

## Análise de controle via algoritmo *particle swarm optimization*

Diogo C. J. Karmouche\* Camilo Rocha Araujo\*\* Eduardo Preto\*\*\*  
Gustavo Luiz C. M. de Abreu\*\*\*\*

Universidade Estadual Paulista - UNESP, FEIS, Dep. de Engenharia  
Mecânica, Av. Brasil, 56  
Ilha Solteira-SP, Brasil, (e-mail: diogo.karmouche@unesp.br\*,  
camilo.rocha@unesp.br\*\*, eduardo.preto@unesp.br\*\*\*,  
gustavo.abreu@unesp.br\*\*\*\*).

---

**Abstract:** Metaheuristic algorithms have become popular lately as an optimization method due to their similarity to natural adaptation events. In this context, such algorithms have the ability to cover several areas, in which for control systems such algorithms present a new way of developing and adapting control laws that are adapted and improved for different situations in which the dynamic system can be found. This work aims to utilize the metaheuristic algorithm particle swarm optimization in which three populations of gains, generated in different ways, are optimized by the algorithm and adapted for different situations of parameters of the analyzed system.

**Resumo:** Algoritmos metaheurísticos se tornaram populares ultimamente como método de otimização devido a sua similaridade com eventos naturais de adaptação. Neste contexto, tais algoritmos possuem a capacidade de abranger diversas áreas, em que para sistemas de controle tais algoritmos apresentam um novo meio de desenvolver e adaptar leis de controle que são adaptadas e melhoradas para diversas situações em que o sistema dinâmico pode ser encontrado. Este trabalho visa utilizar o algoritmo metaheurístico *particle swarm optimization* em que três populações de ganhos, gerados de formas diferentes, são otimizados pelo algoritmo e adaptados para diferentes situações de parâmetros do sistema analisado.

**Keywords:** Control; inverted pendulum; particle swarm optimization; quadratic linear regulator; linear matrix inequality

**Palavras-chaves:** Controle; pêndulo invertido; otimização de enxame de partículas; regulador linear quadrático; desigualdade de matriz linear

---

### 1. INTRODUÇÃO

Nas últimas décadas, algoritmos metaheurísticos tornaram-se populares como método de otimização (Lin, 2006) e (Janprom et al., 2020). Tais algoritmos são inspirados por eventos e comportamentos naturais, o algoritmo *particle swarm optimization* (PSO) tem sido aplicado abertamente em diversas áreas como a biomédica, processamento de sinais, *design* de estruturas, otimização combinatória, sistemas de controle e entre muitas outras de acordo com Poli (2008). Em sistemas de controle, o algoritmo PSO tem sido usado com sucesso para ajustar parâmetros de ganho de modo ótimo para a técnica PID ((Asifa and Vaishnav, 2010) e (Janprom et al., 2020)), alimentando o algoritmo com dados dinâmicos do sistema e a força de controle atualização dos ganhos. Para avaliação do algoritmo PSO, de acordo com Akhtaruzzaman and Shafie (2010), o sistema pêndulo invertido é um ótimo exemplo dinâmico para aplicar e avaliar técnicas e sistemas de controle, pois, sua modelagem é analoga a muitos problemas de engenharia e por ser um sistema não linear e de alta ordem e fase mínima. Este trabalho visa a apresentar a comparação entre três populações de ganhos de controle em espaço de

estados criadas de forma aleatória, de forma ótima e de forma ótima robusta para o algoritmo PSO e comparar a resposta do melhor ganho de cada população e a adaptação do algoritmo para três casos de mudança nos parâmetros de atrito do sistema.

### 2. PARTICLE SWARM OPTIMIZATION

*Particle Swarm Optimization* (PSO) foi introduzido inicialmente por (Kennedy and Eberhart, 1995) como um algoritmo com a intenção de simular o comportamento social representando o movimento de um bando de aves ou um cardume de peixes. Este algoritmo funciona criando uma população de partículas em uma dimensão de parâmetros,  $N$ , de um problema ou função que buscará a melhor posição entre as partículas, sendo refinada a cada iteração por meio da atração das demais partículas para melhor posição (Poli, 2008), mantendo salvo sua individual melhor posição encontrada (Lin, 2006). Para atualização da posição e velocidade das partículas, são consideradas duas equações, respectivamente:

$$x_{k+1,N} = x_{k,N} + v_{k+1,N} \quad (1)$$

$$v_{k+1,N} = w_k v_{k,N} + c_1 r_1 (p_{k,N} - x_{k,N}) + c_2 r_2 (g_{k,N} - x_{k,N}) \quad (2)$$

Os termos das equações 1 e 2 são apresentados pela seguinte tabela:

Tabela 1. Parâmetros de Posição e Velocidade do PSO

Variável	Definição
Posição da partícula	$x_{k,N}$
Velocidade da partícula	$v_{k,N}$
Melhor posição individual	$p_{k,N}$
Melhor posição do <i>swarm</i>	$g_{k,N}$
Constante inercial de peso	$w_k$
Parâmetro cognitivo	$c_1$
Parâmetro Social	$c_2$
Número Aleatório entre 0 e 1	$r_1, r_2$

