

OTIMIZAÇÃO DOS GANHOS DE UM CONTROLADOR PID BASEADO EM PSO APLICADO À SINCRONIZAÇÃO DE SISTEMAS CAÓTICOS

RAFAEL B. GREBOGI¹, LEANDRO S. COELHO²

^{1,2}Departamento de Engenharia Elétrica, Universidade Federal do Paraná (UFPR)
Pós-Graduação em Engenharia Elétrica, Av. Cel. Francisco H. dos Santos, 210
Centro Politécnico, CEP 81530-970, Curitiba, PR, Brasil

²Grupo Produtrônica, Pós-Graduação em Engenharia de Produção e Sistemas (PPGEPS)
Pontifícia Universidade Católica do Paraná (PUCPR)
Rua Imaculada Conceição, 1150, CEP 80215-901, Curitiba, PR, Brasil

E-mails: rafagreboqi@gmail.com, leandro.coelho@pucpr.br

Resumo – Caos é o estado de desordem e irregularidade de um sistema determinístico cuja evolução no tempo, apesar de ser regida por leis exatas e simples, é altamente sensível às condições iniciais. A consequência imediata da sensibilidade às condições iniciais é a falta de previsibilidade do sistema a longo prazo. A aplicação de controladores para sincronização de sistemas caóticos com condições iniciais diferentes pode ser facilitada pela utilização de algoritmos de otimização bio-inspirados. Neste contexto, uma abordagem relevante da área denominada inteligência de enxames (*swarm intelligence*) é a otimização por enxame de partículas (*particle swarm optimization*, PSO). O PSO emergiu de experiências com algoritmos que modelam o “comportamento social” observado em muitas espécies de pássaros e cardumes de peixes, e até mesmo do comportamento social humano. Cada indivíduo (solução potencial de um problema de otimização, também denominado de *partícula* no PSO) de uma população possui sua própria experiência e é capaz de estimar a qualidade dessa experiência. Como os indivíduos são sociais, eles também possuem conhecimentos sobre como seus vizinhos comportam-se. Esses dois tipos de informação correspondem à aprendizagem individual (cognitiva) e à transmissão cultural (social), respectivamente. Portanto, a probabilidade de que um determinado indivíduo tome uma determinada decisão será uma função de seu desempenho no passado e do desempenho de alguns de seus vizinhos. A contribuição deste artigo é da validação de um algoritmo PSO para sintonia de um controlador PID (proporcional, integral e derivativo) com intuito de sincronizar sistemas caóticos. Os resultados obtidos foram promissores em termos da qualidade de resposta do procedimento de otimização e do resultado de controle.

Palavras-chave: Sistemas caóticos, controle PID, otimização por enxame de partículas, sincronização caótica.

1. Introdução

A evolução dos sistemas eletrônicos nos últimos anos permitiu um grande avanço na área de controle digital e também na identificação de sistemas com comportamentos caóticos. Desde a popularização do computador e de telas capazes de mostrar gráficos, cientistas e engenheiros conseguiram identificar comportamentos caóticos em equações importantes [1]. Uma das aplicações da sincronização de sistemas caóticos está na área de telecomunicações, como uma forma de encriptar informações [2]. A idéia de utilizar a sincronização de caos para transportar informações justifica-se por duas razões principais. Por um lado, porque sistemas caóticos possuem várias características desejáveis do ponto de vista criptográfico: ergodicidade, sensibilidade às condições iniciais, complexidade e dinâmica determinística. Isso significa que, além de transportar informações (como sinais periódicos fazem), o sinal caótico é potencialmente um método de *criptografar* [16].

Deve-se enfatizar que uma das características marcantes dos sistemas caóticos é sua grande sensibilidade a pequenas perturbações, sendo esta característica conhecida como “efeito borboleta” [3]. Por serem considerados imprevisíveis, acreditava-se que estes sistemas eram incontroláveis também, mas isto foi recentemente provado ser possível, sendo que uma das formas de controlá-lo pode ser usando as próprias perturbações as quais o sistema é sensível [3], [4].

