **Fim Enquanto**
 - 19: **Para** $j = 1 : n$ **Faça**
 - 20: Aplicar a soma de todos os primeiros elementos do vetor de incrementos de controle no processo, i.e. $u_j(k) = u_j(k-1) + \sum_{i=1}^p \Delta u_j^i(k)$.
 - 21: **Fim Para**
 - 22: Aguardar o término do período de amostragem.
 - 23: Retornar ao passo 1 com $k = k + 1$.
-

4.5.2 Exemplo 7: Controle de um processo de neutralização de pH

O funcionamento do algoritmo PNMPCI-ESNI é demonstrado utilizando o problema de controle do processo de neutralização de pH. Esse exemplo é uma continuação do Exemplo 4, apresentado na Subseção 4.3.2, compartilhando os mesmos procedimentos para simulação, criação e sintonia dos controladores. O modelo ESNI utilizado foi composto por um reservatório com 500 unidades. Foram utilizados os mesmos horizontes de predição ($N = 10$, com $N_1 = 1$ e $N_2 = 10$) e controle $N_u = 3$, e a avaliação inicial utilizou $\lambda/\delta = 0,1$, com o impacto da escolha de λ/δ sendo discutido no decorrer da seção. A avaliação do desempenho do controlador é realizada utilizando o mesmo ensaio de teste usado no Exemplo 4, o qual considera várias trocas de referência entre os valores de pH 6 e 9. Novamente, a IAE é usada como métrica para avaliação do desempenho dos controladores.

A predição iterativa do algoritmo proposto é ajustada escolhendo ϵ e i_{\max} . Ambos os parâmetros são relacionados, já que valores pequenos de ϵ resultam em um número de iterações elevado. Uma boa prática para a sintonia desses parâmetros consiste em selecionar um valor pequeno para ϵ , e um valor de i_{\max} que mantenha o custo computacional dentro do limite desejado, ou seja, escolhendo um limite superior para o número de iterações. Para demonstrar o funcionamento do procedimento iterativo, são apresentados os resultados em malha fechada considerando $\epsilon = 0,1$ e $i_{\max} \in [1; 10]$, sendo que o resultado para as primeiras cinco configurações é mostrado na Figura 35.

Figura 35 – Desempenho em malha fechada do controlador PNMPCI-ESNI considerando cinco configurações para o número máximo de iterações (i_{\max}).



Fonte: dados da pesquisa.

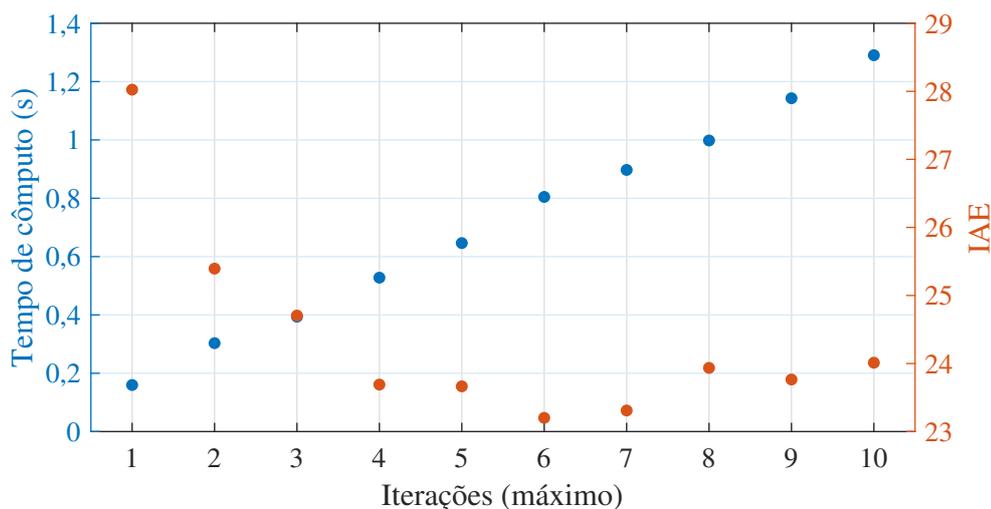
À medida que o número máximo de iterações é incrementado, é verificada uma melhora

no desempenho em malha fechada. Essa melhora é mais significativa para valores baixos de i_{\max} , especialmente quando i_{\max} é aumentado de 1 para 2. Para valores de $i_{\max} \geq 3$ a melhoria de desempenho é menos significativa, o que pode ser verificado na Figura 35, na qual os desempenhos para $i_{\max} = 3$ e $i_{\max} = 5$ são similares.

Como apresentado no terceiro gráfico da Figura 35, um número de iterações alto é observado nos instantes em que variações de referência ocorrem, instantes em que a saída do processo está longe da referência. Nessas situações, a linearização do modelo ESNI para obtenção de $\mathbf{G}_{\text{PNMPCI}}$ resulta em uma representação imperfeita da dinâmica não linear do processo. Devido ao modelo de previsão apresentar imperfeições, os incrementos de controle $\Delta \mathbf{u}$, obtidos pelo controlador, são distantes dos valores ótimos. Utilizando o procedimento iterativo, o controlador é capaz de atualizar a porção base da previsão com um novo $\Delta \mathbf{u}$ candidato, reduzindo o impacto da linearização na resposta em malha fechada do sistema. Isso ocorre pois a importância da porção ótima da previsão é reduzida a cada iteração, fazendo com que essa porção, a qual utiliza a aproximação linear, tenha menor impacto em relação à porção base, que utiliza o modelo não linear completo.

Uma representação gráfica do valor final da métrica IAE e do custo computacional máximo, para cada configuração do controlador PNMPCI-ESNI, é apresentada na Figura 36. O valor final da IAE é menor quando i_{\max} é maior, porém, a taxa de melhora no desempenho diminui à medida que i_{\max} aumenta. Por sua vez, o custo computacional, avaliado por meio do tempo máximo de cômputo, cresce linearmente. Para esse exemplo, o valor $i_{\max} = 6$ apresenta o melhor compromisso entre o desempenho do controlador e o custo computacional.

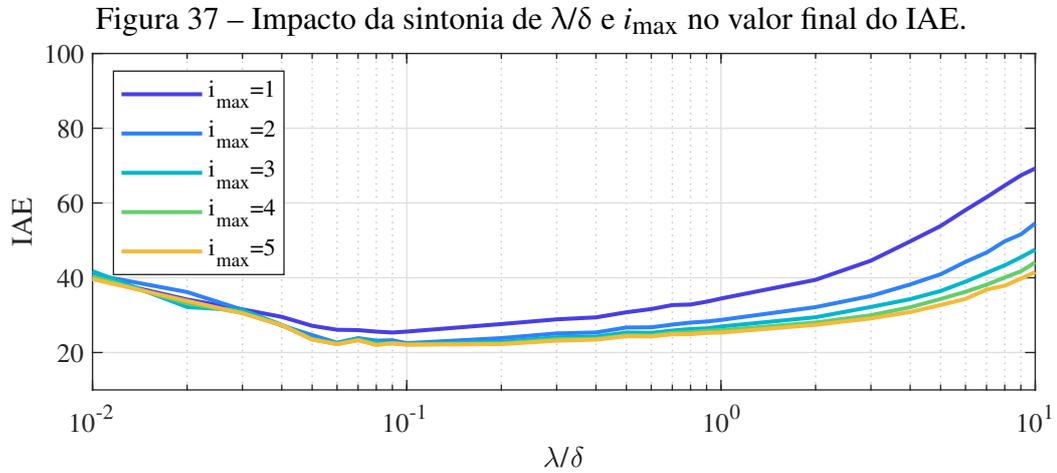
Figura 36 – Impacto do número máximo de iterações no valor final da métrica IAE e no tempo máximo de cômputo.



Fonte: dados da pesquisa.

Esses resultados correspondem ao controlador PNMPCI-ESNI com sintonia $\lambda/\delta = 0, 1$, porém, a sintonia do procedimento iterativo também afeta a sintonia geral do controlador. Para avaliar como a sintonia de λ/δ e de i_{\max} se relacionam, foram realizadas simulações avaliando

o desempenho do controlador para $\lambda/\delta \in [0,01; 10]$ e $i_{\max} \in [1; 5]$. A influência da sintonia de ambos os parâmetros é apresentada por meio do valor final da métrica IAE na Figura 37.



Fonte: dados da pesquisa.

A análise da Figura 37 evidencia a melhoria alcançada com a utilização do procedimento iterativo, a qual é alcançada independentemente da sintonia escolhida. A maior diferença é observada entre $i_{\max} = 1$ e $i_{\max} = 2$, sendo que a sintonia de λ/δ que apresenta o melhor desempenho se manteve similar para todos valores de i_{\max} , próximo de $\lambda/\delta = 0,09$. A avaliação final do controlador e sua comparação com o controlador NMPC é apresentada na Figura 38, utilizando a sintonia $\lambda/\delta = 0,09$ e valores $i_{\max} = 1$, $i_{\max} = 3$ e $i_{\max} = 6$. As métricas de desempenho correspondentes ao ensaio apresentado na Figura 38 são apresentadas na Tabela 7.

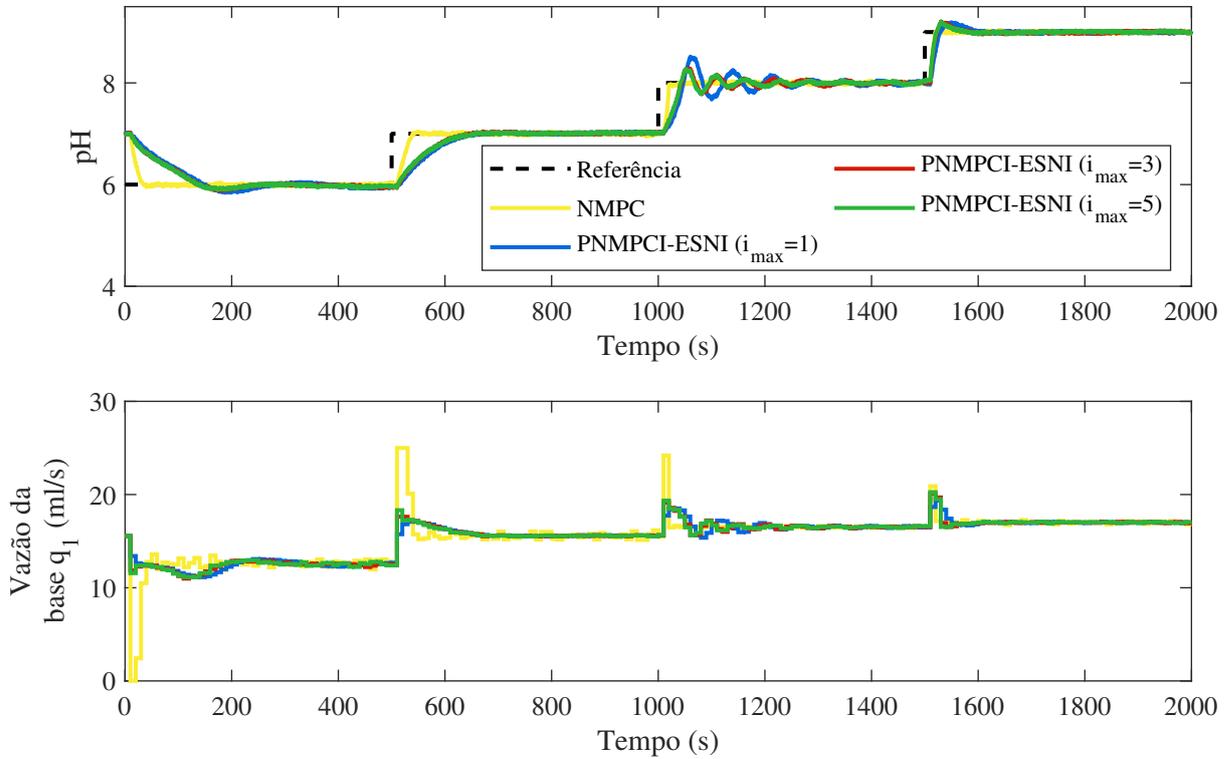
Tabela 7 – Comparação das métricas de desempenho para os controladores PNMPCI-ESNI e NMPC.

Controlador	IAE	ITAE	Tempo de cômputo médio (s)	Tempo de cômputo máximo (s)
PNMPCI-ESNI ($i_{\max} = 1$)	28,02	236,25	0,13	0,16
PNMPCI-ESNI ($i_{\max} = 3$)	24,09	192,27	0,22	0,44
PNMPCI-ESNI ($i_{\max} = 6$)	23,20	188,93	0,63	0,80
NMPC	12,31	95,86	22,71	29,10

Fonte: dados da pesquisa.

O resultado alcançado utilizando $i_{\max} = 6$ é mais próximo do desempenho do NMPC, o que pode ser verificado pela inspeção na Figura 38, e também por meio da IAE, apresentada na Tabela 7. É importante notar que o controlador proposto apresenta comportamentos diferentes, dependendo do ponto de operação do processo, com comportamentos lentos para transições até a referência com valor de pH 6, e comportamentos mais rápidos para a referência com valor de pH 9. Devido ao processo apresentar uma variação grande no seu ganho estático, o que pode ser verificado avaliando a derivada da curva característica estática, na Figura 18, o uso de uma sintonia fixa pelo controlador proposto faz com que existam diferenças no comportamento do controlador em condições de operação distintas. A diferença percentual entre o desempenho

Figura 38 – Resultados do controle de um reator de neutralização de pH com algoritmo PNMPCI-ESNI, e NMPC. O gráfico superior apresenta o sinal de referência (linha tracejada) e os valores de pH (linhas sólidas), enquanto no gráfico inferior são mostrados os sinais de controle.



