

## IDENTIFICAÇÃO DE PARÂMETROS POR ENXAME DE PARTÍCULAS ATRAVÉS DO COMPROMISSO ENTRE VARIÂNCIA E POLARIZAÇÃO

LÚCIA VALÉRIA R. ARRUDA, LUIS ANTONIO D. ISFER, HEITOR SILVERIO LOPES

*Laboratório de Automação e Sistemas de Controle Avançado, Programa de Pós-Graduação em Engenharia Elétrica e Informática Industrial, Universidade Tecnológica Federal do Paraná CPGEI/UTFPR, Av. 7 de Setembro, 3165, Bloco D, 3º andar, 80230-901, Curitiba, PR, Brasil*  
*lvrrarruda@utfpr.edu.br, tomisfer@yahoo.com.br, hslopes@utfpr.edu.br*

**Abstract** - This paper proposes the application of the particle swarm optimization (PSO) technique to the problem of identification of reduced order models for complex plants. The proposed algorithm seeks to represent the real plant by a polynomial model whose poles and zeros are estimated from system's input and output data. This paper presents an alternative proposal based on PSO for the representation of individuals and the fitness function, which uses a compromise between bias and variance errors between the estimated output and the measured data. Finally, the results illustrate the effectiveness of this proposal compared with those of the literature.

**Keywords** – Particle Swarm Optimization, Bias, Variance, System Identification, Reduced Order Model

**Resumo** - O presente trabalho propõe a aplicação da técnica de otimização por enxame de partículas (PSO) ao problema de identificação de modelos de ordem reduzida para plantas complexas. O algoritmo proposto representa a planta real através de um modelo polinomial cujos pólos e zeros são estimados a partir dos dados de entrada e saída do sistema. O presente trabalho apresenta uma proposta alternativa baseada em PSO para a representação dos indivíduos e para a função de *Fitness*, que utiliza o compromisso entre os erros de polarização e de variância entre a saída do modelo estimado e os dados reais. Por fim, os resultados apresentados ilustram a eficiência da presente proposta em comparação com os resultados da literatura.

**Palavras-chave** – Otimização por Enxame de Partículas, Polarização, Variância, Identificação de Sistemas, Modelos de Ordem Reduzida

### 1. Introdução

A cada dia que passa, sistemas industriais estão mais complexos fazendo com que o controle e a supervisão de tais sistemas tornem-se cada vez mais difíceis. Na modelagem de sistemas para fins de controle, tal complexidade ocasiona problemas na escolha da melhor estrutura para o modelo que represente corretamente o sistema nas condições desejadas. As técnicas clássicas de estimação baseadas no método do gradiente (mínimos quadrados, máxima verossimilhança, entre outros) costumam produzir resultados polarizados quando se utiliza modelos de ordem reduzida em relação ao sistema, a fim de diminuir a complexidade do sistema a ser controlado. Neste contexto, uma alternativa para estimação de parâmetros de modelos polinomiais é utilizar técnicas de inteligência computacional que dispensem o cálculo de derivadas da função objetivo e considerações sobre a natureza estocástica da perturbação afetando os dados usados na identificação (Arruda, et al, 2003).

Tendo em vista esta discussão, este artigo desenvolve uma metodologia de obtenção de um modelo polinomial simplificado de um sistema real complexo baseada na técnica metaheurística de otimização por enxame de partículas (Particle Swarm Optimization – PSO). Uma das representações matemáticas de um sistema, mais comumente usadas quando o objetivo do modelo identificado é a síntese de controladores, é o uso de funções de transferência com pólos e zeros (Zhu, 2001). Entretanto, a identificação direta de

pólos e zeros de funções de transferência exige o uso de técnicas avançadas de otimização baseadas nos métodos de Gauss-Newton e Levenberg-Marquadt, pois constitui um problema de estimação não-linear. O método proposto neste trabalho pretende contornar este problema, gerando um algoritmo de estimação que não leva em conta cálculos intermediários que dependam da ordem do modelo, ou de um mau condicionamento da matriz Hessiana da função custo.