A aplicação de algoritmos de otimização vem crescendo junto com a evolução citada anteriormente, onde a capacidade computacional é um fator determinante no tempo necessário para se otimizar problemas cada vez mais

complexos. Os algoritmos baseados em fenômenos naturais se mostraram eficientes para otimização, entre esses algoritmos pode-se citar como exemplo o algoritmo genético [5], o algoritmo de aprendizagem incremental baseado em população, em inglês, PBIL [6],[7], e o algoritmo baseado em enxame de partículas, em inglês, PSO [8]-[11].

O algoritmo PSO é uma abordagem estocástica baseada em população da área denominada de inteligência de enxames. O PSO foi originalmente proposto por Kennedy e Eberhart em 1995 [9] e implementa uma metáfora do comportamento social da interação entre indivíduos (partículas) de um grupo (enxame). A contribuição deste artigo é da validação de um algoritmo PSO para sintonia de um controlador PID (proporcional, integral e derivativo) com intuito de sincronizar sistemas caóticos.

O restante deste artigo está organizado da seguinte forma. Na seção 2 são apresentados os fundamentos de sistemas caóticos e expoente de Lyapunov. A descrição do PSO e controle PID são abordadas nas seções 3 e 4, respectivamente. Na seção 5 são apresentados e discutidos os resultados de controle e otimização. Finalizando, a conclusão é apresentada na seção 6.

2. Fundamentos de sistemas caóticos

Os sistemas caóticos estão presentes em diversos segmentos de estudo [2]. Este comportamento é possível tanto em sistemas complexos, quanto em sistemas muito simples. Um exemplo de sistema simples, mas que possui um comportamento caótico é o sistema de um pêndulo simples, o qual é um dos exemplos mais antigos de osciladores, e, em particular, osciladores não-lineares [1].

A identificação de um regime caótico num sistema não-linear pode ser feita através do cálculo do expoente de Lyapunov [2],[12]. Em geral, os expoentes de Lyapunov podem ser usados para caracterizar a sensibilidade de um sistema às condições iniciais, através da medição da taxa de divergência e convergência das soluções que partem de pontos próximos.

O expoente de Lyapunov pode indicar se um sistema é caótico, quase-periódico ou convergente. O cálculo se baseia na quantificação da separação exponencial entre as trajetórias dos espaços de fase vizinhos, a partir disso, é obtida a taxa média de divergência por passo, no caso de um sistema discreto, entre os pontos vizinhos, ao longo de uma órbita. Os valores obtidos pelo cálculo do expoente de Lyapunov irão indicar qual é o tipo do sistema em questão. Os valores positivos indicam um sistema caótico, valores iguais a zero indicam um sistema quase-periódico, e valores negativos indicam um sistema que converge para um ponto no infinito.

Computacionalmente, o expoente de Lyapunov pode ser calculado conforme a equação (1),

$$\lambda = \frac{1}{\Delta n} \left(\ln \frac{d_n}{d_0} \right), \quad (1)$$

onde Δn é o número de iterações realizadas no total, d_0 é a distância entre as duas condições iniciais e d_n é a distância entre as trajetórias num tempo tendendo ao infinito.

O conceito do cálculo computacional de λ é medir a distância entre duas condições iniciais, d_0 , que deve tender a zero, e a distância entre dois pontos num tempo que deve tender a infinito, d_n , as trajetórias irão divergir se o sistema for caótico, conforme mostrado na Figura 1.

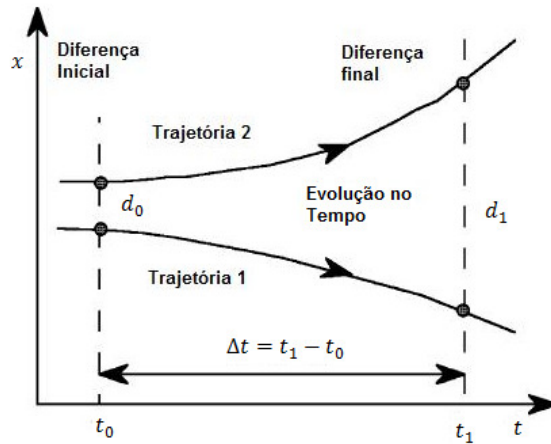


Figura 1 - Trajetória de sistemas divergindo. [12]

O sistema caótico escolhido para este artigo é dado pela equação (2),

$$x(k + 1) = x(k) - \frac{\delta}{m} x(k - m) + \frac{\varepsilon}{m} x^3(k - m), \quad (2)$$

onde, δ e ε são constantes positivas e os valores adotados nesse trabalho são 3,6 e 1, respectivamente, m é o termo de atraso e seu valor é 10, k é o número de iterações e x é a saída do sistema. O comportamento do sistema pode ser definido pelo valor escolhido pelo parâmetro δ [13].

