

Identificação Paramétrica de Motor CC Utilizando Algoritmos Metaheurísticos e Mínimos Quadrados

Ewerton C. L. Oliveira *, Sandio M. Santos*, Francisco E. A. Félix Júnior*,
Marcelino S. Silva * and, Jasmine P. L. Araújo*

Email: ewerton.o43@gmail.com, sandio.maciell@gmail.com, jralbbuquerque@gmail.com,
marcelino@ufpa.br, jasmine.araujo@gmail.com

* Instituto de Tecnologia, Universidade Federal do Pará, Belém, Brasil

Resumo—Um assunto de grande importância na engenharia e na ciência é como obter modelos matemáticos que representem sistemas e que sejam capazes de simular a dinâmica e o funcionamento dos mesmos para possíveis análises e projeto de controladores. Este trabalho tem como objetivo abordar a aplicação de algoritmos metaheurísticos de inteligência de enxame e dos Mínimos Quadrados (MQ), desenvolvidos e simulados no software Matlab, com o propósito de realizar a identificação *off-line* da planta de um motor de corrente contínua (CC) para entrada degrau unitário e sinal binário pseudoaleatório (SBPA), totalizando seis testes. Também é objetivo do trabalho realizar uma análise estatística e comparativa dos resultados das metaheurísticas para os critérios de precisão, acurácia e tempo médio de processamento após 30 simulações de identificação da planta analisada, e para o MQ apenas em termos de acurácia e tempo de processamento. As inteligências de enxame utilizadas neste trabalho correspondem a Otimização por Enxame de Partículas ou PSO (*Particle Swarm Optimization*), e o PSO Adaptativo ou APSO (*Adaptive Particle Swarm Optimization*). Os resultados alcançados mostram que as metaheurísticas obtiveram melhor acurácia em relação ao MQ, todavia, este alcançou o menor tempo de processamento entre as técnicas utilizadas.

Palavras-chave—Identificação *off-line*, Avaliação de Desempenho, Motor CC, PSO, APSO.

I. INTRODUÇÃO

Com o avanço da ciência e da tecnologia, e de suas consequências na engenharia e na sociedade, novos processos cada vez mais complexos são criados, e com isso, surge a necessidade de conhecer os modelos matemáticos que regem suas dinâmicas para auxiliarem no processo de análise, e no projeto de controladores para estes sistemas [3].

Atualmente, motores CC podem ser encontrados em diversas aplicações, sejam elas residências, comerciais ou mesmo industriais [6]. Conhecer os parâmetros deste sistema pode contribuir tanto no processo de controle deste, quanto na supervisão e monitoramento para diagnóstico de falhas.

Na classe de identificação de sistemas com algoritmos determinísticos pode-se citar: Mínimos Quadrados (MQ), Mínimos Quadrados Recursivos (MQR), Métodos de Subespaço, Filtro de Kalman [1], Método da Variável Instrumental Recursiva, Método da Função Correlativa [9]. Embora estes algoritmos possuam relativa eficiência no processo de identificação, em vários casos, eles recaem no mínimo local em seu método de estimação paramétrica para o modelo, podendo gerar erros de estimação significativos. Além de que a identificação por

algoritmos determinísticos é um campo diversificado, pois dependendo da natureza do sistema, existe um algoritmo diferente que melhor se enquadra à resolução do problema [12].

Por outro lado, algoritmos metaheurísticos de busca estocástica apresentam certas vantagens na resolução destes problemas, uma vez que a maioria deles possuem mecanismos que visam contornar o mínimo (ou máximo) local, otimizando a busca pelo valor ótimo em seu intervalo de busca, embora isso demande um tempo computacional maior. Outra vantagem no uso das metaheurísticas recai na característica genérica que elas possuem na resolução de problemas de vários ramos da ciência, da engenharia e outros campos.

Metaheurísticas como Algoritmo Genético (AG), Otimização por Enxame de Partículas (PSO), PSO com Construção (PSO-Co), PSO Adaptativo (APSO) e outras, são bastante utilizadas na identificação de sistemas lineares [4] e não lineares [5], além de suas respectivas aplicações em ajustes de controladores do tipo PID [10] e LQ [8].

Portanto, este trabalho tem por objetivo apresentar a aplicação de algoritmos MQ, PSO e APSO na identificação *off-line* da planta do sistema de um motor de corrente contínua (CC), além de abordar uma análise estatística de desempenho dos resultados de cada algoritmo.

II. MODELAGEM MATEMÁTICA DE UM MOTOR CC

Com a finalidade de converter potência elétrica em potência mecânica, surgiram os primeiros motores elétricos de corrente contínua (CC), com testes realizados inicialmente por Michael Faraday [5]. Motores CC são largamente usados em eletrodomésticos, no setor comercial e industrial. A Fig. 1 mostra o diagrama eletromecânico de um motor CC.

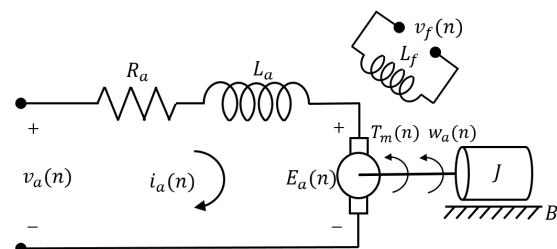


Figura 1. Diagrama eletromecânico do motor CC.

A Tabela I abaixo resume cada um dos parâmetros que compõe o mecanismo eletromecânico do motor mostrado na Fig. acima.

