

Direcionamento de Trajetórias Usando Caos

Saymon Henrique Santos Santana¹, Elbert Einstein Nehrer Macau²

¹Programa de Doutorado em Computação Aplicada – CAP
Instituto Nacional de Pesquisas Espaciais – INPE

²Laboratório Associado de Computação e Matemática Aplicada – LAC
Instituto Nacional de Pesquisas Espaciais – INPE

¹saymonhss@gmail.com

²elbert@lac.inpe.br

Abstract. *In this work the problem of targeting trajectories in chaotic systems is analyzed. Making use of controlled small perturbations in system parameters the time spent by the system to start from an initial state and reach a neighborhood of a desired final state can be substantially reduced. In this study three methods adapted for the analysis of trajectories in the Hénon map were implemented. In the results presented the feasibility regarding the use of sensitivity to initial conditions of chaotic systems can be noted. Through appropriate perturbations the trajectory reaches the neighborhood of a final state in a time considerably shorter than the original one.*

Resumo. *Neste trabalho é analisado o problema do direcionamento de trajetórias em sistemas caóticos. Fazendo uso de pequenas perturbações controladas nos parâmetros do sistema é possível reduzir consideravelmente o tempo gasto pelo sistema para partir de um estado inicial e chegar a uma vizinhança de um estado final desejado. Nesse estudo foram implementados três métodos adaptados para análise de trajetórias no mapa de Hénon. Nos resultados apresentados é possível perceber a viabilidade do uso da sensibilidade às condições iniciais dos sistemas caóticos. Por meio de perturbações adequadas é possível fazer com que a trajetória atinja um determinado estado final em um tempo consideravelmente menor que o tempo original levado para alcançar uma vizinhança do estado final.*

Palavras-chave: *Targeting , Caos, Mapa de Hénon*

1. Introdução

A sensibilidade a variações nas condições iniciais é uma das marcas registradas dos sistemas caóticos. Esta característica, geralmente vista como uma inconveniência, pode ser extremamente útil em diversos problemas de interesse prático.

Diversos trabalhos na literatura [Macau and Grebogi 2001], [Ott and Grebogi 1990a], [Ott and Grebogi 1990b] mostram a eficácia e a eficiência da exploração das próprias características do sistema caótico para se atingir um dado estado final num intervalo de tempo pequeno.

Uma vez que o estado futuro do sistema pode ser drasticamente alterado em virtude de uma pequena variação nas condições iniciais, se o estado for alterado adequadamente, por meio de uma perturbação (ou uma sequência de perturbações) nos parâmetros do sistema, é possível direcionar o sistema para o estado final rapidamente.

Neste trabalho foram implementados três métodos adaptados para o direcionamento de trajetórias. O primeiro, apresentado por [Shinbrot et al. 1990] usa uma abordagem geométrica para encontrar uma perturbação adequada que permita a transferência entre dois pontos no espaço num tempo reduzido; O segundo, um algoritmo evolucionário, *Generalized Extremal Optimization* (GEO) [De Sousa and Ramos 2003], adaptado para a busca por órbitas ótimas; O terceiro [Baptista and Caldas 1998] usa sucessivas perturbações em cada iteração do sistema a fim de atingir um estado final desejado num curto intervalo de tempo.

2. Targeting

Seja o sistema dinâmico tridimensional contínuo no tempo $\frac{d\mathbf{X}}{dt} = \mathbf{F}(\mathbf{X})$ e seja uma superfície (N-1)-dimensional, denota-se as coordenadas desta superfície por ξ , e define-se um mapa de Poincaré de ξ como sendo

$$\xi_{n+1} = \mathbf{f}(\xi_n, \alpha) \quad (1)$$

onde \mathbf{f} é um mapa inversível e α um parâmetro do sistema. Supondo que deseja-se partir de um dado ponto-fonte \mathbf{X}_S para uma pequena região ao redor de um ponto-alvo \mathbf{X}_T . Segundo a trajetória que parte de \mathbf{X}_S e avançando no tempo podemos definir o ponto ξ_s onde ocorre a primeira interseção da trajetória com a superfície da seção. Similarmente iterando para trás no tempo o primeiro ponto de interseção é chamado de ξ_t . Desta forma,

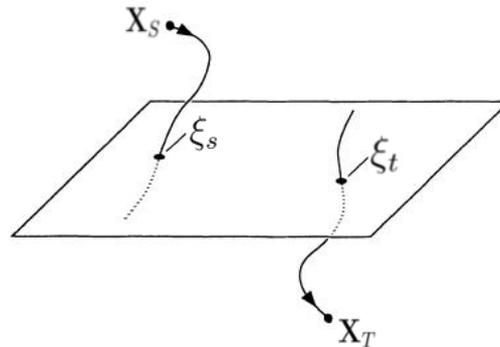


Figura 1. Relação entre a trajetória e o mapa de Poincaré

reduz-se nosso problema a um mapa bidimensional no qual deseja-se sair de ξ_s e chegar em ξ_t .

