

# CONTROLE DE VIBRAÇÕES USANDO LÓGICA FUZZY COM REGRAS OTIMIZADAS POR ALGORITMOS GENÉTICOS

**Rafael Luís Teixeira**

**Prof. Dr. José Francisco Ribeiro**

Laboratório de Sistemas Mecânico, Faculdade de Engenharia Mecânica, Universidade Federal de Uberlândia  
Av. João Naves de Ávila, 2160 – Campus Santa Monica – Cep 38400.089 - Uberlândia, MG, Brasil  
E-mails: rafael@mecanica.ufu.br jribeiro@mecanica.ufu.br

## Resumo

A Lógica Fuzzy é uma técnica moderna de controle que permite trabalhar com sistemas complexos, não lineares, variantes no tempo e que dispensa o conhecimento preciso dos modelos matemáticos envolvidos. Em contrapartida é indispensável que se tenha um entendimento heurístico do sistema a ser controlado, conhecimento este materializado no conjunto de regras do controlador fuzzy. O estabelecimento do conjunto de regras, que normalmente é construído a partir da experiência do operador (ou usuário) do sistema, é tarefa fundamental no projeto do controlador fuzzy. Este trabalho propõe um método de otimização do conjunto de regras via Algoritmos Genéticos. O procedimento parte de um conjunto inicial de regras e pondera cada regra de acordo com sua importância na minimização de um índice de desempenho arbitrado. Apenas as regras mais importantes são retidas no controlador. Os controladores propostos são avaliados numericamente e experimentalmente no controle de um sistema mecânico vibratório de 1 gdl e os resultados obtidos mostram a potencialidade e a eficiência do método.

## Abstract

The fuzzy logic is the technique of modern control for complex, nonlinear and time varying systems. The knowledge of the mathematical models of the plant is not necessary in this technique but the heuristic knowledge of the plant, represented by a set of rules, is necessary. The set of rules, usually built by the expert human, represents the important task for the project of the fuzzy controller. This work proposes an optimization method of the set of rules using Genetic Algorithms (GA's). In the numerical procedure, a set of initial rules is generated and each rule is evaluated by the controller and the best set of rules is used by the fuzzy controller. The proposed fuzzy controller is evaluated numeric and experimentally for the control of a vibratory mechanical system of one degree of freedom and the results show the potentiality and efficiency of the optimization method.

## 1. INTRODUÇÃO

O controle de sistemas físicos tem evoluído rapidamente nos últimos anos, devido ao uso de controladores digitais através das técnicas de Inteligência Artificial. Os Controladores baseados em Lógica Fuzzy, introduzidos por Lofti A. Zadeh (Universidade da Califórnia – Berkeley) em 1965, é um exemplo disto. Tais controladores aplicam em áreas onde decisões objetivas precisam ser tomadas a partir de um conjunto de informações muitas vezes vagas, imprecisas e não adequadamente quantificadas. Nos sistemas fuzzy as incertezas são expressas em variáveis lingüísticas (Ex: muito, pouco, baixo, forte, fraco, quente, frio, etc). A tecnologia fuzzy é natural e intuitiva na sua formulação, pois tenta imitar o comportamento consciente ou a estratégia de controle de um operador humano. Pode ser aplicada a sistemas lineares e não lineares, é de baixo custo e de fácil implementação, apresentando boas características de robustez a incertezas ou variações paramétricas.

A eficiência dos modelos fuzzy, ou seja, a capacidade de mapeamento entre as entradas e saídas de um sistema, depende da escolha de uma série de parâmetros que definem, entre outras coisas, a natureza dos conjuntos fuzzy envolvidos, o universo de variação das variáveis de entrada e saída, a base de regras empregada, os mecanismos de inferência utilizados, etc. Todavia no projeto de um controlador fuzzy a maior dificuldade está no estabelecimento do conjunto de regras, que normalmente são construídas com base na experiência e intuição de um especialista. Estas regras relacionam as entradas com as saídas e, em sistemas não lineares, variantes no tempo e instáveis, com muitas entradas e muitas saídas, a definição do conjunto de regras se torna uma tarefa não muito trivial e um grande desafio. Uma questão crítica é o grande número de combinações que podem ocorrer no projeto de controladores fuzzy, especialmente quando o sistema apresenta muitas variáveis lingüísticas e muitas funções de pertinência. Quanto maior o número de regras, maior será o tempo computacional de processamento, podendo inviabilizar o uso em tempo real do controlador. Há, portanto, um compromisso entre o número de regras, a velocidade de processamento e o desempenho esperado para o controlador.

