

Algoritmo de Controle Preditivo com Predição Iterativa Não Linear e Linearização para Modelos Hammerstein-Wiener

Bernardo B. Schwedersky* Rodolfo C. C. Flesch**
Hiago A. S. Dangui*

* Programa de Pós-Graduação em Automação e Sistemas,
Universidade Federal de Santa Catarina, Florianópolis, Brazil (e-mail:
[bernardo.schwedersky, hiagodangui@gmail.com]).

** Departamento de Automação e Sistemas, Universidade Federal de
Santa Catarina, Florianópolis, Brazil (e-mail: rodolfo.flesch@ufsc.br)

Abstract: This paper proposes a model predictive control algorithm with iterative nonlinear prediction and linearization, using a Hammerstein-Wiener model. In the proposed algorithm, the Hammerstein-Wiener model is used to obtain a base response, which depends on the past and current inputs, while a linearized version of the model is used to obtain the optimal response, which depends only on current and future inputs. Future control efforts are obtained by solving a quadratic optimization problem. Instead of applying the first obtained control effort, an iterative procedure is performed, in which the base input is updated with the control efforts calculated in the previous iteration, until the term corresponding to the base response is large enough compared to the term of the optimal response, reducing the performance deterioration due to linearization. The proposed algorithm was evaluated with the nonlinear model of a pH neutralization process, and the results were used to show the impact of the iterative procedure on closed-loop control performance and on computational effort. The results show that the iterative procedure improves closed-loop performance, while keeping the computational cost acceptable.

Resumo: Esse artigo propõe um algoritmo de controle preditivo com predição iterativa não linear e linearização, usando um modelo do tipo Hammerstein-Wiener. No algoritmo proposto, o modelo Hammerstein-Wiener é usado para obter uma resposta base, o qual depende das entradas passadas e atuais, enquanto uma versão linearizada do modelo é usada para obter a resposta ótima, a qual depende apenas de entradas atuais e futuras. Os esforços de controle futuros são obtidos a partir da solução de um problema de otimização quadrático. Ao invés de aplicar primeiro esforço de controle calculado, um procedimento iterativo é executado, no qual a entrada base é atualizada com os esforços de controle calculados na iteração anterior, até que o termo correspondente à resposta base seja grande o suficiente em comparação ao termo da resposta ótima, reduzindo a influência da deterioração do desempenho decorrente da linearização. O algoritmo proposto foi avaliado com um modelo não linear de um processo de neutralização de pH, e os resultados foram utilizados para mostrar o impacto do procedimento iterativo no desempenho em malha fechada, bem como no esforço computacional. Os resultados obtidos neste estudo mostram que o procedimento iterativo melhora o desempenho em malha fechada, mantendo o custo computacional aceitável.

Keywords: nonlinear model predictive control; Hammerstein-Wiener; nonlinear model; iterative nonlinear prediction; MPC-INPL.

Palavras-chaves: controle preditivo baseado em modelo não linear; Hammerstein-Wiener; modelo não linear; predição não linear iterativa; MPC-INPL.

1. INTRODUÇÃO

Controle preditivo baseado em modelo (MPC, do inglês, *Model Predictive Control*) é uma classe de algoritmos de controle amplamente aplicada na indústria, consistindo na técnica mais empregada em problemas de controle com restrições (Lee, 2011). O interesse no uso de formulações não lineares de MPC vem crescendo, devido ao fato de que as dinâmicas frequentemente encontradas em processos industriais são melhor modeladas utilizando representações

não lineares. Porém, a falta de métodos para obtenção de modelos não lineares com alta fidelidade ainda limita a adoção de tais técnicas (Forbes et al., 2015; Schoukens and Ljung, 2019).

Formulações que utilizam representações baseadas em modelos fenomenológicos não lineares geralmente são empregadas, porém a obtenção de tais representações pode ser custosa devido à complexidade dos processos envolvidos (Foss and Schei, 2007). Outra alternativa é o emprego de

estratégias baseadas em identificação de sistemas. Uma classe de modelos, comumente utilizada para obtenção de modelos não lineares identificados é conhecida como modelos Hammerstein-Wiener (HW). Essa classe é caracterizada por blocos não lineares estáticos associados com blocos lineares dinâmicos (Wills et al., 2013; Quachio and Garcia, 2019; Xu et al., 2019). Algoritmos MPC não lineares (NMPC, do inglês *Nonlinear Model Predictive Control*) que utilizam modelos HW para a previsão do comportamento futuro do processo são explorados na literatura, requerendo a solução *online* de algoritmos de programação não linear (Bloemen et al., 2001; Wellers and Rake, 2000).

Uma alternativa para implementação de controladores NMPC é por meio de algoritmos práticos (ou computacionalmente eficientes), nos quais a solução de um problema de programação não linear é evitada. Tais estratégias se baseiam na separação do modelo de previsão em duas partes, sendo que a parte dependente de ações de controle futuras é linearizada, o que faz com que o problema de otimização se torne quadrático. Esse é um conceito geral, compartilhado por vários algoritmos NMPC, como por exemplo o algoritmo NMPC prático (PNMPC, do inglês *Practical NMPC*), introduzido por Plucenio et al. (2007), e o algoritmo MPC com previsão não linear e linearização (MPC-NPL, do inglês *MPC with Nonlinear Prediction and Linearization*), apresentado por Lawryńczuk (2014). Tal estratégia vem sendo aplicada para algoritmos NMPC baseados em modelos Hammerstein-Wiener, como ns propostos por Lawryńczuk (2015) e Zhang et al. (2018), que apresentaram formulações MPC-NPL para modelos HW, e na formulação PNMPC Hammerstein (PNMPC-H), apresentada por Schwedersky et al. (2018), a qual tratou apenas o caso específico de modelos Hammerstein.