Fonte: dados da pesquisa.

do PNMPCI-ESNI ($i_{\max} = 6$) e o NMPC é de 47%, sendo expressivamente menor do que a diferença entre o controlador PNMPC-ESNI e o NMPC, que é de 56%.

4.6 COMENTÁRIOS FINAIS

Neste capítulo foram apresentadas abordagens de controle preditivo baseado em modelo que utilizam modelos de predição baseados na rede ESNI, proposta no Capítulo 3. Todas as abordagens exploraram formulações NMPC práticas, nas quais o modelo ESNI é utilizado de diferentes maneiras, de forma a permitir que o problema de otimização associado ao controlador seja resolvido com algoritmos de programação quadrática, resultando em algoritmos computacionalmente eficientes. Foi proposta uma formulação PNMPC-ESNI para modelos ESNI identificados, complementada por uma formulação adaptativa, na qual o modelo ESNI é identificado on-line, utilizando o método proposto no Capítulo 3. Por fim, uma proposta para aprimorar o algoritmo PNMPC-ESNI foi apresentada, a qual utiliza um procedimento iterativo de forma a reduzir a importância da porção linearizada no modelo de predição, aprimorando o desempenho final do controlador.

Quanto ao algoritmo PNMPC-ESNI, foi observado que a formulação foi capaz de apresentar resultados em malha fechada superiores aos apresentados pelas abordagens MPC baseada em modelo linear e PNMPC com modelo Hammerstein-Wiener, o que era esperado, devido ao modelo ESNI ter apresentado melhores resultados que o modelo linear e o Hammerstein-Wiener na etapa de identificação. Para o caso do processo de neutralização de pH, quando comparado com um controlador NMPC, foi observado que a abordagem PNMPC-ESNI foi capaz de apresentar tempos de cômputo com valores duas ordens de grandeza inferiores, porém, devido ao modelo de predição ser composto de uma porção linearizada e apresentar erros de modelagem, o resultado em malha fechada do algoritmo PNMPC-ESNI proposto foi 56% inferior ao apresentado pelo NMPC.

A abordagem adaptativa do algoritmo proposto também apresentou resultados superiores aos dos controladores usados para comparação, com exceção do controlador NMPC, em um cenário cujo processo opera em sua condição nominal. Em um cenário com variação paramétrica, o controlador proposto mostrou sua principal vantagem, com o APNMPC-ESNI se adaptando à variação paramétrica do processo e, devido a isso, não sendo verificada uma queda no desempenho em malha fechada, como foi verificado nas abordagens não adaptativas, caso do controlador NMPC. Foram verificados tempos de cômputo máximo uma ordem de grandeza menor que os verificados para o controlador NMPC, e próximos ao verificado para o controlador aGPC, indicando que a abordagem proposta é uma alternativa para o controle adaptativo de processos reais, com desempenho próximo a um NMPC, dependendo do cenário, e custo computacional com a mesma ordem de grandeza de um aGPC.

Com o intuito de reduzir o impacto da utilização de uma representação linearizada no modelo de predição, foi proposta uma extensão do algoritmo PNMPC-ESNI, considerando um mecanismo iterativo, associado a uma interpretação alternativa dos termos do modelo de predição. O algoritmo PNMPCI-ESNI, que utiliza o mecanismo iterativo, apresentou uma melhora significativa no desempenho em malha fechada em relação à variante PNMPC-ESNI, com resultados até 17% melhores para o cenário testado. Foi verificado que o número máximo de

iterações do algoritmo tem um impacto na melhora de desempenho alcançada, sendo que o aprimoramento dos resultados cessa à medida que o número máximo de iterações atinge certo patamar, no caso do exemplo apresentado $i_{\max} = 6$. Foi demonstrado, adicionalmente, que o procedimento iterativo faz parte da sintonia do controlador, havendo uma influência entre i_{\max} e as ponderações da função custo, λ e δ . Por fim, é importante destacar que a abordagem iterativa é geral, podendo ser associada às demais formulações de NMPC prático da literatura, independente do tipo de modelo usado para obtenção do modelo de predição, apresentando um potencial para aplicação e melhoria no desempenho de variadas abordagens NMPC práticas.

5 CONSIDERAÇÕES FINAIS

Neste capítulo são apresentadas as considerações finais deste trabalho. Na Seção 5.1 são apresentadas as conclusões desta tese de doutorado, enquanto na Seção 5.2 são detalhadas as sugestões para trabalhos futuros, e, por fim, na Seção 5.3 são apresentados os artigos publicados durante o desenvolvimento desta tese.

5.1 CONCLUSÕES

Esta tese explorou a utilização da rede de estado de eco para identificação de sistemas dinâmicos não lineares e predição em controladores preditivos baseados em modelo. Por meio de um estudo da literatura, foi identificado que a rede de estado de eco apresenta características interessantes para aplicação em abordagens de identificação e controle, já que ela é capaz de modelar sistemas não lineares com boa precisão com um custo computacional, associado ao seu treinamento, pequeno. Quanto à utilização em identificação e controle, foi constatado que modelos baseados em ESN já foram explorados para identificação de sistemas em abordagens MPC, porém, foram encontradas algumas lacunas. Quanto à utilização de modelos ESN em tarefas de identificação de sistemas, foi identificada uma oportunidade para o desenvolvimento de uma abordagem MRI, a qual apresenta potencial para produzir modelos mais adequados à aplicação em MPC. Por sua vez, no domínio da aplicação de modelos ESN em abordagens MPC, foram identificadas oportunidades relacionadas ao desenvolvimento de abordagens NMPC práticas e também abordagens adaptativas.

Foi proposta uma formulação MRI para identificação de sistemas não lineares, a qual se baseou na ESN cuja arquitetura contém um termo integrador. A utilização da abordagem MRI tem como objetivo fazer com que o modelo resultante apresente características desejáveis para aplicação em MPC, fazendo com que o modelo apresente bom desempenho para a predição de múltiplos instantes à frente. Em conjunto com a abordagem MRI, foram propostos algoritmos para o treinamento do modelo ESNI, seja utilizando dados coletados do processo seja de forma on-line. Para o treinamento do modelo com dados coletados do processo, foi proposto o uso de um método de regressão com regularização, enquanto para o treinamento on-line foi proposto um algoritmo RLS com fator de esquecimento adaptativo.

A avaliação da abordagem MRI-ESNI, em cenários com dados coletados do processo e em cenários on-line, evidenciou o principal benefício do uso da abordagem MRI, que consiste na obtenção de modelos com melhores resultados para a predição de múltiplos instantes à frente. Isso foi verificado em ambos os estudos de caso apresentados, a identificação de um processo de neutralização de pH simulado, utilizando dados obtidos do processo, e a identificação on-line de um tanque de perfil cônico simulado. Foi observado que a melhoria observada para a predição de múltiplos instantes à frente ocorreu em detrimento do desempenho do modelo para a predição de um instante à frente, com os modelos que utilizaram a abordagem MRI apresentando melhores desempenhos para a predição de múltiplos instantes à frente quando comparados com

um modelo com a mesma estrutura e treinado utilizando um método convencional, porém, sendo superados quando a tarefa é de predição de um instante à frente.

O modelo ESNI foi utilizado para a obtenção de um preditor em uma abordagem NMPC prática. A formulação da abordagem PNMPC proposta foi detalhada, sendo apresentado o equacionamento para obtenção dos vetores de resposta livre e forçada, construção do problema de otimização e obtenção do sinal de controle. Foi, também, discutida a importância da utilização do termo integrador na arquitetura ESNI, o qual é responsável por conferir ao controlador resultante ação integral. Esse controlador, denominado PNMPC-ESNI, foi avaliado por meio de dois estudos de caso, o controle de um processo de neutralização de pH simulado e o controle de uma bancada de ensaio de compressores herméticos real. De forma geral, o controlador proposto apresentou resultados em malha fechada superiores aos apresentados por controladores com modelos de predição com precisão menor, casos do GPC e PNMPC-HW, e resultados inferiores aos de controladores mais complexos e com modelos de predição com maior precisão, caso do NMPC que utilizou o modelo fenomenológico para predição, o que era o esperado. Quanto ao custo computacional, novamente, o controlador proposto apresentou os resultados esperados, requerendo menores tempos médios e máximos de cômputo que o NMPC, porém, tempos superiores aos requeridos pelo GPC e PNMPC-HW.

Dentre as oportunidades de contribuição identificadas na revisão da literatura, a utilização da rede ESN em abordagens MPC adaptativas foi uma que se destacou, pois apresentava potencial para aproveitar a principal característica que difere a rede ESN das demais RNAs recorrentes, que é seu treinamento computacionalmente eficiente. Neste contexto, foi proposto um algoritmo PNMPC adaptativo utilizando o modelo ESNI, o qual é estimado on-line utilizando o método MRI proposto, por meio de um algoritmo RLS com fator de esquecimento adaptativo. Esse controlador proposto foi avaliado utilizando dois estudos de caso, o controle de um tanque de perfil cônico e o controle de um motor síncrono de ímãs permanentes, ambos simulados. Os resultados indicaram que o uso da abordagem MRI para treinamento on-line do modelo do controlador APNMPC-ESNI resultou em um melhor desempenho em malha fechada, já que o controlador proposto apresentou melhores métricas de desempenho que controladores semelhantes que utilizaram modelos ESN treinados por abordagens que não eram MRI. A comparação com um controlador NMPC cujo modelo era fenomenológico e não adaptativo evidenciou a principal vantagem de utilizar a abordagem adaptativa, que é a possibilidade do controlador se adaptar às variações paramétricas do processo. O controlador proposto apresentou bons resultados, porém, desempenho inferior ao NMPC em um cenário em que o processo opera com os parâmetros nominais. Por sua vez, em um cenário com variações paramétricas, o controlador proposto foi capaz de se adaptar, apresentando desempenho similar ao alcançado no cenário anterior, o que não foi verificado para controladores não adaptativos, inclusive o NMPC, os quais apresentaram piora no seu desempenho, fazendo com que o controlador proposto apresentasse melhor desempenho inclusive que o NMPC.

Uma das características que definem os controladores PNMPC é a divisão da predição

em duas parcelas, uma utilizando o modelo não linear completo e outra utilizando uma aproximação linear local da dinâmica do modelo. Essa aproximação é imperfeita, pois essa divisão se baseia no princípio da superposição, o qual não é verificado para sistemas não lineares, resultando em perdas no desempenho do controlador. Para tentar reduzir a perda de desempenho decorrente dessa aproximação, foi proposta uma alteração no funcionamento do controlador PNMPC-ESNI, utilizando um procedimento iterativo que atualiza o modelo de predição com as ações de controle obtidas em cada iteração, de forma a reduzir a importância da porção da predição que utiliza a aproximação linear local do modelo ESNI. Esse controlador PNMPC-ESNI foi avaliado, e os resultados indicaram que o procedimento iterativo resulta em melhorias no resultado em malha fechada, com melhorias de até 25% em comparação com o controlador PNMPC-ESNI. Foram discutidos o impacto da sintonia do procedimento iterativo e sua relação com a sintonia das ponderações da função custo e apresentados resultados para o controle do processo de neutralização de pH simulado. O uso do procedimento iterativo teve impacto no custo computacional do controlador, porém o custo ainda é menor que o requerido por um controlador NMPC.

De forma geral, o modelo ESNI e as abordagens de NMPC prático propostas nesta tese evidenciaram que existe um potencial para o uso de modelos baseados na ESN para identificação e controle em aplicações reais. Foi apresentada uma implementação do controlador PNMPC-ESNI para o controle de pressões em uma bancada de ensaio de compressores herméticos real, indicando que a utilização deste tipo de abordagem é viável em processos da indústria, com os benefícios verificados nos estudos simulados sendo verificados também na aplicação real. Os tempos de cômputo requeridos pelo controlador PNMPC-ESNI foram compatíveis com o período de amostragem do processo, o que viabilizaria inclusive a implementação da abordagem adaptativa APNMPC-ESNI.

De acordo com as conclusões apresentadas, é possível afirmar que os objetivos da tese foram alcançados. Foram pesquisadas as técnicas de identificação de sistemas dinâmicos baseadas na ESN e as abordagens de MPC que usam tal modelo para predição. Foi proposta e avaliada por meio de simulações uma abordagem MRI para o modelo ESNI proposto, o qual foi posteriormente utilizado para predição em um algoritmo NMPC prático. Uma abordagem PNMPC adaptativa e uma estratégia para melhoria do desempenho de controladores PNMPC foram também propostas, com seu funcionamento sendo avaliado por meio de simulações.

5.2 SUGESTÕES PARA TRABALHOS FUTUROS

O desenvolvimento desta tese abrangeu vários tópicos, explorando desde a identificação de sistemas com modelos baseados em ESN, os quais foram utilizados em abordagens de NMPC prático. Foram observadas várias oportunidades para continuidade do estudo desenvolvido nesta tese, sendo que, nesta seção, são apresentadas algumas sugestões que apresentam potencial para serem exploradas em trabalhos futuros. As principais sugestões para trabalhos futuros são:

- explorar abordagens MRI para outras arquiteturas de redes recorrentes, por exemplo as redes RVFL (do inglês, *Random Vector Functional Link*) e ELM (do inglês, *Extreme Learning Machine*);
- avaliar outras abordagens para construção do reservatório, de forma a melhorar a precisão do modelo treinado ou reduzir a complexidade do modelo para predição;
- estudar alternativas para *pruning* do modelos ESNI após o treinamento, de forma a reduzir a complexidade requerida para predição com o modelo;
- avaliar abordagens para garantia de robustez do controlador PNMPC-ESNI;
- desenvolver abordagens PNMPCI para modelos baseados em outras arquiteturas de RNA, como por exemplo redes RVFL, ELM e LSTM;
- avaliar a possibilidade de utilização da solução do problema de otimização, obtida a cada período de amostragem pelo PNMPCI-ESNI, para inicialização do procedimento iterativo, quando $N_u > 1$, de forma a reduzir a quantidade de iterações e, por consequência, o custo computacional total do controlador;
- implementar o algoritmo PNMPCI-ESNI para o controle de um processo real;
- avaliar a utilização do modelo ESNI para identificação e controle preditivo de processos não lineares que unem variáveis contínuas e discretas.