Considerando que o parâmetro α é ajustável em cada iteração, de modo que $\alpha_n = \bar{\alpha} + \delta_n$, onde $\bar{\alpha}$ é um valor nominal de α e δ_n uma pequena variação em torno do valor nominal.

Uma vez que a transitividade, inerente aos caos, está intimamente relacionada com a ergodicidade do sistema, para os casos em que $\alpha_n = \bar{\alpha}$, o tempo necessário para viajar de ξ_s até uma pequena vizinhança $\bar{\epsilon}_T$ de tamanho linear ϵ_T em torno do ponto ξ_s é $\tau_0 \sim (1/\epsilon_T)^D$, onde D é a dimensão do mapa.

Considerando que pequenas perturbações em α podem ser aplicadas, após uma iteração do mapa de Poincaré, a variação no estado do sistema devida a uma pequena perturbação δ_1 e relativa ao ponto $f(\xi_s, \bar{\alpha})$ é dada por:

$$\delta\xi \cong \delta_1 \frac{\partial f(\xi_s, \alpha)}{\partial \alpha} \Big|_{(\bar{\alpha})} \quad (2)$$

Onde δ_1 varia dentro de um pequeno intervalo, $|\delta_1| < \delta_*$. A equação 2 define um pequeno segmento $\overline{\delta\xi}$ de comprimento $\delta\xi$ através do ponto $f(\xi_s, \alpha)$. Uma vez que o sistema é caótico, o comprimento da imagem desse segmento irá crescer a cada iteração do mapa. Geralmente o segmento $\overline{\delta\xi}$ tem seu comprimento aumentado em uma ordem após n_1 iterações, de modo que $\delta\xi e^{n_1\lambda_1} \sim 1$, onde λ_1 é o expoente de Lyapunov positivo obtido para condições iniciais no atrator, definindo $\tau_1 = \frac{1}{\lambda_1} \ln(1/\epsilon_T)$, então o comprimento do segmento aumentará em uma ordem após um intervalo de tempo τ_1 .

Similarmente, iterando a região $\overline{\epsilon_T}$ para trás no tempo, a preimagem intersepta a imagem do segmento $\overline{\delta\xi}$ após n_2 iterações, o que corresponde a um tempo $\tau_2 = \frac{1}{|\lambda_2|} \ln(1/\epsilon_T)$, onde λ_2 é o expoente de Lyapunov negativo obtido para condições iniciais no atrator.

Na prática, não é possível iterar nem o segmento de linha $\overline{\delta\xi}$ nem a região $\overline{\epsilon_T}$. As iterações são feitas por meio de distribuições de pontos, sob a linha do segmento fonte e em uma região circular ao redor do alvo. Inicialmente itera-se, respectivamente, o segmento $\overline{\delta\xi}$ e a região $\overline{\epsilon_T}$ ao redor do alvo para frente e para trás no tempo, usando $\alpha_n = \bar{\alpha}$, até que seja detectada a interseção entre as imagens de $\overline{\delta\xi}$ e $\overline{\epsilon_T}$ ¹.

Uma vez detectada a interseção, é realizado um refinamento dos pontos, juntando as imagens dos pontos dos segmentos originais por segmentos de reta até que se obtenha uma precisão desejada. Após refinado, é possível obter um ponto, aqui chamado de ponto ótimo, em $\overline{\delta\xi}$, próximo ao ponto fonte, que é mapeado na região do alvo em $n_1 + n_2$ iterações.

3. Generalized Extremal Optimization

Diferentemente dos algoritmos genéticos conhecidos, no GEO não existe uma população de *strings* (ou soluções), mas sim uma população de bits representada por uma única string conforme mostra a figura 2. Além disso, não há cruzamento entre os indivíduos, como no algoritmo genético, no GEO cada bit (espécie) é “forçada” a sofrer mutação com uma probabilidade proporcional ao seu *fitness* (adaptabilidade).