Uma característica comum em tais implementações é a separação do modelo de previsão em dois termos, o que é uma representação imprecisa do comportamento futuro do processo, já que o princípio da superposição não é verificado para sistemas não lineares. Uma alternativa para mitigar tal problema foi apresentada por De Keyser (2009), fazendo uso de um procedimento iterativo para encontrar a melhor ação de controle. Essa proposta considera a previsão dividida em um termo que depende apenas das ações de controle passadas e atual, denominada resposta base, e um termo que depende apenas das ações de controle futuras, denominada resposta ótima. A estratégia consiste em realizar um procedimento iterativo para encontrar a melhor ação de controle base, fazendo com que o termo correspondente à resposta ótima seja suficientemente menor que o termo referente à resposta base, o qual considera o modelo não linear. Com tal procedimento, o modelo de previsão do controlador é capaz de melhor aproveitar a característica não linear do modelo. Esse procedimento apresenta potencial para para aprimorar o resultado de implementações de NMPC práticas, pois permite que as não linearidades do modelo sejam melhor consideradas no cálculo ação de controle.

Este artigo propõe uma formulação MPC com previsão iterativa não linear e linearização (MPC-INPL, do inglês *MPC with Iterative Nonlinear Prediction and Linearization*) para modelos do tipo Hammerstein-Wiener. Na estratégia de controle proposta, a cada período de amostragem, o modelo não linear HW é usado para obter uma

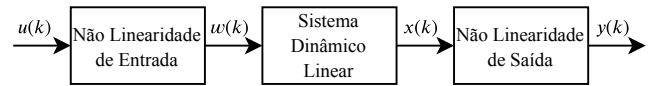


Figura 1. Diagrama de blocos de um modelo não linear do tipo Hammerstein-Wiener.

resposta base do comportamento futuro do processo. Adicionalmente, uma versão linearizada localmente do modelo HW é obtida a cada iteração do algoritmo, sendo utilizada para construir a matriz de dinâmica do sistema, de forma a resolver um problema de otimização quadrático e encontrar o esforço de controle futuro. Ao invés de ser aplicado diretamente no processo, o esforço de controle calculado é usado para encontrar o melhor sinal de controle base, por meio de um procedimento iterativo, o qual, após convergência, retorna o sinal de controle a ser aplicado no processo.

A formulação proposta expande algoritmo NMPC prático para modelos Hammerstein (PNMPC-H, do inglês *Practical NMPC for Hammerstein models*) apresentada por Schwedersky et al. (2018), adicionando o procedimento iterativo que atualiza o sinal de controle base, fazendo com que o algoritmo PNMPC-H possa ser compreendido como um caso especial do algoritmo proposto, no qual o número de iterações máximo para obtenção da resposta base é 1. O algoritmo de controle proposto é avaliado utilizando o modelo de um processo de neutralização de pH, e os resultados são comparados com os obtidos pelo algoritmo PNMPC-H e uma implementação tradicional de MPC linear, o Controle Preditivo Generalizado (GPC, do inglês *Generalized Predictive Control*), proposta por Clarke et al. (1987).

O presente artigo é dividido da seguinte forma. A estrutura dos modelos Hammerstein-Wiener é apresentada na seção 2. A formulação geral do problema de controle preditivo baseado em modelo é introduzida na seção 3. Na seção 4, a formulação do algoritmo MPC-INPL para modelos Hammerstein-Wiener proposto é apresentada. A avaliação de tal algoritmo é apresentada na seção 5, na qual os resultados para o controle em malha fechada do processo de neutralização de pH são apresentados. Por fim, as conclusões do trabalho são apresentadas na seção 6.

2. MODELOS HAMMERSTEIN-WIENER

Hammerstein-Wiener é uma classe de modelos não lineares, construídos com uma estrutura modular, na qual não linearidades estáticas são associadas à entrada e à saída de um sistema dinâmico linear. A estrutura geral do modelo HW é ilustrada na Figura 1, na qual o bloco dinâmico linear é conectado a blocos não lineares estáticos, tanto em sua entrada quanto em sua saída. Caso o bloco não linear da saída seja desconsiderado, o modelo é denominado Hammerstein, e, por sua vez, se o bloco não linear de entrada for desconsiderado, o modelo é denominado Wiener (Eskinat et al., 1991).

A não linearidade estática de entrada é definida pela função não linear $f(\cdot)$, enquanto a de saída é definida por $g(\cdot)$. Por sua vez, o bloco dinâmico linear é definido como uma função de transferência linear. As equações que definem o modelo Hammerstein-Wiener, em tempo discreto, são:

$$w(k) = f(u(k)), \quad (1)$$

$$x(k) = \frac{B(z^{-1})}{A(z^{-1})}w(k), \quad (2)$$

$$y(k) = g(x(k)), \quad (3)$$

em que $u(k)$ e $y(k)$ representam entrada e saída, respectivamente; $A(z^{-1})$ e $B(z^{-1})$ representam polinômios em z^{-1} ; $x(k)$ e $w(k)$ representam variáveis internas.

As funções não lineares comumente usadas são saturação, zona morta, funções sigmóides, funções lineares por partes e funções polinomiais. As polinomiais são as mais utilizadas, principalmente no bloco de entrada, sendo particularmente úteis quando a relação não linear entrada-saída envolve potências e produtos das variáveis de entrada. A formulação da função polinomial $f(u(k)) = P(u(k))$ é:

$$P(u(k)) = c_n u^n(k) + c_{n-1} u^{n-1}(k) + \dots + c_1 u(k) + c_0, \quad (4)$$

na qual c_i , $i = 0 \dots n$, representam os coeficientes do polinômio e n representa o grau do polinômio. Para um dado conjunto de dados entrada-saída, os parâmetros de um modelo HW podem ser estimados por meio de métodos iterativos, por exemplo os algoritmos de Gauss-Newton e Levenberg-Marquardt. Detalhes acerca dos métodos iterativos para solução do problema de mínimos quadrados não linear são apresentados por Golub and Pereyra (2003). Por sua vez, a obtenção de modelos HW adequados a abordagens MPC é detalhada por Quachio and Garcia (2019).