5.3 TRABALHOS PUBLICADOS

No contexto da realização das atividades durante o doutorado e dos desenvolvimentos apresentados nesta tese, foram publicados cinco artigos completos diretamente relacionados com a tese. Três artigos foram publicados em periódicos e dois artigos em congressos, um em um congresso internacional e outro em um congresso nacional. Estes trabalhos são:

- SCHWEDERSKY, B. B.; FLESCHE, R. C. C.; DANGUI, H. A. C. Echo State Networks for Online, Multi-step MPC Relevant Identification. **Engineering Applications of Artificial Intelligence**, Elsevier BV, v. 108, p. 104596, fev. 2022. DOI: 10.1016/j.engappai.2021.104596
- SCHWEDERSKY, B. B.; FLESCHE, R. C. C.; DANGUI, H. A. C. Nonlinear MIMO System Identification with Echo State Networks, **Journal of Control, Automation and Electrical Systems**, Springer Science and Business Media LLC, jan. 2022. DOI: 10.1007/s40313-021-00874-y

- SCHWEDERSKY, B. B.; FLESCHE, R. C. C.; ROVEA, S. B. Adaptive Practical Nonlinear Model Predictive Control for Echo State Network Models. **IEEE Transactions on Neural Networks and Learning Systems**. Institute of Electrical and Electronics Engineers (IEEE), *No prelo*. DOI: 10.1109/TNNLS.2021.3109821.
- SCHWEDERSKY, B. B.; FLESCHE, R. C. C.; DANGUI, H. A. C.; IERVOLINO, L. A. Practical Nonlinear Model Predictive Control Using an Echo State Network Model. In: 2018 INTERNATIONAL JOINT CONFERENCE ON NEURAL NETWORKS (IJCNN). Rio de Janeiro, Brasil. **Proceedings [...]**. New York: IEEE, 2018. DOI: 10.1109/IJCNN.2018.8489446.
- SCHWEDERSKY, B. B.; FLESCHE, R. C. C.; DANGUI, H. A. C. Identificação de Sistemas Dinâmicos Não Lineares Multivariáveis com Redes de Estado de Eco. In: XXIII CONGRESSO BRASILEIRO DE AUTOMÁTICA. Porto Alegre, Brasil. **Anais [...]**. Campinas: SBA, 2020. p. 1-8. DOI: 10.48011/asba.v2i1.1713

Foram publicados quatro artigos completos relacionados parcialmente ao tema da tese. Estes artigos exploram abordagens PNMPC com modelos de predição baseados no modelo Hammerstein-Wiener e na rede LSTM, e também exploram a sintonia de controladores MPC. Destes trabalhos, três foram apresentados em congressos internacionais e um em congresso nacional. Tais artigos são:

- SCHWEDERSKY, B. B.; FLESCHE, R. C. C.; DANGUI, H. A. C. Practical Nonlinear Model Predictive Control Algorithm for Long Short-Term Memory Networks. In: 12th IFAC SYMPOSIUM ON DYNAMIC AND CONTROL OF PROCESS SYSTEMS, INCLUDING BIOSYSTEMS DYCOPS 2019. Florianópolis, Brasil. **Proceedings [...]** New York: IFAC, 2019. p. 1-6 DOI: 10.1016/j.ifacol.2019.06.106
- SCHWEDERSKY, B. B.; FLESCHE, R. C. C.; DANGUI, H. A. C. Practical Nonlinear Model Predictive Control with Hammerstein Model Applied to a Test Rig for Refrigeration Compressors. In: 24th INT. COMPRESSOR ENGINEERING CONF. AT PURDUE. West Lafayette, Indiana. **Proceedings [...]**. West Lafayette: Purdue University Press, 2018. p. 1-8.
- DANGUI, H. A. C.; FLESCHE, R. C. C.; SCHWEDERSKY, B. B. Practical Guidelines for Tuning Model-Based Predictive Controllers for Refrigerant Compressor Test Rigs. In: 24th INT. COMPRESSOR ENGINEERING CONF. AT PURDUE. West Lafayette, Indiana. **Proceedings [...]**. West Lafayette: Purdue University Press, 2018. p. 1-8.
- SCHWEDERSKY, B. B.; FLESCHE, R. C. C.; DANGUI, H. A. C. Algoritmo de Controle Preditivo com Predição Iterativa Não Linear e Linearização para Modelos Hammerstein-Wiener. In: XXIII CONGRESSO BRASILEIRO DE AUTOMÁTICA. Porto Alegre, Brasil. **Anais [...]**. Campinas: SBA, 2020. p. 1-8. DOI: 10.48011/asba.v2i1.1715

Adicionalmente, durante o período de desenvolvimento desta tese, foi publicado um artigo em periódico cujo tema é de uma área afim à de estudo desta tese, porém sem relação direta com a mesma. Este artigo explora a segmentação de imagens com RNAs. Tal artigo é:

- SCHWEDERSKY, B. B.; DE OLIVEIRA, B. C. F.; ALBERTAZZI, A.; FLESCHE, R. C. C. Impact damage characterization in CFRP samples with self-organizing maps applied to lock-in thermography and square-pulse shearography images. **Expert Systems with Applications**, Elsevier BV, v. 192, p. 116297, 2022. DOI: 10.1016/j.eswa.2021.116297

REFERÊNCIAS

ADETOLA, Veronica; DEHAAN, Darryl; GUAY, Martin. Adaptive model predictive control for constrained nonlinear systems. **Systems & Control Letters**, v. 58, p. 320–326, mai. 2009.

ADETOLA, Veronica; GUAY, Martin. Robust adaptive MPC for constrained uncertain nonlinear systems. **International Journal of Adaptive Control and Signal Processing**, v. 25, p. 155–167, fev. 2011.

AKPAN, Vincent A.; HASSAPIS, George D. Nonlinear model identification and adaptive model predictive control using neural networks. **ISA Transactions**, Elsevier BV, v. 50, n. 2, p. 177–194, abr. 2011.

AL SEYAB, Rihab; CAO, Yi. Differential recurrent neural network based predictive control. **Computers & Chemical Engineering**, Elsevier, v. 32, n. 7, p. 1533–1545, 2008a.

AL SEYAB, Rihab; CAO, Yi. Nonlinear system identification for predictive control using continuous time recurrent neural networks and automatic differentiation. **Journal of Process Control**, Elsevier, v. 18, n. 6, p. 568–581, 2008b.

ALEXANDRIDIS, Alex; SARIMVEIS, Haralambos. Nonlinear adaptive model predictive control based on self-correcting neural network models. **AIChE Journal**, Wiley Online Library, v. 51, n. 9, p. 2495–2506, 2005.

ANTONELO, Eric A. **Reservoir Computind Architectures for Modeling Robot Navigation Systems**. 2012. Tese (Doutorado) – Ingenieurswetenschappen: Computerwetenschappen, University Gent, Gent.

ANTONELO, Eric A.; CAMPONOVARA, Eduardo; FOSS, Bjarne. Echo State Networks for data-driven downhole pressure estimation in gas-lift oil wells. **Neural Networks**, v. 85, p. 106–117, 2017.

ARMENIO, Luca B.; TERZI, Enrico; FARINA, Marcello; SCATTOLINI, Riccardo. Model Predictive Control Design for Dynamical Systems Learned by Echo State Networks. **IEEE Control System Letters**, Institute of Electrical e Electronics Engineers (IEEE), v. 3, n. 4, p. 1044–1049, out. 2019.

ASHRAE. **ANSI/ASHRAE 23-2005: Methods of testing for rating positive displacement refrigerant compressors and condensing units**. Atlanta, EUA: AHRAE, 2005.

ÅSTRÖM, Karl J; WITTENMARK, Björn. **Adaptive control**. North Chelmsford, Massachusetts: Courier Corporation, 2013.

BALTAR, Raphael M. S. M.; LIMA, Rafael B. C.; BARROS, Péricles R. EN-PH: a regularization-based model predictive control relevant identification method. **IFAC-PapersOnLine**, Elsevier BV, v. 48, n. 28, p. 1166–1171, 2015.

BERTSEKAS, Dimitri P. Neuro-Dynamic Programming. In: **Encyclopedia of Optimization**. Edição: Christodoulos A. Floudas e Panos M. Pardalos. Boston, MA: Springer US, 2009. P. 2555–2560.

BIEGLER, Lorenz T. Efficient Solution of Dynamic Optimization and NMPC Problems. In: ALLGÖWER, Frank; ZHENG, Alex (Ed.). **Nonlinear Model Predictive Control**. Basel: Birkhäuser Basel, 2000. P. 219–243.

BOBÁL, Vladimír; BÖHM, Josef; FESSL, Jaromír; MACHÁČEK, Jiří. **Digital self-tuning controllers: algorithms, implementation and applications**. London: Springer Science & Business Media, 2006.

BUJARBARUAH, Monimoy; ZHANG, Xiaojing; TANASKOVIC, Marko; BORRELLI, Francesco. Adaptive MPC under Time Varying Uncertainty: Robust and Stochastic. arXiv, p. 1–23, set. 2019.

CAMACHO, Eduardo F.; BORDONS, Carlos. **Model Predictive Control**. Londres: Springer Verlag, 2007. P. 405.

CHEN, Steven W.; WANG, Tianyu; ATANASOV, Nikolay; KUMAR, Vijay; MORARI, Manfred. Large scale model predictive control with neural networks and primal active sets. **Automatica**, Elsevier BV, v. 135, p. 109947, jan. 2022.

CHEN, Yanhua; HE, Zhaoshuang; SHANG, Zhihao; LI, Caihong; LI, Lian; XU, Mingliang. A novel combined model based on echo state network for multi-step ahead wind speed forecasting: A case study of NREL. **Energy Conversion and Management**, Elsevier BV, v. 179, p. 13–29, jan. 2019.

CHI, Qinghua; FEI, Zhengshun; ZHAO, Zhao; ZHAO, Li; LIANG, Jun. A model predictive control approach with relevant identification in dynamic PLS framework. **Control Engineering Practice**, Elsevier BV, v. 22, p. 181–193, jan. 2014.

CLARKE, David W.; MOHTADI, Coorous; TUFFS, P. Simon. Generalized predictive control-Part I. The basic algorithm. **Automatica**, v. 23, n. 2, p. 137–148, 1987. ISSN 0005-1098.

COLIN, Guillaume; CHAMAILLARD, Yann; BLOCH, Grard; CORDE, Gilles. Neural control of fast nonlinear systems: Application to a turbocharged SI engine with VCT. **IEEE Transactions on Neural Networks**, IEEE, v. 18, n. 4, p. 1101–1114, 2007.

CUTLER, Charles R.; RAMAKER, Brian L. Dynamic matrix control: A computer control algorithm. In: 1980. JOINT AUTOMATIC CONTROL CONFERENCE. San Francisco, Ca. **Proceedings [...]**. New York: IEEE, 1980. P. 72.

DA CRUZ, Uriel I. O.; SANTOS, Tito L. M.; ACHY, Acbal R. A.; PAIM, Cristiane C. Controle Preditivo Baseado em um Modelo Neural Aplicado a um Sistema Não-Linear de Tanques Acoplados. In: SIMPÓSIO Brasileiro de Automação Inteligente. Fortaleza, CE. **Anais [...]**. Campinas: IFAC, 2013. P. 1–6.

DAI, Jing; VENAYAGAMOORTHY, Ganesh K.; HARLEY, Ronald G. An Introduction to the Echo State Network and its Applications in Power System. In: 2009 15th INT. CONF. INTELLIGENT SYSTEM APPLICATIONS TO POWER SYSTEMS. Curitiba, PR. **Proceedings [...]**. New York: IEEE, nov. 2009. P. 1–7.

DANGUI, Hiago A. S.; FLESCHE, Rodolfo C. C.; SCHWEDERSKY, Bernardo B. Practical Guidelines for Tuning Model-Based Predictive Controllers for Refrigerant Compressor Test Rigs. In: 24TH INTERNATIONAL COMPRESSOR ENGINEERING CONFERENCE AT PURDUE. West Lafayette, Indiana: **Proceedings [...]**, West Lafayette: Purdue University Press, 2018.

DE KEYSER, Robin. Model based predictive control for linear systems. In: CONTROL Systems, Robotics and Automation. Paris: Eolss Publishers, 2009. v. XI. P. 24–58.

DE OLIVEIRA, Simone L.; NEVISTIĆ, Vesna; MORARI, Manfred. Control of Nonlinear Systems Subject to Input Constraints. In: 3RD IFAC SYMPOSIUM ON NONLINEAR CONTROL SYSTEMS DESIGN. Tahoe City, Ca. **Proceedings [...]**. New York: IFAC, 1995. P. 13–18.

DECLERCQ, Filip; DE KEYSER, Robain. Comparative study of neural predictors in model based predictive control. In: INTERNATIONAL WORKSHOP ON NEURAL NETWORKS

FOR IDENTIFICATION, CONTROL, ROBOTICS AND SIGNAL/IMAGE PROCESSING. Veneza, It. **Proceedings [...]**. New York: IEEE, 1996. P. 20–28.

DEMMERS, Theo G.M.; CAO, Yi; GAUSS, Sophie; LOWE, John C.; PARSONS, David J.; WATHES, Christopher M. Neural predictive control of broiler chicken and pig growth. **Biosystems Engineering**, v. 173, p. 134–142, 2018.

DRAEGER, Andreas; ENGELL, Sebastian; RANKE, Horst. Model predictive control using neural networks. **IEEE Control Systems Magazine**, IEEE, v. 15, n. 5, p. 61–66, 1995.