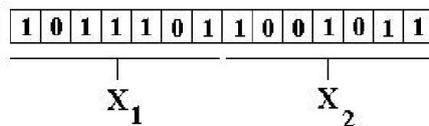


Figura 2. População de espécies no GEO codificando duas variáveis X_1 e X_2

Aqui o GEO foi usado para encontrar um ponto próximo ao ponto fonte (dito *ponto ótimo*), a partir do qual o número de iterações para chegar a uma região próxima ao o alvo fosse o menor possível.

¹Os segmentos fonte e alvo serão iterados respectivamente n_1 e n_2 vezes

O procedimento básico do GEO consiste em:

- 1 - Inicialização aleatória da string binária de comprimento L que codifica N variáveis de projeto de comprimento L_i ($i = 1, 2, \dots, N$). Para esta configuração inicial C de bits, é calculada a função objetivo V e é definido $C_{best} = C$ e $V_{best} = V$.
- 2 - Para cada bit i e em cada iteração:
 - Altera o bit e calcula a função objetivo V_i da configuração C_i ,
 - Associa um valor de fitness usando $\Delta V_i = V_i - V_{best}$, este valor indica o ganho (ou perda) que ocorre em cada mutação em relação ao melhor valor da função objetivo calculada até então.
 - Retorna o bit ao seu valor original.
- 3 - Classifica os bits de acordo com seu valor de fitness, de $k = 1$ para o menos adaptado até $k = L$ para o melhor adaptado. Em problemas de minimização, os maiores valores de ΔV terão maiores valores de fitness.²
- 4 - Escolher com igual probabilidade um bit i candidato à mudança. Gerar um número aleatório RAN distribuição uniforme no intervalo $[0, 1]$. Se a probabilidade de mutação $P(k) = k^{-\tau}$ (onde τ é um parâmetro positivo ajustável) do bit escolhido for maior ou igual a RAN então o bit é confirmado a sofrer a mutação, caso contrário, o processo é repetido até que um bit seja confirmado a mutar.
- 5 - Se $V < V_{best}$ (para problemas de minimização) então é definido $V = V_{best}$ e $C = C_{best}$.
- 6 - Repetir os passos de 2 a 6 até que o critério de parada seja satisfeito.
- 7 - Retornar valores de V_{best} e C_{best} .

4. Método das Perturbações Sucessivas

Considerando um mapa $Z_{n+1} = F(b, Z_n)$, onde b é um parâmetro que pode ser alterado por $\pm\delta$, desta forma, b pode assumir os valores, $b + \delta$, b e $b - \delta$. A idéia do método é mostrar que com o uso desses três valores de parâmetros é possível guiar a trajetória de um ponto inicial P_i até um ponto final P_f .

Inicialmente calcula-se o mapa F fazendo uso dos três valores possíveis do parâmetro b , de modo que ao fim da primeira iteração obtêm-se três pontos: $Z_{1,1} = F(b + \delta, Z_0)$, $Z_{1,2} = F(b, Z_0)$ e $Z_{1,3} = F(b - \delta, Z_0)$.

O procedimento é mantido com os pontos $Z_{1,1}$, $Z_{1,2}$ e $Z_{1,3}$ obtendo novos três pontos com cada um deles por meio dos valores possíveis do parâmetro b e assim sucessivamente até que um dos pontos $Z_{i,j}$ alcance a vizinhança do estado final. Por conseguinte, na N -ésima iteração têm-se 3^N pontos e 3^N trajetórias. A figura 4 mostra as possíveis trajetórias partindo de um ponto P_i .

Nos resultados mostrados a seguir as sucessivas perturbações serão representadas por uma sequência do tipo $[-\delta, 0, +\delta, +\delta, 0, \dots]$, especificando respectivamente qual perturbação foi aplicada em cada iteração, de modo que:

- $-\delta$ indica que $\alpha = \bar{\alpha} + \delta$
- 0 indica que $\alpha = \bar{\alpha}$
- $+\delta$ indica que $\alpha = \bar{\alpha} - \delta$

²Caso dois ou mais bits tenham o mesmo valor de fitness a classificação entre eles é feita de maneira aleatória com probabilidade uniforme.