Os parâmetros  $c_1$ ,  $c_2$  e  $w_k$  são considerados forças que influenciam no movimento de cada partícula sendo perturbadas aleatoriamente por  $r_1$  e  $r_2$ . A posição da partícula na dimensão  $N$  é representado por meio da matriz  $x_{k,N} = [x_{k,1}, x_{k,2}, \dots, x_{k,N}]$  e analogamente a melhor posição individual, a melhor posição do *swarm* e a velocidade são representados similarmente (Zhao et al., 2005). No algoritmo para que as posições não extrapolarem e resultem em valores altos ou baixos é definido condições de limites para dimensão de parâmetros representada por  $x_{k,N}^{(min,max)} = [(x_{k,1}^{min}, x_{k,1}^{max}), (x_{k,2}^{min}, x_{k,2}^{max}), \dots, (x_{k,N}^{min}, x_{k,N}^{max})]$ .

### 3. MODELAGEM

O sistema pêndulo invertido é um problema clássico em teoria de controle devido a sua dinâmica não linear de alta ordem e fase mínima, como a seguir é representado no diagrama de corpo livre do sistema em questão:

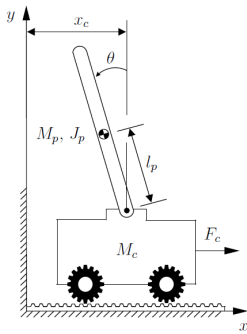


Figura 1. Diagrama de corpo livre do pêndulo invertido (Fonte: Adaptado de Netto et al. (2018))

Para modelagem do sistema é considerado como atuador um motor de corrente contínua fixado ao carro, que desliza sobre uma cremalheira horizontalmente, e como pêndulo uma haste fixada ao carro.

#### 3.1 Motor

Um motor de corrente contínua convencional é um sistema eletromecânico que converte tensão em torque ao redor de seu eixo que pode ser representado por um circuito elétrico onde  $V_m$  é a tensão de entrada,  $R_m$  resistência de

armadura,  $I_m$  é a corrente de armadura,  $L_m$  a indutância do motor,  $\theta_m$  é ângulo de rotação,  $k_g$  é a constante eletromotriz e  $F_c$  é a força de controle exercida no sistema (Netto et al., 2018). A tensão de entrada,  $V_m$ , no motor é convertida na força de controle,  $F_c$ . Deste modo, é possível construir uma relação entre a tensão de entrada no motor e a força que é aplicada no sistema, através da Lei de Kirchoff, é obtido a seguinte equação:

$$V_m - R_m I_m(t) - L_m \dot{I}_m - k_m \dot{\theta}_m = 0 \quad (3)$$

Considerando que a indutância do motor,  $L_m \approx 0$ , é desprezível em comparação ao movimento mecânico do motor a corrente de armadura pode ser isolada tal que:

$$I_m = \frac{V_m - k_m \dot{\theta}_m}{R_m} \quad (4)$$

Em motores de corrente contínua o torque gerado é proporcional à corrente de armadura sendo representado por  $\tau_m = \eta_m k_t I_m$ . Onde  $\eta_m$  é a eficiência do motor e o  $k_t$  é a constante de proporcionalidade corrente-torque. Onde é possível obter uma relação em que o torque esteja em função da tensão de entrada e da velocidade angular:

$$\tau_m = \eta_m k_t \frac{V_m - k_m \dot{\theta}_m}{R_m} \quad (5)$$

O torque pode ser, também, representado em função da força de controle, onde é considerado uma constante de redução realizada pela caixa de transmissão,  $k_g$ , uma eficiência de transmissão de movimento,  $\eta_g$ , e o raio do pinhão,  $r_{pm}$ :

$$\tau_m = \frac{F_c r_{pm}}{\eta_g k_g} \quad (6)$$

Deste modo, a força de controle da equação 6 pode ser isolada e determinando uma relação entre a posição angular e a posição linear,  $\theta_m = \frac{k_g x_c}{r_{pm}}$ , e derivando está relação a velocidade angular é determinada por  $\dot{\theta}_m = \frac{k_g \dot{x}_c}{r_{pm}}$  e substituindo na equação de força de controle a variável do torque, é possível descrever a força de controle em função da tensão de entrada,  $V_m$ , e da velocidade linear,  $x_c$ :

$$F_c = \frac{\eta_g \eta_m k_g k_t}{r_{pm} R_m} \left( V_m(t) - \frac{k_m k_g}{r_{pm}} \dot{x}_c(t) \right) \quad (7)$$