3. FORMULAÇÃO DO PROBLEMA MPC

No paradigma de controle preditivo baseado em modelo (MPC), um modelo explícito do processo é utilizado para prever seu comportamento durante um horizonte futuro, de forma a encontrar uma sequência de ações de controle futuras ótimas. Algoritmos MPC podem ser divididos, geralmente, em dois blocos principais, o modelo de predição e o otimizador. Usando o modelo, as saídas futuras do processo são previstas. Por sua vez, o otimizador lida com o problema de otimização para determinar a sequência ótima de ações futuras. A sequência de predições, tomadas no instante k , para um horizonte de predição $i = N_1 \dots N_2$, é representada como $\hat{y}(k+i | k)$. A sequência de ações ótimas de controle, $u(k+j)$, é determinada para um horizonte de controle $j = 0 \dots N_u - 1$, de forma a minimizar um funcional do erro de seguimento de referência predito e do esforço de controle. Apenas a primeira ação de controle é aplicada ao sistema, antes que o algoritmo avance para a próxima iteração.

Uma expressão para o funcional de custo, o qual é usado em vários algoritmos MPC, como por exemplo o GPC, consiste em:

$$J = \sum_{i=1}^m \sum_{j=N_{1,i}}^{N_{2,i}} \delta_i(j) [\hat{y}_i(k+j | k) - r_i(k+j | k)]^2 + \sum_{i=1}^n \sum_{j=1}^{N_{u,i}} \lambda_i(j) [\Delta u_i(k+j-1)]^2, \quad (5)$$

na qual m e n representam a dimensão das saídas e entradas do sistema; N_1 e N_2 representam o início e o

fim do horizonte de predição, com $N = N_2 - N_1 + 1$; N_u representa o horizonte de controle; $\hat{y}_i(k+j | k)$ representa a trajetória predita da saída i do processo, tomada no instante k ; $r_i(k+j | k)$ representa a sequência de referências futuras, conhecidas no instante k , para a saída i ; e $\Delta u_i(k+j-1)$ representa incrementos de controle, associados à entrada i .

As sequências de ponderações para o erro de seguimento de referência e esforço de controle são definidas pelos coeficientes $\delta_i(j)$ e $\lambda_i(j)$. Valores maiores para a razão $\delta_i(j)/\lambda_i(j)$ priorizam a velocidade das transições entre referências, enquanto razões menores penalizam a variação do sinal de controle, o que resulta em ações de controle mais suaves (Camacho and Bordons, 2007).

O sinal de controle futuro, $u(k)$, é obtido ao minimizar o funcional de custo (5). O problema de otimização é definido como:

$$\begin{aligned} & \min_{\Delta \mathbf{u}} J \\ & \text{sujeito a } \mathbf{A} \Delta \mathbf{u} \leq \mathbf{b}, \end{aligned} \quad (6)$$

no qual a matriz \mathbf{A} e o vetor \mathbf{b} definem as restrições para a tarefa de otimização, na forma de um politopo convexo em $\Delta \mathbf{u}$. Detalhes sobre as restrições tipicamente utilizadas, bem como a estrutura de \mathbf{A} e \mathbf{b} são detalhadas por Camacho and Bordons (2007).

4. CONTROLADOR PROPOSTO

Nesta seção, a formulação de um MPC com predição iterativa não linear e linearização para modelos HW (MPC-INPL-HW) é proposta. Essa formulação estende a formulação originalmente proposta por Schwedersky et al. (2018), utilizando um procedimento iterativo proposto no algoritmo introduzido por De Keyser (2009).

Considerando um processo identificado por meio de um modelo HW, a predição da saída pode ser representada pela soma de duas partes, a saída predita do modelo $\mathbf{y}_p(k)$, obtida da saída do modelo HW, e um termo referente à perturbação, $\mathbf{n}(k)$, o qual representa o erro entre o processo e o modelo. O termo referente à perturbação pode ser descrito como:

$$\mathbf{n}(k) = \mathbf{y}(k) - \mathbf{y}_p(k). \quad (7)$$

O preditor da saída do processo é obtido de forma semelhante ao apresentado no algoritmo NEPSAC (do inglês, *Nonlinear Extended Prediction Self-Adaptive Control*) (De Keyser, 2009). O modelo não linear é usado para prever a resposta base do sistema, a qual depende apenas das ações de controle passadas e de uma estimativa da ação de controle atual, enquanto uma versão linearizada do modelo é usada para obter a parte ótima, a qual depende das ação atual e futuras de controle. Com tal estratégia, é possível construir o problema de otimização, definido em (6), de forma a ser possível o uso de algoritmos de programação quadrática, já que a predição da saída do processo se torna afim em relação aos incrementos de controle. A predição da saída i de um sistema, em relação à entrada j , pode ser escrita como:

$$\begin{aligned} \hat{\mathbf{y}}_{ij}(k) &= \mathbf{y}_{\text{opt},ij}^p(k) + \mathbf{y}_{\text{base},i}^p(k) \\ &= \mathbf{G}_{ij}^p(k) \Delta \mathbf{u}_j(k) + \mathbf{y}_{\text{base},i}^p(k), \end{aligned} \quad (8)$$

na qual o vetor de predições da saída e o vetor de incrementos de controle são, respectivamente

$$\hat{\mathbf{y}}_{ij}(k) = [\hat{\mathbf{y}}_{ij}(k + N_{1,i}) \dots \hat{\mathbf{y}}_{ij}(k + N_{2,i})]^T \quad (9)$$

e

$$\Delta \mathbf{u}_j(k) = [\Delta \mathbf{u}_j(k) \dots \Delta \mathbf{u}_j(k + N_{u,j} - 1)]^T \quad (10)$$

e p representa o número da iteração. O vetor de resposta base $\mathbf{y}_{\text{base},i}^p(k)$ é obtido a partir da trajetória não linear do modelo considerando como sua entrada

$$\mathbf{u}_{\text{base},j}^p(k) = \mathbf{u}_{\text{base},j}^{p-1}(k) + \Delta \mathbf{u}_j^{p-1}(k), \quad (11)$$

na qual $\mathbf{u}_{\text{base},j}^1(k) = \mathbf{u}_j(k - 1)$.