Tabela I
PARÂMETROS ELETROMECÂNICOS DO MOTOR CC.

Descrição	Sigla	Unidades
Resistência da armadura	R_a	Ω
Indutância da armadura	L_a	H
Momento de inércia	J	Nms^2
Coefficiente de atrito viscoso	B	Nm/rad
Corrente de armadura	i_a	A
Tensão de armadura	v_a	V
Tensão induzida	E_a	V
Torque	T_m	Nm
Velocidade angular do eixo	w_a	rad/s

A tensão induzida nas espiras do motor é definida como sendo diretamente proporcional à velocidade de rotação, $w_a(n)$, Equação (1), onde K_e definida como constante de força contra eletromotriz, sendo medida em [Vs/rad].

$$E_a(n) = K_e w_a(n) \quad (1)$$

O torque desenvolvido no motor CC é diretamente proporcional à intensidade da corrente que circula na armadura, Equação (2), sendo K_t definida como constante de torque, medida em [Nm/A].

$$T_m(n) = K_t i_a(n) \quad (2)$$

Aplicando a segunda lei de Kirchhoff no circuito da Fig. 1 e a segunda lei de Newton para sistemas rotacionais obtém-se, as Equações (3) e (4) respectivamente.

$$v_a(n) = R_a i_a(n) + L_a \frac{di_a(n)}{dn} + E_a(n) \quad (3)$$

$$T_m(n) = J \frac{dw_a(n)}{dn} + Bw(n) \quad (4)$$

Ajustando as Equações 1 a 4, tem-se a função de transferência do motor CC na equação 5, onde $V_a(s)$ e $\Phi(s)$ é a entrada e a saída da planta respectivamente.

$$\frac{\Phi(s)}{V_a(s)} = \frac{\frac{K_t}{JL_a}}{s^2 + s \left(\frac{B}{J} + \frac{R_a}{L_a} \right) + \left(\frac{BR_a + K_e K_t}{JL_a} \right)} \quad (5)$$

III. INTELIGÊNCIAS DE ENXAME

Descreve-se a seguir os algoritmos de Inteligência de Enxame (IE) utilizados neste trabalho no processo de identificação de sistemas.

A. PSO

O algoritmo PSO foi primeiramente descrito nos trabalhos de Eberhart e Kennedy em 1995, o qual trata de um algoritmo inspirado no comportamento de bandos de pássaros. Este algoritmo consiste basicamente em um conjunto de partículas, enxame, onde a posição de cada uma representa uma solução possível para um problema, e cada partícula possui uma velocidade associada, que é atualizada conforme um modelo matemático baseado nas informações compartilhadas pelas partículas [7]. A Fig. 2 mostra de forma completa o fluxograma do algoritmo PSO.

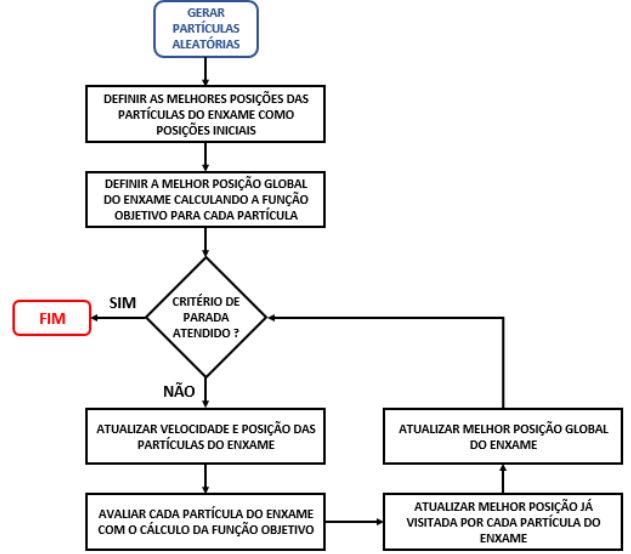


Figura 2. Fluxograma do algoritmo PSO.

No PSO, cada partícula do enxame é formada por: um vetor posição, $\vec{X}_i(t)$, que representa uma solução possível para o problema; um vetor velocidade, $\vec{V}_i(t)$, que pondera a velocidade que uma partícula se move no espaço de busca; um vetor de memória cognitiva da partícula, $\vec{P}_i(t)$, o qual indica a melhor posição individual já ocupada por determinada partícula e $\vec{G}(t)$ é a melhor posição global do exame. Onde: i é a i -ésima partícula do enxame de N dimensões e t é a iteração do algoritmo. As Equações (6) e (7) permitem atualizar as velocidades e posições das partículas respectivamente.

$$\vec{V}_i(t+1) = w\vec{V}_i(t) + c_1 R_1(t)(\vec{P}_i(t) - \vec{x}_i(t)) + c_2 R_2(t)(\vec{G}_i(t) - \vec{x}_i(t)) \quad (6)$$

$$\vec{x}_i(t+1) = \vec{x}_i(t) + \vec{v}_i(t+1) \quad (7)$$

Na Equação (6) c_1 e c_2 são o fator cognitivo e social respectivamente, também conhecidos como coeficientes de aceleração, e $R_1(t)$ e $R_2(t)$ são parâmetros aleatórios iniciados uniformemente entre 0 e 1, e w é coeficiente de inércia.