### 2. Modelagem do problema

Considerando que o sistema a ser modelado pode ser descrito por um modelo ARX (Ljung, 1999), dado na equação abaixo:

$$A(q^{-1})y(t) = B(q^{-1})u(t-d) + e(t) \quad (2.1)$$

onde,  $A(q^{-1})$  e  $B(q^{-1})$  são polinômios mônicos, representados através do operador deslocamento  $q^{-1}$ , onde  $d$  é o atraso de transporte, e os sinais  $u(t)$ ,  $e(t)$  e  $y(t)$  representam respectivamente a entrada, o ruído e a saída. O ruído é uma sequência aleatória com distribuição de probabilidade supostamente conhecida. Em geral esta distribuição é assumida normal com média zero e variância ( $\sigma_e^2$ ). A ordem do modelo ARX é composta pela soma das ordens dos polinômios  $A(q^{-1})$  e  $B(q^{-1})$ . O objetivo da identificação é obter as raízes dos polinômios  $A(q^{-1})$  e  $B(q^{-1})$ , isto é os pólos e zeros do modelo a partir dos dados de entrada  $u(t)$  e saída  $y(t)$  (Ljung, 1999).

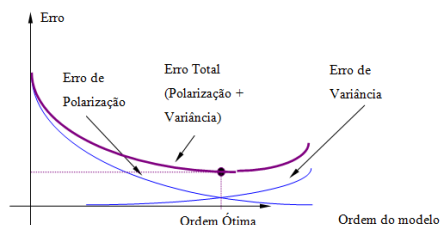
Para observar a eficiência de representação do modelo em relação ao sistema real, define-se a sequência  $\eta(t)$ , também chamada de erro “mismatch”, que representa o erro previsto um passo a frente entre a saída real e a saída estimada (Ljung, 1999):

$$\eta(t) = y(t) - \hat{y}(t) \quad (2.2)$$

$$\hat{A}(q^{-1})\hat{y}(t) = \hat{B}(q^{-1})u(t-d) \quad (2.3)$$

onde  $\hat{y}(t)$  é a saída de um sistema ARX estimado quando aplicada a entrada  $u(t)$  e  $\hat{A}(q^{-1})$  e  $\hat{B}(q^{-1})$  são as estimativas dos polinômios  $A(q^{-1})$  e  $B(q^{-1})$  encontradas para minimizar o critério escolhido.

O erro “mismatch” em geral é formado por duas parcelas: erro de polarização e erro de variância (Ljung, 1999). A primeira parcela ocorre se o sistema real não pode ser representado integralmente pela estrutura do modelo escolhido, como é o caso dos modelos de ordem reduzida. A segunda parcela é devida à presença do ruído sobreposto aos dados observados. Segundo Ljung (1999), o erro de polarização diminui com o aumento da ordem do modelo. Em contrapartida, o erro de variância aumenta com a ordem do modelo, apesar de diminuir com o número de dados observados. A figura (2.1) demonstra este comportamento (Arruda, et al, 2003), onde observa-se a existência de uma ordem de modelo ótima que equilibra a diminuição do erro de polarização com o aumento do erro de variância.



**Figura 2.1:** Compromisso entre os erros de polarização e variância

Para encontrar o equilíbrio entre os dois erros, em geral é necessário executar várias vezes o algoritmo de estimação, utilizando diferentes ordens dos polinômios  $A$  e  $B$ , obtendo diferentes modelos. Feito isto, deve-se construir um gráfico com o erro previsto para estes diferentes modelos, a fim de se determinar visualmente o ponto de inflexão da curva, que corresponde à ordem ótima do modelo. Por outro lado, ao se utilizar o estimador por enxame de partículas, tal busca pelo ponto de inflexão é feita diretamente, identificando ao mesmo tempo a ordem ótima (estrutura) e os parâmetros (pólos e zeros) do modelo.

Para a representação do modelo por pólos e zeros, os polinômios  $A(q^{-1})$  e  $B(q^{-1})$  são fatorados, conforme as equações (2.4) e (2.5):

$$A(q^{-1}) = (1 - p_1 q^{-1}) \dots (1 - p_n q^{-1}) \quad (2.4)$$

$$B(q^{-1}) = (1 - z_1 q^{-1}) \dots (1 - z_m q^{-1}) \quad (2.5)$$

onde:

$p_i$  = pólo de  $A(q)$ ,  $i = 1, 2, \dots, n$

$z_j$  = zero de  $B(q)$ ,  $j = 1, 2, \dots, m$

$n$  = n° máximo de raízes (pólos) de  $A(q^{-1})A(q^{-1})$

$m$  = n° máximo de raízes (zeros) de  $B(q^{-1})B(q^{-1})$

Substituindo e rearranjando a equação (2.1) pode-se definir a função de transferência na forma de pólos e zeros como:

$$G(q^{-1}) = \frac{B(q^{-1})}{A(q^{-1})} = \frac{(1 - z_1 q^{-1}) \dots (1 - z_m q^{-1})}{(1 - p_1 q^{-1}) \dots (1 - p_n q^{-1})} \quad (2.6)$$

Deve-se ressaltar que os pólos e zeros podem assumir tanto valores reais como complexos, e que a existência de pólo ou zero complexo implica automaticamente na existência de seu conjugado.