Uma técnica que tem sido utilizada com frequência, no intuito de otimização de sistemas, são os Algoritmos Genéticos (AG), cujos conceitos básicos foram desenvolvidos por Holland [1992]. Os algoritmos evolutivos são métodos baseados na seleção natural pela sobrevivência, conforme a teoria de Charles Darwin, que considera na evolução das gerações cruzamentos e mutações. A formulação matemática desta evolução configura o algoritmo genético.

Dentro das perspectivas de que os Sistemas Fuzzy, envolvem variáveis de projeto passíveis de otimização, segundo critérios de desempenho estabelecidos, os AG tem sido empregados como ferramentas de otimização no projeto de controladores fuzzy, tanto otimizando o conjunto de regras, conforme Baitinger *et al.* [1993] e Yván *et al.* [1999], como na escolha ótima das funções de pertinência, segundo Homaifar *et al.* [1995] .

Este trabalho propõe um procedimento que - via algoritmos genéticos - hierarquiza as regras de um controlador fuzzy, de acordo com o grau de importância de cada uma no desempenho do sistema, segundo um critério de desempenho estabelecido. Assim é possível extrair de um conjunto grande de regras um conjunto menor, com as regras mais importantes, facilitando a implementação física do controlador. Este trabalho está assim organizado: na seção 2 são apresentados os fundamentos do algoritmo genético e sua utilização no problema em questão, na seção 3 é apresentado o aparato experimental onde o procedimento proposto é avaliado, na seção 4 são apresentados os resultados dos ensaios realizados e na última seção apresenta as conclusões e desdobramentos.

## 2. O ALGORITMO GENÉTICO E O CONTROLADOR FUZZY

O controlador fuzzy envolve na sua formulação, a definição de um conjunto de variáveis lingüísticas, relacionadas por operadores lógicos (e/ou) através de sentenças gramaticais (se/então), que configuram as regras do controlador. Um exemplo típico de regra é:

1. **SE** [ Deslocamento é *Z* ] **E** [ Velocidade é *PG* ] **ENTÃO** [ Controle é *NG* ]
2. **SE** [ Deslocamento é *NG* ] **OU** [ Velocidade é *NB* ] **ENTÃO** [ Controle é *PB* ] .....

A coleção de regras é chamada base de regras. Elas são normalmente constituídas de duas parcelas, uma relacionada a condição **SE**, que é a parte antecedente da regra, e outra a condição **ENTÃO**, que é a parcela conseqüente da regra. As variáveis *Z*, *NG*, *PG*, *NB* e *PB* são variáveis lingüísticas associadas às palavras Zero, Negativo Grande, Positivo Grande, Negativo Grande e Positivo Baixo, respectivamente. Assim, nas regras do controlador fuzzy “os números são substituídos por palavras e as equações por sentenças gramaticais”.

A escolha das variáveis lingüísticas e a forma matemática associadas a estas variáveis (denominadas funções de pertinência), são importantes instrumentos de projeto na construção dos controladores fuzzy. A eficiência destes controladores depende também do mecanismo utilizado na inferência (associados aos conectivos **E** / **OU**) e na forma de agregação de todas as regras para gerar a(s) saída(s) do controlador, segundo Tsoukalas e Uhrig [1997].

Uma das maiores dificuldades no controlador fuzzy, no entanto, é a elaboração das regras e o estabelecimento do número de regras adequadas, para formarem a base de regras do controlador.

Propõe-se neste trabalho uma estratégia de otimização da base de regras usando programação genética. Seja, por exemplo, uma base com 49 regras como mostra a Tabela 01. Para cada campo definido por uma regra atribui-se um peso, que pondera a importância daquela regra dentro do conjunto de regras. O algoritmo genético busca a melhor valor para estes pesos segundo uma função de otimização estabelecida pelo projetista. Cada peso é codificado numa palavra binária de *n* bits, configurando um gene do indivíduo. No exemplo em questão, se cada peso for descrito por uma palavra de 5 bits então o cromossomo terá 245 bits como ilustra na Tabela 1.