Após a atualização das posições e velocidades das partículas, é verificado se os limites de busca desses dois

vetores foram extrapolados. Em seguida, atualiza-se a memória cognitiva conforme a Equação (8).

$$\vec{P}_i(t+1) = \begin{cases} \vec{x}(t+1) & \text{se } f(\vec{x}_i(t+1)) > f(\vec{P}_i(t)) \\ \vec{P}_i(t) & \text{se } f(\vec{x}_i(t+1)) \leq f(\vec{P}_i(t)) \end{cases} \quad (8)$$

A melhor partícula global é definida como sendo a partícula dentre as melhores posições cognitivas atualizadas que possui o melhor valor da função objetivo f , ou seja, o maior valor para problemas de maximização, e o menor valor para problemas de minimização. Em seguida, o algoritmo finaliza seu ciclo até convergir para um valor esperado.

B. APSO

A primeira proposta de PSO adaptativo (APSO) ocorreu nos trabalhos em [13], e é esta estrutura que será utilizada neste trabalho conforme o fluxograma da Fig. 3.

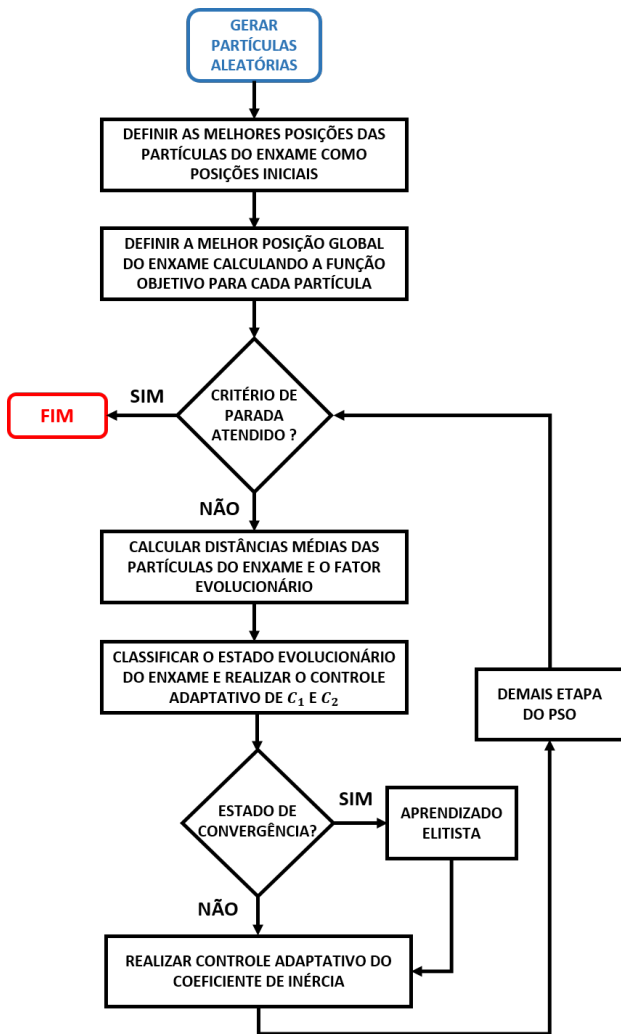


Figura 3. Fluxograma do algoritmo APSO.

Tal proposta consiste em contornar a deficiência do PSO canônico que recai em sua baixa velocidade de convergência e a capacidade limitada de escapar dos ótimos locais, sendo estes dois os principais objetivos do APSO [11].

Para realizar os seus objetivos, o APSO apresenta duas principais etapas. Primeiramente o algoritmo realiza uma avaliação da distribuição populacional do enxame e de seus *fitness*, pois com essa informação o algoritmo estima o coeficiente conhecido como fator evolucionário (*fevol*), o qual é usado para definir em cada iteração em qual estado evolucionário está o enxame, cujos estados são: *exploration*, *exploitation*, *convergence* e *jumping out*. Além controlar o coeficiente de inércia conforme a Equação (9).

$$W(f_{evol}) = \frac{1}{1 + 1.5e^{-2.6f_{evol}}} \quad (9)$$

O estado evolucionário é definido pelo conjunto de funções *fuzzy* mostradas na Fig. 4. A tabela II mostra o controle dos coeficientes de aceleração, onde o incremento é uma variável aleatória entre 0.05 e 0.10.

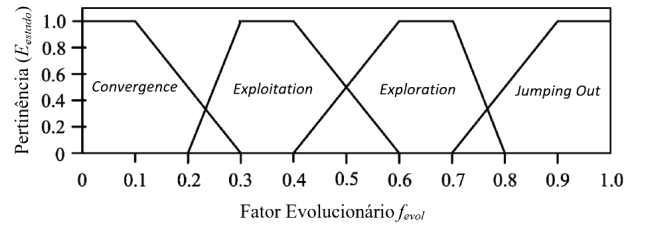


Figura 4. Funções de pertinência *fuzzy* dos estados evolucionários do APSO.

Tabela II
ATUALIZAÇÃO ADAPTATIVA DOS COEFICIENTES DE ACELERAÇÃO.

