

MPC NEURO FUZZY: UM ESTUDO COMPARATIVO ENTRE DUAS DIFERENTES ESTRATÉGIAS

TEÓFILO P. G. MENDES*, TAKASHI YONEYMA*

**Divisão de Engenharia Eletrônica, Departamento de Sistemas e Controle
Instituto Tecnológico de Aeronautica, ITA-IEE-IEES
Praça Mal. Eduardo Gomes, 50, Vila das Acácias, 12228-900
São José dos Campos, São Paulo, Brasil*

E-mails: teopgmendes@uol.com.br, takashi@ita.br

Abstract— Recently, non linear model based predictive controllers (MPC) which use Takagi and Sugeno neuro-fuzzy nets have appeared. Starting from these nets properties, such controllers can be classified in two groups. One group applies the linear controller for each one of the multiple models and combines the control laws at end of each sample instant. The other applies only one linear controller at every moment in the instantaneous linearized global model. Differently of previous works, were it is usual to compare the performance of non linear controllers MPC with the linear ones, this article gives emphasis to comparison between different classifications of neuro-fuzzy MPC. By simulations of a nonlinear level control problem, the advantages and disadvantages of each method are presented.

Keywords— Generalized Predictive Control, Takagi-Sugeno, Multiple Models, Global Model.

Resumo— Recentemente, surgiram controladores preditivos não lineares baseado em modelo (MPC) que empregam redes neuro-fuzzy do tipo Takagi e Sugeno. Partindo das propriedades dessas redes, tais controladores podem ser classificados em dois grupos. Um grupo aplica um controlador o linear para cada um dos múltiplos modelos e combina as leis de controle ao final de cada instante, enquanto o outro aplica apenas um linear no modelo global instantaneamente linearizado. Diferentemente de trabalhos anteriores que geralmente confrontam o desempenho dos controladores MPC não lineares com os lineares, este artigo enfatiza a comparação entre diferentes classificações de MPC neuro-fuzzy. A partir de diversas simulações de um problema não-linear de controle de nível, as vantagens e desvantagens de cada método são mostradas.

Palavras-chave— Controle Preditivo Generalizado, Takagi-Sugeno, Múltiplos Modelos, Modelo Global.

1 Introdução

Atualmente, MPC tem grande número de algoritmos implementados e o uso largamente difundido na indústria devido à sua capacidade de lidar com difíceis problemas de controle, os quais envolvem interações de processos multivariáveis, restrições e atrasos de transportes, Camacho e Bordons (1999).

O termo MPC é associado a um grupo de controladores com mesma filosofia: a lei de controle depende de um comportamento predito, um modelo do processo é o responsável por prever a saída, a entrada é determinada a cada instante por uma otimização baseada na performance da predição, além de existir um horizonte de predição retrocedente. Enfatiza-se neste estudo o controlador preditivo generalizado (GPC) proposto por Clarke et al. (1987).

Apesar dos processos industriais geralmente apresentarem comportamento não-linear, aplicações bem sucedidas do GPC com modelos linearizados são encontradas em Camacho e Bordons (1999). Contudo, há casos que seu desempenho não é o desejável e modelos não-lineares são necessários para aumentar a performance dos controladores através de predições mais precisas. Todavia, esse método geralmente conduz a problemas de otimização não convexa, alta demanda computacional e soluções com mínimos locais.

Uma das formas de contornar estes problemas é o uso de redes neuro-fuzzy do tipo Takagi e Sugeno (1985). Essas redes ganharam grande importância na modelagem de sistemas em função da sua propriedade de aproximadores universais, especialmente no mapeamento de entradas e saídas de sistemas não-lineares. As redes do tipo Takagi e Sugeno (TS) se destacam pela capacidade de dividir um sistema não-linear em múltiplos modelos (MM) lineares ou a possibilidade de serem tratadas como uma aproximação de um modelo global (MG) linear com parâmetros variáveis (LPV).