#### 3.2 Carro

O carro é movimentado através da rotação de seu pinhão sobre a cremalheira gerando uma força de atrito,  $B_c$ . Com este movimento, uma força inercial de armadura,  $F_a$ , é gerado devido ao torque de inércia e junto ao deslocamento do carro a haste do pêndulo acomplada rotaciona gerando uma força,  $T_x$ , sobre o carro. Sendo assim adotando os referenciais da figura 1 e aplicando o princípio de D'Alembert a equação de movimento do carro é dado por:

$$M_c \ddot{x}_c + F_a + T_x + B_c \dot{x}_c = F_c \quad (8)$$

A força inercial da armadura é equacionada em função do torque de inércia :

$$F_a = \frac{\eta_g k_g \tau_{inercia}}{r_{pm}} \quad (9)$$

Sendo o torque de inércia representado pelo produto entre o momento de inércia do motor e a aceleração de seu eixo, que quando substituído na equação 9 é possível escrever a força inercial da armadura em:

$$F_a = \frac{\eta_g k_g^2 J_m \ddot{x}_c}{r_{pm}^2} \quad (10)$$

Substituindo a força de controle, equação 7, e a força inercial de armadura, equação 10, na equação de movimento do carro, equação 8, é possível obter:

$$M_c \ddot{x}_c + \frac{\eta_g k_g^2 J_m \ddot{x}_c}{r_{pm}^2} + T_x + B_c \dot{x}_c = \frac{\eta_g \eta_m k_g k_t}{r_{pm} R_m} \left( V_m(t) - \frac{k_m k_g}{r_{pm}} \dot{x}_c(t) \right) \quad (11)$$

Simplificando a equação 11 é possível resumir em:

$$M_{eq} \ddot{x}_c + B_{eq} \dot{x}_c + T_x = A_{eq} V_m \quad (12)$$

Onde,  $M_{eq}$ ,  $B_{eq}$  e  $A_{eq}$  são a massa equivalente, coeficiente de atrito viscoso equivalente e o ganho de atuação equivalente, respectivamente, tais quais são representados por:

$$M_{eq} = M_c + \frac{\eta_g k_g^2 J_m}{r_{pm}^2} \quad (13)$$

$$B_{eq} = B_c + \frac{\eta_g \eta_m k_g^2 k_t k_m}{r_{pm}^2 R_m} \quad (14)$$

$$A_{eq} = \frac{\eta_g \eta_m k_g k_t}{r_{pm} R_m} \quad (15)$$

### 3.3 Pêndulo

Diferente do carro, que apenas movimentava-se horizontalmente, o pêndulo possui dois tipos de movimento, rotação e translação. Como seu momento de inércia,  $J_p$ , é localizado a uma distância  $l_p$  do seu centro de gravidade ao ponto de pivoteamento, o movimento de translação pode ser equacionado como:

$$M_p \frac{d^2}{dt^2} (x_c - l_p \text{sen}(\theta)) - T_x = 0 \quad (16)$$

E seu movimento de rotação gera um atrito no ponto de pivoteamento com coeficiente de atrito,  $B_p$ , resultando na seguinte equação:

$$(J_p + M_p l_p^2) \ddot{\theta} - M_p l_p \cos(\theta) \ddot{x}_c - M_p g l_p \text{sen}(\theta) + B_p \dot{\theta} = 0 \quad (17)$$

### 3.4 Espaço de Estados

Com as equações 12, 16 e 17 a dinâmica do sistema é descrita completamente, entretanto as equações 12 e 16 possuem um coeficiente em comum,  $T_x$ , e podem ser simplificadas em uma equação, onde  $M_1 = M_p + M_{eq}$  e  $M_2 = M_p l_p^2 + J_p$  e  $M_3 = M_p l_p$ , as equações não lineares que descrevem completamente a dinâmica do sistema são obtidas:

$$M_1 \ddot{x}_c + B_{eq} \dot{x}_c - M_3 \cos(\theta) \ddot{\theta} + M_3 \text{sen}(\theta) \dot{\theta}^2 = A_{eq} V_m \quad (18)$$

$$M_2 \ddot{\theta} + B_p \dot{\theta} - M_3 \cos(\theta) \ddot{x}_c - M_3 \text{sen}(\theta) \dot{\theta}^2 = 0 \quad (19)$$

Para o caso não linear do espaço de estados as matrizes  $\mathbf{A}$  e  $\mathbf{B}$  são dependentes das variáveis de estado,  $\mathbf{x}(t) = [x_c \ \theta \ \dot{x}_c \ \dot{\theta}]'$ , assim, as equações 18 e 19 são descritas em forma matricial (Netto et al. (2018)):

$$\Phi(x) \dot{\mathbf{x}}(t) + \Gamma(x) \mathbf{x}(t) + \Omega(x) = \Psi(x) u(t) \quad (20)$$

Sendo  $\Phi$ ,  $\Gamma$ ,  $\Omega$  e  $\Psi$  respectivamente:

$$\Phi(x) = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & M_1 & -M_3 \cos(x_2) \\ 0 & 0 & -M_3 \cos(x_2) & M_2 \end{bmatrix} \quad (21)$$

$$\Gamma(x) = \begin{bmatrix} 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & -1 \\ 0 & 0 & B_{eq} & M_3 \text{sen}(x_2) x_4 \\ 0 & 0 & 0 & B_p \end{bmatrix} \quad (22)$$

$$\Omega(x) = \begin{bmatrix} 0 \\ 0 \\ 0 \\ -M_3 g \text{sen}(x_2) \end{bmatrix} \quad (23)$$

$$\Psi(x) = \begin{bmatrix} 0 \\ 0 \\ 0 \\ A_{eq} \end{bmatrix} \quad (24)$$

$$\dot{\mathbf{x}}(t) = -\Phi(x)^{-1} \Gamma(x) \mathbf{x}(t) + \Phi(x)^{-1} \Psi(x) u(t) - \Phi(x)^{-1} \Omega(x) \quad (25)$$