### 3. Estimador por Enxame de Partículas

Os principais aspectos inovadores do algoritmo de estimação por enxame de partículas propostos neste trabalho são o espaço de busca, a representação utilizada e a função de *Fitness* desenvolvida. A relação entre estes fatores e a qualidade do modelo é nítida: o espaço de busca proporciona a precisão desejada ao modelo estimado; a representação por sua vez, acarreta ao estimador a versatilidade em termos de estrutura do modelo; por fim, a função de *Fitness* garante que o modelo estimado é de ordem ótima, através do compromisso entre os erros de polarização e de variância (Arruda, et al, 2003).

#### 3.1 Representação dos indivíduos

Os sistemas a serem modelados são discretos e estáveis, apresentando assim, pólos e zeros dentro do círculo unitário mostrado na figura (3.1). A partir desta figura, o espaço definido para a busca por pólos e zeros reais é a reta EF, no eixo dos números reais, enquanto que para os pólos e zeros complexos o espaço de busca corresponde à área compreendida no quadrado unitário ABCD. Este espaço de busca é maior que o círculo unitário, mas o uso de uma representação em coordenadas cartesianas, ao

invés de coordenadas polares, simplifica o problema.

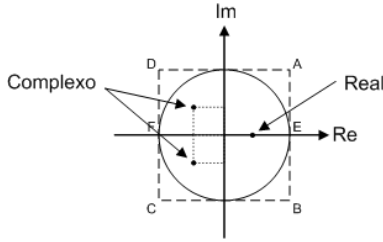


Figura 3.1: Representação do espaço de busca

O modelo proposto para identificação através de enxame de partículas é então:

$$G(q^{-1}) = K \cdot \frac{[1 - m_1 \cdot (a_1 \pm j \cdot b_1) \cdot q^{-1}]}{[1 - m_3 \cdot (c_1 \pm j \cdot d_1) \cdot q^{-1}]} \cdots \frac{[1 - m_2 \cdot (a_2 \pm j \cdot b_2) \cdot q^{-1}]}{[1 - m_4 \cdot (c_2 \pm j \cdot d_2) \cdot q^{-1}]} \quad (3.1)$$

onde:

$K$  = ganho em malha aberta

$a_i$  = parte real do  $i$ -ésimo zero

$b_i$  = parte imaginária do  $i$ -ésimo zero

$c_i$  = parte real do  $i$ -ésimo pólo

$d_i$  = parte imaginária do  $i$ -ésimo pólo

$m_l$  = multiplicador do  $l$ -ésimo pólo/zero

$z_i$  =  $i$ -ésimo zero

$p_i$  =  $i$ -ésimo pólo

$i$  = índice para pólos e zeros

$l$  = índice para os multiplicadores de pólos/zeros

Cada partícula do enxame será composta de treze parâmetros e fará sua busca nas treze dimensões do espaço de parâmetro:

Partícula  $\Rightarrow (a_1, a_2, b_1, b_2, c_1, c_2, d_1, d_2, K, m_1, m_2, m_3, m_4)$

A partir desta partícula, é possível representar quatro famílias de modelos, de acordo com a presença ou não de pólos ou zeros reais ou por pares complexos conjugados. Os multiplicadores " $m_l$ " permitem a modificação da estrutura do modelo, para ser adaptado conforme a estrutura do sistema real e podem assumir somente valores nulos ou unitários.