Dessa forma, as estratégias de controle preditivo baseado em modelos neuro-fuzzy podem ser classificadas em dois grupos quanto à utilização das propriedades das redes TS. Um grupo de estratégias aplica o MPC linear separadamente nos MM, e combina as leis de controle ao final de cada instante, como Zhang e Morris (2000), Ning Li et al. (2004) e Ming He et al. (2005). No outro grupo utiliza-se a outra propriedade da rede, a de modelo global LPV, como Matko, D. e Kavsek B. K. (2000) e também Škrjanc, I. e Blažic, S. (2005).

Não é da ciência dos autores trabalhos focados em confrontar essa duas classificações de controlador preditivo generalizado neuro-fuzzy (NFGPC). Nos trabalhos citados é comum apenas a comparação entre o controlador não-linear proposto pelos autores e o MPC linear. Sendo assim, o presente artigo investiga o desempenho e a robustez das duas estratégias, bem como suas vantagens e desvantagens.

Este artigo segue explicando brevemente a estrutura de um modelo TS na seção 2 e revisando rapidamente o controlador GPC linear na seção 3. Na seção 4, os algoritmos das duas estratégias NFGPC são mostrados e, na seção 5, simulações são realizadas para os casos com e sem distúrbios não modelados. Por fim, conclui-se esse trabalho com uma análise comparativa entre as técnicas na seção 6.

2 Modelos neuro-fuzzy tipo Takagi-Sugeno

Seja um sistema não-linear de uma entrada e uma saída (SISO), ou múltiplas entradas e múltiplas saídas (MIMO), tal que:

$$y(k+1) = F(y(k), \dots, y(k-n_y), u(k), \dots, u(k-n_u)) \quad (1)$$

Este sistema pode ser descrito aproximadamente por vários modelos lineares locais, válidos para regimes de operação caracterizados por agrupamentos nebulosos Λ_i . Considera-se que esses agrupamentos nebulosos são divisões do conjunto de possíveis valores da entrada U e da saída Y , tal que a entrada $u \in U \subset \mathbf{R}$ e a saída $y \in Y \subset \mathbf{R}$. Define-se ainda, $\varphi_a(k) = [y(k), \dots, y(k-nay), u(k), \dots, u(k-nau)]$ como sendo o regressor do antecedente, com $\varphi_a(k) \in \Lambda_i$, e nay e nau , respectivamente, os números de atrasos da saída e da entrada usados nesse regressor.

Neste caso, em cada um dos i regimes de operação, tem-se a saída do sistema da forma em (2), com $\varphi_c(k) = [y(k), \dots, y(k-ncy), u(k), \dots, u(k-ncu)]$ definido como o regressor do conseqüente e ncy e ncu , respectivamente, os números de atrasos da saída e da entrada usados nesse regressor.

$$y_i(k+1) = f_i(\varphi_c(k)) \quad (2)$$

Note que a função f_i em (2) pode ser arbitrada desde que descreva suficientemente bem o sistema no regime de operação i , ou seja, satisfaça o critério de erro adotado. Neste trabalho, adota-se o modelo do tipo auto-regressivo com média móvel e entrada exógena (ARMAX) para representar essa função.

O modelo Takagi-Sugeno, então, descreve o comportamento desse sistema na forma de regras Se <condições> - Então <ações>, como mostra (3).

$$\begin{aligned} R_i: & \text{ Se } \varphi_a(k) \in \Lambda_i; \\ \text{Então } & y_i(k+1) = f_i(\varphi_c(k)) \quad i = 1, 2, \dots, R \end{aligned} \quad (3)$$

Dessa forma, cada regra i do modelo representa um diferente regime de operação, caracterizado por um agrupamento nebuloso Λ_i no antecedente e um sub-modelo ARMAX no conseqüente.

O processamento do termo antecedente consiste em calcular o peso normalizado das regras $\omega_i(k)$ dados por (4).

$$\omega_i(k) = \frac{\beta_i(\varphi_a(k))}{\sum_1^R \beta_i(\varphi_a(k))} \quad (4)$$

O peso não normalizado $\beta_i(k)$ em (4) é resultado da aplicação da norma T às funções de pertinência μ_{Λ_i} e seu valor é computado por (5).

$$\beta_i(\varphi_a(k)) = \prod \mu_{\Lambda_{i,0}}(y_k) \dots \mu_{\Lambda_{i,nay}}(y_{k-nay}) \cdot \prod \mu_{\Lambda_{i,0}}(u_k) \dots \mu_{\Lambda_{i,nau}}(u_{k-nau}) \quad (5)$$

Note que na expressão (5), e na próxima subseção, foi utilizado o abuso de notação $x(k) = x_k$ para preservar a clareza das equações.