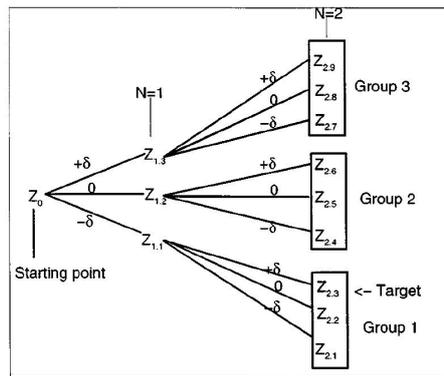


Figura 3. Trajetórias possíveis para atingir o estado final no método das sucessivas perturbações

5. Mapa de Hénon

O objeto de estudo neste trabalho são as trajetórias no Mapa de Hénon [Henon 1976]. Este mapa, que consiste de um sistema de duas variáveis e dois parâmetros, é um modelo simplificado da seção de Poincaré do modelo de Lorenz, as escolhas particulares dos parâmetros produzem o comportamento caótico na dinâmica do mapa criando um atrator em formato de bumerangue conhecido como o *Atrator de Hénon*.

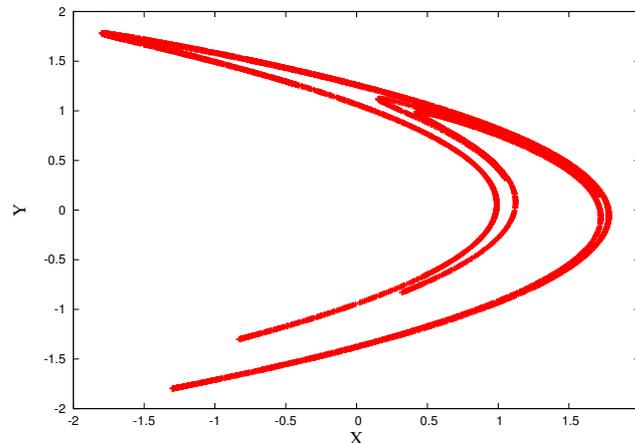


Figura 4. Atrator de Hénon para $\alpha = 1.4$ e $\beta = 0.3$

O mapa de Hénon consiste da iteração

$$\begin{aligned} x_{n+1} &= 1 - \alpha x_n^2 + y_n \\ y_{n+1} &= \beta x_n \end{aligned} \tag{3}$$

no espaço de fase (x, y) onde os parâmetros são α e β . O Jacobiano é dado por

$$J = \det \begin{bmatrix} -2\alpha x & 1 \\ \beta & 0 \end{bmatrix} = -\beta \tag{4}$$

Portanto, valores de $|\beta| < 1$ provocarão uma redução na área original a cada iteração por um fator β , para $|\beta| > 1$ ocorrerá um aumento na área e $|\beta| = 1$ a área no espaço de fase será conservada.

O mapa de Hénon também é inversível, de modo que $H^{-1}(x, y)$ (iteração para trás) é dada por

$$\begin{aligned} x_n &= \frac{y_{n+1}}{\beta} \\ y_n &= x_{n+1} + \frac{\alpha}{\beta^2} y_{n+1}^2 - 1 \end{aligned} \quad (5)$$

6. Resultados

A seguir são apresentados os resultados obtidos com as implementações dos métodos citados anteriormente e aplicados ao mapa de Hénon.

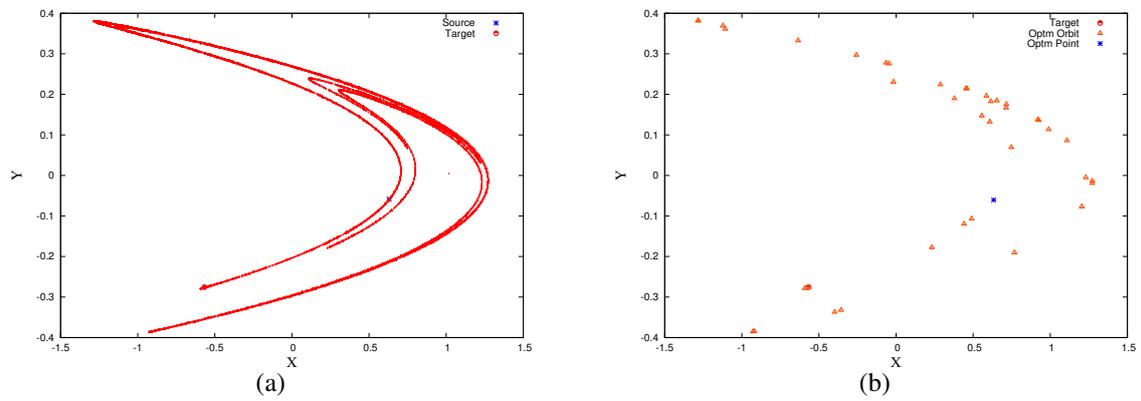


Figura 5. Caso 1 Targeting. Órbita original em (a) com 10^6 iterações, órbita ótima (triângulos) em (b) obtida com 37 iterações.