### 3.2 Espaço de Busca

O espaço de busca utilizado compreende as treze dimensões e seus limites inferiores e superiores, como representado a seguir:

$$\begin{aligned} 0.0001 &\leq K \leq 100 \\ -1 &\leq a_i \leq 1 \\ -1 &\leq b_i \leq 1 \\ -1 &\leq c_i \leq 1 \\ -1 &\leq d_i \leq 1 \\ m_l &\in \{0,1\} \end{aligned} \quad (3.2)$$

Considerando uma precisão,  $\epsilon$ , de 0,0001, a dimensão do espaço de busca pode ser calculada como:

$$\text{Dimensão} \approx (2 \times 10^4)^8 \times 10^6 \times 2^4 \approx 4,1 \times 10^{41}$$

### 3.3 Função de Fitness

Tomando como base a função de *Fitness* proposta por Arruda, et al, (2003), foi desenvolvida uma nova função, composta por duas parcelas, representando os erros de polarização e de variância:

$$F = \left( \sum_{i=1}^N |\eta_i| \right) \cdot \left( e^{\|1 - C_{xy}\|} - 1 \right) \quad (3.3)$$

O primeiro termo da equação representa os erros de polarização, onde  $N$  é o número de amostras e  $\eta$  é o erro "mismatch", descrito na equação (2.2). O segundo termo corresponde ao coeficiente de correlação  $C_{xy}$  entre os dados de saída experimentais e os obtidos pelo modelo PSO, representado assim o erro de variância.

### 3.4 Características do Algoritmo PSO

A população inicial utilizada foi gerada de forma aleatória dentro dos limites inferiores e superiores do espaço de busca estipulados em . As velocidades máximas para cada dimensão da partícula foram definidas como o intervalo para cada dimensão da partícula. O critério de parada adotado foi o número máximo de iterações, pois o valor mínimo de *Fitness* é difícil de ser calculado *a priori*.

A equação de atualização das coordenadas de cada partícula compreende a realização da movimentação da partícula no espaço de busca com base na velocidade calculada. O cálculo da velocidade utiliza a posição atual da partícula, a velocidade na iteração anterior e também a posição da melhor partícula de toda a população. A equação é empregada para a obtenção da velocidade da partícula na iteração atual (Eberhart e Shi, 2001). De posse da velocidade pode-se então mover a partícula para um novo ponto do espaço de busca, utilizando a equação.

$$V_i^{t+1} = \omega_{atual} \cdot V_i^t + c_1 r_1 (pbest_i^t - x_i^t) \dots + c_2 r_2 (gbest_i^t - x_i^t) \quad (3.4)$$

$$x_i^{t+1} = x_i^t + V_i^{t+1} \quad (3.5)$$

A variável  $x$  representa a solução candidata atual, onde o índice  $i$  se refere a cada

uma das treze dimensões do problema, e o índice  $t$  denota as iterações. Os parâmetros  $r_1$  e  $r_2$  são valores aleatórios com distribuição normal entre zero e um. Os parâmetros  $pbest$  e  $gbest$  são as melhores soluções armazenadas tanto pela partícula (*personal best*) como pela população inteira (*global best*). As variáveis  $c_1$  e  $c_2$ , chamadas respectivamente de parâmetros de atração cognitiva e social, são duas constantes de aceleração, e possuem grande influência sobre o comportamento das partículas durante o processo de busca (Eberhart e Shi, 2001).

Na equação o primeiro termo representa o momento de inércia das partículas no instante  $t$  e é composto pelo fator de inércia  $\omega$  e pela velocidade de cada dimensão  $i$  da partícula no instante  $t$  (Eberhart e Shi, 2001). O segundo termo é chamado de componente cognitivo e possibilita que uma partícula seja atraída para o melhor ponto por ela encontrado. A constante de aceleração  $c_1$  presente neste termo, em caso de um valor muito alto, ocasiona na divisão da população em pequenos grupos, gerando vários pontos ótimos locais.

Já o último termo, chamado de componente social, reflete o comportamento coletivo da partícula, quantificando a distância entre a partícula e o melhor ponto encontrado por toda a população. A presença da constante de aceleração  $c_2$  neste termo influencia na velocidade de convergência das partículas. Valores muito altos para esta constante podem causar uma convergência prematura devido ao deslocamento forçado a que as partículas são submetidas.

Para a aplicação da técnica de PSO, foi utilizado a toolbox de PSO do MATLAB, na sua versão 7.9.0, desenvolvido por Sam Chen (2009).