Estado Evolucionário	c_1	c_2
<i>Exploration</i>	Incrementar	Decrementar
<i>Exploitation</i>	Incrementar levemente	Decrementar levemente
<i>Convergence</i>	Decrementar levemente	Incrementar levemente
<i>Jumping Out</i>	Decrementar	Incrementar

A segunda etapa somente ocorre quando o enxame é classificado no estado *convergence*, o que consiste em executar uma estratégia elitista sobre a melhor partícula global, de forma a alterar um de seus parâmetros aleatoriamente com uma distribuição gaussiana de média zero e variância σ^2 , de forma a aumentar o espaço de busca do algoritmo, e assim, impedindo que a busca fique presa em um mínimo (ou máximo) local. A Equação (10) mostra o cálculo deste desvio padrão, onde $\sigma_{max} = 0.1$ e $\sigma_{min} = 1.0$, e t_{max} é a última iteração do algoritmo.

$$\sigma = \sigma_{max} - (\sigma_{max} - \sigma_{min}) \left(\frac{t}{t_{max}} \right) \quad (10)$$

Caso o enxame não seja classificado como *convergence*, o algoritmo apenas ajusta os coeficientes de aceleração e o fator de inércia, sem realizar a estratégia elitista, e segue com as próximas etapas.

IV. IDENTIFICAÇÃO DE SISTEMAS COM METAHEURÍSTICAS POPULACIONAIS

A. Identificação de sistemas de 1ª e 2ª ordem

A identificação *off-line* de sistemas lineares de tempo contínuo com IE demanda das informações do sinal de entrada $u(n)$, e do sinal de saída $y(n)$ da planta a ser identificada, conforme a Fig. 5.

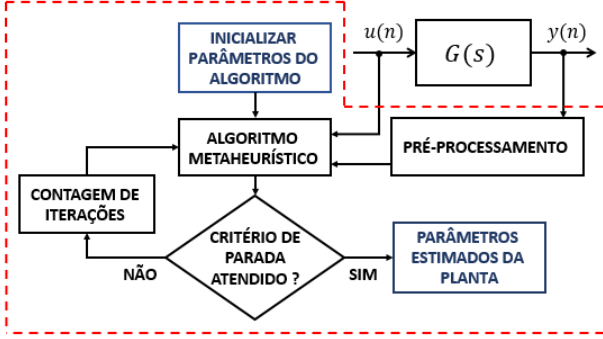


Figura 5. Processo de identificação *off-line* de sistemas com algoritmo metaheurístico.

Na Fig. 5, $G(s)$ representa um sistema linear de 1ª ou 2ª a ordem a ser identificado, o bloco pré-processamento é responsável por extrair algumas informações da resposta da planta na identificação.

Para sistemas de 1º ordem, equação 11, as partículas representam uma solução possível para cada um de seus parâmetros a ser identificado, Equação (12). Sendo K o ganho da planta, T é a constante de tempo e θ é o atraso de resposta da planta.

$$G(s) = \frac{K}{Ts + 1} e^{-\theta s} \quad (11)$$

$$\vec{x}_i(t) = [K_i, T_i, \theta_i] \quad (12)$$

Os limites de busca para os parâmetros K, T e θ são mostrados nas inequações (13), (14) e (15). O tempo total de resposta da planta t_{resp} é extraído na fase de pré-processamento e os demais limites devem ser definidos antes da execução do algoritmo.

$$K_{inf} \leq L_{k1} \leq K_{sup} \quad (13)$$

$$T_{inf} \leq L_T \leq T_{sup} \quad (14)$$

$$0 \leq L_{\theta 1} \leq \frac{t_{resp}}{2} \quad (15)$$

A Equação (16) mostra a função de transferência de um sistema de 2ª ordem. Onde K é o ganho, ξ é o coeficiente de amortecimento, w_n é frequência natural de oscilação e θ é o atraso de resposta da planta.

As partículas das IE que representam uma solução possível para cada um de seus parâmetros a ser identificado são mostradas na Equação (17).

$$G(s) = \frac{Kw_n^2}{s^2 + 2\xi w_n s + w_n^2} e^{-\theta s} \quad (16)$$

$$\vec{x}_i(t) = [K_i, \xi_i, w_{ni}, \theta_i] \quad (17)$$

Os limites de busca para os parâmetros K, ξ, w_n e θ são mostrados nas inequações (18), (19), (20) e (21) respectivamente. Nessas equações, Y_{max} é o valor de pico da resposta e t'_p é o tempo que a planta leva para chegar ao valor de pico. Tais informações são colhidas no pré-processamento. Os demais limites são definidos antes da execução das IE.

$$K_{inf} \leq L_{K2} \leq Y_{max} \quad (18)$$

$$0 \leq L_{\xi} \leq 1 \quad (19)$$

$$W_{inf} \leq Lw_n \leq \frac{\pi}{t_p \sqrt{1 - 0.99^2}} \quad (20)$$

$$0 \leq L_{\theta 2} \leq t'_p \quad (21)$$

B. Parametrização e funções de avaliação das IE

A parametrização inicial das IE (PSO e APSO) depende das seguintes variáveis: número, dimensão e aceleração das partículas; limites de velocidades; fator de inércia; número de iterações e erro quadrático médio (EQM) máximo.

Cada um destes parâmetros deve ser inicializado antes da execução do algoritmo e também antes da aquisição dos dados de entrada e saída da planta que será identificada.

Neste trabalho, propõe-se o uso do EQM, Equação (22), como função custo das IE e estas terão como objetivo sua minimização [4].

$$EQM = \frac{1}{S_n} \sum_{n=1}^{S_n} (y(n) - y_i(n))^2 \quad (22)$$

Onde n é o tempo discreto da resposta da planta ao degrau unitário $u(n)$, $y_i(n)$ é a resposta da planta estimada pela i -ésima partícula e S_n é número total de amostras do vetor $y(n)$.