As funções de pertinência μ_{Λ_i} dos agrupamentos nebulosos são calculadas por (6), com o vetor $z(k) = \varphi_a(k)$; o sub-índice $j = 0, 1, 2, \dots, nay+nau$; $\sigma_{i,j}$ e $c_{i,j}$ as variâncias e os centros dos agrupamentos respectivamente.

$$\mu_{\Lambda_{i,j}}(z_j) = \exp\left(-0.5 \frac{(z_{i,j}(k) - c_{i,j})^2}{\sigma_{i,j}^2}\right) \quad (6)$$

No termo conseqüente, a saída y_i dos sub-modelos ARMAX, expressa por (7), corresponde à versão linear de (2) em torno do centro c_i .

$$y_i(k+1) = A_i(q^{-1})y(k) + B_i(q^{-1})u(k) + r_i \quad (7)$$

Define-se o operador atraso de tempo q^{-1} , e tendo os polinômios A_i dados por (8), B_i por (9) e o termo de *offset* r_i por (10).

$$A_i(q^{-1}) = 1 + A_{i,1}q^{-1} \dots + A_{i,ncy}q^{-ncy} \quad (8)$$

$$B_i(q^{-1}) = B_{i,1} + B_{i,2}q^{-1} \dots + B_{i,ncu}q^{-ncu} \quad (9)$$

$$r_i = F(c_i) \quad (10)$$

A saída final da rede TS é uma média ponderada das saídas de cada sub-modelo com os pesos normalizados da regra correspondente, como mostra (11).

$$y(k+1) = \sum_1^R \omega_i(k) y_i(k+1) \quad (11)$$

Para um determinado instante, os diversos sub-modelos podem ser condensados em um único modelo global (12), onde se calcula os polinômios A, B e R por (13), podendo-se interpretar a rede como uma aproximação em LPV.

$$y(k+1) = A(k, q^{-1})y(k) + B(k, q^{-1})u(k) + R(k) \quad (12)$$

$$A = \sum_1^R \omega_i(k) A_i; B = \sum_1^R \omega_i(k) B_i; R = \sum_1^R \omega_i(k) r_i \quad (13)$$

Vale frisar que os parâmetros do modelo LPV dependem dos pesos normalizados $\omega_i(k)$, que por sua vez são funções não-lineares do regressor do antecedente $\varphi_a(k)$.

Cabe citar também que na etapa de identificação e modelagem do sistema, deve-se determinar o número de regras, a quantidade de atrasos nos regressores, os parâmetros das funções de pertinência, bem como os métodos de treinamento da rede. Em razão do espaço limitado não se descrevem tais tópicos devido a sua extensão.

3 Controle preditivo generalizado linear

O algoritmo GPC básico proposto em Clarke et al. (1987) é descrito a seguir. Ressalta-se que nessa seção utilizou-se a notação vetorial \vec{x}_{k+1} e $\overset{\leftarrow}{x}_{k-n}$ (14) sugerida por Rossiter J.A. (2003).

$$\begin{aligned}\vec{x}_{k+1} &= [x_{k+1} \ x_{k+2} \ \dots \ x_{k+n}]^T \\ \overset{\leftarrow}{x}_k &= [x_k \ x_{k-1} \ \dots \ x_{k-n}]^T\end{aligned}\quad (14)$$

Considera-se um modelo linear (15), sua forma incremental (16), com $A(q^{-1}) = a(q^{-1})\Delta$ e ainda o operador diferença dado por $\Delta = (1 - q^{-1})$. Sendo que os polinômios A e B possuem estruturas semelhantes aos apresentados em (8) e (9) respectivamente.

$$a(q^{-1})y_k = B(q^{-1})u_k \quad (15)$$

$$A(q^{-1})y_k = B(q^{-1})\Delta u_k \quad (16)$$

No controlador GPC, utiliza-se o modelo (16) para realizar previsões do comportamento do sistema, aplicando-se a primeira ação de controle de uma seqüência ótima, encontrada a partir da minimização de uma função custo (17), com $J: \mathbf{R}^{hc} \rightarrow \mathbf{R}$.

$$J = \left\| \vec{r} - \vec{y} \right\|_2^2 + \lambda \left\| \overset{\rightarrow}{\Delta u} \right\|_2^2 \quad (17)$$

A sintonia do controlador é feita através dos parâmetros dessa função custo, horizonte de predição hp, horizonte de controle hc e penalização do esforço de controle λ .