O fator de inércia do PSO é corrigido ao longo das iterações. No início, quando as partículas estão bastante espaçadas umas das outras e com altíssima diversidade, o fator de inércia é maior, fazendo com que elas tenham longos deslocamentos no espaço de busca, proporcionando uma varredura maior e melhorando a busca global. À medida que as iterações vão aumentando, o fator de inércia vai diminuindo, de modo que cada partícula passa a se deslocar com menor intensidade, refinando a busca local. A equação abaixo rege a correção do fator de inércia:

$$\omega_{atual} = \omega_{inicial} - \frac{f \cdot (I_{atual} - 1)}{I_{total} - 1} \quad (3.6)$$

onde:

$\omega_{atual}$  é o fator de inércia na iteração corrente;

$\omega_{inicial}$  é o fator de inércia inicial;

$f$  é um fator de ponderação que expressa a porcentagem que deve ser diminuída do fator de inércia inicial;

$I_{atual}$  é a iteração corrente;

$I_{total}$  é o número total de iterações.

Tabela 3.1: Parâmetros do PSO

Parâmetros	Valor
População de Partículas	100
Numero de Iterações	600
Parâmetro de atração cognitiva - $c_1$	1,5
Parâmetro de atração social - $c_2$	2,0
Fator de Inércia inicial - $\omega$	0,5
Velocidade Máxima - $a_i, b_i, c_i, d_i$	2,0
Velocidade Máxima - $K$	100,0
Velocidade Máxima - $m_i$	1,0
Fator de Ponderação - $f$	0,2

A escolha dos parâmetros para construção do algoritmo de busca por PSO se baseou nos valores usuais encontrados na literatura.

Para quantificar a discrepância final entre os pontos obtidos pelo estimador e os dados reais, foi utilizado um cálculo de erro percentual sobre todos os dados utilizados:

$$erro(\%) = 100 \sum_{i=1}^{no.iteração} \frac{|\hat{Y}(i) - Y|}{no.iteração} \quad (3.7)$$

onde  $\hat{Y}(i)$  corresponde ao vetor com as saídas estimadas pelo modelo PSO na iteração corrente  $i$ ,  $Y$  corresponde ao vetor de saídas reais, e  $no.iterações$  é o valor final de iterações.

#### 4. Resultados e Discussões

O sistema utilizado para a aplicação do estimador por enxame de partículas foi o mesmo proposto em (Arruda, et al, 2003), e consiste de um sistema de 9ª ordem (seis pólos e três zeros) e apresenta um atraso de 5, e deve ser estimado por um modelo com no máximo 4 pólos. Tal fato induz a presença de dinâmicas que não podem ser modeladas por limitações do modelo. A função de transferência do sistema real corresponde à equação (4.1). O sinal de entrada é um ruído branco de amplitude 10, média 0,2483 e variância 10,7108, enquanto que o distúrbio corresponde à 10% da amplitude do sinal de entrada, valor usual em plantas industriais, com variância de 1,6066 e média 0,0962. Com isto é possível comparar o modelo estimado com PSO com o modelo estimado usando algoritmos genéticos (GA) desenvolvido por Arruda e colaboradores (2003), o qual é dado na equação (4.3).

Após 600 iterações, e tendo levado aproximadamente 20 minutos para realizar a

busca, a melhor resposta obtida após várias simulações foi:

Partícula final => [0,46657 0,98914 -0,35309 -0,2536 0,84198 0,61276 0,045794 -0,26337 0,38786 1,0000 1,0000 1,0000 1,0000]

$$G(z) = \frac{0,01 \cdot z^3 + 0,0074 \cdot z^2 + 0,000924 \cdot z - 0,00001762}{z^8 - 2,14 \cdot z^7 + 1,3049 \cdot z^6 + 0,0963 \cdot z^5 - 0,3467 \cdot z^4 + 0,1097 \cdot z^3 - 0,0105 \cdot z^2} \quad (4.1)$$

$$G_{PSO}(z) = 0,3879 \cdot \frac{(z^2 - 1,978 \cdot z + 1,043) \cdot (z^2 - 0,9331 \cdot z + 0,3424)}{(z^2 - 1,684 \cdot z + 0,711) \cdot (z^2 - 1,226 \cdot z + 0,4448)} \quad (4.2)$$

$$G_{AG}(z^{-1}) = \frac{0,0055}{[1 - (0,7064 \pm 0,3111 \cdot i) \cdot z^{-1}] \cdot [1 - (0,8678 \pm 0,0641 \cdot i) \cdot z^{-1}]} \quad (4.3)$$

