

Algoritmo matemático para um agente em sistemas dinâmicos

Elmer Rolando Llanos Villarreal,¹
Marcelino Pereira dos Santos Silva

Universidade do Estado do Rio Grande do Norte
Departamento de Informática, Telefone/FAX: (084) 3315-2235
BR 110, Km 48, Bairro Costa e Silva, Mossoró RN.
e-mail: evillarrea@hotmail.com; marcelinopereira@uern.br

Abstract

Resumo. No presente artigo trata-se o problema de controle de sistemas dinâmicos. Assim é proposto um novo conceito de três agentes inteligentes para controlar sistemas complexos e sofisticados. O primeiro agente é heurístico baseado gerador de modelo referencial fuzzy múltiplo. O segundo agente é um controlador rede neural baseado em (*RBFINN*) (Radial Basis Function Neural Network) que é usado para aumentar o tradicional (*MRAC*) (Model Reference Adaptive Controller) na presença do sistema funcional incerto. O terceiro agente é um algoritmo matemático baseado sobre o MRAC, é o MRAC tradicional que controla de forma adaptativa o sistema, lineariza os parâmetros sobre um domínio específico e força a saída ou outras variáveis da planta para uma estrutura de modelo referência apropriada.

Palavras-chave: Controlador Multiagente, Sistemas dinâmicos, controlador adaptativo, Algoritmo.

1 Introdução

Controladores adaptativos multimodal e os sistemas de salto são extremamente difíceis devido à mudança drástica na planta dinâmica que pode ser representada por sistemas não lineares [7], [5]. Neste artigo é proposto um novo conceito de três agentes inteligentes para controlar sistemas complexos e sofisticados. O primeiro agente é heurístico baseado gerador de modelo referencial fuzzy múltiplo o qual move o modelo referencial, traçando o sistema estado auxiliar. Este agente é autônomo e trabalha dentro de seu próprio domínio fazendo testes com entradas auxiliares e gerando a estrutura de modelo de referência apropriado em cada instante de tempo. Assim comparado com MRAC matemático e com o sistema controlador reconfigurável, este processo de "switching" não demonstra intensidade computacional. O segundo agente é um controlador rede neural baseado em (*RBFINN*) (Radial Basis Function Neural Network) que é usado para aumentar o tradicional (*MRAC*) (Model Reference Adaptive Controller) na presença do sistema funcional incerto. A consideração principal mais importante aqui é para usar uma rede neural para aproximar a dinâmica inversa da planta que trabalha em paralelo com uma lei de controle adaptativo linear. A vantagem de tal característica quando comparada a um esquema de aprendizagem do erro é sua habilidade para o controle coordenado, que fornece uma lei estável durante o parâmetro de incertezas. O terceiro agente é o MRAC tradicional que controla de forma adaptativa o sistema, lineariza os parâmetros sobre um domínio específico e força a saída ou outras variáveis da planta a uma estrutura de modelo referência apropriada [1],[2]. Para validar o esquema, o controlador é usado para seguir da posição de um manipulador flexível da única ligação em várias circunstâncias e as simulações são comparadas com os esquemas duplos do agente. Por outro lado a base teórica deve permitir implementar uma plataforma de controle multiagente [3]. O artigo é organizado como segue. Na seção (2 e 3) são apresentados os conceitos básicos. Na seção (4) é apresentado a formulação matemática, os agentes inteligentes são abordados na seção (5) e finalmente as conclusões finais.

2 Conceitos matemáticos

Usando a teoria do controle, estuda-se neste artigo, como os agentes são usados para o controle de uma planta [9]. Assim este artigo consiste de três agentes baseado em algoritmos que são usados

para sistemas de controle que mostra saltos repentinos na dinâmica não modeladas de um sistema de controle. Assim na presença de uma dinâmica não modelada e do balanço modal, o controlador paralelo com *RBFNN* aumentam o controlador adaptativo do modelo de referência direta e assim o sistema controlado ficar estável [8]. Entretanto quando o projeto da planta mostra saltos repentinos, a estrutura do modelo da referência precisam serem mudados. Se estas mudanças forem conhecidas, o motor de inferencia fuzzy a priori pode gerar uma estrutura do modelo de referência múltipla em [11], [10].

Existem dois agentes inteligentes baseados em malha supervisores; o primeiro gera uma estrutura de modelo referencia em mudança, e a segunda aumenta o controlador adaptativo do modelo de referencia direto estável. O terceiro agente é um algoritmo MRAC tradicional próprio. Se a operação do sistema sob consideração pode ser modelado usando um modelo de referência, então o controlador adaptativo pode ser utilizado para controlar o sistema para seguir a saída desejada do modelo da referência [12].