V. AVALIAÇÃO DE DESEMPENHO DAS METAHEURÍSTICAS

As métricas de avaliação são as ferramentas de avaliação de desempenho de cada algoritmo metaheurístico, e elas têm a importância de indicar se um algoritmo é “bom” em relação aos demais em determinado critério.

As métricas utilizadas para este trabalho são: precisão, acurácia e tempo médio de processamento. Todas essas métricas são aplicadas para avaliar o desempenho individual das metaheurísticas utilizadas para identificação de sistemas. Cada algoritmo é executado um número n_p de vezes, em geral algumas dezenas de vezes, para resolver o mesmo problema, e obter dados suficientes para uma análise global das IE.

A. Precisão

A precisão, P , é uma medida estatística que indica o quanto as medidas de um dado problema estão próximas ou dispersas entre si. Ela é calculada utilizando a média σ_m entre os desvios padrões, conforme as Equações (23) e (24), onde N é o número de parâmetros da planta.

$$P = \frac{1}{\sigma_m} \quad (23)$$

$$\sigma = \frac{\sigma_1, \sigma_2, + \dots + \sigma_N}{N} \quad (24)$$

B. Acurácia

A acurácia indica o quanto os dados são próximos do valor real que se deseja alcançar, quantificando a exatidão dos dados. A Equação (25) mostra seu cálculo, onde o termo $bEQM_h$ indica o menor erro médio quadrático alcançado pela IE na h -ésima simulação, e para o MQ, apenas seu EQM em uma simulação. Numa análise geral, utiliza-se a média e o desvio padrão das acurácias para todas as n_p simulações de cada metaheurística.

$$Acuracia_h = \frac{1}{\Delta + bEQM_h} \quad (25)$$

C. Tempo médio de processamento (TMP)

Neste trabalho o tempo de processamento é contado a partir do momento em que cada algoritmo inicializa as partículas, e termina de ser contado quando o algoritmo interrompe a sua execução devido a algum dos critérios de parada. Como cada IE é simulada n_p vezes, calcula-se a média dos tempos de simulação.

VI. IDENTIFICAÇÃO DE SISTEMAS COM MQ

O método dos Mínimos Quadrados (MQ), formulado por Carl Friedrich Gauss em 1775, tem por objetivo encontrar o melhor ajuste para um conjunto de dados tentando minimizar a soma dos quadrados das diferenças entre o valor estimado e os dados observados [3]. Atualmente, esta técnica é bastante utilizada para identificação paramétrica de sistemas lineares representados por modelos discretos.

A. Formulação matemática do MQ

Considerando um processo físico caracterizado por uma entrada $u(n)$, uma saída $y(n)$ e uma perturbação $e(n)$. A planta do sistema no modelo ARX e seus polinômios $A(q)$ e $B(q)$ são mostrados respectivamente nas Equações (26), (27) e (28).

$$A(q)y(n) = B(q)u(n) + e(n) \quad (26)$$

$$A(q) = 1 - a_1q^{-1} - \dots - a_{dy}q^{-dy} \quad (27)$$

$$B(q) = b_0q^{-1} + \dots + b_{du}q^{-du} \quad (28)$$

Representando o sistema no modelo de regressão linear, equação 29, onde $\psi^T(n)$ e $\varphi^T(n)$ são o vetor de medidas e o

vetor de parâmetros, mostrados respectivamente nas Equações (30) e (31).

$$y(n) = \psi^T(n)\varphi(n) + e(n) \quad (29)$$

$$\psi^T(n) = [-y(n-1) - y(n-2) \dots - y(n-dy)u(n-d) \dots u(n-d-dy)] \quad (30)$$

$$\varphi^T(n) = [a_1 \ a_2 \ \dots \ a_{ny} \ b_0 \ b_1 \ \dots \ b_{mu}] \quad (31)$$

Portanto, minimizando o erro entre os valores medidos e os valores estimados, tem-se a estimador de Mínimos Quadrados na Equação (32).

$$\hat{\psi} = [\psi^T \ \psi]^{-1} \psi^T Y \quad (32)$$

B. Considerações sobre Sinal Binário Pseudo Aleatório (SBPA)

O SBPA ou PRBS (*Pseudorandom Binary Sequence*) é um sinal periódico que alterna entre dois níveis pré-fixados. Como características relevantes, o sinal de entrada PRBS apresenta média nula e é robusto ao ruído, pois sua covariância se aproxima ao ruído branco, na medida em que a sua duração tende ao infinito. Este sinal é capaz de estimular a dinâmica da planta em frequências desejadas de atuação, com isso pode excitar os modos dos sistemas que corresponde ao conteúdo espectral [1].

O PRBS também possui a vantagem de ser aplicado na entrada de um sistema no qual já tem uma entrada atuante.

VII. RESULTADOS

Neste tópico são aplicados e analisados os algoritmos PSO, APSO e o MQ na identificação *off-line* da planta de um motor CC modelo DC054B-2 (14202) 30,3V / 36W e 430 rad/s, os dados dos parâmetros do motor são mostrados na Tabela III.

Tabela III
PARÂMETROS DO MOTOR CC MODELO DC054B-2 (14202).