Perturbação requerida $\delta(x, y) = (-3.109 \times 10^{-2}, 5.195 \times 10^{-4})$

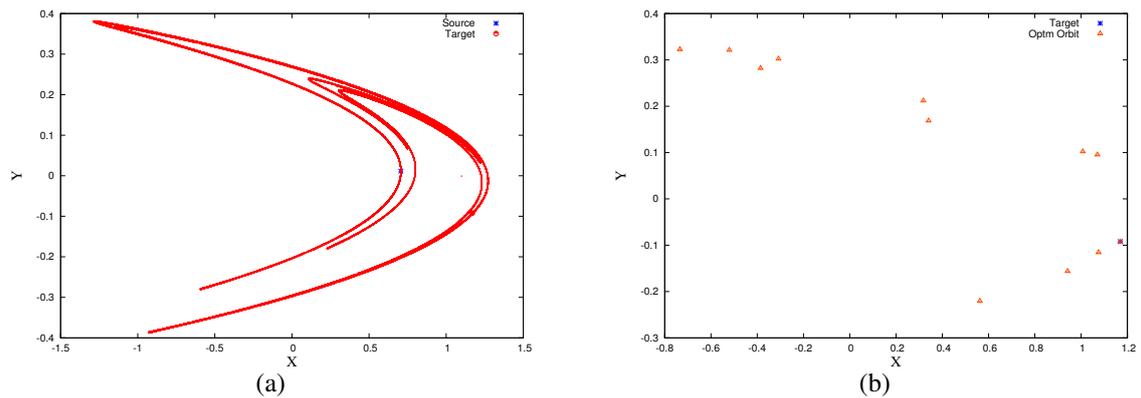


Figura 6. Caso 1 - Sucessivas perturbações. Órbita original em (a) com 10^6 iterações, órbita ótima (triângulos) em (b) com 12 iterações.

Sequência de perturbações: $-\delta, 0, +\delta, 0, -\delta, -\delta, 0, +\delta, -\delta, -\delta, 0$

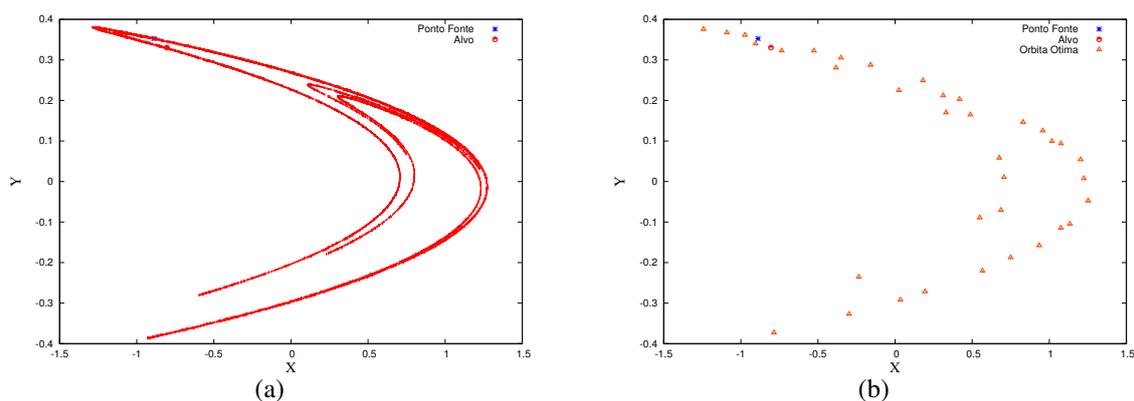


Figura 7. Caso 1 - GEO. Órbita original em (a) com 10^6 iterações, órbita ótima (triângulos) em (b) com 36 iterações.
Perturbação requerida $\delta(x, y) = (-1.416 \times 10^{-2}, -6.069 \times 10^{-2})$

7. Conclusões

Este trabalho tratou de um breve estudo sobre o problema de direcionamento de trajetórias em sistemas caóticos, em particular o problema aplicado ao mapa caótico de Hénon. Nas figuras a seguir são mostradas comparações entre os métodos implementados neste trabalho. Para cada um dos métodos foram simulados 25 pontos fonte (e alvo) para diferentes valores de raio do alvo. Todos os métodos implementados mostraram-se eficientes na busca de um valor mínimo de iterações para atingir uma vizinhança do alvo. A figura 8(a) mostra uma relação obtida para o número médio de iterações em função do raio do alvo para cada um dos métodos implementados. Em 8(b) é apresentada a comparação entre os valores de perturbação média em função do raio do alvo para os casos do GEO e do Targeting.