O modelo (16) tem a saída predita um passo a frente expressa por (18), onde n representa a ordem do modelo.

$$\begin{aligned}y_{k+1} &= -[A_1 q^{-1} \ \dots \ A_n q^{-n}] \overset{\leftarrow}{y}_k \\ &+ [B_1 q^{-1} \ \dots \ B_n q^{-n}] \overset{\leftarrow}{\Delta u}_{k-1} + B_1 \Delta u_k\end{aligned}\quad (18)$$

Escrevendo as equações de predição do modelo para hp passos à frente e usando a notação matricial de toeplitz/hankel também sugerida por Rossiter J.A. (2003) tem-se (19). Com definidas em (20) e as matrizes C_{zb} e H_{zb} construídas de forma análoga.

$$C_{\vec{z}b} \vec{y} + H_{\overset{\leftarrow}{z}b} \overset{\leftarrow}{y} = C_{\vec{z}b} \overset{\rightarrow}{\Delta u} + H_{\overset{\leftarrow}{z}b} \overset{\leftarrow}{\Delta u} \quad (19)$$

$$C_A = \begin{bmatrix} A_1 & 0 & \dots & 0 \\ A_2 & A_1 & \dots & 0 \\ A_3 & A_2 & \ddots & \vdots \\ \vdots & \vdots & \vdots & \ddots \end{bmatrix} \quad H_A = \begin{bmatrix} A_2 & A_3 & \dots & A_n \\ A_3 & A_4 & \dots & 0 \\ A_4 & A_5 & \ddots & 0 \\ \vdots & \vdots & \vdots & \vdots \end{bmatrix} \quad (20)$$

Isolando o vetor \vec{y} em (19) tem-se as hp previsões do modelo computadas por (21).

$$\vec{y} = H \overset{\rightarrow}{\Delta u} + P \overset{\leftarrow}{\Delta u} + Q \overset{\leftarrow}{y} \quad (21)$$

Sendo que as matrizes de predição H, P, Q são calculadas a partir de (22).

$$H = C_A^{-1} C_{zb}; \quad P = C_A^{-1} H_{zb}; \quad Q = -C_A^{-1} H_A \quad (22)$$

Substituindo (19) na expressão da função custo tem-se a equação (23), a qual, no caso sem restrições, tem solução analítica dada por (24).

$$J = \left\| \vec{r} - H \overset{\rightarrow}{\Delta u} - P \overset{\leftarrow}{\Delta u} - Q \overset{\leftarrow}{y} \right\|_2^2 + \lambda \left\| \overset{\rightarrow}{\Delta u} \right\|_2^2 \quad (23)$$

$$\overset{\rightarrow}{\Delta u} = (H^T H + \lambda I)^{-1} H^T (\vec{r} - P \overset{\leftarrow}{\Delta u} - Q \overset{\leftarrow}{y}) \quad (24)$$

Em função da estratégia de horizonte retrocedente, aplica-se apenas o primeiro valor da seqüência $\overset{\rightarrow}{\Delta u}$, expresso em (25), a qual pode-se interpretar como um ganho fixo.

$$\Delta u_k = P_r \vec{r} - D_k \overset{\leftarrow}{\Delta u} - N_k \overset{\leftarrow}{y} \quad (25)$$

No caso de um sistema linear e otimização sem restrições, o controlador GPC propriamente dito (G) pode ser computado *off-line* pelas matrizes P_r , N_k e D_k calculadas respectivamente por (26), (27), (29) e o vetor $e^T = [1, 0, \dots, 0]$.

$$P_r = e^T (H^T H + \lambda I)^{-1} H^T \quad (26)$$

$$N_k = e^T (H^T H + \lambda I)^{-1} H^T Q \quad (27)$$

$$D_k = e^T (H^T H + \lambda I)^{-1} H^T P \quad (28)$$

4 Controle preditivo generalizado neuro-fuzzy

Após uma identificação offline de um mesmo modelo TS, pode-se combinar suas propriedades de aproximador universal, com a técnica de controle preditivo usando duas estratégias distintas descritas abaixo, as quais envolvem tanto a etapa de análise quanto a etapa de projeto de controladores.