DRGOŇA, Ján; PICARD, Damien; KVASNICA, Michal; HELSEN, Lieve. Approximate model predictive building control via machine learning. **Applied Energy**, Elsevier BV, v. 218, p. 199–216, mai. 2018.

ELMAN, Jeffrey L. Finding structure in time. **Cognitive Science**, Wiley-Blackwell, v. 14, n. 2, p. 179–211, jun. 1990.

FORBES, Michael G.; PATWARDHAN, Rohit S.; HAMADAH, Hamza; GOPALUNI, R. Bhushan. Model Predictive Control in Industry: Challenges and Opportunities. In: 9TH IFAC SYMPOSIUM ON ADVANCED CONTROL OF CHEMICAL PROCESSES (ADCHEM). Whistler, BC. **Proceedings [...]**. New York: IFAC, 2015. P. 531–538.

FOSS, Bjarne A; SCHEI, Tor S. Putting nonlinear model predictive control into use. In: FINDEISEN, Rolf; ALLGÖWER, Frank; BIEGLER, Lorenz T. (Ed.). **Assessment and Future Directions of Nonlinear Model Predictive Control**. Freudenstadt-Lauterbad, Alemanha: Springer, 2007. P. 407–417.

FUKUSHIMA, Hiroaki; KIM, Tae-Hyoung; SUGIE, Toshiharu. Adaptive model predictive control for a class of constrained linear systems based on the comparison model. **Automatica**, v. 43, n. 2, p. 301–308, 2007.

GARCIA, Claudio. Quadratic dynamic matrix control of nonlinear processes. An application to a batch reaction process. In: 1984. AICHE ANNUAL MEETING. San Francisco, CA. **Proceedings [...]**. New York: AIChE, 1984.

GATTU, Gangadhar; ZAFIRIOU, Evangelos. Nonlinear quadratic dynamic matrix control with state estimation. **Industrial & Engineering Chemistry Research**, v. 31, n. 4, p. 1096–1104, 1992.

GATTU, Gangadhar; ZAFIRIOU, Evangelos. Observer based nonlinear quadratic dynamic matrix control for state space and input/output models. **The canadian journal of chemical engineering**, Wiley Online Library, v. 73, n. 6, p. 883–895, 1995.

GLOROT, Xavier; BORDES, Antoine; BENGIO, Yoshua. Deep sparse rectifier neural networks. In: 14TH INTERNATIONAL CONFERENCE ON ARTIFICIAL INTELLIGENCE AND STATISTICS (AISTATS). Fort Lauderdale, FL. **Proceedings [...]**: PMLR, 2011. P. 315–323.

GOMEZ, Juan C.; JUTAN, Arthur; BAEYENS, Enrique. Wiener model identification and predictive control of a pH neutralisation process. **IEE Proceedings-Control Theory and Applications**, IET, v. 151, n. 3, p. 329–338, 2004.

GOODFELLOW, Ian; BENGIO, Yoshua; COURVILLE, Aaron. **Deep Learning**. Cambridge: MIT Press, 2016.

GOPALUNI, Ratna B.; PATWARDHAN, Rohit S.; SHAH, Sirish L. MPC relevant identification-tuning the noise model. **Journal of Process Control**, Elsevier BV, v. 14, n. 6, p. 699–714, set. 2004.

GOPALUNI, Ratna B.; PATWARDHAN, Rohit S.; SHAH, Sirish L. The nature of data pre-filters in MPC relevant identification - open and closed-loop issues. **Automatica**, Elsevier, v. 39, n. 9, p. 1617–1626, 2003.

GRABEN, Peter b.; LIEBSCHER, Thomas; KURTHS, Jürgen. Neural and Cognitive Modeling with Networks of Leaky Integrator Units. In: GRABEN, Peter b.; ZHOU, Changsong; THIEL, Marco; KURTHS, Jürgen (Ed.). **Understanding Complex Systems**. Berlin, Heidelberg: Springer, 2008. P. 195–223.

GRAVES, Alex; JAITLEY, Navdeep; MOHAMED, Abdel-rahman. Hybrid speech recognition with deep bidirectional LSTM. In: IEEE. 2013 IEEE WORKSHOP ON AUTOMATIC SPEECH RECOGNITION AND UNDERSTANDING. Olomouc, CZ. **Proceedings [...]**. New York: IEEE, 2013. P. 273–278.

HAN, Hong-Gui; ZHANG, Lu; HOU, Ying; QIAO, Jun-Fei. Nonlinear model predictive control based on a self-organizing recurrent neural network. **IEEE Transactions on Neural Networks and Learning Systems**, IEEE, v. 27, n. 2, p. 402–415, 2015.

HASHIMOTO, Kazumune; ADACHI, Shuichi; DIMAROGONAS, Dimos V. Self-Triggered Model Predictive Control for Nonlinear Input-Affine Dynamical Systems via Adaptive Control Samples Selection. **IEEE Transactions on Automatic Control**, v. 62, n. 1, p. 177–189, 2017.

HAYKIN, S. **Neural Networks and Learning Machines**. 3. ed. New Jersey: Pearson, 2009. P. 900.

HAYKIN, S. **Redes neurais. Princípios e práticas**. 2. ed. Porto Alegre: Bookman, 2001. P. 900.

HEIRUNG, Tor A. N.; YDSTIE, B. Erik; FOSS, Bjarne. Dual adaptive model predictive control. **Automatica**, Elsevier BV, v. 80, p. 340–348, jun. 2017.

HENSON, Michael A. Nonlinear model predictive control: current status and future directions. **Computers & Chemical Engineering**, Elsevier, v. 23, n. 2, p. 187–202, 1998.

HENSON, Michael A.; SEBORG, Dale E. Adaptive nonlinear control of a pH neutralization process. **IEEE Transactions on Control Systems Technology**, IEEE, v. 2, n. 3, p. 169–182, 1994.

HERNANDAZ, Evelio; ARKUN, Yaman. Neural Network Modeling and an Extended DMC Algorithm to Control Nonlinear Systems. In: 1990 American Control Conference. San Francisco, CA. **Proceedings [...]**. Dayton: AACC, mai. 1990. P. 2454–2459.

HERTNECK, Michael; KOHLER, Johannes; TRIMPE, Sebastian; ALLGOWER, Frank. Learning an Approximate Model Predictive Controller With Guarantees. **IEEE Control Systems Letters**, Institute of Electrical e Electronics Engineers (IEEE), v. 2, n. 3, p. 543–548, jul. 2018.

HOCHREITER, Sepp. The vanishing gradient problem during learning recurrent neural nets and problem solutions. **International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems**, World Scientific, v. 6, n. 02, p. 107–116, 1998.

HOCHREITER, Sepp; SCHMIDHUBER, Jürgen. Long short-term memory. **Neural computation**, MIT Press, v. 9, n. 8, p. 1735–1780, 1997.

HUANG, Biao; MALHOTRA, Ashish; TAMAYO, Edgar C. Model predictive control relevant identification and validation. **Chemical Engineering Science**, Elsevier BV, v. 58, n. 11, p. 2389–2401, jun. 2003.

HUANG, Biao; WANG, Zidong. The Role of data prefiltering for integrated identification and model predictive control. In: 14TH IFAC WORLD CONGRESS. Beijing, CN. **Proceedings [...]**. New York: IFAC, 1999. P. 6751–6756.

HUANG, Jian; QIAN, Jin; LIU, Lei; WANG, Yongji; XIONG, Caihua; RI, Songhyok. Echo state network based predictive control with particle swarm optimization for pneumatic muscle actuator. **Journal of Franklin Institut**, Elsevier BV, v. 353, n. 12, p. 2761–2782, ago. 2016.

ISERMANN, Rolf; MÜNCHHOF, Marco. **Identification of Dynamic Systems**. Berlim: Springer, 2011. P. 705.

JAEGER, Herbert. Adaptive nonlinear system identification with echo state networks. In: ADVANCES IN NEURAL INFORMATION PROCESSING SYSTEMS. Vancouver, Canadá. **Proceedings [...]**. Cambridge: MIT Press, 2003. P. 609–616.

JAEGER, Herbert. 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**, Bonn, v. 148, n. 34, p. 13, 2001.

JAEGER, Herbert. Tutorial on training recurrent neural networks, covering BPPT, RTRL, EKF and the “echo state network” approach. Edição: GMD-Forschungszentrum Informationstechnik. **GMD Report 159, German National Research Center for Information Technology**, 2002.

JAEGER, Herbert; HAAS, Harald. Harnessing nonlinearity: Predicting chaotic systems and saving energy in wireless communication. **Science**, American Association for the Advancement of Science, v. 304, n. 5667, p. 78–80, 2004.

JAEGER, Herbert; LUKOŠEVIČIUS, Mantas; POPOVICI, Dan; SIEWERT, Udo. Optimization and applications of echo state networks with leaky- integrator neurons. **Neural Networks**, Elsevier BV, v. 20, n. 3, p. 335–352, abr. 2007.

JEON, Byung-Ki; KIM, Eui-Jong. LSTM-Based Model Predictive Control for Optimal Temperature Set-Point Planning. **Sustainability**, MDPI AG, v. 13, n. 2, p. 894, jan. 2021.

JIN, Yingyi; SU, Chengli. Adaptive model predictive control using diagonal recurrent neural network. In: IEEE. 2008 4th INTERNATIONAL CONFERENCE ON NATURAL COMPUTATION. Jinan, China. **Proceedings [...]**. New York: IEEE, 2008. P. 276–280.

JORDAN, Michael I. **Serial Order: A parallel Distributed Processing Approach**. San Diego, 1986. Ca, EUA.

JORDANOU, Jean P.; CAMPONOGARA, Eduardo; ANTONELLO, Eric Aislan; AGUIAR, Marco Aurélio Schmitz. Nonlinear Model Predictive Control of an Oil Well with Echo State Networks. In: 2018. 3RD IFAC WORKSHOP ON AUTOMATIC CONTROL IN OFFSHORE OIL AND GAS PRODUCTION OOGP 2018. Esbjerg, Dinamarca. **Proceedings [...]**. Laxenburg: IFAC, 2018. P. 13–18.

KARIMANZIRA, Divas; RAUSCHENBACH, Thomas. Deep Learning Based Model Predictive Control for a Reverse Osmosis Desalination Plant. **Journal of Applied Mathematics and Physics**, Scientific Research Publishing, Inc., v. 08, n. 12, p. 2713–2731, 2020.

KARLA, V. R.; BAKKER, Huub C. Neural-network-based model predictive control: a case study. In: IEEE. 2ND NEW ZEALAND INTERNATIONAL TWO-STREAM CONFERENCE ON ARTIFICIAL NEURAL NETWORKS AND EXPERT SYSTEMS. Dunedin, Nova Zelândia. **Proceedings [...]**. New York: IEEE, 1995. P. 355–358.

KULHAVÝ, Rudolf. Restricted exponential forgetting in real-time identification. **Automatica**, v. 23, n. 5, p. 589–600, 1987.

KUMAR, Kunal; PATWARDHAN, Sachin C.; NORONHA, Santosh. Development of an adaptive and explicit dual model predictive controller based on generalized orthogonal basis filters. **Journal of Process Control**, v. 83, p. 196–214, 2019a.

KUMAR, Kunal; PATWARDHAN, Sachin C.; NORONHA, Santosh. Tracking Economic Optimum of a Continuous Fermenter using Adaptive Dual Nonlinear MPC. In: 2019 58th ANNUAL CONFERENCE OF THE SOCIETY OF INSTRUMENT AND CONTROL ENGINEERS OF JAPAN (SICE). Hiroshima, Japão. **Proceedings [...]**. New York: IEEE, 2019b. P. 581–586.

KURTZ, Michael J.; HENSON, Michael A. Input-output linearizing control of constrained nonlinear processes. **Journal of Process Control**, v. 7, n. 1, p. 3–17, 1997.

LANG, Kevin J; WAIBEL, Alex H; HINTON, Geoffrey E. A time-delay neural network architecture for isolated word recognition. **Neural Networks**, Elsevier, v. 3, n. 1, p. 23–43, 1990.

- LAURÍ, David; MARTÍNEZ, Miguel; SALCEDO, José V.; SANCHIS, J. PLS-based model predictive control relevant identification: PLS-PH algorithm. **Chemometrics and Intelligent Laboratory Systems**, Elsevier BV, v. 100, n. 2, p. 118–126, fev. 2010.
- ŁAWRYŃCZUK, Maciej. A Family of Model Predictive Control Algorithms With Artificial Neural Networks. **International Journal of Applied Mathematics and Computer Science**, Walter de Gruyter GmbH, v. 17, n. 2, jan. 2007.
- ŁAWRYŃCZUK, Maciej. **Computationally Efficient Model Predictive Control Algorithms**. Heidelberg: Springer International Publishing, 2014.
- ŁAWRYŃCZUK, Maciej. Practical nonlinear predictive control algorithms for neural Wiener models. **Journal of Process Control**, v. 23, n. 5, p. 696–714, 2013.
- ŁAWRYŃCZUK, Maciej. Computationally efficient nonlinear predictive control based on neural Wiener models. **Neurocomputing**, Elsevier, v. 74, n. 1-3, p. 401–417, 2010.
- ŁAWRYŃCZUK, Maciej; TATJEWSKI, Piotr. A computationally efficient nonlinear predictive control algorithm with RBF neural models and its application. In: INTERNATIONAL CONFERENCE ON ROUGH SETS AND INTELLIGENT SYSTEMS PARADIGMS. Warsaw, Polónia. **Proceedings [...]**. New York: Springer, 2007. P. 603–612.
- LEE, Jay H. Modeling and Identification for Nonlinear Model Predictive Control: Requirements, Current Status and Future Research Needs. In: ALLGÖWER, Frank; ZHENG, Alex (Ed.). **Nonlinear Model Predictive Control**. Basel: Springer, 2000. P. 269–293.
- LEE, Jay H.; MORARI, Manfred; GARCIA, Carlos E. State-space interpretation of model predictive control. **Automatica**, Elsevier BV, v. 30, n. 4, p. 707–717, abr. 1994.
- LEE, Jay H.; RICKER, N Lawrence. Extended Kalman Filter Based Nonlinear Model Predictive Control. In: 1993 AMERICAN CONTROL CONFERENCE. San Francisco, Califórnia. **Proceedings [...]**. New York: IEEE, 1993. P. 1895–1899.
- LEVINE, William S. **The Control Handbook, (Three Volume Set) (Electrical Engineering Handbook)**. 2. ed. Boca Raton: CRC Press, Inc., 2010.
- LI, Decai; HAN, Min; WANG, Jun. Chaotic time series prediction based on a novel robust echo state network. **IEEE Transactions on Neural Networks and Learning Systems**, IEEE, v. 23, n. 5, p. 787–799, 2012.