Devido ao modelo ser não linear, a predição em (8) é uma aproximação da resposta do modelo, já que o princípio da superposição não é verificado neste caso. Essa aproximação é válida quando o termo $\mathbf{y}_{\text{opt},i}^p$ é suficientemente pequeno, comparado com o termo $\mathbf{y}_{\text{base},j}^p$. A estratégia utilizada para alcançar tal condição consiste em computar iterativamente $\Delta \mathbf{u}_j(k)$, atualizando $\mathbf{u}_{\text{base},j}^p$, até que $\Delta \mathbf{u}_j^p(k) \approx 0$, o que resulta no termo $\mathbf{y}_{\text{opt},i}^p \approx 0$. Dessa forma, o algoritmo é repetido enquanto $\|\Delta \mathbf{u}_j^p(k)\| \geq \epsilon$ e o número de iterações $p \leq i_{\text{max}}$. Quando $i_{\text{max}} = 1$, e o modelo empregado é Hammerstein, o algoritmo é equivalente à formulação original do algoritmo PNMPC-Hammerstein, descrita por Schwedersky et al. (2018).

Para obter $\Delta \mathbf{u}_j^p$, a matriz de dinâmica $\mathbf{G}_{ij}^p(k) \in \mathbb{R}^{N_i \times N_{u,j}}$ é construída com os coeficientes da resposta ao degrau, obtidos a partir da simulação do modelo não linear considerando como entrada um degrau em volta da entrada atual. A matriz $\mathbf{G}_{ij}^p(k)$ é descrita como:

$$\mathbf{G}_{ij}^p(k) = \begin{bmatrix} s_{11}(k) & 0 & \dots & 0 \\ s_{21}(k) & s_{12}(k) & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ s_{N1}(k) & s_{(N-1)2}(k) & \dots & s_{(N-N_u+1)(N_u)}(k) \end{bmatrix}, \quad (12)$$

na qual o vetor $[s_{11} \dots s_{N1}]^T$ corresponde aos primeiros N coeficientes da resposta ao degrau, após a aplicação de um degrau em volta de $\mathbf{u}_{\text{base},j}^p$, com amplitude $\Delta \mathbf{u}_j^p/10$, no instante k . As demais colunas podem ser obtidas considerando o mesmo procedimento, aplicando degraus em volta de $\mathbf{u}_{\text{base},j}^p$ nos instantes $k + 1$ a $k + N_u - 1$.

No caso com múltiplas entradas e saídas (MIMO, do inglês *Multiple Input Multiple Output*), a resposta base é definida como:

$$\mathbf{y}_{\text{base}}^p(k) = \begin{bmatrix} \mathbf{y}_{\text{base},1}^p(k) \\ \vdots \\ \mathbf{y}_{\text{base},m}^p(k) \end{bmatrix}, \quad (13)$$

na qual cada bloco $\mathbf{y}_{\text{base},i}^p(k)$ representa a resposta base correspondente à saída i , para $i \in [1, m]$.

A matriz dinâmica $\mathbf{G}^p(k)$ é definida, para o caso MIMO, como:

$$\mathbf{G}^p(k) = \begin{bmatrix} \mathbf{G}_{11}^p(k) & \dots & \mathbf{G}_{1n}^p(k) \\ \vdots & \ddots & \vdots \\ \mathbf{G}_{m1}^p(k) & \dots & \mathbf{G}_{mn}^p(k) \end{bmatrix}, \quad (14)$$

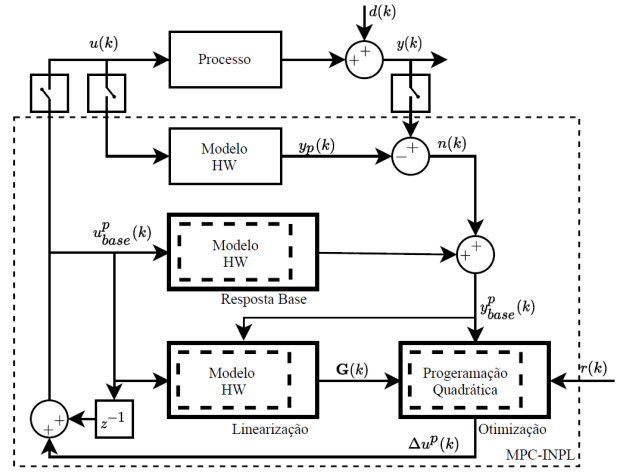


Figura 2. Diagrama de blocos do algoritmo MPC-INPL para modelos HW. O procedimento dentro do bloco tracejado é computado iterativamente enquanto $\|\Delta \mathbf{u}^p\| \geq \epsilon$ e $p \leq i_{\text{max}}$.

na qual cada bloco $\mathbf{G}_{ij}^p(k)$ representa a matriz \mathbf{G} obtida para o caso monovariável (SISO, do inglês *Single Input Single Output*), para o par saída i e entrada j , considerando $i \in [1, m]$ e $j \in [1, n]$.

Em cada iteração, o funcional custo, descrito na equação (5), é construído usando o vetor de predições, $\hat{\mathbf{y}}(k)$, e o vetor de incrementos de controle, $\Delta \mathbf{u}(k)$. Devido à matriz \mathbf{G} representar uma aproximação linear local do modelo não linear, o problema de otimização definido em (6) é quadrático, com o caso sem restrições apresentando solução analítica, e o caso com restrições sendo solucionado utilizando algoritmos de programação quadrática. A solução do problema de otimização retorna o vetor $\Delta \mathbf{u}_j(k)$, sendo que o primeiro elemento de cada vetor é usado para atualizar a resposta base da iteração futura, até que a condição de parada seja satisfeita. Quando satisfeita, o sinal de controle futuro é definido como

$$\mathbf{u}_j(k) = \mathbf{u}_j(k - 1) + \sum_{s=1}^p \Delta \mathbf{u}_j^s(k). \quad (15)$$

Um diagrama de blocos da estratégia de controle proposta é apresentado na Figura 2. O algoritmo de controle é sumarizado no Algoritmo 1.

5. RESULTADOS DE SIMULAÇÃO

Esta seção apresenta a avaliação do algoritmo MPC-INPL-HW, por meio da simulação de um reator de neutralização de pH. O modelo fenomenológico do reator de neutralização de pH é apresentado na seção 5.1. A identificação de um modelo Hammerstein-Wiener é apresentada na seção 5.2. Por fim, o desenvolvimento do controlador proposto e os resultados em malha fechada, são apresentados na seção 5.3.