4.1 Controlador NFGPC com Múltiplos Modelos (NFGPC MM)

Partindo da capacidade da rede neuro-fuzzy de dividir o sistema não-linear em diferentes regimes e sub-modelos lineares, o controlador NFGPC com MM organiza-se da seguinte maneira:

- Analisa-se o modelo por regimes de operação e projeta-se um controlador para cada regime.
- Combinam-se os diferentes controladores usando, por exemplo, o centro de gravidade de suas saídas individuais.

Como citado anteriormente, o controlador GPC é fixo para um modelo linear sem restrições nas condições de operação, podendo assim ser computado *off-line*. Dessa forma, o controlador NFGPC com MM é semelhante a um estratégia de escalonamento de ganhos fixos. O escalonamento é proporcionado pelos pesos $\omega_i(k)$ e pelos ganhos de cada controlador dos sub-modelos. Resumindo esse algoritmo em passos tem-se:

Algoritmo 1

Etapa Off-line

Passo 1: Computar os controladores G_i para cada modelo linear, usando o algoritmo clássico GPC dado por (26), (27) e (28).

Etapa On-line

Passo 2: Medir o valor das variáveis que compõem o regressor do antecedente e encontrar os pesos normalizados $\omega_i(k)$ de cada regra usando (4), (5) e (6).

Passo 3: Usando os controladores C_i , processar o incremento das entradas $\Delta u_i(k)$ para os múltiplos modelos com (25).

Passo 4: Calcular o incremento da entrada final $\Delta u(k)$ usando (29).

$$\Delta u(k) = \sum_i^R \omega_i(k) \cdot \Delta u_i(k) \quad (29)$$

Passo 5: Encontrar o valor de $u(k)$ a ser aplicado com (30) e voltar ao passo 2.

$$u(k) = u(k-1) + \Delta u(k) \quad (30)$$

4.2 Controlador NFGPC com Modelo Global (NFGPC MG)

A linearização de sistemas é comumente utilizada para fugir de análises e projetos de controladores não lineares. A propriedade da rede neuro-fuzzy em aproximar o sistema não-linear em um modelo LPV é usada no algoritmo NFGPC com MG. A cada instante, podem ser usados métodos lineares de análise e projeto de controladores para o modelo LPV, porém o trabalho deve ser refeito para o instante seguinte. Dessa forma, esse algoritmo se organiza na forma abaixo:

- Combinam-se os diferentes ARMAX, e analisa-se o comportamento global do modelo, baseando-se nas mudanças de peso que cada regra possui ao longo dos diferentes regimes.
- Projeta-se um controlador que leve em consideração as mudanças no comportamento do modelo global.

É importante salientar, que diferente das linearizações geralmente feitas em um ponto de operação específico, o algoritmo NFGPC com MG realiza uma “linearização instantânea”, ou seja, em cada período de amostragem a rede TS “congela” os parâmetros LPV para poder aplicar o GPC clássico. Assim, resume-se esse algoritmo em passos.

Algoritmo 2

Etapa On-line

Passo 1: Medir o valor das variáveis que compõem o regressor do antecedente e encontrar os pesos normalizados $\omega_i(k)$ de cada regra usando (4), (5) e (6).

Passo 2: Calcular o modelo global linearizado na forma LPV usando (11).

Passo 3: Computar o controlador G para o modelo global LPV, usando o algoritmo clássico GPC dado por (26), (27) e (28).

Passo 4: Calcular o incremento da entrada final $\Delta u(k)$, usando (25).

Passo 5: Encontrar o valor de $u(k)$ a ser aplicado com (30) e voltar ao passo 2.