Tabela 01 : Base de regras inicial e representação binária do peso de cada regra para formação de um indivíduo

		DESLOCAMENTO												
		NG	NM	NB	Z	PB	PM	PG						
VELOCIDADE	NG	NM	Z	NG	Z	PM	PB	PM	1ª regra	1	1	0	0	0
	NM	PB	PM	PB	PM	PB	PM	PM	2ª regra	1	0	1	0	1
	NB	PB	NG	PM	PM	PB	PB	PM	3ª regra	1	1	1	0	1
	Z	NB	PM	PM	Z	PB	PG	PB	4ª regra	0	0	0	1	1
	PB	NM	PG	NB	PM	PB	PB	PG	.....					
	PM	NM	PB	NG	NM	Z	PB	PB	47ª regra	0	1	0	1	0
	PG	NB	PM	NG	NB	NM	NG	NB	48ª regra	0	1	1	1	1
									49ª regra	0	0	0	0	1
1 cromossomo ( 1 indivíduo ) = 49 genes ( 49 pesos ) = 245 bits ( 245 alelos )														
										.....				

Portanto o valor inteiro de cada peso varia, neste exemplo, de 0 à 31, correspondendo aos genes [ 00000 ] e [ 11111 ].

Considerando a base de regras da tabela 01, que foi arbitrada sem nenhuma consideração física preliminar, o algoritmo genético parte de uma população inicial aleatória de indivíduos (cromossomos), isto é, várias palavras de 245 bits, cada um representando um conjunto de pesos diferentes. Cada indivíduo é avaliado segundo a seguinte função do erro quadrático:

$$Erro = \frac{1}{2} \sum_k (\Phi_{AG}(k) - \Phi_{ótimo}(k))^2 \quad (1)$$

onde  $\phi_{ótimo}$  é definido pelo projetista, a partir da resposta desejada para o sistema a ser controlado.

Após obter o aptidão (*fitness*) de cada indivíduo, seleciona-se os mais aptos, que são os que minimizam o erro quadrático da Equação 01. Os demais indivíduos são extintos. Com os indivíduos selecionados, escolhe-se ao acaso, n pares de indivíduos para sofrerem cruzamento. Por exemplo, sejam dois indivíduos de 25 bits escolhidos para o cruzamento. A estratégia do algoritmo genético, para cada cruzamento, é escolher aleatoriamente um ponto da palavra binária, para então trocar o material genético entre os indivíduos, como ilustrado na Figura 01. Este novo conjunto de indivíduos serão também avaliados pelo AG.

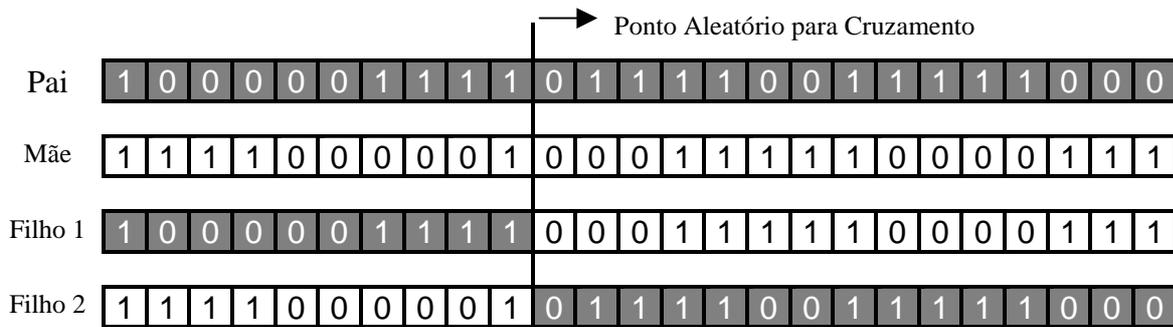


Figura 01. Esquema utilizado pelo Algoritmo Genético para realização dos Cruzamentos

Feito os cruzamento e obtido o aptidão dos descendentes, basta ao algoritmo genético realizar as mutações, com intuito de imitar as leis naturais da teoria da evolução. A mutação ocorre em n indivíduos escolhidos aleatoriamente, e consiste na troca de um ou mais bits (alelos), também escolhido ao acaso. Por exemplo, seja um indivíduo escolhido para sofrer mutação do 4º e 22º alelo (bit). Basta trocar esses alelos, se for 1 troca-se por 0 e vice e versa, conforme mostra a figura 02.