3. Otimização através do PSO

Os algoritmos de otimização são métodos de procura, onde o objetivo é achar uma solução para um problema de otimização. Geralmente as soluções estão sujeita a uma ou mais restrições. Os problemas de otimização nem sempre são simples, as soluções podem consistir de diferentes tipos de dados, podem ter a área de procura restrita por restrições não-lineares, as características do problema podem variar com o tempo e os objetivos do problema podem ser conflitantes [8]. De maneira geral, pode se definir um problema de otimização da seguinte forma,

$$\begin{aligned} & \min_{\vec{x}} f(\vec{x}) \\ & \text{sujeito a } \vec{g}(\vec{x}) \geq 0 \text{ e } \vec{h}(\vec{x}) = 0 \\ & \text{para } \vec{x}_{min} \leq \vec{x} \leq \vec{x}_{max} \end{aligned} \quad (3)$$

Neste trabalho, a métrica escolhida como função objetivo a ser otimizada foi o ITAE (*Integral Time-weighted Absolute Error*), dado pela equação (4),

$$ITAE = \int_0^t t|e(t)|dt, \quad (4)$$

onde o ITAE é o valor absoluto da integral em relação ao tempo, do erro multiplicado pelo tempo.

O algoritmo de otimização por enxame de partículas se baseia no comportamento social de um bando de pássaros procurando por comida, em que cada partícula é um pássaro, e no caso do algoritmo, é uma possível solução do problema [8],[9].

Conceituando de uma forma simplificada o funcionamento do algoritmo, pode-se dizer que as partículas “voam” pelo espaço de procura multidimensional, onde a posição de cada partícula é ajustada de acordo com a sua própria experiência e a experiência das partículas vizinhas.

A posição da partícula e a sua velocidade são dadas por (5) e (6), respectivamente,

$$x_i(t + 1) = x_i(t) + v_i(t + 1), \quad (5)$$

$$v_i(t + 1) = w \cdot v_i(t) + c_1(lP_i(t) - x_i(t)) + c_2(gP(t) - x_i(t)), \quad (6)$$

onde x_i é a posição da partícula i , v_i é a velocidade da partícula i , t é a iteração atual do algoritmo, w é a ponderação de inércia da partícula, lP_i é a melhor posição que a partícula alcançou até a iteração t e gP é a melhor posição alcançada entre todas as partículas do enxame até a iteração t .

Esta equação da velocidade representa uma topologia do enxame onde todas as partículas se comunicam entre si, por meio da melhor posição alcançada pelo enxame até o instante atual [8].

Um parâmetro de grande importância no algoritmo é o peso de inércia, denotado w . Ele é um mecanismo usado para controlar a habilidade de exploração do espaço de procura. O peso de inércia controla o momento da partícula ponderando a contribuição da velocidade anterior, ou seja, controlando quanto da memória da direção anterior de vôo irá influenciar a nova velocidade. O valor deste parâmetro é extremamente importante para garantir um comportamento convergente ao algoritmo. Usando-se valores maiores que 1, a partícula irá acelerar em direção a sua velocidade máxima e o enxame irá divergir. Para valores menores que 1, a velocidade irá diminuir. Valores altos e menores que 1 do peso de inércia facilitam a procura e com uma maior diversidade, já valores menores facilitam uma busca local [8].

Os coeficientes de aceleração c_1 e c_2 controlam a influência estocástica dos componentes social e cognitivo na velocidade da partícula. Estes coeficientes também são chamados de coeficientes de confiança, pois c_1 expressa quanto a partícula confia em si própria e c_2 expressa quanto a partícula confia nos seus vizinhos. Valores pequenos para c_1 e c_2 garantem uma trajetória suave para a partícula, já valores altos causam maior aceleração com movimentos bruscos. Normalmente c_1 e c_2 são estáticos e seus valores são achados empiricamente. Uma iniciação errada de c_1 e c_2 pode provocar um comportamento divergente ou cíclico do algoritmo [8].