O modelo final obtido através de enxame de partículas corresponde à equação (4.2). O ganho, os zeros e os pólos desse modelo são:

$$K = 0,38786$$

$$z_1 = 0,46657 \pm 0,35309 \cdot i \quad e \quad z_2 = 0,98914 \pm 0,2536 \cdot i$$

$$p_1 = 0,84198 \pm 0,045794 \cdot i \quad e \quad p_2 = 0,61276 \pm 0,26337 \cdot i$$

$$m_1 = m_2 = m_3 = m_4 = 1,0000 \quad (4.4)$$

Os resultados qualitativos da função de *Fitness* podem ser visualizados na figura (4.1), onde se observa o comportamento das partículas para a melhor, a média e a pior resposta da função de *Fitness*, ao longo das iterações. A figura (4.2) mostra uma aproximação para melhor visualização da curva dos melhores pontos, em preto. Nestas duas figuras pode-se perceber uma grande flutuação inicial, pois o fator de inércia neste período é maior, e também que à medida que as iterações foram acontecendo este valor de inércia foi sendo diminuído.

Para comparar os resultados obtidos pelo estimador por enxame de partículas, foi produzido um gráfico com os dados de saída das funções de transferência, equações (4.1), (4.2) e (4.3). O resultado está na figura (4.3). Nota-se nesta figura a precisão dos estimadores GA e PSO, pois ambos interpolam bem o sistema real. Observa-se ainda nesta figura que o erro *mismatch* do modelo PSO é extremamente baixo, evidenciando uma boa estimação de parâmetros e também uma boa escolha da estrutura do modelo.

O valor final obtido para a função de *Fitness*, equação (3.3), foi de  $9,4279 \cdot 10^{-5}$ , sendo que o erro de polarização encontrado foi de 1,7058 e o coeficiente de correlação ( $C_{xy}$ ) 0,999, gerando um erro de variância de  $5,5271 \cdot 10^{-5}$ . Desta forma pode-se afirmar que a estrutura do modelo é a melhor possível. O erro percentual final, equação (3.7), obtido entre os dados de saída do sistema real e do modelo gerado foi de  $3,7978 \cdot 10^{-4}$ , enquanto que para o modelo da literatura (GA) o erro foi de  $4,3 \cdot 10^{-3}$ . A média e a variância dos dados de saída do modelo obtido foram respectivamente 0,3182 e 1,8279 e dos dados reais foram 0,3181 e 1,8255, enquanto que para o modelo GA a média foi de 0,3314 e a

variância 1,8435, o que mostra que os resultados via estimador PSO foram mais próximos da realidade.

O modelo obtido com PSO possui quatro pólos e quatro zeros, todos complexos (eq. 4.2) enquanto que o modelo GA correspondendo à equação (4.3) possui quatro pólos complexos e nenhum zero. Já o sistema real possui, como dito acima, 6 pólos, sendo 4 reais e um par complexo, e 3 zeros. Os pólos e zeros das funções de transferência (4.1), (4.2) e (4.3) podem ser observados através do diagrama da figura (4.4). Observa-se nesta figura que ambos os modelos encontraram pólos em regiões próximas, entre si e em relação aos pólos do sistema, o que comprova a boa aproximação das dinâmicas principais do sistema real.

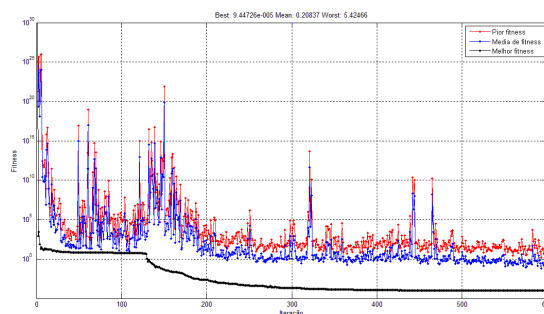


Figura 4.1: Curva de *Fitness* versus Iteração

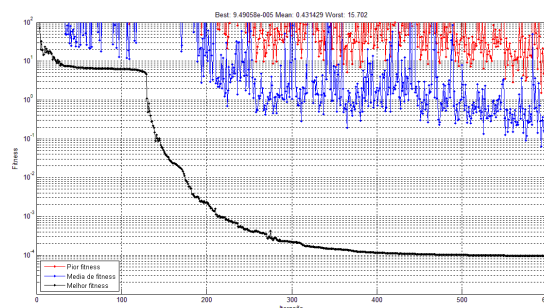


Figura 4.2: Curva de *Fitness* versus Iteração aproximada

Por fim, a figura (4.5) apresenta o comportamento dos dados de saída no domínio da



freqüência, onde se observa uma pequena diferença entre as curvas do sistema real e do modelo obtido em relação às do modelo da literatura.