3 Conceito do Projeto

A estrutura do controlador paralelo é baseado no fato que a rede neural pode ser usado para aprender do erro de retroção do controlador do projeto (Planta) em malha fechada. O projeto é controlado por um controlador adaptativo que segue a saída do modelo de referência. Isto trabalha em um domínio autônomo por se próprio, olha a saída e fornece a ação de controle requerida. Em um evento de distorsão de saída, o controlador *RBFNN* aprende do sistema dinâmico e aumenta o *MRAC* olheando o sistema em movimento. Isto é de fato realizado continuamente tal que o erro de saída é zero e a precisão seguida é atingida.

Quando no sistema, o controlador adaptativo não é executado com um único modelo da referência. Consequentemente, há uma necessidade para gerar as estruturas do modelo de referência o qual seja o processo executado pelo agente fuzzy. Assim pode-se mostrar as interações do agente e o método do projeto para um esquema total. Desta o primeiro agente está em uma forma hierárquica monitorando indiretamente o desempenho do sistema através do terceiro agente. No evento das mudanças drásticas devido aos parâmetros do sistema detetados pelas entradas fuzzy do sistema, o motor de inferência seleciona as regras apropriadas que mudam por sua vez com estrutura do modelo de referência. Isto permite para uma mudança suave na saída do modelo da referência, para assim o terceiro agente trabalhe eficazmente. O agente dois tem como função monitorar o sistema de saída e aumentar e regular os parâmetros do agente três quando existe uma mudança funcional na saída do sistema. Neste contexto esta arquitetura é uma abordagem híbrida com estrutura hierarquica entre o agente um versus agente três.

4 Formulação Matemática

O sistema não linear pode ser controlado com entrada U e saída y_i pode ser representado como

$$\dot{x} = A_{imode(t)}f(X) + b_{imode(t)}g(X)U \quad (1)$$

$$y_i(t) = h(X) \quad (2)$$

onde y_i é uma planta de saída em um modo específico, U é uma entrada de controle, X é um vetor estado onde $\begin{bmatrix} X_1(t), \dots, X_n(t) \end{bmatrix}' \in \mathbb{R}^n$; $A_i(t) = \begin{bmatrix} a_{1i}, a_{2i}, \dots, a_{ni} \end{bmatrix}' \in \mathbb{R}^{n \times n}$; $f(*)$, $g(*)$ e $h(*)$ são funções não lineares e b_i toma valores do conjunto de elementos constantes de H indexado por índice $i \in \{1, 2, \dots, H\}$. Por outro lado, baseado na mudança de valores, sejam $f(*)$ e $g(*)$ duas funções não lineares tal que elas são contínuas. Primeiro, assume-se que o sistema é operado em uma

modalidade sem nenhuma interrupção da modalidade. Sob tal circunstância pode ser escrito como

$$\dot{x} = Af(X) + bg(X)U \quad (3)$$

$$y(t) = h(X) \quad (4)$$

Assumindo o controle *SISO* e o sistema de segunda ordem, a equação genérica acima será expressa por

$$x_1' = x_2 \quad (5)$$

$$x_2' = G(x_1, x_2, U) \text{ ou} \quad (6)$$

$$x_1'' = G(x_1, x_1', U) \quad (7)$$

Assumindo que a função G é invertível, então

$$U = G \left[x_1, x_1' x_1'' \right] \quad (8)$$

Seja U_d o valor desejado do controlador, então

$$U_d = G \left[x_{1d}, x_{1d}' x_{1d}'' \right] \quad (9)$$

Se

$$x_1 = y, U_d = G \left[y_d, y_d' y_d'' \right] \quad (10)$$

Isto significa na presença de um controlador U_d , o sistema pode ser controlado eficazmente através da aprendizagem se for invertida. Um controlador *RBFNN* é utilizado para controlar o sistema não linear com dinâmica não modelada, o qual aprende do sistema dinâmico on-line e então gera o valor de U_d . Como pode ser visto a partir de (10), as entradas para o controlador *RBFNN* são a saída da planta y e outros estados desejados. O erro de trajetória da saída e_1 é reduzida pela técnica de aprendizagem do gradiente assintoticamente e o sistema aprende durante o decorrer do tempo, gerando o valor U_d .