Uma forma de aperfeiçoar a procura no algoritmo PSO é implementando uma ponderação de inércia com variação dinâmica [10]. Uma das formas possíveis é variar o valor do parâmetro w de forma linear, começando num valor alto e diminuindo ao longo da operação do algoritmo. O objetivo disso é aumentar a procura por todo o espaço de busca durante as primeiras iterações e quando o algoritmo convergir para uma solução, o valor de w estará pequeno, melhorando a procura local. Estudos indicam que um bom valor inicial é 0,9 e um bom valor final seria de 0,4 [8].

Para este artigo, os valores escolhidos, depois da realização de diversos testes, para os parâmetros do algoritmo PSO são dados na Tabela 1.

Tabela 1 - Parâmetros de ajuste do PSO.

Variável	Valor
População	50
Ponderação de inércia inicial	0,9
Ponderação de inércia final	0,4
Número máximo de iterações	100000

4. Controlador PID

O controlador Proporcional-Integral-Derivativo (PID) é um dos controladores mais utilizados atualmente, onde 98% das indústrias químicas médias utilizam estes controladores [14]. Podem-se citar algumas de suas vantagens perante outros controladores, tais como, a fácil sintonia e a obtenção de resultados satisfatórios, sua simplicidade matemática, e seu algoritmo tende a ser robusto para a maioria dos processos.

O controlador PID é a união de três controladores, um proporcional, um integrador e um derivativo. O termo proporcional ajuda a rastrear a entrada e responder apropriadamente, o termo derivativo reduz o sobre-sinal da saída e o tempo de resposta, e o termo integrador elimina o erro em regime permanente.

A função de transferência de um controlador PID, na forma de equações a diferenças no domínio do tempo é descrita na equação (7), tal que

$$u(k) = u(k-1) + K_p[e(k) - e(k-1)] + K_i \frac{T_s}{2} [e(k) - e(k-1)] + K_d \frac{T_s}{2} [e(k) - 2e(k-1) + e(k-2)], \quad (7)$$

onde T_s é o período de amostragem, K_p é o ganho proporcional, K_i é o ganho do integrador, K_d é o ganho derivativo, $u(k)$ é o sinal de controle e $e(k)$ é o erro [7], [15].

5. Resultados de simulação e sua análise

A simulação consiste em sincronizar dois sistemas caóticos com condições iniciais diferentes, onde o sistema mestre foi dado pela equação (2) e o sistema escravo, que recebe o sinal de controle é dado pela equação (8), tal que

$$y(k+1) = y(k) - \frac{\delta}{m} y(k-m) + \frac{\varepsilon}{m} y^3(k-m) + u(k), \quad (8)$$

onde δ e ε são constantes positivas, m é o termo de atraso, k é o número de iterações e x é a saída do sistema. Os valores dos parâmetros para os sistemas mestre e escravo, bem como suas condições iniciais, são mostrados na Tabela 2.

Tabela 2 - Parâmetros dos sistemas caóticos.

Variável	Valor
δ	3,6
ε	1
m	10
$x(1)$	0,2
$y(1)$	-1
T_s	1

Segundo os conceitos apresentados no início deste trabalho, valores positivos do expoente de Lyapunov indicam um sistema caótico, valores negativos indicam um sistema periódico e valores iguais a zero indicam um sistema quase-periódico. Conforme é apresentado na Figura 2 para δ com o valor igual a 3,6, em ambas as condições iniciais, o expoente de Lyapunov é positivo, confirmando que o sistema terá comportamento caótico.