- LI, Pengfei; WANG, Tao; KANG, Yu; ZHAO, Yun-Bo. Event-Triggered Adaptive Horizon Model Predictive Control for Perturbed Nonlinear Systems. In: 2020 59th IEEE CONFERENCE ON DECISION AND CONTROL (CDC). Jeju Island, Republic of Korea. **Proceedings [...]**. New York: IEEE, 2020. P. 2392–2397.
- LI, Zhijun; XIAO, Hanzhen; YANG, Chenguang; ZHAO, Yiwen. Model Predictive Control of Nonholonomic Chained Systems Using General Projection Neural Networks Optimization. **IEEE Transactions on Systems, Man, and Cybernetics: Systems**, Institute of Electrical and Electronics Engineers (IEEE), v. 45, n. 10, p. 1313–1321, out. 2015.
- LIU, GP; KADIRKAMANATHAN, V. Predictive control for non-linear systems using neural networks. **International Journal of Control**, Taylor & Francis, v. 71, n. 6, p. 1119–1132, 1998.
- LJUNG, Lennart. **System identification**. 2. ed. Englewood Cliffs, New Jersey: Prentice Hall, 1999.
- LUKOŠEVIČIUS, Mantas. A practical guide to applying echo state networks. In: MONTAVON, Grégoire; ORR, Geneviève B.; MÜLLER, Klaus-Robert (Ed.). **Neural Networks: Tricks of the Trade**. 2. ed. Berlin: Springer, 2012. P. 659–686.
- MAASS, Wolfgang; BISHOP, Christopher M. **Pulsed Neural Networks**. Cambridge: MIT Press, 11 jan. 2001.
- MAASS, Wolfgang; NATSCHLÄGER, Thomas; MARKRAM, Henry. Real-Time Computing Without Stable States: A New Framework for Neural Computation Based on Perturbations. **Neural Computation**, MIT Press - Journals, v. 14, n. 11, p. 2531–2560, nov. 2002.
- MAHMOODI, Sanaz; POSHTAN, Javad; JAHED-MOTLAGH, Mohammad Reza; MONTAZERI, Allahyar. Nonlinear model predictive control of a pH neutralization process based on Wiener–Laguerre model. **Chemical Engineering Journal**, v. 146, n. 3, p. 328–337, 2009.
- MARAFIOTI, Giancarlo; BITMEAD, Robert R.; HOVD, Morten. Persistently exciting model predictive control. **International Journal of Adaptive Control and Signal Processing**, Wiley, v. 28, n. 6, p. 536–552, mai. 2014.
- MAYNE, David Q. Model predictive control: Recent developments and future promise. **Automatica**, Elsevier BV, v. 50, n. 12, p. 2967–2986, dez. 2014.

- MAYNE, David Q. Nonlinear Model Predictive Control: Challenges and Opportunities. In: **Nonlinear Model Predictive Control**. Edição: Frank Allgöwer e Alex Zheng. Basel: Birkhäuser Basel, 2000. P. 23–44.
- MAYNE, David Q.; MICHALSKA, Hannah. Adaptive receding horizon control for constrained nonlinear systems. In: 32ND IEEE CONFERENCE ON DECISION AND CONTROL. San Antonio, Texas. **Proceedings [...]**. New York: IEEE, 1993. P. 1286–1291.
- MCCULLOCH, Warren S.; PITTS, Walter. A logical calculus of the ideas immanent in nervous activity. **The Bulletin of Mathematical Biophysics**, v. 5, n. 4, p. 115–133, 1943.
- MCINTOSH, Andrew R.; SHAH, Sirish L.; FISHER, D. Grant. Analysis and tuning of adaptive generalized predictive control. **Can. J. Chem. Eng.**, Wiley, v. 69, n. 1, p. 97–110, fev. 1991.
- MU, Junxia; REES, David; LIU, GP. Advanced controller design for aircraft gas turbine engines. **Control Engineering Practice**, Elsevier, v. 13, n. 8, p. 1001–1015, 2005.
- MYNAR, Zbynek; VESELY, Libor; VACLAVEK, Pavel. PMSM Model Predictive Control With Field-Weakening Implementation. **IEEE Transactions on Industrial Electronics**, Institute of Electrical e Electronics Engineers (IEEE), v. 63, n. 8, p. 5156–5166, ago. 2016.
- NAJIM, Kaddour; RUSNAK, A; MÉSZÁROS, Alojz; FIKAR, Miroslav. Constrained long-range predictive control based on artificial neural networks. **International Journal of Systems Science**, Taylor & Francis, v. 28, n. 12, p. 1211–1226, 1997.
- NARENDRA, Kumpati S.; PARTHASARATHY, K. Identification and Control of Dynamical Systems Using Neural Networks. **IEEE Transactions on Neural Networks**, IEEE Press, Piscataway, NJ, USA, v. 1, n. 1, p. 4–27, mar. 1990.
- NELLES, O. **Nonlinear System Identification**. London: Springer, 2001. P. 785.
- PAN, Yunpeng; WANG, Jun. Model Predictive Control of Unknown Nonlinear Dynamical Systems Based on Recurrent Neural Networks. **IEEE Transactions on Industrial Electronics**, Institute of Electrical e Electronics Engineers (IEEE), v. 59, n. 8, p. 3089–3101, ago. 2012.
- PAN, Yunpeng; WANG, Jun. Two neural network approaches to model predictive control. In: 2008 AMERICAN CONTROL CONFERENCE. Seattle, WA. **Proceedings [...]**. New York: IEEE, 2008. P. 1685–1690.

PARISINI, Thomas; ZOPPOLI, Riccardo. A receding-horizon regulator for nonlinear systems and a neural approximation. **Automatica**, Elsevier BV, v. 31, n. 10, p. 1443–1451, out. 1995.

PARLOS, A.; ATIYA, A.; CHONG, K.; TSAI, W.; FERNANDEZ, B. Recurrent multilayer perceptron for nonlinear system identification. In: 3. IJCNN-91-SEATTLE International Joint Conference on Neural Networks. Seattle: International Joint Conference on Neural Networks, jul. 1991. P. 537–540.

PČOLKA, Matej; ŽÁČEKOVÁ, Eva; ROBINETT, Rush; ČELIKOVSKÝ, Sergej; ŠEBEK, Michael. Bridging the gap between the linear and nonlinear predictive control: Adaptations for efficient building climate control. **Control Engineering Practice**, Elsevier, v. 53, p. 124–138, 2016.

PENG, Hui; NAKANO, Kazushi; SHIOYA, Hideo. Nonlinear predictive control using neural nets-based local linearization ARX model: Stability and industrial application. **IEEE Transactions on Control Systems Technology**, IEEE, v. 15, n. 1, p. 130–143, 2006.

PICHE, Stephen; KEELER, James D; MARTIN, Greg; BOE, Gene; JOHNSON, Doug; GERULES, Mark. Neural network based model predictive control. In: ADVANCES IN NEURAL INFORMATION PROCESSING SYSTEMS. Denver, CO. **Proceedings [...]**. Cambridge: MIT Press, 2000. P. 1029–1035.

PLUCENIO, A. **Desenvolvimento de técnicas de controle não linear para elevação de fluidos multifásicos**. 2010. Tese – Universidade Federal de Santa Catarina, Florianópolis.

PLUCENIO, Agostinho; PAGANO, Daniel .J.; BRUCIAPAGLIA, Augusto H.; NORMEY-RICO, Julio E. A Practical Approach to Predictive Control for Nonlinear Processes. In: 12. 7TH IFAC SYMPOSIUM ON NONLINEAR CONTROL SYSTEMS. Pretoria, South Africa. **Proceedings [...]**. New York: IFAC, 2007. P. 210–215.

POTTS, Alain S.; ROMANO, Rodrigo A.; GARCIA, Claudio. Improving performance and stability of MPC relevant identification methods. **Control Engineering Practice**, Elsevier BV, v. 22, p. 20–33, jan. 2014.

PSICHOGIOS, Dimitris C.; UNGAR, Lyle H. Nonlinear internal model control and model predictive control using neural networks. In: 5TH IEEE INTERNATIONAL SYMPOSIUM ON INTELLIGENT CONTROL. Philadelphia, PA. **Proceedings [...]**. New York: IEEE, set. 1990. P. 1082–1087.

PUSKORIUS, Gintaras V.; FELDKAMP, Lee A. Neurocontrol of nonlinear dynamical systems with Kalman filter trained recurrent networks. **IEEE Transactions on Neural Networks**, Institute of Electrical e Electronics Engineers (IEEE), v. 5, n. 2, p. 279–297, mar. 1994.

QIAO, Junfei; WANG, Lei; YANG, Cuili. Adaptive lasso echo state network based on modified Bayesian information criterion for nonlinear system modeling. **Neural Computing and Applications**, Springer Science e Business Media LLC, v. 31, n. 10, p. 6163–6177, mar. 2018.

QIN, S. Joe; BADGWELL, Thomas A. A survey of industrial model predictive control technology. **Control Engineering Practice**, v. 11, n. 7, p. 733–764, 2003.

QUACHIO, Raphael; GARCIA, Claudio. Application of the PLS-PH algorithm for identification of closed-loop systems. In: 2011. 9TH IEEE INTERNATIONAL CONFERENCE ON CONTROL AND AUTOMATION (ICCA). Santiago, Chile. **Proceedings [...]**. New York: IEEE, dez. 2011.

QUACHIO, Raphael; GARCIA, Claudio. Application of the PLS-PH method for identifying Polynomial NARX Models. **Journal of Control, Automation and Electrical Systems**, v. 25, n. 2, p. 184–194, abr. 2014.

QUACHIO, Raphael; GARCIA, Claudio. MPC relevant identification method for Hammerstein and Wiener models. **Journal of Process Control**, v. 80, p. 78–88, 2019.

QUACHIO, Raphael; GARCIA, Claudio. MPC relevant identification method for hammerstein models. **IFAC-PapersOnLine**, Elsevier BV, v. 50, n. 2, p. 47–52, dez. 2017.

REDDI, Kamesh; RANI, K. Yamuna. Novel Formulation of Adaptive MPC as EKF Using ANN Model: Multiproduct Semibatch Polymerization Reactor Case Study. **IEEE Transactions on Neural Networks and Learning Systems**, PP, p. 1–13, out. 2016.

ROSENBLATT, Frank. The perceptron: a probabilistic model for information storage and organization in the brain. **Psychological review**, American Psychological Association, v. 65, n. 6, p. 386, 1958.

RUMELHART, David E.; HINTON, Geoffrey E.; WILLIAMS, Ronald J. Learning representations by back-propagating errors. **Nature**, Springer Science e Business Media LLC, v. 323, n. 6088, p. 533–536, out. 1986.

SCHMIDHUBER, Jürgen. Deep learning in neural networks: An overview. **Neural Networks**, v. 61, p. 85–117, 2015.

SCHWEDERSKY, Bernardo B.; FLESCHE, Rodolfo C. C.; DANGUI, Hiago A. C. Practical Nonlinear Model Predictive Control Algorithm for Long Short-Term Memory Networks. In: 12TH IFAC SYMPOSIUM ON DYNAMICS AND CONTROL OF PROCESS SYSTEMS, INCLUDING BIOSYSTEMS – DYCOPS. Florianópolis, SC. **Proceedings [...]**. Laxenburg: IFAC, 2019. P. 1–6.

SEKI, Hiroya; OYAMA, Satoshi; OGAWA, Morimasa. Nonlinear Model Predictive Control Using Successive Linearization. **Transactions of the Society of Instrument and Control Engineers**, The Society of Instrument and Control Engineers, v. 38, n. 1, p. 61–66, 2002.

SHENG, Chunyang; ZHAO, Jun; LIU, Ying; WANG, Wei. Prediction for noisy nonlinear time series by echo state network based on dual estimation. **Neurocomputing**, Elsevier BV, v. 82, p. 186–195, abr. 2012.

SHI, Zhiwei; HAN, Min. Ridge regression learning in ESN for chaotic time series prediction. **Control and Decision**, v. 22, n. 3, p. 258, 2007.

SHOOK, D.S.; MOHTADI, C.; SHAH, S.L. A control-relevant identification strategy for GPC. **IEEE Transactions on Automatic Control**, Institute of Electrical and Electronics Engineers (IEEE), v. 37, n. 7, p. 975–980, jul. 1992.

SMOLA, Alex J; SCHÖLKOPF, Bernhard. A tutorial on support vector regression. **Statistics and computing**, Springer, v. 14, n. 3, p. 199–222, 2004.

SOLOWAY, Don; HALEY, Pamela J. Neural generalized predictive control. In: 1996 IEEE INTERNATIONAL SYMPOSIUM ON INTELLIGENT CONTROL. Dearborn, MI. **Proceedings [...]**. New York: IEEE, set. 1996. P. 277–282.