Resistência da armadura	$R_a = 2.74 \ \Omega$
Indutância da armadura	$L_a = 0.00405H$
Momento de inércia	$B = 1.14 \times 10^{-5} \ Nms/rad$
Coefficiente de atrito viscoso	$K_e = 0.07 \ V s/rad$
Constante de força eletromotriz	$K_t = 0.07 \ V s/rad$

A planta do motor foi submetida a testes com entrada degrau unitário e com entrada PRBS com $T_b = 10 \ ms$, e o tempo de amostragem do sistema para o uso do MQ foi de $T_s = 10 \ ms$. No total, obtiveram-se seis resultados, três para entrada degrau unitário para cada algoritmo, e mais três para entrada PRBS.

Para a análise dos algoritmos PSO e APSO foram feitas 30 simulações de identificação da planta para cada um, e cada IE foi submetida aos três critérios de avaliação de desempenho. Em relação aos critérios de parada das IE, o valor do EQM máximo utilizado é 4.5×10^3 , e o número de iterações das IE usado é 100. O ruído usado nas análises é o gaussiano branco

de média zero e variância 4×10^3 , e este ruído é somado na saída da planta a ser identificada.

O parâmetro Δ é definido com o valor 0.01 para o cálculo das acurácias dos algoritmos. Deste modo, quando o $bEQM_h$ for igual a zero, condição ideal de identificação, o valor da acurácia será de 100%.

Para o uso das metaheurísticas no processo de identificação, optou-se por identificar os parâmetros da planta do motor CC para um modelo de 1ª ordem, uma vez que sua resposta deste sistema é superamortecida. Os limites de busca de K e T são mostrados na Tabela IV, considerando o tempo de reposta total da planta igual a 10 segundos. E os parâmetros de inicialização das IE são mostrados na tabela V. Todos os resultados são oriundos de simulações feitas no *software* Matlab.

Tabela IV
LIMITES DE BUSCA PARAMÉTRICA DA PLANTA PELAS IE.

Parâmetro da Planta	Límite Inferior	Límite Superior
K	0	50
T	0	50

Tabela V
PARÂMETROS DE INICIALIZAÇÃO DAS IE.

Inteligência de Enxame	Enxame	w	$c_{1,2}$	V_{min}	V_{max}
PSO	60	0.8	2.0	-2.5	2.5
APSO	60	0.8	2.0	-2.5	2.5

A função de transferência do motor CC com os parâmetros mostrados na Tabela III é mostrado na Equação (33).

$$\frac{\Phi(s)}{V_a(s)} = \frac{7515}{s^2 + 676.5s + 529.4} \quad (33)$$

A função de transferência acima possui polos reais em -0.78 e -675.71, e coeficiente de amortecimento igual a 14.7, o que indica que o sistema é superamortecido, e assim, a identificação do mesmo pode ser aproximado para uma dinâmica de 1ª ordem.

A. Identificação com entrada degrau unitário

A Tabela VI mostra os valores de média e desvio padrão dos parâmetros K e T estimados pelas metaheurísticas, além dos valores de a_1 e b_0 estimados pelo MQ para o teste de entrada degrau. A Tabela VII mostra os valores de média e desvio padrão da acurácia, e as precisões das IE, além da acurácia para a estimação com MQ.

Tabela VI
PARÂMETROS ESTIMADOS PELAS IE E PELO MQ.

	Média de K	Desvio Padrão de K	Média de T	Desvio Padrão de T
PSO	14.1986	0.0092	1.278	0.0043
APSO	14.1947	0.0095	1.2774	0.0054
	a_1	b_0	-	-
MQ	-0.9918	0.1155		

Tabela VII
ACURÁCIA DOS ALGORITMOS E PRECISÃO DAS IE.

	Média da Acurácia	Desvio Padrão da Acurácia	Precisão
PSO	71.1874	0.6036	148.14
APSO	70.9850	0.5697	134.22
	Acurácia	-	-
MQ	47,6190		

Conforme mostra a Tabela VII, o PSO obteve a maior precisão e a melhor acurácia, pois alcançou a maior média (menor EQM), além de um desvio padrão próximo ao do APSO. Embora os demais algoritmos também tenham alcançado valores pequenos de EQM, o MQ obteve o maior erro para uma aproximação de 1ª ordem. A Tabela VIII mostra o tempo médio de processamento dos algoritmos.

Tabela VIII
TMP DOS ALGORITMOS.

	Tempo Médio de Processamento (s)
PSO	19.2366
APSO	16.7227
MQ	0.0748

As Equações (34), (35) e (36) mostram as funções de transferência das plantas no domínio discreto estimadas respectivamente pelo PSO, APSO e MQ com entrada degrau. As funções das IE foram convertidas do domínio contínuo para o discreto usando a transformada Z com $T_s = 10$ ms e o segurador de ordem zero (Z.O.H).

$$G_{PSO}(Z) = \frac{0.1107}{z - 0.9922} \quad (34)$$

$$G_{APSO}(Z) = \frac{0.1106}{z - 0.9922} \quad (35)$$

$$G_{MQ}(Z) = \frac{0.115}{z - 0.9918} \quad (36)$$

As Fig. 6, 7 e 8 mostram a resposta ao degrau unitário da planta identificada pelo PSO, APSO e MQ respectivamente.

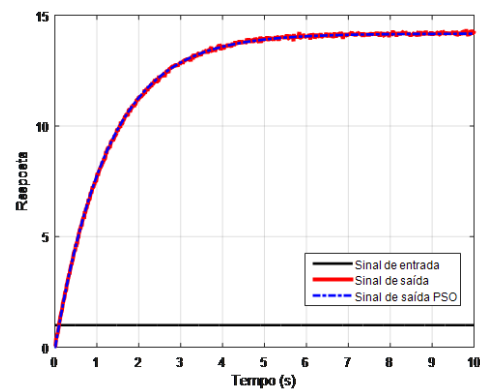


Figura 6. Resposta ao degrau unitário do motor CC identificado com PSO.