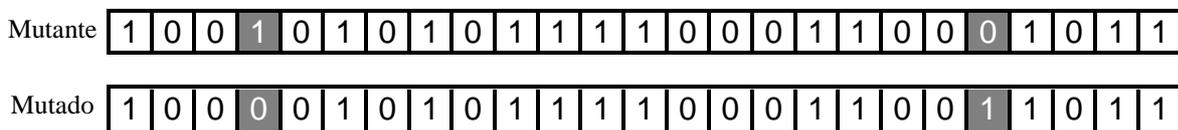


Figura 02. Esquema utilizado pelo Algoritmo Genético para realização das Mutações

Com isso o processo se repete através das gerações sucessivas. Para cada geração é feito a devida seleção natural, os cruzamentos e as mutações.

O procedimento se encerra quando um erro mínimo definido pelo projetista é atingido.

### 3. O SISTEMA VIBRATÓRIO A SER CONTROLADO

Para testar o procedimento de seleção de regras proposto, foi construído um sistema vibratório com 1 gdl., segundo Abreu *et al.* [1999]. Tal sistema é composto por uma mesa vibratória sujeita à ação de um atuador eletromagnético. A mesa é instrumentada com um sensor de proximidade, que gera informações dos deslocamentos laterais. Estas informações são usadas pelo controlador fuzzy, que gera as ações de comandos. Os sinais de comandos são enviados para um driver de corrente que alimenta um transdutor eletromagnético, que por sua vez, transforma o sinal de entrada em forças eletromagnéticas de controle. O controlador é implementado num microcomputador PC, conforme ilustra a figura abaixo.

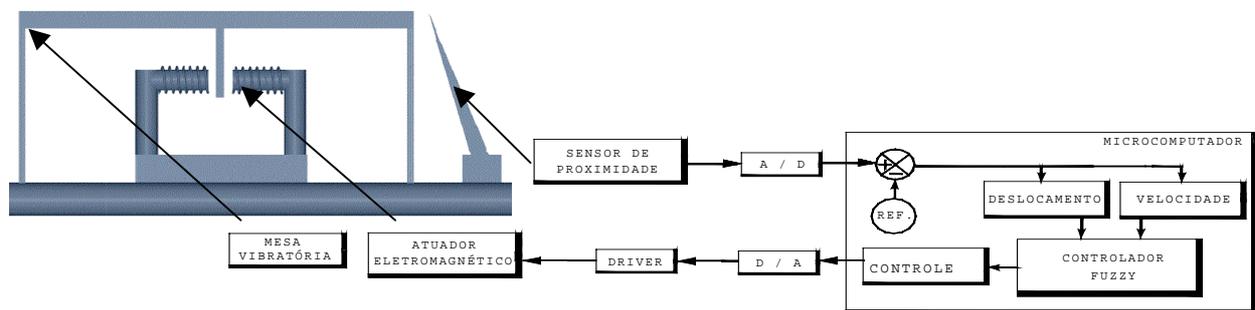


Figura 03: Componentes do sistema vibratório

O problema em questão é introduzir um amortecimento no sistema vibratório utilizando controladores fuzzy. Para tanto definiu-se 7 funções de pertinência, tanto para as 2 entradas (deslocamento e velocidade), como para saída (força de controle), todas representadas por curvas gaussianas. Cada variável de E/S foi classificada em 7 variáveis lingüísticas: Negativo Grande (NG), Negativo Médio (NM), Negativo Baixo (NB), Zero (Z), Positivo Baixo (PB), Positivo Médio (PM) e Positivo Grande (PG). O universo do discurso de cada parâmetro, isto é, o domínio de variação de cada parâmetro, foi escolhido dentro de valores reais permitido para a planta.