5.1 Reator de neutralização de pH

Na literatura, o controle de um reator de neutralização de pH é uma aplicação comumente abordada para estudo e avaliação de técnicas de controle não linear. A modelagem

Algoritmo 1 Algoritmo MPC-INPL para modelos HW

- 1: Medição da saída atual 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 a resposta base SISO $\mathbf{y}_{\text{base},i}^p(k)$ da simulação do modelo HW, para $\mathbf{u}_{\text{base},i}^p(k)$, como descrito em (11).
 - 6: **Fim Para**
 - 7: Construir a resposta base MIMO concatenando o vetor $\mathbf{y}_{\text{base}}^p(k)$ como em (13).
 - 8: **Para** $i = 1 : m$ **Faça**
 - 9: **Para** $j = 1 : n$ **Faça**
 - 10: Linearizar o modelo HW aplicando um degrau em volta da $\mathbf{u}_{\text{base},j}^p(k)$, obtendo a matriz de dinâmica SISO $\mathbf{G}_{ij}^p(k)$, descrita em (12) com os coeficientes da resposta ao degrau obtidos da simulação do modelo HW.
 - 11: **Fim Para**
 - 12: **Fim Para**
 - 13: Construir a matriz de dinâmica MIMO $\mathbf{G}^p(k)$ como em (14).
 - 14: Solucionar o problema QP descrito em (6) para encontrar o incremento de controle futuro $\Delta\mathbf{u}^p$.
 - 15: Atualizar $\mathbf{u}_{\text{base}}^p = \mathbf{u}_{\text{base}}^{p-1} + \Delta\mathbf{u}^p(k)$.
 - 16: Atualizar $p = p + 1$
 - 17: **Fim Enquanto**
 - 18: Aplicar no processo a soma de todos os primeiros elementos do vetor de incrementos de controle, como descrito em (15).
 - 19: Retornar para a etapa 1 com $k = k + 1$.
-

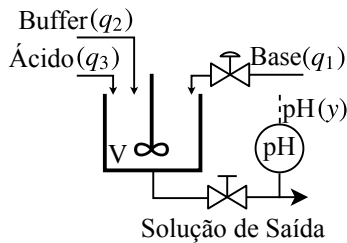


Figura 3. Diagrama de processo e instrumentos de um processo de neutralização de pH.

desse processo, bem como o seu controle em malha fechada, foi explorado em diversos trabalhos, como os de Henson and Seborg (1994), Gomez et al. (2004), Mahmoodi et al. (2009) e Ławryńczuk (2013). Este 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, considerando que apenas ácidos fortes e bases fortes estão presentes (Gomez et al., 2004).

O processo de neutralização de pH consiste na reação entre uma base q_1 (NaOH), um *buffer* q_2 (NaHCO₃), e um ácido q_3 (HNO₃). Essa reação é realizada em um tanque de volume constante com mistura contínua. O

modelo fenomenológico no domínio do tempo contínuo desse reator é descrito por duas equações diferenciais ordinárias, definidas como:

$$\frac{dW_a(t)}{dt} = \frac{q_1(t)(W_{a_1} - W_a(t))}{V} + \frac{q_2(t)(W_{a_2} - W_a(t))}{V} + \frac{q_3(t)(W_{a_3} - W_a(t))}{V} \quad (16)$$

$$\frac{dW_b(t)}{dt} = \frac{q_1(t)(W_{b_1} - W_b(t))}{V} + \frac{q_2(t)(W_{b_2} - W_b(t))}{V} + \frac{q_3(t)(W_{b_3} - W_b(t))}{V} \quad (17)$$

sendo que W representa concentrações. A saída do processo, pH, é representada por y , sendo definida pela seguinte equação algébrica:

$$W_a(t) + 10^{y(t)-14} - 10^{-y(t)} + W_b(t) \frac{1 + 2 \times 10^{y(t)-K_2}}{1 + 10^{K_1-y(t)} + 10^{y(t)-K_2}} = 0. \quad (18)$$

Os parâmetros do modelo fenomenológico são definidos na Tabela 1. A saída do processo, y , é manipulada pela vazão q_1 (ml/s), assumindo que as vazões q_2 e q_3 são mantidas constantes. Os valores iniciais para a simulação da operação do processo são apresentados na Tabela 2.

5.2 Identificação do processo de neutralização de pH

O modelo fenomenológico do reator de neutralização de pH foi utilizado para o desenvolvimento de modelos identificados a partir de dados e para o teste em malha fechada dos controladores. Para a resolução das equações diferenciais ordinárias, definidas em (17), foi utilizado o método de integração numérica Runge-Kutta 45, sendo adicionado um ruído branco com desvio padrão 0,02 na equação de saída (18), de forma a emular o ruído de medição.

Uma simulação em malha aberta foi realizada, com o objetivo de obter um conjunto de dados para a identificação

Tabela 1. Parâmetros do modelo fenomenológico do reator de neutralização de pH.

Parâmetro	Valor
K_1	6,35
K_2	10,25
V	2900 ml
W_{a_1}	$-3,05 \times 10^{-3}$ mol
W_{a_2}	-3×10^{-2} mol
W_{a_3}	3×10^{-3} mol
W_{b_1}	5×10^{-5} mol
W_{b_2}	3×10^{-2} mol
W_{b_3}	0

Tabela 2. Condições iniciais de operação para o modelo fenomenológico do reator de neutralização de pH.

Variável	Valor Inicial
q_1	15,55 ml/s
q_2	0,55 ml/s
q_3	16,60 ml/s
W_a	$-4,32 \times 10^{-4}$ mol
W_b	$5,28 \times 10^{-4}$ mol

de sistemas. Foi considerado um período de amostragem de 10 segundos para o modelo. Para excitação do processo foi usado um sinal APRBS, projetado de forma a levar o sistema a todas as condições de operação de interesse, 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.