SONG, Qingsong; ZHAO, Xiangmo; FENG, Zuren; SONG, Baohua. Recursive least squares algorithm with adaptive forgetting factor based on echo state network. In: 2011 9th WORLD CONGRESS ON INTELLIGENT CONTROL AND AUTOMATION. Taipei, Taiwan. **Proceedings [...]**. New York: IEEE, jun. 2011a. P. 295–298.

SONG, Qingsong; ZHAO, Xiangmo; FENG, Zuren; SONG, Baohua. Recursive least squares algorithm with adaptive forgetting factor based on echo state network. In: 2011 9th World Congress on Intelligent Control and Automation. New York: IEEE, jun. 2011b.

STEIL, Jochen J. Backpropagation-decorrelation: online recurrent learning with $O(N)$ complexity. In: INTERNATIONAL JOINT CONFERENCE ON NEURAL NETWORKS. Budapest, Hungary. **Proceedings [...]**. New York: IEEE, jul. 2004. P. 843–848.

SU, Chengli; WU, Yun. Adaptive neural network predictive control based on PSO algorithm. In: 2009 CHINESE CONTROL AND DECISION CONFERENCE. Guilin, China. **Proceedings [...]**. New York: IEEE, 2009. P. 5829–5833.

SU, Hong-Te; MCAVOY, Thomas J. Neural model predictive control of nonlinear chemical processes. In: 1993 IEEE INTERNATIONAL SYMPOSIUM ON INTELLIGENT CONTROL. Chicago, IL. **Proceedings [...]**. New York: IEEE, ago. 1993. P. 358–363.

SUN, Xiaochuan; LI, Yingqi; LIU, Jiayu; ZHANG, Minghui. Echo State Wavelet-sigmoid Networks and Their Application to Nonlinear System Identification. **Engineering Letters**, v. 25, n. 3, p. 301–311, 2017.

SUN, Zhongqi; DAI, Li; LIU, Kun; DIMAROGONAS, Dimos V.; XIA, Yuanqing. Robust Self-Triggered MPC With Adaptive Prediction Horizon for Perturbed Nonlinear Systems. **IEEE Transactions on Automatic Control**, v. 64, n. 11, p. 4780–4787, 2019.

SUNDERMEYER, Martin; SCHLÜTER, Ralf; NEY, Hermann. LSTM neural networks for language modeling. In: 13TH annual conference of the international speech communication association. Portland, Oregon, USA: ISCA, 2012.

SUSSILLO, David; ABBOTT, L.F. Generating Coherent Patterns of Activity from Chaotic Neural Networks. **Neuron**, Elsevier BV, v. 63, n. 4, p. 544–557, ago. 2009.

TANASKOVIC, Marko; FAGIANO, Lorenzo; GLIGOROVSKI, Vojislav. Adaptive model predictive control for linear time varying MIMO systems. **Automatica**, Elsevier BV, v. 105, p. 237–245, jul. 2019.

TANASKOVIC, Marko; FAGIANO, Lorenzo; SMITH, Roy; MORARI, Manfred. Adaptive receding horizon control for constrained MIMO systems. **Automatica**, v. 50, nov. 2014.

TATJEWSKI, Piotr; ŁAWRYŃCZUK, Maciej. Soft computing in model-based predictive control. **International Journal of Applied Mathematics and Computer Science**, v. 16, p. 7–26, 2006.

TEMENG, Kwaku O; SCHNELLE, Phillip D; MCAVOY, Thomas J. Model predictive control of an industrial packed bed reactor using neural networks. **Journal of Process Control**, Elsevier, v. 5, n. 1, p. 19–27, 1995.

TERZI, Enrico; BONASSI, Fabio; FARINA, Marcello; SCATTOLINI, Riccardo. Learning model predictive control with long short-term memory networks. **International Journal of Robust and Nonlinear Control**, p. 1–20, 2021.

TOWNSEND, S.; LIGHTBODY, Gordon; BROWN, Michael D.; IRWIN, George W. Nonlinear Dynamic Matrix Control using Local Models. **Transactions of the Institute of Measurement and Control**, SAGE Publications, v. 20, n. 1, p. 47–56, jan. 1998.

TRAJANOSKI, Zlatko; WACH, Paul. Neural predictive controller for insulin delivery using the subcutaneous route. **IEEE Transactions on Biomedical Engineering**, IEEE, v. 45, n. 9, p. 1122–1134, 1998.

VICENTE, Bernardo A. H.; TRODDEN, Paul A. Stabilizing predictive control with persistence of excitation for constrained linear systems. **Systems & Control Letters**, v. 126, p. 58–66, 2019.

WANG, H; WANG, Y. Neural-network-based fault-tolerant control of unknown nonlinear systems. **IET Proceedings-Control Theory and Applications**, IET, v. 146, n. 5, p. 389–398, 1999.

WANG, Jiasen; WANG, Jun; HAN, Qing-Long. Neurodynamics-based Model Predictive Control of Continuous-time Under-actuated Mechatronic Systems. **IEEE/ASME Transactions on Mechatronics**, Institute of Electrical e Electronics Engineers (IEEE), p. 1–1, 2020.

WERBOS, Paul J. Backpropagation through time: what it does and how to do it. **Proceedings of the IEEE**, IEEE, v. 78, n. 10, p. 1550–1560, 1990.

WITTENMARK, Björn. Adaptive Dual Control Methods: An Overview. In: 1995 ADAPTIVE SYSTEM IN CONTROL AND SIGNAL PROCESSING. Budapest, Hungary. **Proceedings [...]**. Amsterdam: Elsevier BV, jun. 1995. P. 67–72.

WYSOCKI, Antoni; ŁAWRYŃCZUK, Maciej. Jordan neural network for modelling and predictive control of dynamic systems. In: IEEE. 20TH INTERNATIONAL CONFERENCE ON METHODS AND MODELS IN AUTOMATION AND ROBOTICS (MMAR). Miedzyzdroje, Polônia. **Proceedings [...]**. New York: IEEE, 2015. P. 145–150.

XU, Jing; LI, Chuandong; HE, Xing; HUANG, Tingwen. Recurrent neural network for solving model predictive control problem in application of four-tank benchmark. **Neurocomputing**, Elsevier BV, v. 190, p. 172–178, mai. 2016.

YAN, Zheng; WANG, Jun. Model Predictive Control for Tracking of Underactuated Vessels Based on Recurrent Neural Networks. **IEEE Journal of Oceanic Engineering**, Institute of Electrical e Electronics Engineers (IEEE), v. 37, n. 4, p. 717–726, out. 2012a.

YAN, Zheng; WANG, Jun. Model predictive control for tracking of underactuated vessels based on recurrent neural networks. **IEEE Journal of Oceanic Engineering**, IEEE, v. 37, n. 4, p. 717–726, 2012b.

YANG, Cuili; QIAO, Junfei; AHMAD, Zohaib; NIE, Kaizhe; WANG, Lei. Online sequential echo state network with sparse RLS algorithm for time series prediction. **Neural Networks**, v. 118, p. 32–42, 2019.

YANG, Y.; HARLEY, R. G.; DIVAN, D.; HABETLER, T. G. Overhead conductor thermal dynamics identification by using Echo State Networks. In: INTERNATIONAL JOINT CONFERENCE ON NEURAL NETWORKS. Atlanta, GA. **Proceedings [...]**. New York: IEEE, jun. 2009. P. 3436–3443.

YILDIZ, Izzet B.; JAEGER, Herbert; KIEBEL, Stefan J. Re-visiting the echo state property. **Neural Networks**, Elsevier BV, v. 35, p. 1–9, nov. 2012.

YU, DL; GOMM, JB. Implementation of neural network predictive control to a multivariable chemical reactor. **Control Engineering Practice**, Elsevier, v. 11, n. 11, p. 1315–1323, 2003.

ŽÁČEKOVÁ, Eva; PRÍVARA, Samuel; VÁŇA, Zdeněk. Model predictive control relevant identification using partial least squares for building modeling. In: 2011 AUSTRALIAN CONTROL CONFERENCE. Melbourne, Austrália. **Proceedings [...]**. New York: IEEE, 2011. P. 422–427.

ŽÁČEKOVÁ, Eva; PRÍVARA, Samuel; VÁŇA, Zdeněk Jiří Cigler. Towards the real-life implementation of MPC for an office building: Identification issues. **Applied Energy**, Elsevier BV, v. 135, p. 53–62, dez. 2014.

ZARZYCKI, Krzysztof; ŁAWRYŃCZUK, Maciej. LSTM and GRU Neural Networks as Models of Dynamical Processes Used in Predictive Control: A Comparison of Models Developed for Two Chemical Reactors. **Sensors**, MDPI AG, v. 21, n. 16, p. 5625, ago. 2021.

ZHANG, Changhao; GUO, Yu; WANG, Fei; CHEN, Badong. Generalized Maximum Correntropy-Based Echo State Network for Robust Nonlinear System Identification. In: 2018. INTERNATIONAL JOINT CONFERENCE ON NEURAL NETWORKS (IJCNN). Rio de Janeiro, Brasil. **Proceedings [...]**. New York: IEEE, jul. 2018.

ZHENG, Alex. A computationally efficient nonlinear MPC algorithm. In: 1997 AMERICAN CONTROL CONFERENCE. Albuquerque, New Mexico. **Proceedings [...]**. New York: IEEE, 1997. P. 1623–1627.

ZHENG, Alex. Some Practical Issues and Possible Solutions for Nonlinear Model Predictive Control. In: **Nonlinear Model Predictive Control**. Edição: Frank Allgöwer e Alex Zheng. Basel: Birkhäuser Basel, 2000. P. 129–143.

APÊNDICE A – DESCRIÇÃO DOS PROCESSOS EM ESTUDO

Neste apêndice são detalhados os processos utilizados como estudo de caso neste trabalho. Para cada um deles, é apresentado um esquema geral de funcionamento, seu equacionamento e é descrito o problema de controle. São apresentados quatro processos, dois usados para a identificação de modelos a partir de dados coletados da operação do processo e dois que são usados para avaliação de algoritmos de identificação on-line. Os processos usados para identificação a partir de dados coletados do processo são o processo de neutralização de pH, apresentado na Seção A.1, e a bancada de controle de pressões, apresentada na Seção A.2. Para esses processos são detalhados os procedimentos para obtenção dos modelos convencionais, os quais são usados para comparação com os obtidos utilizando as abordagens de identificação propostas no trabalho, e são apresentados os respectivos resultados. Já o tanque de perfil cônico, detalhado na Seção A.3, e o motor síncrono de ímãs permanentes, detalhado na Seção A.4, são os processos usados para avaliação de abordagens de identificação on-line, sendo apresentados apenas o funcionamento geral e o equacionamento do processo, já que os modelos, para tais processos, são obtidos on-line.

A.1 REATOR DE NEUTRALIZAÇÃO DE PH

O reator de neutralização de pH é um processo usado, por várias fontes na literatura, para avaliação do desempenho de métodos para identificação de sistemas dinâmicos não lineares e algoritmos de controle não linear (HENSON; SEBORG, 1994; GOMEZ *et al.*, 2004; MAHMOODI *et al.*, 2009; ŁAWRYŃCZUK, 2013). Esse processo desperta interesse e é bastante estudado devido à sua relevância na indústria química e ao seu comportamento não linear desafiador, que o torna um excelente estudo de caso para a avaliação de controladores não lineares. O maior desafio para o controle em malha fechada deste processo consiste na operação do mesmo em uma faixa de pH neutra, com valores entre 6 e 8, no caso em que apenas ácidos fortes e bases fortes estão presentes (GOMEZ *et al.*, 2004).

O processo de neutralização de pH consiste na mistura, em um tanque com volume constante V , de uma base NaOH, com vazão q_1 , uma solução tampão NaHCO_3 , com vazão q_2 , e uma ácido HNO_3 , com vazão q_3 . Este processo é apresentado, por meio de um diagrama de processo e instrumentos, na Figura 39. A saída do processo, y , é o pH da solução efluente q_4 do reator, sendo manipulada pela solução básica q_1 e pela solução tampão q_2 , a qual, é considerada neste estudo como uma perturbação não medida. Adicionalmente, é assumido que a solução ácida q_3 apresenta vazão constante. Todas as vazões são expressas em mililitros por segundo.