O dinâmica do erro durante a aprendizagem do sistema pode ser escrito como

$$e = G \left[x_1, \dot{x}_1, U \right] - G \left[x_{1d}, \dot{x}_{1d}, U_d \right] \quad (11)$$

Aplicando expansão de série de Taylor e coletando os termos de primeira ordem

$$\dot{e} \cong A(t)e + B(t)[U - U_d] + N(t)\varepsilon \quad (12)$$

Considerando $\varepsilon = 0$ (aprendizagem perfeita)

$$\dot{e} \cong A(t)e \quad (13)$$

Pode ser visto de (13) que o erro derivativo é dependente em $A(t)$ que é função dos estados e do valor de controle. Uma lei de controle adaptativa é usado para reduzir este erro assintoticamente para zero. A lei de controle total é

$$U = U_{mr} + U_{nn} \quad (14)$$

onde

$$\begin{aligned} U_{mr} &= \theta' * w \\ \theta &= [k, \theta_0, \theta_1, \theta_2], \\ w &= [r, y_p, w_1, w_2]' \end{aligned} \quad (15)$$

com

$$\begin{aligned} U_{nn} &= \sum_{j=1}^{nodo} \left(\exp - \left(\frac{d_j^2}{r_j^2} \right) \right) * w_j \\ d_j &= \sqrt{\sum_1^{Nm} (X_k - C_{(j,k)})^2} \end{aligned} \quad (16)$$

d_j é a distância e r_j , W_j são raio e peso de cada nodo respectivo. Os pesos da rede neural são ajustados em ordem para reduzir o gradiente do erro de saída, somente que a dinâmica não-linear são mantidos delimitadas. O controlador do modelo de referência adaptativa identifica o valor paramétrico online da planta e controla eficaz.

A mudança estrutural no modelo de referência é requerida se não existe um procedimento de identificação explícita disponível para a planta.

5 Agentes

5.1 Agente inteligente 1

A base teórica do gerador do modelo de referência múltiplo fuzzy é a seguir. Considera-se um sistema fuzzy saída denotada pela função $f(\Omega)$ e representado como

$$f(\Omega) = \frac{\sum_{i=1}^p p_i \mu_i}{\sum_{i=1}^p \mu_i} \quad (17)$$

O sistema fuzzy mencionado acima tem p regras e μ_i é uma função pertencente do antecedente da i -ésima regra dado pela entrada Ω .

Onde $\Omega \in \mathbb{R}^m$ é um vetor contendo estados auxiliares relevantes.

A função de aproximação pelo esquema fuzzy é igual ao produto de um vetor parâmetro ϕ e a matriz de peso Θ .

A função de transferência do modelo de referência pode ser a função da lógica fuzzy de saída.

$$y_{mi}(t) = (M_i' P_i \Theta_i) * W_{mi} * S(t) \equiv (\phi_i * \Theta) * W_{mi} * S(t) \quad (18)$$

Para um comando de sinal constante (18) pode ser reescrito como

$$y_{mi}(t) = v(\phi_i, \Theta_i, W_{mi}) \quad (19)$$

onde:

$y_{mi}(t)$: é a saída do modelo de referência

ϕ_i : é umj vetor parâmetro desenvolvido pelo sistema fuzzy

Θ_i é a função de peso, v é um coeficiente indexado e W_{mi} é a função de transferência do modelo de referência.

5.2 Agente inteligente 2

A estrutura *RBFNN* é usado junto com o MRAC para capturar a funcional não-linear, através da aprendizagem on-line. Os passos envolvidos na projeto rede *RBFNN* é a seguir.

A rede *RBFNN* crescente proposta terão início com três nodos. Os centros desses nodos serão inicializados como zero. A rede *RBFNN* é "nova" sobre o sistema que está sendo aproximado ou controlado.

Seleção da função básica:

Mais comumente usada é a função básica, expressa em função Gaussiana como

$$O_j = \exp\left(\frac{-d_j^2}{r_j^2}\right) \quad (20)$$

onde O é a rede de saída. Esta função é escolhida devido ao fato de dar maior peso para os pontos que estão próximos do centro nodal e exponencialmente menor peso aos pontos, que se movem, mais longe do centro do nodo.

5.3 Agente inteligente 3

O terceiro agente é um algoritmo matemático baseado sobre o MRAC. É o MRAC tradicional que controla de forma adaptativa o sistema, lineariza os parâmetros sobre um domínio específico e força a saída ou outras variáveis da planta a uma estrutura de modelo referência apropriada.