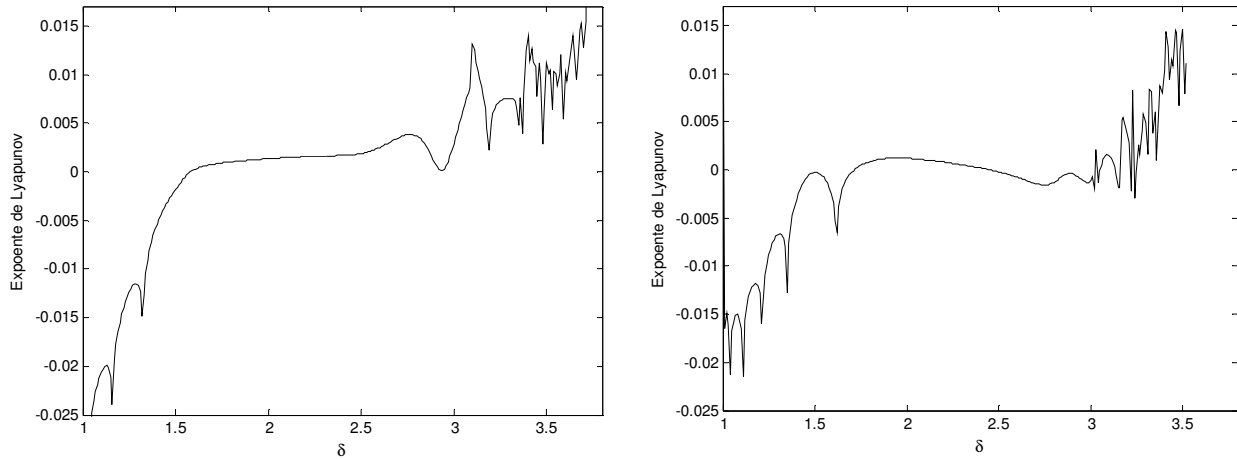


Figura 2 – Valores do expoente de Lyapunov da equação (1) para condição inicial igual a 0,2 e -1, consequentemente.

Foram realizados 100 diferentes experimentos na simulação, onde os resultados podem ser observados na Tabela 3.

Tabela 3 - Valor do ITAE para o sistema caótico utilizando o PSO.

Índice	ITAE
Valor Mínimo	249,061918
Valor Máximo	249,066584
Valor Médio	249,064674
Desvio Padrão	0,002303

Na Tabela 4 estão os valores dos parâmetros otimizados para o menor valor de ITAE.

Tabela 4 - Valores dos parâmetros otimizados pelo PSO.

Parâmetros	Valores
K_p	0,599032
K_i	0,098816
K_d	0,287309

Conforme é mostrado na Figura 3, a sincronização entre os dois sistemas foi efetuada com sucesso antes de 100 iterações.

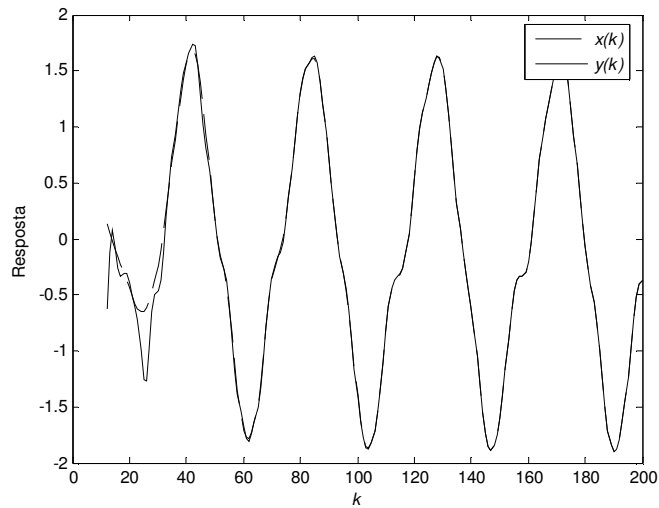


Figura 3 - Sincronização dos sistemas descritos pelas equações (2) e (7).

A Figura 4 apresenta o sinal de controle do controlador PID, mostrando o grande esforço inicial de controle e aproximadamente a partir da 80ª iteração, o controlador só faz pequenos ajustes, indicando que os sistemas já estão praticamente sincronizados.

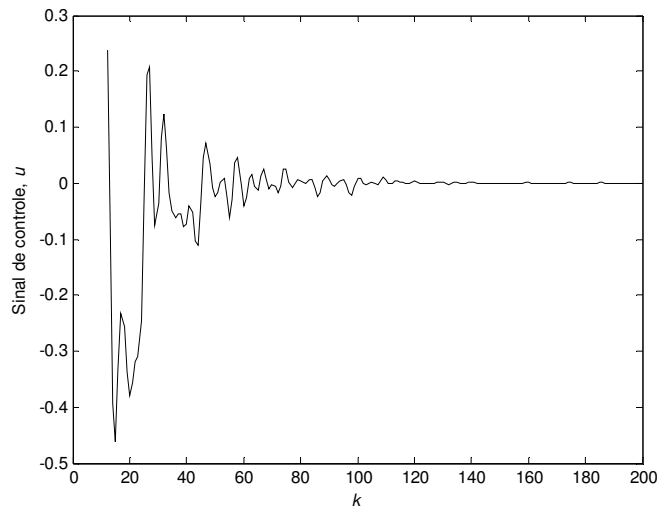


Figura 4 - Sinal de controle para a sincronização dos sistemas caóticos.