Assim, analisando a equação 25 é possível indentificar as matrizes  $\mathbf{A}(x) = -\Phi(x)^{-1} \Gamma(x)$  e  $\mathbf{B}(x) = \Phi(x)^{-1} \Psi(x)$  e uma nova matriz  $\mathbf{E}(x) = -\Phi(x)^{-1} \Omega(x)$ , sendo elas:

$$\mathbf{A}(x) = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & \frac{-B_{eq} M_2}{J_{Tn}} & \frac{-M_3 (B_p \cos(x_2) + x_4 \text{sen}(x_2) M_2)}{J_{Tn}} \\ 0 & 0 & \frac{-B_{eq} M_3 \cos(x_2)}{J_{Tn}} & \frac{-B_p M_1 - M_3^2 \cos(x_2) \text{sen}(x_2) x_4}{J_{Tn}} \end{bmatrix} \quad (26)$$

$$\mathbf{B}(x) = \frac{A_{eq}}{J_{Tn}} \begin{bmatrix} 0 \\ 0 \\ M_p l_p^2 \\ M_3 \cos(x_2) \end{bmatrix} \quad (27)$$

$$\mathbf{E}(x) = \frac{1}{J_{Tn}} \begin{bmatrix} 0 \\ 0 \\ M_3^2 g \text{sen}(x_2) \cos(x_2) \\ M_3 g \text{sen}(x_2) M_1 \end{bmatrix} \quad (28)$$

Sendo que  $J_{Tn} = -M_3^2 (\cos(x_2))^2 + M_1 M_2$ , assim, a dinâmica do sistema não linearizado é representado em espaço de estados por meio das matrizes  $\mathbf{A}(x)$ ,  $\mathbf{B}(x)$  e  $\mathbf{E}(x)$ . A seguir são apresentados os valores utilizados no modelo do sistema:

Tabela 2. Dados do Sistema Pêndulo Invertido

Variável	Valor	Unidade
$M_c$	0.94	[kg]
$M_p$	0.23	[kg]
$l_p$	0.3302	[m]
$J_p$	$7.88 \cdot 10^{-3}$	[kg.m <sup>2</sup> ]
$J_m$	$3.9 \cdot 10^{-7}$	[kg.m <sup>2</sup> ]
$r_{pm}$	$6.35 \cdot 10^{-3}$	[m]
$B_{eq}$	5.4	[N.m/m <sup>2</sup> ]
$B_p$	0.0024	[N.m/m <sup>2</sup> ]
$r$	2.6	[ohm]
$k_t$	0.00767	[N.m]
$k_m$	0.00767	[V/(rad/s)]
$k_g$	3.71	-
$\eta_g$	1	-
$\eta_m$	1	-

#### 4. POPULAÇÃO DE PARTÍCULAS

Neste trabalho, três populações de ganhos foram criadas, uma de forma aleatória, outra com a técnica ótima via *linear quadratic regulator* (LQR) e a última com a técnica robusta ótima via *linear matrix inequality* (LMI) para o controle de feedback:

$$u = -\mathbf{K}\mathbf{x} \quad (29)$$

Tais populações foram avaliadas utilizando a função de custo ótima:

$$J = \int_0^{\infty} \mathbf{x}^T \mathbf{Q}\mathbf{x} + u^T \mathbf{R}u \quad (30)$$

Onde,

$$\mathbf{Q} = \begin{bmatrix} 0.75 & 0 & 0 & 0 \\ 0 & 4 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \quad \mathbf{R} = 0.0012 \quad (31)$$

A população aleatória foi gerada com uma distribuição uniforme com o limite mínimo de -40 e máximo de 200.

##### 4.1 População Ótima

Para o cálculo do ganho LQR foi necessário encontrar uma matriz  $\mathbf{P}$  que satisfaça a equação de riccati minimizando a equação 30 (Ogata, 2003), onde as matrizes  $\mathbf{A}$  e  $\mathbf{B}$  se encontram em sua forma linear:

$$\mathbf{A}^T \mathbf{P} + \mathbf{P}\mathbf{A} - \mathbf{P}\mathbf{B}^T \mathbf{R}^{-1} \mathbf{P}\mathbf{B} + \mathbf{Q} = 0 \quad (32)$$

E seja inserida na equação para obter o ganho:

$$\mathbf{K}_{LQR} = \mathbf{R}^{-1} \mathbf{B}^T \mathbf{P} \quad (33)$$

Para a geração da população LQR foi utilizado a mesma matriz  $\mathbf{Q}$ , de 31, e valores variando de 0.0006 a 0.0024 para  $\mathbf{R}$ . Estes valores foram selecionados com o propósito da força de controle não ultrapassar +-10 volts.