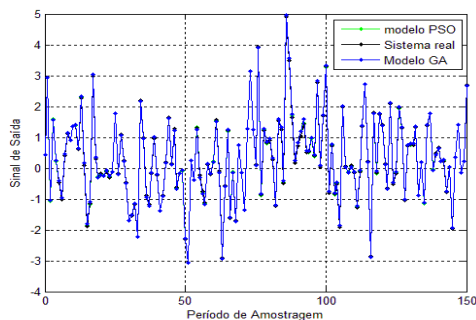


Figura 4.3: Sinal de saída versus Período de Amostragem

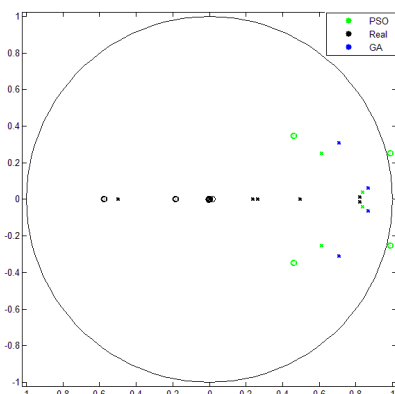


Figura 4.4: Diagrama do lugar das raízes para pólos e zeros

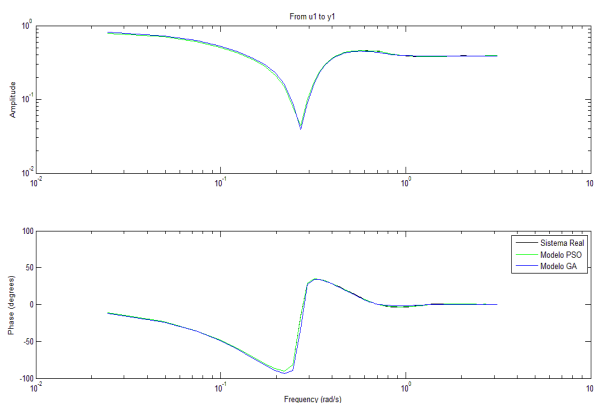


Figura 4.5: Análise Espectral dos Modelos

## 5. Conclusão

Este artigo apresentou um método para estimação de pólos e zeros de funções de transferência baseado em otimização por enxame de partículas. O modelo produzido por este

estimador possui uma estrutura que minimiza o compromisso entre os erros de polarização e de variância, graças a uma boa escolha da representação dos indivíduos e também ao desenvolvimento da função de *Fitness* utilizada, que incorpora satisfatoriamente este compromisso.

Apesar de o exemplo apresentado ter sido aplicado à identificação de um sistema linear, o método proposto aproxima bem sistemas não lineares e de ordem altas e constitui uma alternativa confiável e de fácil implementação frente aos métodos tradicionais de identificação não-linear de parâmetros, baseados em derivadas de funções objetivo e matrizes Hessianas.

Os resultados obtidos comprovam a eficiência do método proposto em comparação com o desenvolvido na literatura, pois foi possível representar todo o comportamento do sistema real, sem registrar nenhuma perda significativa no modelo encontrado.

Por fim, destaca-se o sucesso do método em termos de esforço computacional e tempo de processamento, mostrando a viabilidade de aplicação em problemas industriais.

## 6. Agradecimentos

Os autores agradecem o apoio da Coordenação de Aperfeiçoamento de Pessoal de Nível Superior (CAPES).

## 7. Referências Bibliográficas

- Arruda, L. V. R.; Fávaro, S.; Neves-Jr., F.; Identificação Genética de Modelos por Pólos e Zeros baseada no Compromisso entre os Erros de Polarização e Variância. *Revista Controle e Automação*, Vol. 2 nº2, pp 93-102, 2003.
- Eberhart, R. C.; Shi, Y.; Particle swarm optimization: developments, applications and resources. *Proceedings of the IEEE Congress on Evolutionary Computation (CEC 2001)*, Seoul, Korea, 2001.
- Ljung, L.; *System Identification: Theory for the User*. New Jersey: Prentice Hall, Inc., 2<sup>nd</sup> ed., 1999.
- Zhu, Y.; *Multivariable System Identification for Process Control*. Eindhoven: Elsevier Science & Technology Books, 1<sup>st</sup> ed., 2001.