Uma vez que o GEO “desconhece” a dinâmica do sistema e embora o número de iterações seja (em média) menor que o encontrado com o Targeting, em alguns casos o ponto ótimo encontrado encontrava-se fora da bacia de atração, exigindo assim, um valor maior de perturbação requerida.

O *Targeting*, por sua vez, embora tenha em média encontrado a trajetória ótima com um número superior de iterações, é capaz de encontrar pontos consideravelmente próximos do ponto fonte, exigindo uma perturbação de ordem uma ou duas vezes menor que a encontrada no GEO ou no método das sucessivas perturbações.

No método das sucessivas perturbações, embora a perturbação aplicada no parâmetro possa ser ajustada de modo a assumir valores pequenos, em cada nível são realizadas 3^N análises, conseqüentemente, a quantidade de memória necessária pode ser inviável para um caso de mapas nos quais é preciso um número elevado de iterações para atingir o alvo.

Nos resultados apresentados é possível perceber a viabilidade do uso da sensibilidade às condições iniciais que os sistemas caóticos possuem. O uso adequado dessa característica, associada à transitividade, permite que a trajetória atinja um dado estado final em um tempo consideravelmente pequeno.

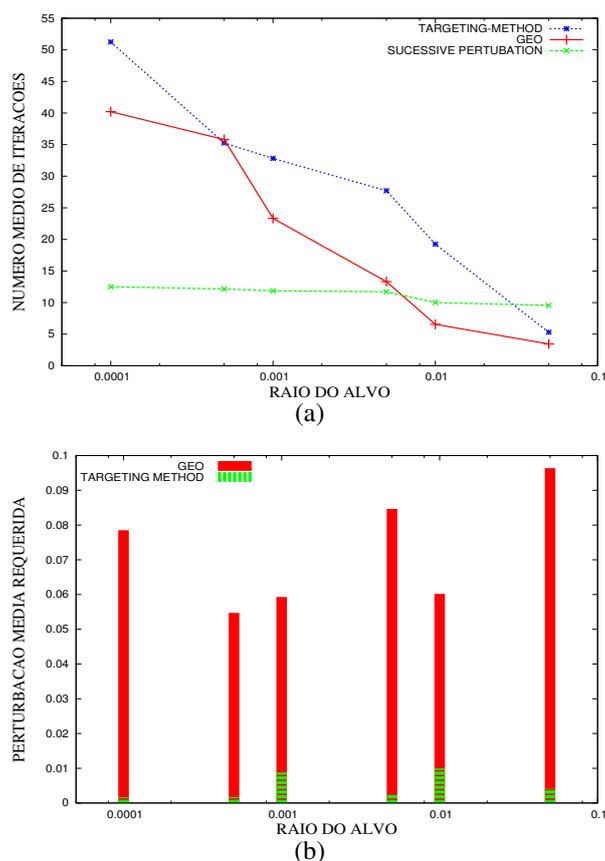


Figura 8. Relação entre o número médio de iterações em função do raio do alvo para cada um dos métodos implementados em (a) e comparação entre os valores de perturbação média em função do raio do alvo para os casos do GEO e do Targeting em (b).

Referências

- Baptista, M. S. and Caldas, I. L. (1998). Easy-to-implemente method to target nonlinear systems. *CHAOS*, 8(1):290–299.
- De Sousa, F. L. and Ramos, F. M. (2003). New stochastic algorithm for design optimization. *AIAA Journal*, 41(9):1808–1818.
- Henon, M. (1976). A two-dimensional mapping with a strange attractor. *Communications in Mathematical-Physics*, 50:69–77.
- Macau, E. E. N. and Grebogi, C. (2001). Driving trajectories in chaotic systems. *International Journal of Bifurcation and Chaos*, 11(5):1423–1442.
- Ott, E. and Grebogi, C. and Yorke, J. A. (1990a). Controlling chaos. *Physical Review Letters*, 64(11):1196–1199.
- Ott, E. and Grebogi, C. and Yorke, J. A. (1990b). Controlling chaotic dynamical systems. *Perspective on Nonlinear Science*, ed Campbell, D., pages 153–172.
- Shinbrot, T., Ott, E., Grebogi, C., and Yorke, J. A. (1990). Using chaos to direct trajectories to targets. *Physical Review Letters*, 65(26):3215–3218.