Observação 1: Uma hipótese importante nas estratégias apresentadas, mas não explícita, é considerar os pesos normalizados $\omega_i(k)$ constantes ao longo do horizonte de predição h_p nas predições do modelo (13).

Observação 2: Cabe ressaltar que o uso dos sub-modelos na forma incremental (16) proporciona a eliminação do termo de *offset* r_i ou $R(k)$, permitindo assim o uso do controlador linear GPC.

Observação 3: Neste artigo, enfatiza-se o algoritmo GPC sem restrições, apesar de existirem controladores NFGPC que as utilizam, como Ning Li et al. (2004), pois a composição ou decomposição de restrições dos sub-modelos não é direta para o caso não linear, além de depender do tipo da não linearidade apresentada pelo sistema.

5 Resultados

Para demonstrar as diferenças de desempenho entre as estratégias dos controladores NFGPC, considerou-se o problema apresentado por Zhang e Morris (2000). Nele, os autores propõem controlar o nível h de um tanque com formato cônico, usando como variável manipulada uma vazão de entrada Q_1 , tendo-se uma vazão de saída Q_0 .

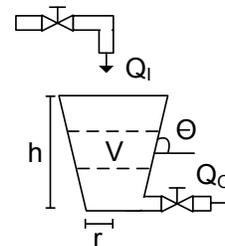


Figura 1. Tanque em forma de cone.

A equação dinâmica do processo tem seu comportamento descrito por (31).

$$\frac{dh}{dt} = \frac{Q_1 - k\sqrt{h}}{\pi \left(r^2 + \frac{2hr}{\tan \theta} + \frac{h^2}{3(\tan \theta)^2} \right)} \quad (31)$$

Assim como em Zhang e Morris, o período de amostragem usado na simulação foi de 10 s. Além disso, foram usados os mesmos parâmetros de tanque: $r = 10\text{cm}$, $\theta = 60^\circ$ e $k = 37,44\text{cm}^{2.5}/\text{s}$, com h em cm e Q_1 em cm^3/s .

O modelo neuro-fuzzy encontrado divide o processo em três regimes de operação: nível do tanque

baixo, médio e alto. A Figura 2 mostra as funções de pertinência do termo antecedente do modelo.

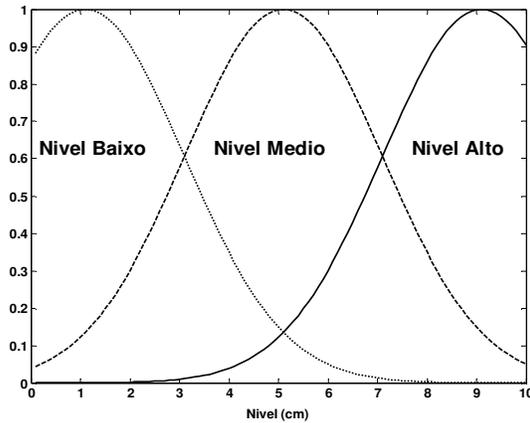


Figura 2. Funções de Pertinência.

A saída de cada regra R_i é dada pelo sub-modelo ARMAX i do conseqüente correspondente:

$$R_1: \text{ Se } y(k) \in \text{"Nível Baixo"}; \text{ Então} \\ y_1(k+1) = -0.1118y(k) - 0.0746y(k-1) \\ + 0.0384u(k) - 0.0026u(k-1) + 1.227$$

$$R_2: \text{ Se } y(k) \in \text{"Nível Médio"}; \text{ Então} \\ y_2(k+1) = 0.1571y(k) + 0.4918y(k-1) \\ + 0.0213u(k) + 0.0219u(k-1) - 1.381$$

$$R_3: \text{ Se } y(k) \in \text{"Nível Alto"}; \text{ Então} \\ y_3(k+1) = 0.1850y(k) + 0.0786y(k-1) \\ + 0.0023u(k) + 0.0154u(k-1) - 0.795$$

A primeira simulação foi realizada sem introduzir distúrbios no processo. Os parâmetros de sintonia usados nos dois controladores foram os mesmos, $h_p = 4$, $h_c = 2$ e $\lambda = 0.02$. Os resultados desta simulação podem ser visto nas figuras 3 e 4.