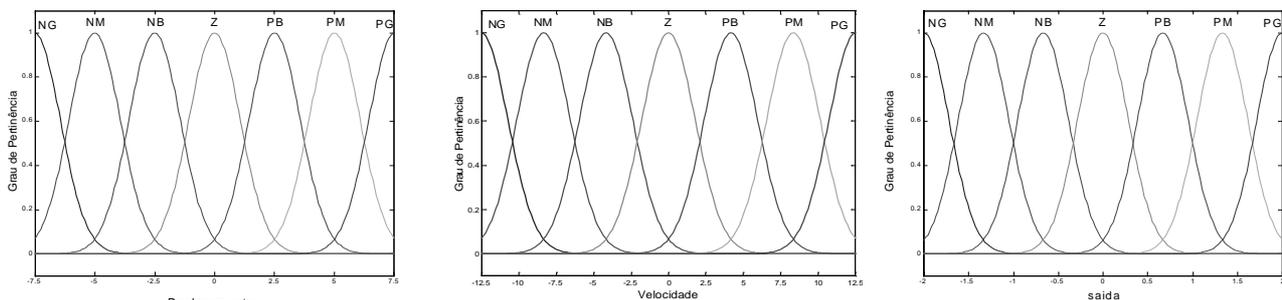


Figura 04. Curvas de Pertinência das entradas e saída do Controlador Fuzzy

O controlador utilizado é do tipo Mandani, onde a Implicação é feita com operador MIN (mínimo dos máximos), a Agregação utiliza o MAX (máximo dos mínimos), e a Defuzzificação é obtida através do cálculo do centróide. A matriz de regras inicial utilizada foi a mostrada na tabela 01. Definido estes conjuntos de variáveis do controlador, utilizou-se o AG, para otimizar o conjunto de regras.

#### 4. RESULTADOS NUMÉRICOS E EXPERIMENTAIS

A estratégia de otimização do AG demanda a seleção da função  $\phi_{\text{ótimo}}$ . Neste trabalho, adotou-se o perfil que representa o deslocamento desejado de um sistema de 2ª ordem com uma condição inicial dada, ilustrado na Figura 05. Para investigar a convergência foi preciso avaliada Equação 01, para todos os indivíduos de cada geração da evolução. Para isto foi desenvolvido um programa em Matlab, cujo fluxograma é mostrado na figura 05.