##### 4.2 População Ótima Robusta

Para o cálculo deste ganho ótimo o problema precisa ser formulado em LMI. Sendo assim, dado o sistema em espaço de estados em sua forma linear, utilizando o operador traço matricial  $T_r(\cdot)$ , que satisfaz  $\mathbf{a}^T \mathbf{X}\mathbf{b} = T_r(\mathbf{X}\mathbf{b}\mathbf{a}^T)$ , na equação 30 (Olalla et al., 2009) a função de custo é equivalente a:

$$J = \int_0^{\infty} T_r((\mathbf{Q} + \mathbf{K}_{LMI}^T \mathbf{R}\mathbf{K}_{LMI})\mathbf{x}^T \mathbf{x}) dt \quad (34)$$

Resumindo o termo  $\int_0^{\infty} \mathbf{x}^T \mathbf{x}$  em uma matriz  $\mathbf{P}$  que satisfaz a seguinte equação:

$$(\mathbf{A} - \mathbf{B}\mathbf{K}_{LMI})\mathbf{P} + \mathbf{P}(\mathbf{A} - \mathbf{B}\mathbf{K}_{LMI})^T + \mathbf{x}_0 \mathbf{x}_0^T = 0 \quad (35)$$

A equação 34 é simplificada para:

$$J = T_r((\mathbf{Q} + \mathbf{K}_{LMI}^T \mathbf{R}\mathbf{K}_{LMI})\mathbf{P}) \quad (36)$$

Desse modo o ganho do controle ótimo pode ser encontrado minimizando a equação 36 sujeita a equação 35. Entretanto, esta relação não é linear devido a multiplicação entre as matrizes  $\mathbf{K}_{LMI}$  e  $\mathbf{P}$  (Olalla et al., 2009), então uma nova quantidade é definida,  $\mathbf{Y} = \mathbf{K}_{LMI}\mathbf{P}$  (Feron et al., 1992), sendo  $\mathbf{P} = \mathbf{P}^T > 0$  e adicionalmente, o novo termo gerado  $\mathbf{R}^{1/2} \mathbf{Y}\mathbf{P}^{-1} \mathbf{Y}^T \mathbf{R}^{1/2}$  pode ser substituído por uma segunda variável auxiliar  $\mathbf{X}$ , que pode ser decomposto com o complemento de Shur. Com o problema formulado em LMI, o ganho ótimo pode ser obtido com  $\mathbf{K}_{LMI} = \mathbf{P}^{-1} \mathbf{Y}$ . Por ser um controlador robusto que abrange incertezas

em seu desenvolvimento os termos de incerteza são agrupados em um vetor  $\mathbf{p}$  que consiste em  $n_p$  parâmetros incertos,  $\mathbf{p} = (p_1, \dots, p_{n_p})$ , onde cada um é limitado a um valor máximo e mínimo. Generalizando, os valores admissíveis para o vetor  $\mathbf{p}$  são restritos em um hiper-retângulo no espaço de parâmetro  $\mathbb{R}^{n_p}$  com  $N = 2^{n_p}$  vértices,  $\{v_1, \dots, v_N\}$ . A imagem da matriz  $[\mathbf{A}(p) \ \mathbf{B}(p)]$ , para cada vértice  $v_i$ , corresponde a um conjunto  $\{G_1, \dots, G_N\}$  que são os extremos de um politopo convexo que contém as imagens para todos os valores de  $\mathbf{p}$  se a matriz  $[\mathbf{A}(p) \ \mathbf{B}(p)]$  depender linearmente de  $\mathbf{p}$  (Sarkawi and Ohta, 2018). Para esta população de ganhos são considerados fontes de incertezas os parâmetros de atrito,  $B_p$  e  $B_{eq}$ , e os demais parâmetros conhecidos, logo,  $\mathbf{p} = [B_p B_{eq}]$ . Onde  $B_p \in [B_{p,min}, B_{p,max}]$  e  $B_{eq} \in [B_{eq,min}, B_{eq,max}]$ . Para obter os máximos e mínimos, as incertezas foram calculadas em termos de porcentagem de incertezas(0-100%). O politopo é definido com  $N = 2^2$  vértices contendo os possíveis valores de incerteza. Como as matrizes  $\mathbf{A}$  e  $\mathbf{B}$  do sistema em sua forma linear dependem linearmente dos parâmetros  $B_p$  e  $B_{eq}$ , os vértices do modelo politópico são  $A_1(B_{p,min}, B_{eq,min})$ ,  $A_2(B_{p,max}, B_{eq,max})$ ,  $A_3(B_{p,min}, B_{eq,max})$ ,  $A_4(B_{p,max}, B_{eq,min})$  e  $B_{1,2,3,4} = B$ . Deste modo, a formulação em LMI pode ser escrita considerando o modelo politópico com condições iniciais zero e que quando solucionada produzirá duas matrizes que satisfazem a inequação para todos os vértices do politopo convexo:

$$\begin{aligned} & \min_{\mathbf{P}, \mathbf{Y}, \mathbf{X}} T_r(\mathbf{Q}\mathbf{P}) + T_r(\mathbf{X}) \\ & \text{sujeito a} \\ & \mathbf{A}_i \mathbf{P} - \mathbf{B}_i \mathbf{Y} + \mathbf{P}\mathbf{A}_i^T - \mathbf{Y}^T \mathbf{B}_i^T < 0. \quad (i = 1, 2, 3, 4) \quad (37) \\ & q \begin{bmatrix} \mathbf{X} & \mathbf{R}^{1/2} \mathbf{Y} \\ \mathbf{Y}^T \mathbf{R}^{1/2} & \mathbf{P} \end{bmatrix} > 0 \quad \mathbf{P} > 0 \end{aligned}$$

#### 5. CONTROLE COM PSO

Com a população de partículas inicial o algoritmo irá avaliar a performance do sistema para cada partícula e determinar a melhor dentre elas a melhor partícula para que as demais sejam atualizadas e busquem a melhor performance na duração de 30 iterações. O sistema será imposto em três situações onde o atrito do sistema não será alterado inicialmente, na décima iteração os atritos do sistema serão duplicados e, por fim, na vigésima iteração os atritos serão anulados. Para a simulação os parâmetros cognitivo e social,  $c_1$  e  $c_2$ , do algoritmo foram definidos como 2.05, a constante inercial de peso,  $w_k$ , foi definida como 0.5 e para os limites foram definidos como  $x_{k,N}^{(min,max)} = [-40, -10), (50, 200), (-40, -20), (10, 30)]$ .

##### 5.1 Aleatório

Nesta seção é apresentado os resultados da população gerada de forma aleatória. O primeiro gráfico da Figura 2 apresenta a resposta para as partículas iniciais desta população que algumas geram respostas instáveis para o sistema. O algoritmo após a análise atualiza as partículas e busca convergir para o ganho com melhor desempenho e explorar o espaço de busca. Após a décima iteração os atritos,  $B_p$  e  $B_{eq}$ , são duplicados e a resposta do sistema para os ganhos obtidos do algoritmo diante a está mudança são analisados no segundo gráfico da Figura 2 e

é possível analisar a estabilização do sistema. Na vigésima iteração os parâmetros de atrito são mudados novamente zerando os mesmos e no terceiro gráfico da Figura 2, é perceptível que os ganhos das partículas começaram a convergir. Na trigésima iteração a grande maioria das partículas convergiram para a posição da partícula com melhor desempenho como observado no quarto gráfico da Figura 2.

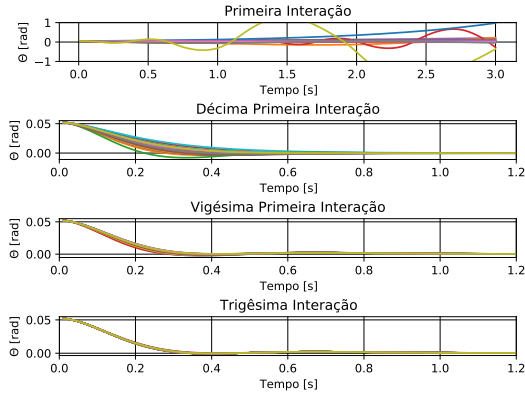


Figura 2. Aleatório: Resposta do sistema na 11<sup>o</sup>, 21<sup>o</sup> e 30<sup>o</sup> iteração. Fonte: Autor

No final da iteração do algoritmo a melhor partícula, representando uma matriz de ganho, foi:

$$\mathbf{K}_{Ale,melhor} = [-40 \ 64.39 \ -10 \ 15.55118547] \quad (38)$$

Por meio de quatro gráficos a progressão das partículas é apresentado nos planos  $k_1$ ,  $k_2$ ,  $k_3$  e  $k_4$ :

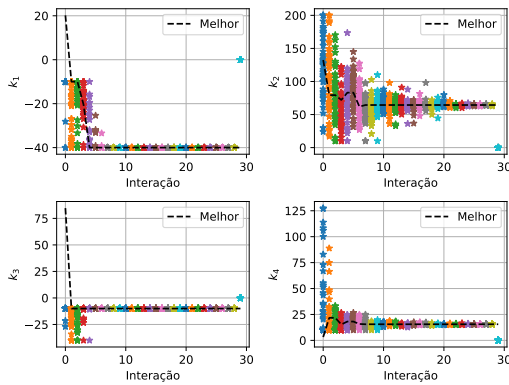


Figura 3. Aleatório: Progressão das partículas no espaço  $k_1$ ,  $k_2$ ,  $k_3$  e  $k_4$ . Fonte: Autor