Nota-se que o controlador NFGPC com MG obtém maior sucesso em rastrear o sinal de referência do que o controlador com MM. A diferença de desempenho se dá especialmente no transitório, ao se começar encher o tanque, e em pontos de operação do processo que não existe um regime de operação único ou dominante.

Na segunda simulação, foi introduzida uma outra vazão de entrada, como um distúrbio não modelado. Para verificar a sensibilidade dos controladores foram mantidos os mesmos parâmetros de sintonia. Observa-se novamente o melhor rastreamento do controlador com MG, e nesse caso maior robustez, como mostrado nas figuras 5 e 6.

Explica-se a diferença de desempenho dos controladores nas simulações realizadas pelo efeito da propriedade da rede TS adotada por cada um. O fato do algoritmo com MG atualizar a cada momento seu modelo permite um controlador mais flexível. Para cada instante, a lei de controle global é resultado dos

pesos das regras e sub-controladores variáveis. Isso proporciona maior robustez e capacidade de rastreamento, contudo o custo disso é uma carga computacional também maior, devido a inversão de matrizes nos controladores.

Por sua vez, como a estratégia NFGPC com MM transfere o cálculo dos sub-controladores para a etapa off-line, tem-se um controlador mais rígido. Nesse caso, a lei de controle global é resultado dos pesos das regras e sub-controladores fixos. O efeito disso fica claro nas simulações: carga computacional menor em detrimento de capacidade de rastreamento e robustez também menores.

Analisando especificamente a discrepância de desempenho no transitório, percebe-se que as linearizações instantâneas realizadas pelo algoritmo NFGPC com MG permitem captar melhor a tendência do processo.

Nos casos nos quais o regime de operação não está bem definido, o controlador com MM apresenta desempenho inferior, pois ao calcular os sub-controladores na etapa off-line leva-se em consideração apenas os sub-modelos dos regimes de operação nominais. Nos regimes de operação intermediários o simples escalonamento dos ganhos pelos pesos das regras não garantem o desempenho similar dos casos anteriores.

Vale frisar que a hipótese adotada, dos pesos normalizados $\omega_i(k)$ serem constantes ao longo do horizonte de predição h_p , é um ponto fraco de ambos controladores, porém acredita-se que as diferenças de comportamento se devem essencialmente à diferença de estratégias, ou propriedades da rede TS usada.

6 Conclusão

Dois diferentes controladores preditivos generalizados usando modelos neuro fuzzy (NFGPC) foram revisados e comparados. Um deles utiliza-se da capacidade da rede TS de dividir o sistema não-linear em múltiplos modelos (MM) lineares, enquanto outro aproveita a interpretação da rede como um modelo global (MG) linear com parâmetros variáveis.

Assim, baseado nos resultados dessa simulação, recomenda-se o uso do controlador NFGPC com MG para aplicações onde uma maior carga computacional não é um problema. Uma vez que essa carga seja uma limitação, e ainda assim seja necessário o uso de um modelo não linear, pode-se fazer uso do controlador NFGPC com MM.

Agradecimentos

Os autores agradecem o apoio financeiro da CAPES e do ITA.

Referências Bibliográficas

- Camacho, E.F., Bordons, C., (1999). Model Predictive control, *Springer Verlag*.
- Clarke, D.W., Mohtadi, C. and Tuffs, P.S. (1987). Generalized predictive control - Partes 1 e 2, *Automatica*, vol. 23, n. 2, p. 859–875.
- Matko, D., Kavsek B. K. (2000). Generalized predictive control of a heat exchanger using fuzzymodel. *26th Annual Conference of the IEEE*, Nagoya, Japão.
- Ming He, Wen-Jian Cai, Shao-Yuan Li (2005). Multiple fuzzy model-based temperature predictive control for HVAC systems. *Information Sciences*, n. 169, p. 155–174.
- Ning Li, Shao-Yuan Li, Yu-Geng Xi (2004). Multi-model predictive control based on the Takagi–Sugeno fuzzy models: a case study. *Information Sciences*, n.165, p. 247–263.
- Rossiter J.A. (2003) Model-Based Predictive Control, A practical Approach, *CRC press*.
- Škrjanc, I., Blažic, S. (2005). Predictive Functional Control Based on Fuzzy Model: Design and Stability Study, *Journal of Intelligent and Robotic Systems*, n. 43, p. 283–299
- Takagi, T., Sugeno, M. (1985). Fuzzy identification of systems and its applications to modeling and control, *IEEE Trans. Systems Man Cybernetic*. n. 15, p. 116–132.
- Zhang, J., Morris, A.J. (2000). Long Range Predictive Control of Nonlinear Processes Based on Recurrent Neuro-Fuzzy Network Models, *Neural Computing and Applications*, *Springer Verlag London Limited*, n. 9,p. 50–59.