A partir do conhecimento acerca da dinâmica geral do processo a ser identificado, foram avaliadas arquiteturas de modelos HW com saturação como não linearidade de saída, e polinômios de diferentes graus como não linearidade de entrada. Os parâmetros dos modelos sob análise foram estimados considerando o conjunto de treinamento, e tiveram seu desempenho avaliado no conjunto de testes. Como métricas de desempenho, foram utilizados o erro médio quadrático (EQM) e o coeficiente de determinação (R^2).

De todas as configurações avaliadas, foi escolhida para o modelo final, utilizado no projeto de controle, a configuração que associou um boa precisão no ajuste com uma baixa complexidade. O modelo escolhido consistiu em uma estrutura com $B(z^{-1})$ de grau 1 e $A(z^{-1})$ representando uma constante, resultando em um modelo linear de primeira ordem, associada a um polinômio de grau 5 como não linearidade $f(\cdot)$ e a uma saturação da saída, como não linearidade $g(\cdot)$. Esse modelo apresentou um EQM de 0,13 e $R^2 = 0,98$. Para fins de comparação, se utilizada apenas a porção linear, o EQM foi 0,38 e $R^2 = 0,93$.

5.3 Controle MPC-INPL-HW

O controle em malha fechada do processo de neutralização de pH foi projetado considerando o modelo Hammerstein-Wiener identificado na seção 5.2. O cenário desenvolvido para teste dos controladores consistiu em um ensaio com duas trocas de referência, considerando uma perturbação de carga aplicada em cada uma das referências. A perturbação de carga foi aplicada na vazão de ácido q_2 , por meio de degraus com amplitude $+0,5$ ml/s e $-0,5$ ml/s. A avaliação do desempenho em malha fechada dos controladores foi realizada por meio da integral dos erros absolutos (IEA).

Para a sintonia do controlador, foi aplicado um procedimento de sintonia baseado nas características em malha aberta do processo, semelhante ao apresentado por Danguì et al. (2018). Foi utilizado um horizonte de predição que considerasse a totalidade da resposta em malha aberta do sistema, correspondendo a $N = 50$, no qual $N_1 = 1$ e $N_2 = 50$. Como horizonte de controle, foi escolhido $N_u = 5$. Os parâmetros que ponderam os termos da função custo foram escolhidos de forma a equilibrar a importância de cada termo na função objetivo, resultando em $\lambda = 1$ e $\delta = 2$.

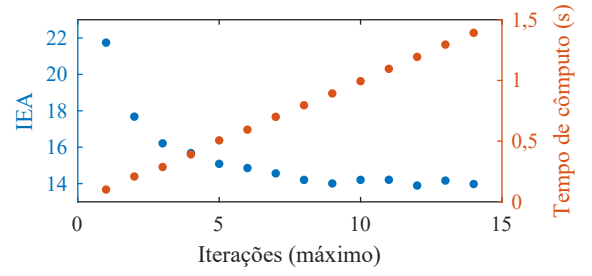


Figura 4. Impacto do número de iterações no valor final da IEA e no tempo máximo de cômputo.

O procedimento iterativo, usado para encontrar o melhor sinal \mathbf{u}_{base} , foi ajustado ao sintonizar ϵ e i_{max} . Ambos os parâmetros são relacionados, já que valores pequenos para ϵ resultam em um número de iterações elevado. Uma boa prática para ajustar ambos os valores consiste em selecionar um valor pequeno para ϵ , e um valor i_{max} que mantenha o custo computacional dentro do limite aceitável, já que i_{max} limita a quantidade máxima de iterações do algoritmo. Para ilustrar tal comportamento, foram realizadas simulações do cenário de teste considerando sintonias com $\epsilon = 0,1$ e $i_{\text{max}} \in [1; 15]$. O IEA final de cada simulação, bem como o tempo máximo de computo de cada iteração, ambos em função de i_{max} , é apresentado na Figura 4.

O tempo máximo de uma iteração é aproximadamente 100 ms, sendo maior do que os 40 ms requeridos para uma iteração do GPC, devido ao custo adicional para obter o modelo linearizado. É observado que o IEA é menor quanto maior for i_{max} , porém o custo computacional máximo cresce linearmente. Para este caso, $i_{\text{max}} = 5$ apresenta o melhor compromisso entre o desempenho em malha fechada do controlador e o custo computacional.

O algoritmo proposto foi avaliado no cenário de teste, considerando as sintonias com $i_{\text{max}} = 5$ e $i_{\text{max}} = 1$, sendo que o caso no qual $i_{\text{max}} = 1$ equivale ao algoritmo PNMPC-H, apresentado por Schwedersky et al. (2018), porém considerando um modelo HW. De forma a fornecer uma base para comparação, um controlador GPC foi desenvolvido, considerando a mesma sintonia e utilizando o polinômio $T = \frac{1}{1-0,4z^{-1}}$. Os resultados em malha fechada para o algoritmo proposto e os métodos de comparação são apresentados na Figura 5.

Os controladores não lineares apresentaram desempenho em malha fechada superior ao apresentado pela alternativa linear, o que é o esperado, já que o modelo não linear representa de forma mais precisa a dinâmica do processo. O resultado alcançado pelo algoritmo PNMPC-H é melhor que para o GPC, com menor sobressinal e tempo de acomodação. O resultado alcançado pelo algoritmo proposto, por sua vez, é claramente melhor que o alcançado por ambos os algoritmos usados como base de comparação. Uma análise do número de iterações realizadas pelo algoritmo proposto evidencia que são realizadas mais iterações em situações nas quais mudanças de referência ou perturbações são inseridas. Esses são os instantes em que o controlador necessita informações acerca do comportamento do processo em regiões de operação distintas da atual. O controlador proposto, devido ao procedimento iterativo, é capaz de aproveitar de forma melhor a informação contida no mo-