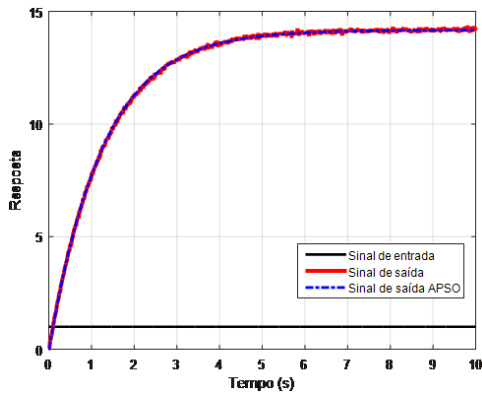


Figura 7. Resposta ao degrau unitário do motor CC identificado com APSO.

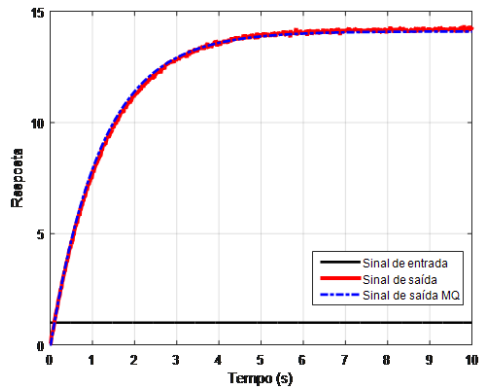


Figura 8. Resposta ao degrau unitário do motor CC identificado com MQ.

Os resultados para a identificação com entrada degrau mostram que todos os algoritmos são capazes de fazer a identificação do sistema com uma boa acurácia. Entretanto, o MQ, por ser uma técnica analítica e não de busca estocástica, este obteve o menor tempo médio de processamento.

B. Identificação com entrada PRBS

A Tabela IX mostra os valores de média e desvio padrão dos parâmetros K e T estimados pelas metaheurísticas, além dos valores de a_1 e b_0 estimados pelo MQ para o teste de entrada degrau. A Tabela X mostra os valores de média e desvio padrão da acurácia, e as precisões das IE, além da acurácia para a estimação com MQ.

Tabela IX
PARÂMETROS ESTIMADOS PELAS IE E PELO MQ.

	Média de K	Desvio Padrão de K	Média de T	Desvio Padrão de T
PSO	14.2169	0.0582	1.2828	0.0082
APSO	14.211	0.0684	1.2807	0.0087
	a_1	b_0	-	-
MQ	-0.9919	0.1103		

Tabela X
ACURÁCIA DOS ALGORITMOS E PRECISÃO DAS IE.

	Média da Acurácia	Desvio Padrão da Acurácia	Precisão
PSO	69.7243	0.9040	30.1204
APSO	69.5992	0.6041	25.9403
	Acurácia	-	-
MQ	41.6666		

Conforme mostra a Tabela X, o APSO obteve a a melhor acurácia, pois alcançou uma das maiores médias (menores EQM) e o menor desvio padrão. Todavia, o PSO alcançou a maior precisão. A Tabela XI mostra o tempo médio de processamento dos algoritmos.

Tabela XI
TMP DOS ALGORITMOS.

	Tempo Médio de Processamento (s)
PSO	14.2488
APSO	7.6748
MQ	0.0223

As Equações (37), (38) e (39) mostram as funções de transferência das plantas no domínio discreto estimadas respectivamente pelo PSO, APSO e MQ com entrada PRBA. As funções das IE foram convertidas do domínio contínuo para o discreto usando a transformada Z com $T_s = 10$ ms e o segurador de ordem zero (Z.O.H).

$$G_{PSO}(Z) = \frac{0.1104}{z - 0.9922} \quad (37)$$

$$G_{APSO}(Z) = \frac{0.1105}{z - 0.9922} \quad (38)$$

$$G_{MQ}(Z) = \frac{0.1103}{z - 0.9919} \quad (39)$$

As Fig. 9, 10 e 11 mostram a resposta ao degrau unitário da planta do motor CC identificada pelo PSO, APSO e MQ respectivamente.

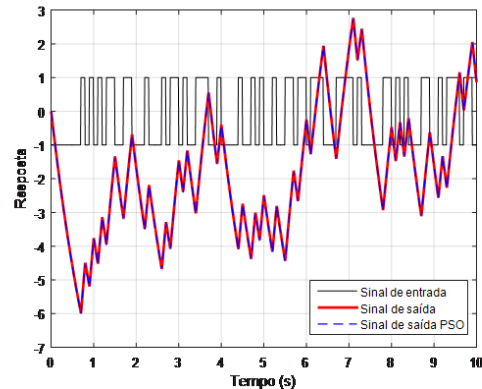


Figura 9. Resposta ao PRBS do motor CC identificado com PSO.