### 5.2 Controle Ótimo

Nesta seção é apresentado os resultados da população gerada de forma ótima através da técnica de controle LQR. No primeiro gráfico da Figura 4 é apresentado a resposta do sistema para as partículas iniciais da população ótima. Diferente da resposta da primeira iteração da população aleatória a população ótima não gera respostas instáveis, resultando em uma resposta inicial melhor quando comparado a resposta do primeiro gráfico na Figura 2. A seguir, igual para população aleatória na décima iteração, os parâmetros de atrito,  $B_p$  e  $B_{eq}$ , são duplicados e a resposta do sistema e do algoritmo,

diante a essa mudança são analisados, podendo observar, no segundo gráfico da Figura 4, uma melhora no tempo de acomodação e *overshoot* da respostas das partículas. Na vigésima iteração os parâmetros de atrito são zerandos e a resposta a está mudança é observada no terceiro gráfico da Figura 4. Na trigésima iteração, observada no quarto gráfico da Figura 4, é percebido que a grande maioria das partículas convergiram para a posição da partícula com melhor desempenho.

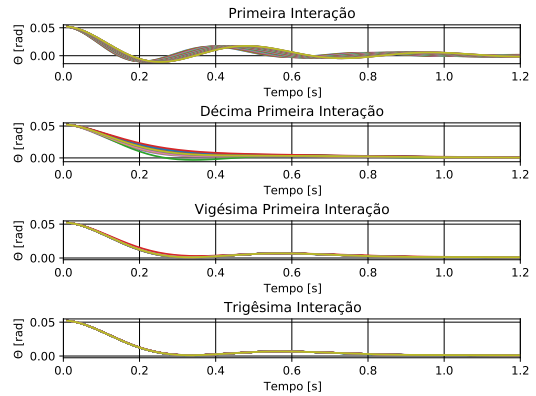


Figura 4. Ótimo: Resposta do sistema na 11<sup>o</sup>, 21<sup>o</sup> e 30<sup>o</sup> iteração. Fonte: Autor

No final da iteração do algoritmo a melhor partícula, representando uma matriz de ganho, foi:

$$\mathbf{K}_{LQR,melhor} = [-40 \ 63.4 \ -20 \ 17.7] \quad (39)$$

Através de quatro gráficos a progressão das partículas é apresentado nos planos  $k_1$ ,  $k_2$ ,  $k_3$  e  $k_4$ .

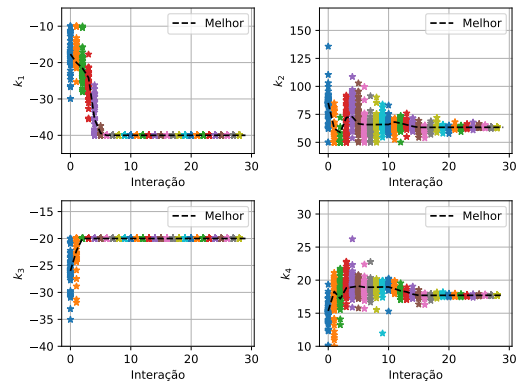


Figura 5. Ótimo: Progressão das partículas no espaço  $k_1$ ,  $k_2$ ,  $k_3$  e  $k_4$ . Fonte: Autor

### 5.3 Ótimo Robusto

Nesta seção é apresentado os resultados da população gerada de forma ótima robusta por meio da técnica de controle LMI. O primeiro gráfico na Figura 6 apresenta a resposta e a força de controle para as partículas iniciais da população ótima robusta. Igual a população aleatória, os parâmetros de atrito,  $B_p$  e  $B_{eq}$ , são duplicados na décima iteração e a resposta do sistema e do algoritmo diante a essa mudança é analisado e é observado que no segundo gráfico da Figura 6 que o tempo de acomodação e *overshoot* da resposta foram reduzidos. Na vigésima

iteração os parâmetros de atrito são zerados e com esta segunda mudança nos parâmetros do sistema, terceiro gráfico da Figura 6, a dinâmica das partículas convergiram. Na trigésima iteração, observada no quarto gráfico da Figura 6, a grande maioria das partículas convergiram para posição da partícula com melhor desempenho. No final da

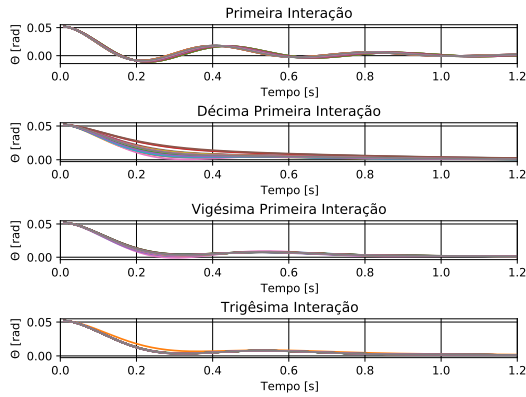


Figura 6. Ótimo Robusto: Resposta do sistema na 1<sup>o</sup>, 11<sup>o</sup>, 21<sup>o</sup> e 30<sup>o</sup> iteração. Fonte: Autor

iteração do algoritmo a melhor partícula, representando uma matriz de ganho, foi:

$$K_{LMI,melhor} = [-26.022 \ 67.876 \ -20 \ 18.215] \quad (40)$$

Por meio de quatro gráficos a progressão das partículas através dos planos  $k_1$ ,  $k_2$ ,  $k_3$  e  $k_4$ :