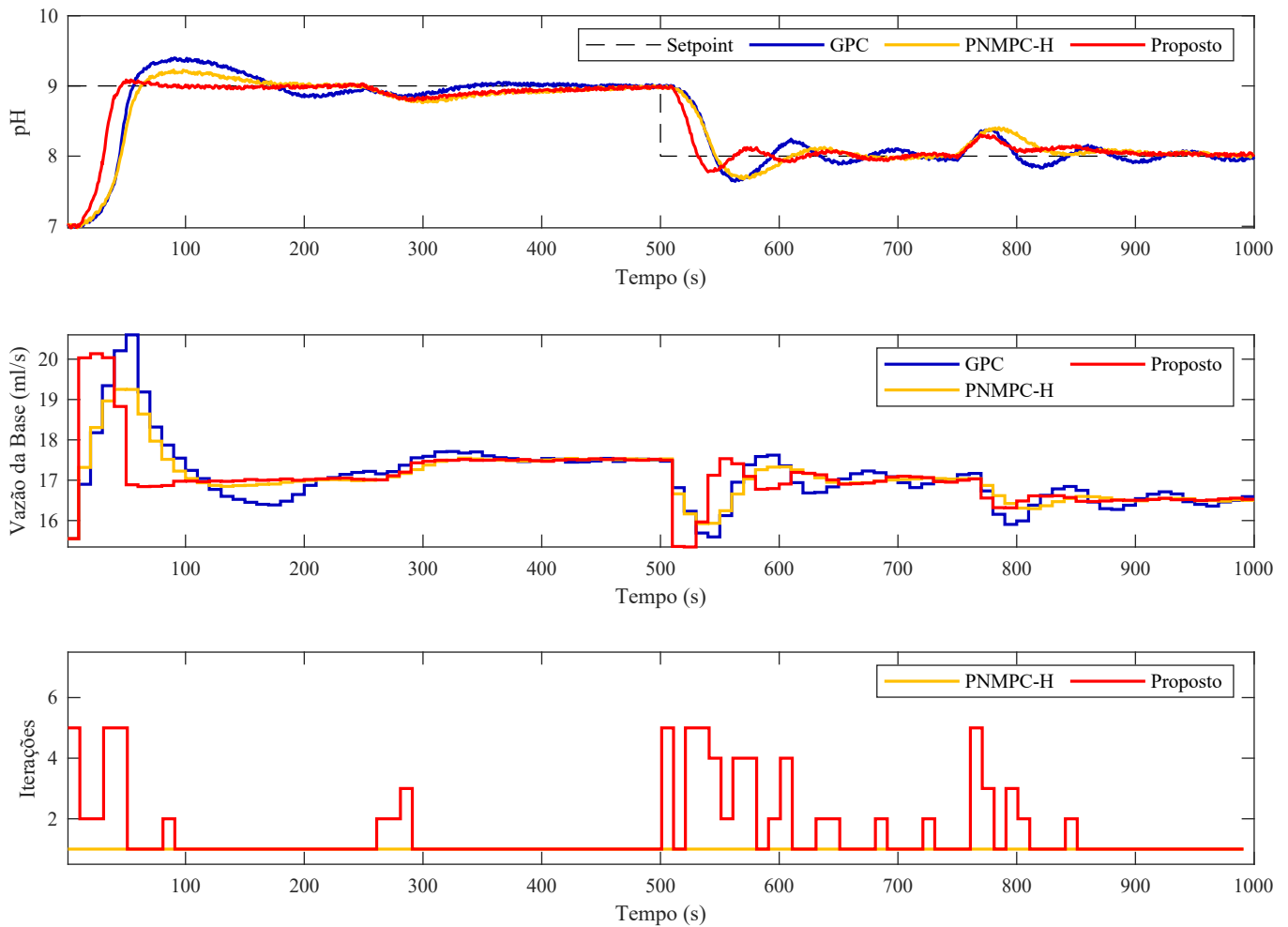


Figura 5. Resultados do controle do processo de neutralização de pH, para os algoritmos MPC-INPL-HW, PNMPC-HW e GPC. O gráfico superior apresentado a referência (linha tracejada) e o pH (linha sólida), com linha vermelha para MPC-INPL-HW, linha verde para PNMPC-HW e linha vermelha para GPC. O segundo gráfico apresenta os sinais de controle, e o terceiro gráfico apresenta a quantidade de iterações do algoritmo proposto.

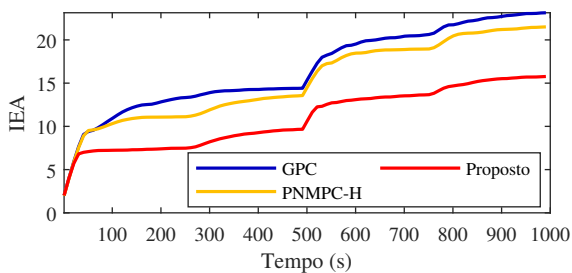


Figura 6. Evolução da integral do erro absoluto para o modelo proposto e modelos de comparação.

delo não linear, o que resulta em rápidas transições entre referências, com os menores sobressinais. Isso acontece devido ao algoritmo proposto atualizar iterativamente a parte do problema de otimização proveniente da linearização, convergindo para uma representação linearizada mais próxima do modelo não linear, considerando não só a dinâmica para o estado atual do processo, mas também para os estados futuros, preditos a cada passo do procedimento iterativo. A comparação entre o desempenho dos controladores pode ser melhor avaliada observando a evolução do IEA, a qual é apresentada na Figura 6.

O IEA final do algoritmo PNMPC-H é 21,6, valor esse 7% menor que o obtido pelo GPC, que apresentou um IEA final de 23,2. O algoritmo proposto, por sua vez, apresentou um EQM final de 15,3, representando uma melhoria de 34% em relação ao GPC, e de 29% em relação ao PNMPC-H. Essa melhoria, alcançada pelo algoritmo proposto, é expressiva, e deu-se em detrimento do custo computacional, o qual pode representar um aumento de 5 vezes em relação ao PNMPC-H.

6. CONCLUSÃO

Este artigo propõe um algoritmo MPC com predição iterativa não linear e linearização para modelos do tipo Hammerstein-Wiener. No processo iterativo, uma entrada base é aplicada ao modelo HW, de forma a obter uma predição não linear do sistema, e uma versão linearizada do modelo HW é usada para construir uma matriz dinâmica do sistema. Por meio da solução de um problema de programação quadrática, o esforço de controle ótimo é encontrado, o qual é usado para construir a próxima entrada base. Esse procedimento é conduzido até que o termo associado com a entrada base seja suficientemente

grande em comparação ao termo associado com a entrada ótima.