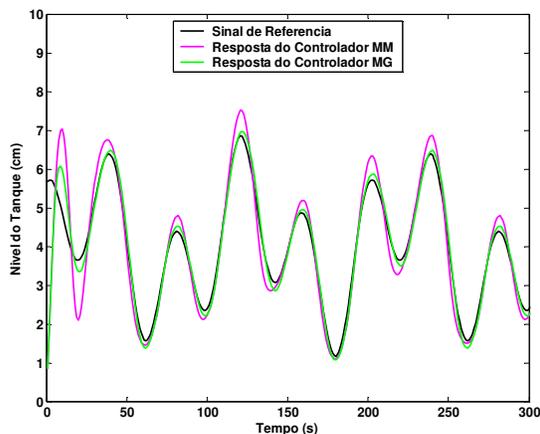


Figura 3. Simulação sem adição de distúrbio.

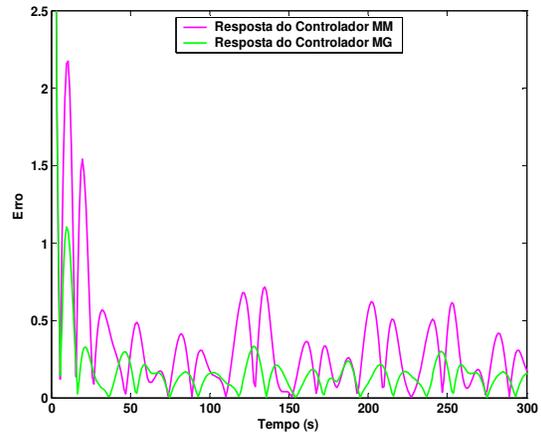


Figura 4. Erro na simulação sem adição distúrbio.

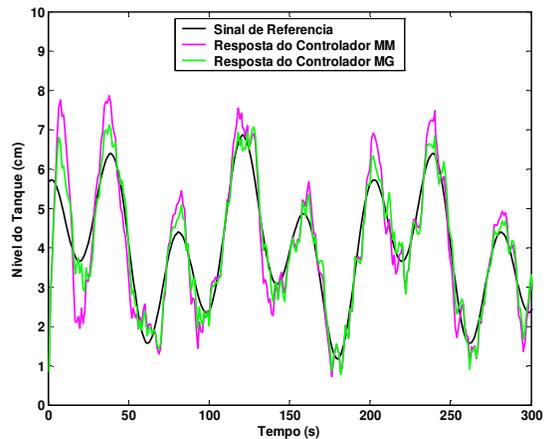


Figura 5. Simulação com adição de distúrbio.

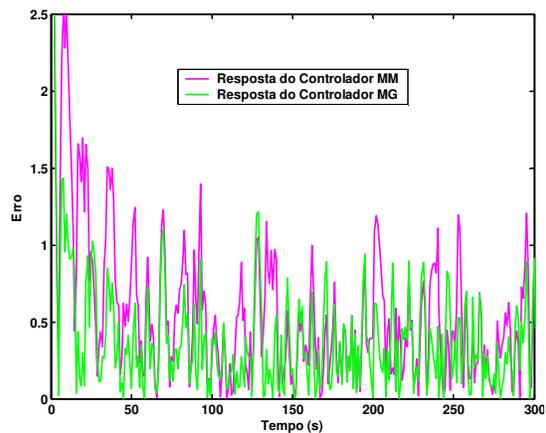


Figura 6. Erro na simulação com adição distúrbio.