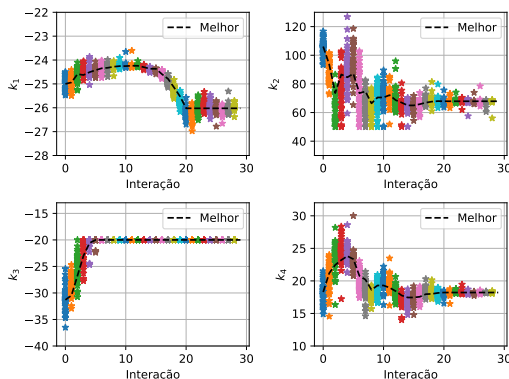


Figura 7. Ótimo Robusto: Progressão das partículas no espaço  $k_1$ ,  $k_2$ ,  $k_3$  e  $k_4$ . Fonte: Autor

## 6. CONCLUSÃO

Neste trabalho, pôde-se verificar o desempenho do algoritmo PSO para as três populações, impostas mudanças nos parâmetros do sistema, convergirem a um melhor ganho. É observado uma similaridade nas melhores matrizes de ganho das populações ótima e aleatória e que a progressão das partículas, nos espaços  $k_1$  e  $k_3$ , ficaram rentes aos limites determinados inicialmente. Para população ótima robusta apenas as partículas do espaço  $k_3$  ficaram no limite. A convergência das partículas para o melhor ganho não foi afetado devido as mudanças de parâmetros no sistema, mas a variação das partículas nos planos  $k_1$ ,  $k_2$ ,  $k_3$  e  $k_4$  para as três populações foi evidente. Nas três populações é observado que a exploração das partículas foi

mais evidente nos espaços  $k_2$  e  $k_3$ , representando o peso do deslocamento e a velocidade angular para o desempenho do sistema. Para população aleatória, foi observado que seus resultados iniciais fornecerem respostas instáveis em sua grande maioria o que força o algoritmo a dedicar uma busca de estabilidade e depois otimizar o resultado. Ao analisar a população ótima e a população ótima robusta foi observado que no plano  $k_1$  a variação das partículas é mínima para população robusta e, para ambas populações, as partículas possuem um bom acompanhamento de convergência para a melhor resposta. Como proposta de trabalho futuro, propõe-se a utilização de um sistema caótico não linear para o algoritmo PSO desenvolver leis de controle devido a dinâmica caótica ser sensível a suas condições iniciais e implementar uma variação dos parâmetros do algoritmo PSO durante as iterações.

## REFERÊNCIAS

- Akhtaruzzaman, M. and Shafie, A.A. (2010). Modeling and control of a rotary inverted pendulum using various methods, comparative assessment and result analysis. In *2010 IEEE International Conference on Mechatronics and Automation*, 1342–1347. IEEE.
- Asifa, H. and Vaishnav, S. (2010). Particle swarm optimisation algorithm based pid controller. In *2010 3rd International Conference on Emerging Trends in Engineering and Technology*, 628–631. IEEE.
- Feron, E., Balakrishnan, V., Boyd, S., and El Ghaoui, L. (1992). Numerical methods for h 2 related problems. In *1992 American Control Conference*, 2921–2922. IEEE.
- Janprom, K., Permpoonsinsup, W., and Wangnipparnto, S. (2020). Intelligent tuning of pid using metaheuristic optimization for temperature and relative humidity control of comfortable rooms. *Journal of Control Science and Engineering*, 2020.
- Kennedy, J. and Eberhart, R. (1995). Particle swarm optimization. In *Proceedings of ICNN'95-international conference on neural networks*, volume 4, 1942–1948. IEEE.
- Lin, J.H. (2006). Particle swarm optimization for control of nonlinear dynamics. In *First International Conference on Innovative Computing, Information and Control-Volume I (ICICIC'06)*, volume 1, 542–545. IEEE.
- Netto, L., Lima, J., et al. (2018). Controle cooperativo h2 e hinf via rede de comunicação: teoria e implementação prática em pêndulos invertidos.
- Ogata, K. (2003). Engenharia de controle moderno, ed. São Paulo: 4<sup>o</sup>.
- Olalla, C., Leyva, R., El Aroudi, A., and Queinnec, I. (2009). Robust lqr control for pwm converters: An lmi approach. *IEEE Transactions on industrial electronics*, 56(7), 2548–2558.
- Poli, R. (2008). Analysis of the publications on the applications of particle swarm optimisation. *Journal of Artificial Evolution and Applications*, 2008.
- Sarkawi, H. and Ohta, Y. (2018). Uncertain dc-dc zeta converter control in convex polytope model based on lmi approach. *International Journal of Power Electronics and Drive Systems*, 9(2), 829.
- Zhao, J., Li, T., and Qian, J. (2005). Application of particle swarm optimization algorithm on robust pid controller tuning. In *International Conference on Natural Computation*, 948–957. Springer.