O algoritmo proposto foi avaliado utilizando um processo de neutralização de pH, no qual foi mostrado que o algoritmo iterativo proposto é capaz de retornar melhores resultados em malha fechada, com transições entre referências mais rápidas e menores sobressinais, em comparação com os algoritmos PNMPC-H e GPC. Tais resultados indicam que a formulação proposta é capaz de aproveitar melhor o conhecimento armazenado em modelos HW, em um algoritmo computacionalmente eficiente. Trabalhos futuros irão estudar a relação entre a abordagem proposta e abordagens NMPC que resolvem diretamente o problema de programação não linear utilizando algoritmos de programação quadrática sequencial (SQP, do inglês *Sequential Quadratic Programming*).

AGRADECIMENTOS

Este trabalho foi financiado em parte pelo Conselho Nacional de Desenvolvimento Científico e Tecnológico (CNPq/Brasil) por meio das concessões 140283/2018-8 e 309244/2018-8, e em parte pela Coordenação de Aperfeiçoamento de Pessoal de Nível Superior (CAPES/Brasil) por meio do projeto PrInt/UFSC.

REFERÊNCIAS

- Bloemen, H.H.J., Van Den Boom, T.J.J., and Verbruggen, H.B. (2001). Model-based predictive control for Hammerstein-Wiener systems. *International Journal of Control*, 74(5), 482–495.
- Camacho, E.F. and Bordons, C. (2007). *Model Predictive Control*. Springer Verlag, London.
- Clarke, D.W., Mohtadi, C., and Tuffs, P. (1987). Generalized predictive control—part i. the basic algorithm. *Automatica*, 23(2), 137–148.
- Dangui, H.A.S., Flesch, R.C.C., and Schwedersky, B.B. (2018). Practical guidelines for tuning model-based predictive controllers for refrigerant compressor test rigs. In *24th International Compressor Engineering Conference at Purdue*.
- De Keyser, R. (2009). Model based predictive control for linear systems. In *Control Systems, Robotics and Automation*, volume XI, 24–58. Eolss Publishers, Paris.
- Eskinat, E., Johnson, S.H., and Luyben, W.L. (1991). Use of Hammerstein models in identification of nonlinear systems. *AIChE Journal*, 37(2), 255–268.
- Forbes, M.G., Patwardhan, R.S., Hamadah, H., and Gopaluni, R.B. (2015). Model predictive control in industry: Challenges and opportunities. In *9th IFAC Symposium on Advanced Control of Chemical Processes ADCHEM 2015*, volume 48, 531–538.
- Foss, B.A. and Schei, T.S. (2007). Putting nonlinear model predictive control into use. In *Assessment and Future Directions of Nonlinear Model Predictive Control*, 407–417. Springer.
- Golub, G. and Pereyra, V. (2003). Separable nonlinear least squares: the variable projection method and its applications. *Inverse Problems*, 19(2), R1–R26.
- Gomez, J., Jutan, A., and Baeyens, E. (2004). Wiener model identification and predictive control of a pH neutralisation process. *IEE Proceedings-Control Theory and Applications*, 151(3), 329–338.
- Henson, M.A. and Seborg, D.E. (1994). Adaptive nonlinear control of a pH neutralization process. *IEEE Transactions on Control Systems Technology*, 2(3), 169–182.
- Lawryńczuk, M. (2013). Practical nonlinear predictive control algorithms for neural Wiener models. *Journal of Process Control*, 23(5), 696–714.
- Lawryńczuk, M. (2014). *Computationally Efficient Model Predictive Control Algorithms*. Springer International Publishing, 1 edition.
- Lawryńczuk, M. (2015). Nonlinear predictive control for Hammerstein–Wiener systems. *ISA Transactions*, 55, 49–62.
- Lee, J.H. (2011). Model predictive control: Review of the three decades of development. *International Journal of Control, Automation and Systems*, 9(3), 415.
- Mahmoodi, S., Poshtan, J., Jahed-Motlagh, M.R., and Montazeri, A. (2009). Nonlinear model predictive control of a pH neutralization process based on Wiener–Laguerre model. *Chemical Engineering Journal*, 146(3), 328–337.
- Plucenio, A., Pagano, D.J., Bruciapaglia, A.H., and Normey-Rico, J.E. (2007). A practical approach to predictive control for nonlinear processes. *IFAC Proceedings Volumes*, 40(12), 210–215. 7th IFAC Symposium on Nonlinear Control Systems.
- Quachio, R. and Garcia, C. (2019). MPC relevant identification method for Hammerstein and Wiener models. *Journal of Process Control*, 80, 78–88.
- Schoukens, J. and Ljung, L. (2019). Nonlinear system identification: A user-oriented road map. *IEEE Control Systems Magazine*, 39(6), 28–99.
- Schwedersky, B.B., Flesch, R.C.C., and Dangui, H.A.S. (2018). Practical nonlinear model predictive control with Hammerstein model applied to a test rig for refrigeration compressors. In *24th International Compressor Engineering Conference at Purdue*.
- Wellers, M. and Rake, H. (2000). Nonlinear model predictive control based on stable Wiener and Hammerstein models. In F. Allgöwer and A. Zheng (eds.), *Nonlinear Model Predictive Control*, 357–366. Birkhäuser, Basel.
- Wills, A., Schön, T.B., Ljung, L., and Ninness, B. (2013). Identification of Hammerstein-Wiener models. *Automatica*, 49(1), 70–81.
- Xu, K.K., Yang, H.D., and Zhu, C.J. (2019). A novel extreme learning machine-based hammerstein-wiener model for complex nonlinear industrial processes. *Neurocomputing*, 358, 246–254.
- Zhang, J., Chin, K.S., and Lawryńczuk, M. (2018). Nonlinear model predictive control based on piecewise linear Hammerstein models. *Nonlinear Dynamics*, 92(3), 1001–1021.