A modelagem do processo de neutralização de pH é apresentada com detalhes em Henson e Seborg (1994). O modelo dinâmico é obtido usando equações de conservação e relações de equilíbrio, considerando uma mistura perfeita, densidade constante e a solubilidade completa

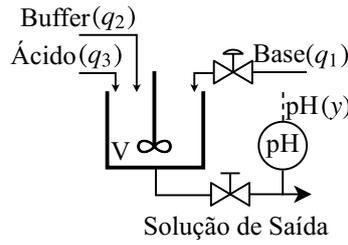
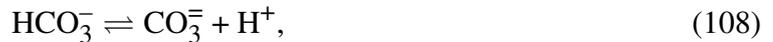


Figura 39 – Diagrama de processo e instrumentos de um processo de neutralização de pH.

dos íons. A reação química é definida como



sendo que as constantes de equilíbrio são representadas por

$$K_{a_1} = \frac{[\text{HCO}_3^-][\text{H}^+]}{[\text{H}_2\text{CO}_3]}, \quad (110)$$

$$K_{a_2} = \frac{[\text{CO}_3^{2-}][\text{H}^+]}{[\text{HCO}_3^-]}, \quad (111)$$

$$K_w = [\text{H}^+][\text{OH}^-]. \quad (112)$$

Se forem definidos dois elementos invariantes de reação, W_a , o qual representa uma quantidade relacionada à carga, e W_b , o qual representa a concentração de íons CO_3^{2-} , é possível definir o equilíbrio químico para cada solução $i \in [1, 4]$ como

$$W_{a_i} = [\text{H}^+]_i - [\text{OH}^-]_i - [\text{HCO}_3^-]_i - 2[\text{CO}_3^{2-}]_i, \quad (113)$$

$$W_{b_i} = [\text{H}_2\text{CO}_3]_i + [\text{HCO}_3^-]_i + [\text{CO}_3^{2-}]_i. \quad (114)$$

Utilizando os elementos W_a e W_b , o pH da solução de saída q_4 pode ser obtido pela equações

$$W_b \frac{\frac{K_{a_1}}{[\text{H}^+]} + \frac{2K_{a_1}K_{a_2}}{[\text{H}^+]^2}}{1 + \frac{K_{a_1}}{[\text{H}^+]} + \frac{K_{a_1}K_{a_2}}{[\text{H}^+]^2}} + W_a + \frac{K_w}{[\text{H}^+]} - [\text{H}^+] = 0, \quad (115)$$

$$\text{pH} = -\log([\text{H}^+]). \quad (116)$$

Como o volume do tanque é constante, o balanço de massa resulta em

$$q_1 + q_2 + q_3 - q_4 = 0, \quad (117)$$

o qual pode ser combinado com o balanço de massa para cada espécie iônica para a obtenção das equações diferenciais para cada invariante W_{a_4} e W_{b_4} , as quais são escritas como

$$\frac{dW_{a_4}(t)}{dt} = \frac{q_1(t)(W_{a_1} - W_{a_4}(t))}{V} + \frac{q_2(t)(W_{a_2} - W_{a_4}(t))}{V} + \frac{q_3(t)(W_{a_3} - W_{a_4}(t))}{V} \quad (118)$$

$$\frac{dW_{b_4}(t)}{dt} = \frac{q_1(t)(W_{b_1} - W_{b_4}(t))}{V} + \frac{q_2(t)(W_{b_2} - W_{b_4}(t))}{V} + \frac{q_3(t)(W_{b_3} - W_{b_4}(t))}{V}. \quad (119)$$

O comportamento dinâmico do processo pode ser descrito por meio de uma formulação no espaço de estados pelas expressões

$$\dot{\mathbf{x}} = \mathbf{r}(\mathbf{x}) + \mathbf{g}(\mathbf{x})q_1 + \mathbf{p}(\mathbf{x})q_2, \quad (120)$$

$$h(\mathbf{x}, y) = 0, \quad (121)$$

nas quais as variáveis de estado do processo são

$$\mathbf{x} = \begin{bmatrix} x_1 & x_2 \end{bmatrix}^T = \begin{bmatrix} W_{a_4} & W_{b_4} \end{bmatrix}^T, \quad (122)$$

e

$$\mathbf{r}(\mathbf{x}) = \begin{bmatrix} \frac{q_3(t)(W_{a_3} - x_1)}{V} & \frac{q_3(t)(W_{b_3} - x_2)}{V} \end{bmatrix}^T, \quad (123)$$

$$\mathbf{g}(\mathbf{x}) = \begin{bmatrix} \frac{(W_{a_1} - x_1)}{V} & \frac{(W_{b_1} - x_2)}{V} \end{bmatrix}^T,$$

$$\mathbf{p}(\mathbf{x}) = \begin{bmatrix} \frac{(W_{a_2} - x_1)}{V} & \frac{(W_{b_2} - x_2)}{V} \end{bmatrix}^T, \quad (124)$$

$$h(\mathbf{x}, y) = x_1 + 10^{y(t)-14} - 10^{-y(t)} + x_2 \frac{1 + 2 \times 10^{y(t)-K_2}}{1 + 10^{K_1-y(t)} + 10^{y(t)-K_2}}, \quad (125)$$

nas quais K_1 e K_2 representam as constantes de dissociação primárias e secundárias do ácido fraco H_2CO_3 , respectivamente.

As condições de operação nominal do processo são apresentadas na Tabela 8.

A.1.1 Identificação do processo de neutralização de pH

Para o desenvolvimento de modelos e controladores, a simulação do modelo fenomenológico foi utilizada como o modelo real. Para a resolução das equações diferenciais ordinárias, definidas em (124), foi utilizado o método de integração numérica Runge-Kutta 45 e foi adicionado um ruído na equação de saída (125), de forma a emular um ruído de medição.

Uma simulação em malha aberta foi realizada, com o objetivo de obter um conjunto de dados para a identificação de sistemas. Foi considerado um período de amostragem de 10

Tabela 8 – Condições nominais de operação do processo de neutralização de pH.

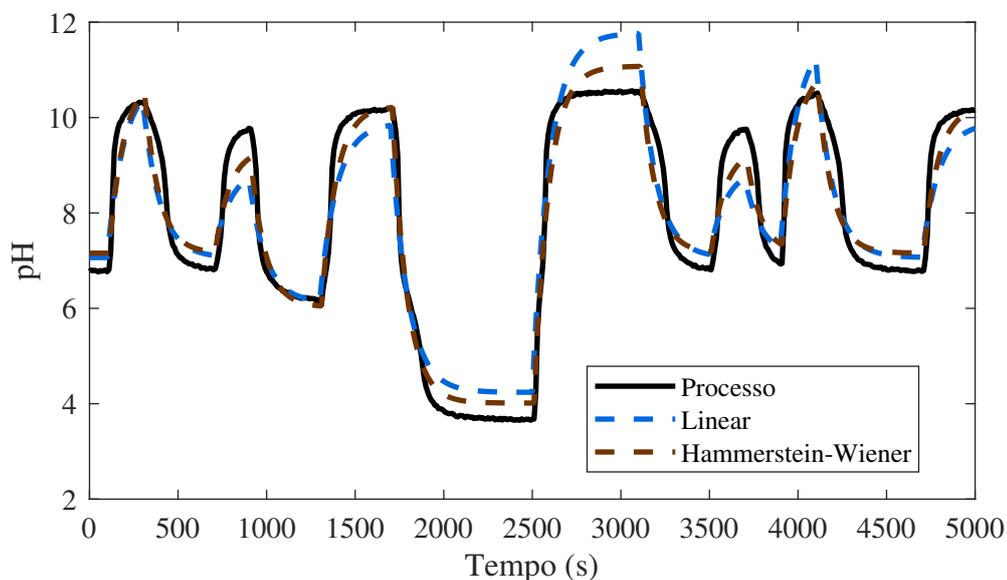
$q_1 = 15,55 \text{ mL s}^{-1}$	$W_a = -4.32 \times 10^{-4} \text{ mol L}^{-1}$
$q_2 = 0,55 \text{ mL s}^{-1}$	$W_b = 5.28 \times 10^{-4} \text{ mol L}^{-1}$
$q_3 = 16,60 \text{ mL s}^{-1}$	$W_{a1} = -3.05 \times 10^{-3} \text{ mol L}^{-1}$
$K_1 = 6.35$	$W_{a3} = 3 \times 10^{-3} \text{ mol L}^{-1}$
$K_2 = 10.25$	$W_{b1} = 5 \times 10^{-5} \text{ mol L}^{-1}$
$V = 2900 \text{ mL}$	$W_{b2} = 3 \times 10^{-2} \text{ mol L}^{-1}$
	$W_{b3} = 0$

segundos para o modelo. Para excitação do processo foi usado um sinal APRBS, projetado de forma a levar o sistema a todas as possíveis condições de operação, com pH variando entre 5 e 10. O conjunto de dados obtido foi dividido em três porções distintas, usadas para treinar, desenvolver e testar modelos. O primeiro conjunto é usado diretamente para obtenção dos parâmetros ajustáveis dos modelos. O conjunto de desenvolvimento é usado para verificar a capacidade dos modelos, sendo útil para a seleção de arquiteturas e hiperparâmetros no processo de obtenção do modelo final. O conjunto de teste é usado para a comparação entre os diversos modelos, sendo utilizado apenas após a seleção dos modelos finais.

Foram desenvolvidos dois modelos, com a finalidade de propiciar uma forma de comparação para as abordagens de identificação e controle propostas neste trabalho. Os modelos desenvolvidos consistem em um modelo linear de primeira ordem com atraso e um modelo não linear do tipo Hammerstein-Wiener. Para o modelo do tipo Hammerstein-Wiener foi selecionado um modelo de primeira ordem com atraso, para a parte dinâmica linear, uma função polinomial de segunda ordem para a parte não linear estática associada à entrada, e uma saturação associada à saída do modelo. Essas estruturas foram escolhidas experimentalmente, por apresentarem resultados melhores comparadas com modelos de ordens maiores. Os resultados obtidos por tais modelos são apresentados na Figura 40, na qual é apresentados os resultados obtidos pelos modelos para uma porção do conjunto de testes.

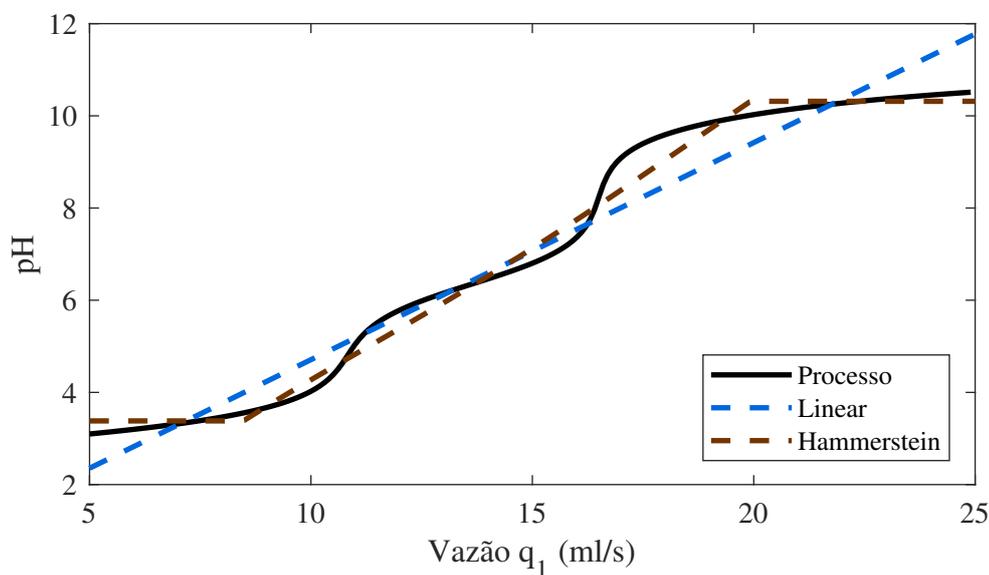
Para o problema de identificação e controle do reator de neutralização de pH, realizado por meio da simulação das equações fundamentais que regem o processo, é possível fazer uma análise aprofundada da qualidade dos modelos nas diversas regiões de operação. Para realizar essa análise, foi utilizada a curva característica estática, a qual apresenta o comportamento não linear estático do processo, representando o mapeamento entre as entradas e as saídas do processo. Essa curva é interessante para avaliar parte do comportamento do modelo para todos os possíveis pontos de operação, os quais podem não ser cobertos pela série de dados de teste. Na Figura 41 são apresentadas as curvas de característica estática do processo e as obtidas para os modelos linear e Hammerstein-Wiener.

Figura 40 – Resultados da identificação do processo de neutralização de pH para modelos linear (azul) e Hammerstein-Wiener (marrom).



Fonte: autor.

Figura 41 – Comparação entre as curvas de característica estática dos modelos linear (azul), Hammerstein-Wiener (marrom) e a curva real do processo de neutralização de pH (preto).



Fonte: autor.

A.2 BANCADA DE CONTROLE DE PRESSÕES

Compressores herméticos são dispositivos vastamente utilizados na indústria de refrigeração. Esse tipo de equipamento é submetido a diversos ensaios durante seu ciclo de desenvolvimento, bem como para garantia de conformidade das unidades produzidas em uma linha de produção. Diversos desses ensaios dependem do controle das pressões nas vias de sucção e de descarga do compressor, de forma a emular as condições de funcionamento do mesmo quando

aplicado a um sistema de refrigeração. Exemplos são os ensaios de desempenho energético de compressores, os quais são usados para determinar parâmetros como o consumo energético e a capacidade de refrigeração do compressor (ASHRAE, 2005).

O processo utilizado neste estudo é composto por duas válvulas localizadas nas vias de sucção e de descarga de fluido refrigerante do compressor de refrigeração. Entre as duas válvulas existe um tanque de armazenamento de fluido, usado para o desacoplamento entre as pressões de sucção e de descarga. Um diagrama de processo e instrumentos desta bancada é ilustrado na Figura 42.

O problema de controle consiste em manter as pressão na sucção, y_1 , e na descarga, y_2 , do compressor de refrigeração em patamares de referência. Tais variáveis de processo são manipuladas pela tensão aplicada nas válvulas u_1 e u_2 . Devido à natureza de operação do compressor hermético de refrigeração, existem alguns desafios para o controle em malha fechada das pressões associadas à sua sucção e descarga, destacando-se a não linearidade de y_2 e o forte acoplamento entre as duas pressões, em especial quando a variável u_1 é manipulada.

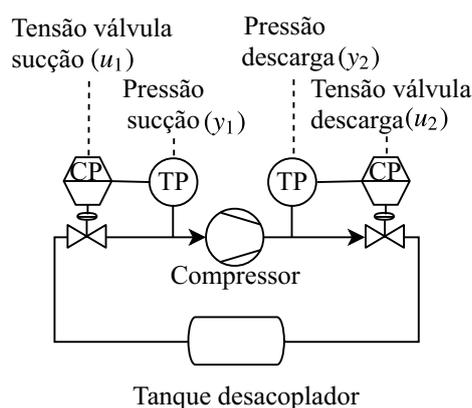


Figura 42 – Diagrama de processo e instrumentos de uma bancada de controle de pressão de compressores de refrigeração.