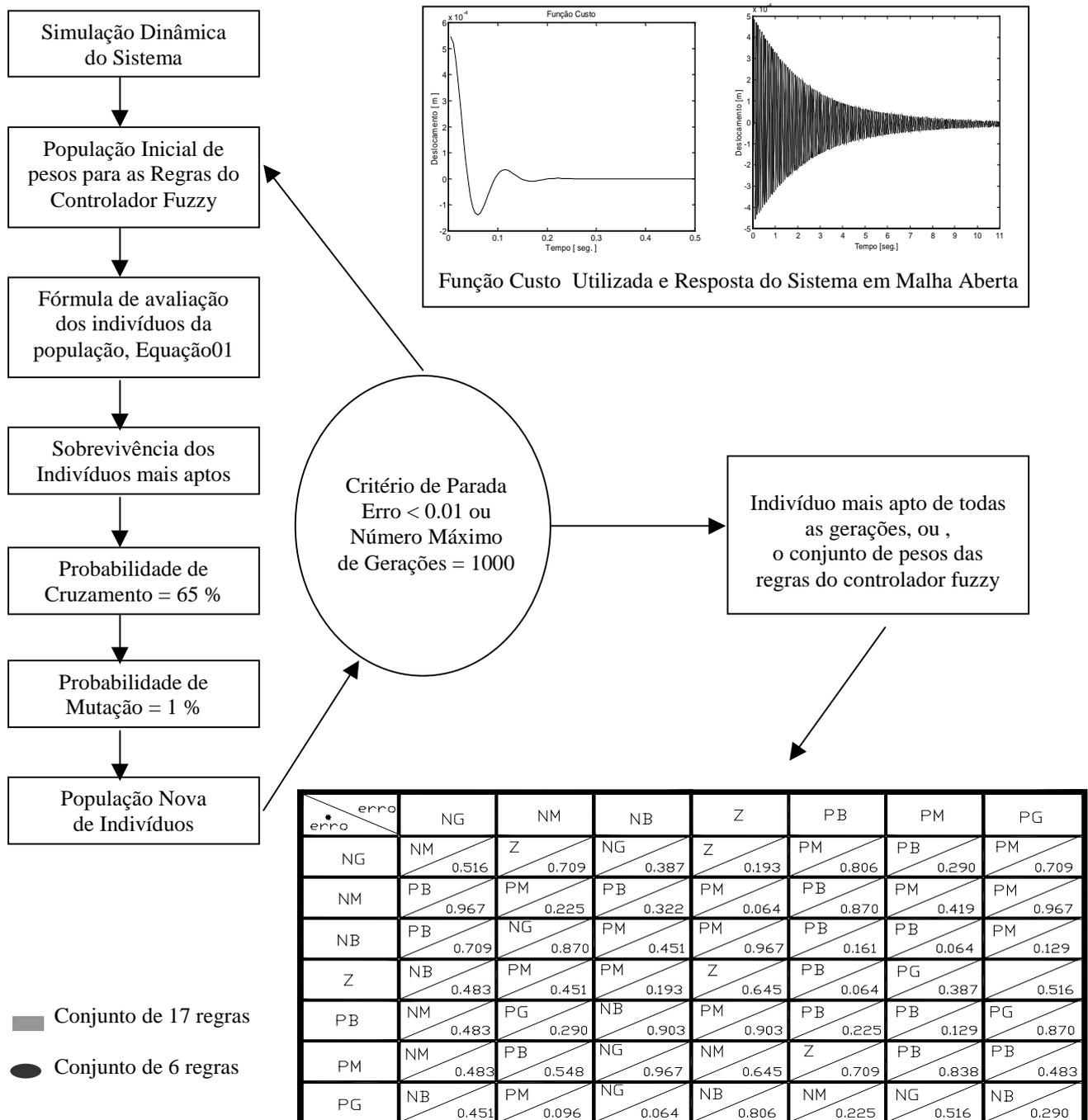


Figura 05: Fluxograma do Algoritmo Proposto, Função Custo e Resposta do Sistema em Malha Aberta e Diagrama das Regras com os Pesos Otimizados

Em posse do conjunto de regras com seus pesos otimizados, foi simulado e ensaiado experimentalmente, o controlador proposto, com diferentes números de regras. Isto é ilustrado na Figura 06.