Qualquer lei de controle adaptativo de certeza equivalente pode ser usado como um controlador básico, neste caso. Usa-se uma abordagem *MRAC* direto deescrito em (??). Considere uma planta *SISO* linear desconhecido descrito pelo

$$x_p = A_p x_p + b_p u, \quad (21)$$

$$y = h_p' x_p \quad (22)$$

onde μ é a entrada, e y_p é a saída da planta, x_p é um vetor de estado de n -ésima ordem, A_p é uma matriz $n \times n$.

Considera-se um modelo de referência representado como

$$e = y_p - y_m \quad (23)$$

se aproxima de zero assintoticamente para condições iniciais arbitrárias.

A estrutura do controlador para o esquema é

$$u = \Theta' w \quad (24)$$

onde $\Theta = [k \ \Theta_0 \ \Theta_1' \ \Theta_2']'$ é um controlador vetor parâmetro e $w = [r \ y_p \ w_1' \ w_2']'$ é um vetor regresivo. Os vetores de regressão são atualizados online baseados na seguinte equação

$$w_1 = \Lambda w_1 + L u \quad (25)$$

$$w_2 = \Lambda w_2 + L y_p \quad (26)$$

onde Λ é uma matriz estável de ordem $(n-1) \times (n-1)$ tal que o determinante $|S_i - \Lambda| = Z_m(s)$ e o vetor L é definido como $L' = [0, \dots, 0, 1]$. Portanto, o controlador μ é estruturado como

$$k = -\text{sgn}(K_p) e r, \quad (27)$$

$$\theta = -\text{sgn}(K_p) e y_p \quad (28)$$

$$\theta_1' = -\text{sgn}(K_p) e w_1' \quad (29)$$

$$\theta_2' = \text{sgn}(K_p) e w_2' \quad (30)$$

As equações (24-30) citadas acima são usadas no processo de controle.

6 Conclusões

Neste artigo, foi estudado um novo conceito do agente supervisor inteligente para o controle em malha que é proposto para sistemas dinâmicos. O projeto proposto consiste de três agentes autônomos dos quais duas são baseados em técnicas artificiais inteligentes. Além disso, devido a abordagem do projeto híbrido, o esquema é factível e pode ser implementado usando um controle embutido e uma rede.

References

- [1] N. Hovakimya, Flavio. Nardi, Anthony J. Calise, Hungu Lee, *Adaptive Output Feedback Control of a Class of Nonlinear Systems Using Neural Networks* , The International Journal of Control, 1999.
- [2] Rolf Rysdyk, Antonhy J. Calise, *Robust Nonlinear Adaptive Flight Control for Consistent Handling Qualities*, IEEE, Transactions on Control Systems Technology, vol. 13 No 6, November, 2005.
- [3] E. R. LL. Villarreal, M. P. S. Silva, D. P. F. Pedrosa and J. D. Lima *Abordagem de integração do sistema computacional para redução de perdas em redes de distribuição de energia com uma plataforma de controle multiagente*, Aceito Revista de Ciência Sempre, FAPERN, Natal, RN, 2008.
- [4] P. A. Ioannou and J. Sun, *Robust Adaptive Control* , Upper Saddle River NJ: Prentice Hall, 1996, ch. 2-6.
- [5] A. Isidori *Nonlinear Control Systems* Third Edition, Springer Verlag, London, 1995.
- [6] Jennings, N. R., *Cooperation in Industrial Multiagent Systems*, World Scientific, 1994.
- [7] H. Khalil, *Nonlinear Systems*, MacMillan, New York, 1992.
- [8] K. S. Narendra, Y. H. Lin, L. S. Valavani, *Stable adaptive controller design, part 1 and 2, and stable discrete adaptive control*, IEEE Trans. Autom. Control, vol. 25, No. 3, pp. 433-461, Jun. 1980.
- [9] K. Ogata, *Engenharia de Controle Moderno*, Prentice Hall, 2004.
- [10] J. T. Spooner and K. M. Passino, *Stable Adaptive Control Using Fuzzy Systems and Neural Networks* , IEEE Trans. on Fuzzy systems, vol. 4 No. 3, pp.339-359, 1996.
- [11] T. Takagi, and M. Sugeno, *Fuzzy Identification of Systems and its Applications to Modeling and Control* , IEEE Trans. System Man Cybernet, vol. 15, No. 1, pp. 116-132, Jan, 1992.
- [12] Sukumar Kamalasandan, *A Novel Multi-Agent Controller for Dynamic Systems based on Supervisory Loop Approach*, Engineering Letters, 14:2, EL14-2-10, 2007.