A.2.1 Identificação da bancada de controle de pressões

Neste trabalho foi empregada uma bancada experimental real, usada para o ensaio de compressores herméticos, localizada no Laboratório de Instrumentação e Automatização de Ensaio (LIAE). Em tal bancada foram realizados ensaios para obtenção da série de dados para identificação. Esses ensaios foram projetados considerando um sinal de excitação APRBS, o qual foi aplicado em momentos distintos para cada entrada do processo, considerando diversos pontos de operação para ambas as pressões. De forma semelhante ao realizado na obtenção da série de dados para o reator de neutralização de pH, as séries de dados obtidas nesse ensaio foram divididas em três porções distintas, usadas para treinar, desenvolver e testar modelos.

Para este estudo de caso, foram desenvolvidos, também, modelos lineares e não lineares do tipo Hammerstein-Wiener. Como o processo é MIMO, foram obtidos modelos para cada saída do processo, considerando como entradas ambas as tensões u_1 e u_2 . Para o modelo linear

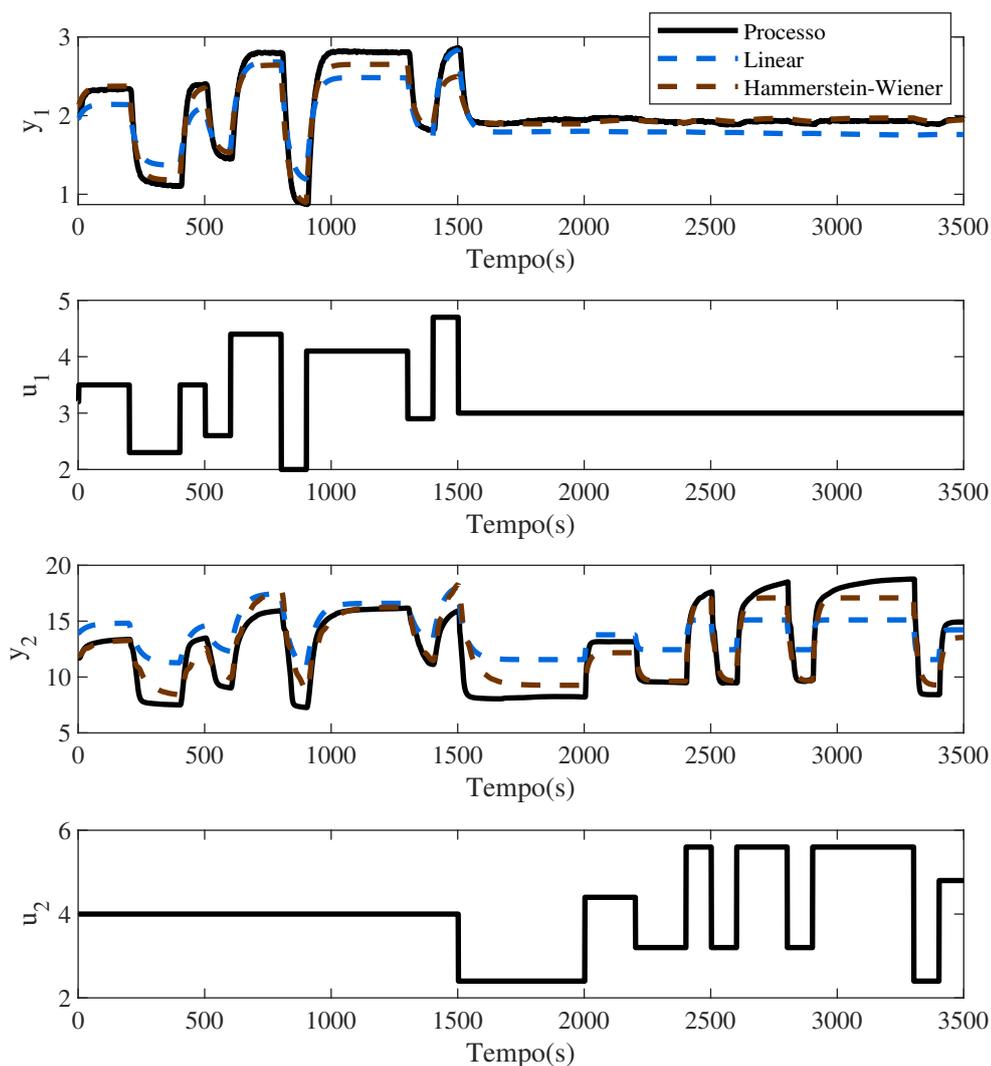


Figura 43 – Resultado da identificação da bancada de ensaio de compressores herméticos para modelos linear (azul) e Hammerstein-Wiener (marrom).

foi escolhida uma estrutura de primeira ordem com atraso discreta, para ambas as saídas. Já para o modelo não linear Hammerstein-Wiener foi escolhida para a porção linear dinâmica uma estrutura de segunda ordem, para a porção não linear estática associada à entrada uma função polinomial de quarta ordem, e para a porção não linear estática associada à saída foi usada a função saturação.

O resultado obtido por esses modelos é apresentado na Figura 43, na qual são apresentados os resultados obtido pelos modelos para apenas uma porção do conjunto de testes.

A.3 TANQUE DE PERFIL CÔNICO

O tanque de perfil cônico consiste em um processo no qual o nível do tanque, $h(t)$, é alterado pela manipulação de uma vazão de entrada, $q_1(t)$, considerando uma vazão de saída, $q_2(t)$, a qual despeja fluido por meio de uma restrição não constante. O objetivo do controle deste processo é a regulação do nível $h(t)$, manipulando $q_1(t)$ e considerando $q_2(t)$ uma perturbação

não medida. O processo é apresentado, por meio de uma representação esquemática, na Figura 44.

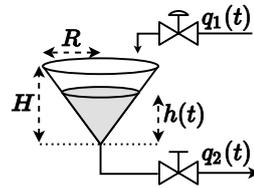


Figura 44 – Representação esquemática do sistema de controle de um tanque de perfil cônico.

O modelo fenomenológico do tanque de perfil cônico, em domínio de tempo contínuo, é descrito por meio da equação diferencial

$$\frac{dh(t)}{dt} = \frac{q_1 - v\sqrt{h(t)}}{\pi\left(\frac{Rh(t)}{H}\right)^2}, \quad (126)$$

a qual considera a vazão efluente como $q_2(t) = v\sqrt{h(t)}$, com v representando o ganho da válvula; H como o nível máximo do tanque; e R como o raio do tanque.

Foram considerados uma altura $H = 5$ m, um raio $R = 1$ m e um ganho da válvula de efluente $v = 0,1 \frac{\text{m}^3}{\text{min}\sqrt{\text{m}}}$. Assim como para os demais processos, a simulação do modelo fenomenológico foi utilizada como o processo real. A resolução da equação diferencial ordinária (126), que define o comportamento dinâmico do processo, foi realizada utilizando o método de integração numérica Runge-Kutta 45, sendo adicionado um ruído de $0,05 \text{ m}^2$ na equação de saída (125), de forma a emular ruído de medição.

A.4 MOTOR SÍNCRONO DE ÍMÃS PERMANENTES

Este processo consiste em um motor síncrono com ímãs permanentes, no qual o objetivo é o controle da velocidade angular do rotor e da corrente. O comportamento dinâmico das variáveis elétricas e mecânicas do PMSM em um referencial giratório pode ser descrito por um conjunto de equações diferenciais (MYNAR *et al.*, 2016). O comportamento elétrico é modelado por meio da expressão

$$\frac{di_d}{dt} = \frac{u_d}{L_d} + \frac{L_q}{L_d} P_p \omega_m i_q - \frac{R_s}{L_d} i_d, \quad (127a)$$

$$\frac{di_q}{dt} = \frac{u_q}{L_q} - \frac{L_d}{L_q} P_p \omega_m i_d - \frac{R_s}{L_q} i_q - \frac{\Psi_{PM}}{L_q} P_p \omega_m, \quad (127b)$$

na qual u_d e u_q representam a tensão do estator no referencial rotativo; i_d e i_q representam a corrente do estator no referencial rotativo; R_s representa a resistência de enrolamento; L_d e L_q representam os componentes de indutância; Ψ_{PM} representa a constante de força eletromotriz (EMF, do inglês *ElectroMotive Force*); P_p representa o número de par de polos; e ω_m representa a velocidade angular do rotor.

O comportamento mecânico é descrito pela expressão

$$\frac{d\omega_m}{dt} = \frac{1}{J}(T_{el} - T_l), \quad (128)$$

na qual J representa o momento de inércia; T_{el} representa o torque eletromagnético; e T_l representa o torque da carga. O torque eletromagnético produzido pelo PMSM é modelado pela expressão (MYNAR *et al.*, 2016)

$$T_{el} = \frac{3}{2}P_p[\Psi_{PM}i_q + (L_d - L_q)i_d i_q]. \quad (129)$$

O problema de controle consiste no seguimento de referências para ω_m , ao mesmo tempo que se busca a regulação da corrente i_d próxima de 0 A. As equações (127a), (127b) e (128) foram resolvidas utilizando o método de integração numérica Runge-Kutta 45. Os parâmetros considerados para a simulação do PMSM e a condição de operação nominal é apresentada na Tabela 9.

Tabela 9 – Condições de operação nominal do motor de indução com ímãs permanentes.

$R_s = 0,38 \cdot$	$L_d = L_q = 535 \mu\text{H}$
$\Psi_{PM} = 0,02594 \text{ Wb}$	$P_p = 3$
$J = 65 \times 10^{-6} \text{ kg m}^{-2}$	
$U_{\max} = 8,6 \text{ V}$	$I_{\max} = 2 \text{ A}$

APÊNDICE B – MÉTRICAS DE DESEMPENHO

Este apêndice apresenta uma compilação das métricas de desempenho utilizadas neste trabalho. Ele está dividido em duas seções: a seção B.1 apresenta as métricas de desempenho que são usadas neste trabalho para avaliar modelos, enquanto a seção B.2 apresenta as métricas de desempenho usadas neste trabalho para avaliar controladores.

B.1 MÉTRICAS DE DESEMPENHO PARA MODELOS

A qualidade de modelos obtidos por meio de identificação de sistemas pode ser avaliada de diferentes maneiras. De forma geral, são usadas métricas de desempenho usadas comparativamente em conjunto com métricas que apresentam interpretação estatística, as quais possibilitam avaliação da qualidade do modelo. Uma métrica útil para a comparação da qualidade de modelos é o erro quadrático médio (MSE, do inglês *Mean Squared Error*), o qual é calculado usando a equação

$$\text{MSE} = \frac{1}{n} \sum_{i=1}^n (y_{m,i} - y_i)^2, \quad (130)$$

na qual y_m representa a saída do modelo e y o valor que se assume como real.

Para avaliação da qualidade de cada modelo, uma métrica comumente usada é o coeficiente de determinação R^2 . Essa métrica apresenta, geralmente, valores entre 0 e 1, indicando quão bem os valores preditos se aproximam dos valores reais, com $R^2 = 1$ indicando que o modelo se ajusta perfeitamente aos dados. O coeficiente de determinação é obtido por meio da equação

$$R^2 = 1 - \frac{\sum_{i=1}^n (y_i - y_{m,i})^2}{\sum_{i=1}^n (y_i - \bar{y}_i)^2}. \quad (131)$$

A terceira métrica usada neste trabalho é o erro médio simétrico absoluto percentual (SMAPE, do inglês *Symmetric Mean Absolute Percentage Error*), que se baseia nos erros relativos, sendo aplicada na literatura para problemas de predição. Essa métrica é útil para a comparação de modelos por considerar o erro percentual, o qual pode ser interessante para avaliação de modelos que operam em diversas faixas de operação. A SMAPE é obtida por meio da equação

$$\text{SMAPE} = \frac{100\%}{n} \sum_{i=1}^n \frac{|y_{m,i} - y_i|}{|y_i| + |y_{m,i}|}. \quad (132)$$

B.2 MÉTRICAS DE DESEMPENHO PARA CONTROLADORES

Diversas métricas podem ser utilizadas para comparar quantitativamente os resultados em malha fechada de controladores. Uma comumente aplicada é a integral dos erros de seguimento de referência absolutos (IAE, do inglês *Integral of Absolute Error*). A IAE é calculada, em

tempo discreto, usando a equação

$$\text{IAE} = \sum_{k=0}^{\infty} |r(k) - y(k)|, \quad (133)$$

e é idealmente utilizada em sistemas SISO.

Para problemas em que o processo é MIMO e as variáveis de processo apresentam escalas distintas, a métrica de desempenho deve considerar o erro de forma proporcional, estabelecendo uma comparação correta entre os erros. Uma métrica que pode ser usada é a integral do erro de seguimento de referência percentual médio (IAPE, do inglês *Integral of the Average Percentual Errors*). Essa métrica considera de forma similar os erros de todas as variáveis, sendo calculada, em tempo discreto, por meio da equação

$$\text{IAPE} = \sum_{k=0}^{\infty} \frac{|r(k) - y(k)|}{r(k)}. \quad (134)$$

A IAPE é calculado para cada saída e pode ser consolidado em uma única métrica, a média da IAPE (MIAPE, do inglês *Mean IAPE*), através da média das m saídas com a equação

$$\text{MIAPE} = \sum_{i=1}^m \sum_{k=0}^{\infty} \frac{|r_i(k) - y_i(k)|}{r_i(k)}. \quad (135)$$

Outra métrica que pode ser utilizada para comparação quantitativa de controladores consiste na integral dos erros absolutos ponderados no tempo (ITAE, do inglês *Integral of Time Absolute Error*). Essa métrica pondera o erro absoluto considerando a distância entre cada saída do processo e a última mudança de referência, ou inserção de uma perturbação. A métrica ITAE é obtida por meio da expressão

$$\text{ITAE} = \sum_{k=0}^{\infty} |k(r(k) - y(k))|. \quad (136)$$