6. Conclusão

Baseado nos resultados apresentados na seção anterior, a sincronização dos sistemas caóticos foi efetuada com sucesso pelo controlador PID, mostrando a sua robustez quando bem projetado. Neste estudo de caso, a otimização utilizando o algoritmo PSO, mais especificamente sua variação com ponderação de inércia dinâmico, obteve um promissor desempenho. O estudo de sistemas caóticos mostrou o potencial futuro desta área, a qual possui fortes relações com processos naturais. A sua aplicação em diversas finalidades, desde o envio seguro de informações, até na melhora do desempenho de técnicas de otimização, dá um exemplo deste potencial futuro.

7. Referências bibliográficas

- [1] B. V. Chirikov, A universal instability of many-dimensional oscillator systems. **Physics Reports**. 52(1979), 263 - 379.
- [2] K. T. Alligood, T. Sauer, J. A. Yorke, Chaos – An Introduction to Dynamical Systems. **Springer**. (1996).
- [3] T. Shinbrot, C. Grebogi, J. A. Yorke, E. Ott, Using small perturbations to control chaos. **Nature**. 363(1993), 411 – 417.
- [4] E. Ott, C. Grebogi, J. A. Yorke, Controlling Chaos. **Physical Review Letters**. 64(1990), 1196 – 1199.
- [5] G. R. Harik, F. G. Lobo, D. E. Goldberg. The compact genetic algorithm. **IEEE Transactions on Evolutionary Computation**. 3(1999), 287 – 297.
- [6] E. J. Hughes, Optimisation using population based incremental learning (PBIL). **In: IEE Colloquium on Optimisation in Control: Methods and Applications (Ref. No. 1998/521)**. (1998), 2/1 – 2/3.
- [7] L. S. Coelho, R. B. Grebogi, Chaotic synchronization using PID control combined with population based incremental learning algorithm. **Expert Systems with Applications**. 37(2010), 5347 – 5352.
- [8] A. P. Engelbrecht, Fundamentals of Computational Swarm Intelligence. **John Wiley & Sons**. (2006).
- [9] J. Kennedy, R. C. Eberhart, Particle swarm optimization. **In: Proceeding of the IEEE Conference on Neural Networks**. (1995), 1942 – 1948.
- [10] B. Jiao, Z. Lian, X. Gu, A dynamic inertia weight particle swarm optimization algorithm. **Chaos, Solitons & Fractals**. 37(2008), 698 – 705.
- [11] C. V. Mariani, V. J. Neckel, R. B. Grebogi, L. S. Coelho, Cauchy particle swarm optimization with dynamic adaptation applied to inverse heat transfer problem. **In: IEEE International Conference on Systems, Man, and Cybernetics**. 1(2010), 1 – 6.
- [12] W. Kinsner, Characterizing chaos through Lyapunov metrics. **IEEE Transactions on Systems, Man and Cybernetics**. 36(2006), 141 – 151.
- [13] W. D. Chang, PID control for chaotic synchronization using particle swarm optimization. **Chaos, Solitons & Fractals**. 39(2009) 910 – 917.
- [14] L. D. Desborough, R. M. Miller, Increasing customer value of industrial control performance monitorings Honeywell's experience. **Proceedings of CPC VI**. (2001).
- [15] C. T. Chen, Analog and digital control system design: transfer-function, state-space, and algebraic methods. **Oxford University Press**. (1993).
- [16] J. M. V. Grzybowski, M. Rafilov, Sincronização do sistema caótico unificado via controle ótimo linear feedback com aplicação em comunicação. **XXX Congresso Nacional de Matemática Aplicada e Computacional (XXX CNMAC)**, Florianópolis, SC, 2007.