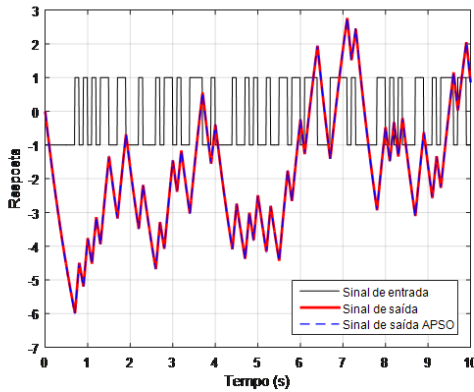


Figura 10. Resposta ao PRBS do motor CC identificado com APSO.

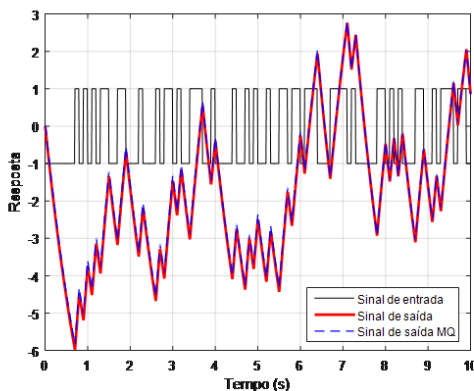


Figura 11. Resposta ao PRBS do motor CC identificado com MQ.

Os resultados para a identificação com entrada PRBS mostram que todos os algoritmos são capazes de fazer a identificação do sistema com um erro quadrático médio relativamente pequeno. Entretanto, o MQ novamente obteve o menor tempo médio de processamento, embora tenha alcançado o maior erro.

VIII. CONCLUSÃO

Este trabalho apresentou as principais características dos algoritmos PSO, APSO e MQ, ressaltando a aplicação destes na identificação *off-line* de sistemas dinâmicos lineares.

Realizou-se a comparação das duas IE e do MQ na identificação paramétrica de uma planta de um motor CC modelo DC054B-2 (14202). Esta comparação foi feita com base em métricas de avaliação de desempenho como precisão, acurácia e tempo médio de processamento, as quais são de fundamental importância na análise do comportamento dos algoritmos.

Por meio dos resultados foi possível verificar que tanto as metaheurísticas quanto os mínimos quadrados possuem bom desempenho e podem ser usadas na identificação *off-line* de sistemas lineares. Também se observou que o algoritmo MQ alcançou, em uma análise geral dos dois casos, o melhor

desempenho em relação ao tempo de processamento, embora a acurácia do PSO e do APSO tenha sido maior nos dois testes.

Propõem-se como trabalhos futuros uma análise estatística mais apurada para os resultados das metaheurísticas utilizando teste ANOVA (Análise da Variância) ou testes não paramétricos, assim como também o planejamento de experimentos para calibração destes algoritmos.

AGRADECIMENTOS

O presente trabalho foi realizado com apoio da Coordenação de Aperfeiçoamento de Pessoal de Nível Superior - Brasil (CAPES) - Código de Financiamento 001

REFERÊNCIAS

- [1] L. A. Aguirre. Introdução à identificação de sistemas: técnicas lineares e não lineares: teoria e aplicação / - 3. ed. rev. - Belo Horizonte: Editora UFMG, 2007.
- [2] M. Clerc and J. Kennedy. The particle swarm – explosion, stability, and convergence in a multidimensional complex space. IEEE transactions on evolutionary computation, vol. 6, no. 1, 2002.
- [3] A. A. R. Coelho. Identificação de Sistemas Dinâmicos Lineares. Florianópolis: 1 ed. Editora UFSC, 2004.
- [4] E. C. L. Oliveira. Análise Comparativa de Metaheurísticas na Identificação de Sistemas Dinâmicos Lineares de 1° e 2° Ordem. Trabalho de Conclusão de Curso - UFPA, 2018.
- [5] E. C. L. Oliveira, J. P. L. Araujo, A. S. Silveira, O. F. Silva, J. F. Vidal and A. F. Silva. Quadrotor Black-Box System Identification using Metaheuristics. Proceedings of the XV International Symposium on Dynamic Problems of Mechanics (DINAME), 2019.
- [6] A. E. Fitzgerald, Jr, C. Kingsley and A. Kusko. Máquinas Elétricas. São Paulo: McGraw-Hill, 2006.
- [7] J. Kennedy and R. Eberhart. Particle swarm optimization. Proceedings of 1995 IEEE International Conference on Neural Networks, pp. 1942 -1948, 1995.
- [8] V. E. Kumar and J. Jerome . An adaptive particle swarm optimization algorithm for robust trajectory tracking of a class of under actuated system. Archives of Electrical Engineering, vol. 63(3), pp. 345-365, 2014.
- [9] H. Modares, A. Alfi and M. N. Sistani. Parameter estimation of bilinear systems based on an adaptive particle swarm optimization. Engineering Applications of Artificial Intelligence 23, 1105–1111, 2010.
- [10] H. Modares and A. Alfi. System identification and control using adaptive particle swarm optimization. Applied Mathematical Modelling 35, 1210–1221. 2011.
- [11] M. R. Pontes. Otimização por Enxames de Partículas utilizando Clãs com Comportamento Adaptativo. Dissertação (mestrado) – Universidade de Pernambuco. Programa de Pós-Graduação em Engenharia da Computação, 2010.
- [12] H. Unbehauen and G.P. Rao. Continuous-time approaches to system identification — a survey. Automatica, 26, (1), pp. 23–35, 1990.
- [13] Z. Zhan et al. Adaptive Particle Swarm Optimization. IEEE Transactions on Systems, Man, and Cybernetics—part b: Cybernetics, vol. 39, no. 6, 2009.