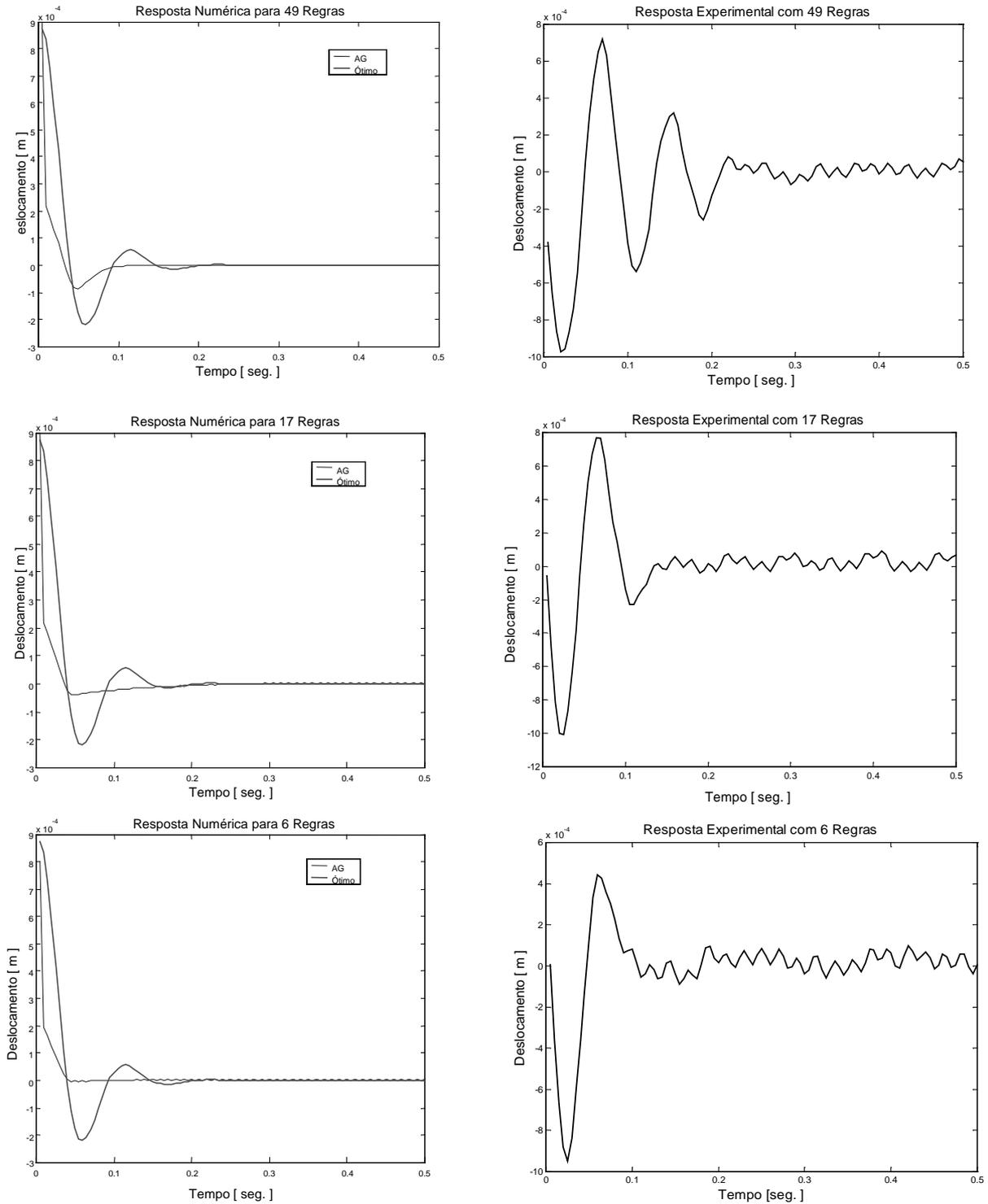


Figura 6 : Resposta Numérico e Experimental para o Controlador Proposto

Faz-se a seguir os comentários dos resultados obtidos.

## 5. CONCLUSÕES E DESDOBRAMENTOS

A análise das figuras revelam que o objetivo proposto, qual seja, introduzir amortecimento no sistema, foi plenamente satisfeito, o que pode ser comprovado quando se compara os resultados obtidos com a resposta do sistema em malha aberta ( Figura 05 ).

O tempo de acomodação do sistema foi reduzido de 11 seg. para 0.3 seg., para todos conjuntos de regras.

Observa-se a presença de um comportamento ruidoso, nos resultados experimentais, embora de pequenas amplitudes, em regime permanente. Isto já era esperado devido ao ruído do sensor, destacando que a velocidade, por limitação de hardware, foi obtida a partir de um sensor de posição.

Ressalta-se o excelente desempenho com um conjunto de regras relativamente pequeno (6 regras), o que significa em termos práticos economia de memória e aumento de velocidade de processamento.

Outro aspecto importante é que não houve preocupação preliminar em relação ao conjunto inicial de regras, ela foi definida de forma arbitrária. Caso tivesse usado algum conhecimento empírico do problema na construção da base de regras, os resultados poderiam ser ainda melhores dos que foram.

Como perspectiva para um futuro trabalho, pretende-se implementar um controlador fuzzy não somente com as regras otimizadas, mas também otimizando as funções de pertinência.

## 6. BIBLIOGRAFIA

- Abreu, G. L. C. M., Ribeiro, J. F, *Amortecedores Ativos de Vibrações Baseados em Controladores Fuzzy*, 4o SBAI – Simpósio de Automação Inteligente, SP, pp. 376-381, setembro de 1999.
- Baitinger, U.G. , K. Kropp, *Optimization of fuzzy logic controller inference rules using a genetic algorithm*. Proceedings of 1st. European Congress on Fuzzy and Intelligent Technologies, v.2, p. 1090-1096, 1993.
- Holland J. H., *Genetic Algorithms*, Scientific America, Vol. 267, pp.66-72,1992.
- Homaifar, A, E. McCormick . *Simultaneous desing of membership functions and rule sers for fuzzy controllers using genetic algorithms*. IEEE Transactions on Fuzzy System, v.3, p. 129-139, 1995.
- Uhrig, R.E., Tsoukala, *The approaches of Neuro-Fuzzy in Engineering*, 1997.
- Yván J. Túpac, Marco Aurélio Pacheco, Marley Velasco, Ricardo Tanscheit, *Geração do conjunto de regras de interferência para um controlador nebuloso usando algoritmos genéticos*, 4o SBAI – Simpósio de Automação Inteligente, SP, Setembro de 1999.
- Zadeh, L. A., 1965, *Fuzzy Sets, Information and Control*, Vol. 8, pp. 338-353.