

We are IntechOpen, the world's leading publisher of Open Access books Built by scientists, for scientists

4,800

Open access books available

122,000

International authors and editors

135M

Downloads

Our authors are among the

154

Countries delivered to

TOP 1%

most cited scientists

12.2%

Contributors from top 500 universities



WEB OF SCIENCE™

Selection of our books indexed in the Book Citation Index
in Web of Science™ Core Collection (BKCI)

Interested in publishing with us?
Contact book.department@intechopen.com

Numbers displayed above are based on latest data collected.

For more information visit www.intechopen.com



Particle Swarm Optimization: Dynamical Analysis through Fractional Calculus

E. J. Solteiro Pires¹, J. A. Tenreiro Machado² and P. B. de Moura Oliveira¹

¹*Universidade de Trás-os-Montes e Alto Douro,*

²*Instituto Superior de Engenharia do Porto
Portugal*

1. Introduction

This chapter considers the particle swarm optimization algorithm as a system, whose dynamics is studied from the point of view of fractional calculus. In this study some initial swarm particles are randomly changed, for the system stimulation, and its response is compared with a non-perturbed reference response. The perturbation effect in the PSO evolution is observed in the perspective of the fitness time behaviour of the best particle. The dynamics is represented through the median of a sample of experiments, while adopting the Fourier analysis for describing the phenomena. The influence upon the global dynamics is also analyzed. Two main issues are reported: the PSO dynamics when the system is subjected to random perturbations, and its modelling with fractional order transfer functions.

2. Particle Swarm Optimization Basics

Evolutionary algorithms have been successfully applied to solve complex optimization engineering problems. Together with genetic algorithms, the particle swarm optimization (PSO) algorithm, proposed by (Kennedy & Eberhart, 1995), has achieved considerable success in solving optimization problems. While PSO algorithms and related variants have been extensively studied (Clerk & Kennedy, 2002), the influence of perturbations signals over the operation conditions is not yet well known.

The PSO algorithm was proposed originally by Kennedy and Eberhart (1995). This optimization technique is inspired in the way swarms behave and its elements move in a synchronized way, both as a defensive tactic and for searching food. An analogy is established between a particle and a swarm element. The particle movement is characterized by two vectors, representing its current position x and velocity v . Since 1995, many techniques were proposed to refine and/or complement the original canonical PSO algorithm, namely regarding its tuning parameters (Shi and Eberhart, 1999) and by considering hybridization with other evolutionary techniques (Lovbjerg et al., 2001).

In this study a standard elementary PSO algorithm is considered (see Fig. 1). The basic algorithm begins by initializing the swarm randomly in the search space. As it can be seen in Fig. 1, where t and $t + 1$ represent two consecutive iterations, the position x of each particle is changed during the iterations by adding a new velocity v . This velocity is evaluated by

summing an increment to the previous velocity value. The increment is a function of two components representing the cognitive and the social knowledge.

The cognitive knowledge of each particle is included by evaluating the difference between the current position x and its best position so far b . The social knowledge of each particle is incorporated through the difference between its current position x and the best swarm global position achieved so far g . The cognitive and social knowledge factors are multiplied by randomly uniformly generated terms ϕ_1 and ϕ_2 , respectively. The particles velocity is restricted, in order to keep velocities from exploding, through the inertia term I (Clerk and Kennedy, 2002).

```

Initialize Swarm
  forAll particles
    calculate fitness f
  endfor
Repeat
  forAll particles
     $v_{t+1} = I v_t + \phi_1 (b - x_t) + \phi_2 (g - x_t)$ 
     $x_{t+1} = x_t + v_{t+1}$ 
  endfor
  forAll particles
    calculate fitness f
  endfor
until Stopping criteria

```

Figure 1. Particle swarm optimization algorithm

3. Fractional Calculus

Fractional Calculus (FC) goes back to the beginning of the theory of differential calculus. Nevertheless, the application of FC just emerged in the last two decades, due to the progress in the area of chaos that revealed subtle relationships with the FC concepts. In the field of dynamical systems theory some work has been carried out but the proposed models and algorithms are still in a preliminary stage of establishment.

The fundamentals aspects of FC theory are addressed in (Gement, 1938; Méhauté, 1991; Oustaloup, 1991; Podlubny, 1999). Concerning FC applications research efforts can be mentioned in the area of viscoelasticity, chaos, fractals, biology, electronics, signal processing, diffusion, wave propagation, percolation, modelling, control and irreversibility (Ross, 1974; Tenreiro Machado, 2001; Torvik, 1984; Vinagre, 2002; Westerlund, 2002).

The FC is a generalization of the classical differential calculus to a non-integer order $\alpha \in \mathbf{C}$. Since its foundation has been the subject of distinct approaches. Due to this reason there are several alternative definitions of fractional derivatives. For example, the Laplace definition of a derivative of order $\alpha \in \mathbf{C}$ of the signal $x(t)$, $D^\alpha[x(t)]$, is a 'direct' generalization of the classic integer-order scheme yielding equation (1):

$$L\{D^\alpha[x(t)]\} = s^\alpha X(s) \quad (1)$$

for zero initial conditions, where s represents the Laplace operator. This means that frequency-based analysis methods have a straightforward adaptation.

An alternative approach, based on the concept of fractional differential, is the Grünwald-Letnikov definition given by equation (2) where h represents the time increment.

$$D^\alpha[x(t)] = \lim_{h \rightarrow 0} \left[\frac{1}{h^\alpha} \sum_{k=0}^{+\infty} \frac{(-1)^k \Gamma(\alpha+1) x(t-kh)}{\Gamma(k+1)(\alpha-k+1)} \right] \quad (2)$$

An important property revealed by equation (2) is that while an integer-order derivative implies just a finite series, the fractional-order derivative requires an infinite number of terms. This means that integer derivatives are 'local' operators in opposition with fractional derivatives which have, implicitly, a 'memory' of all past events.

The characteristics revealed by fractional-order models make this mathematical tool well suited to describe phenomena such as irreversibility and chaos, because of its inherent memory property. In this line of thought, the propagation of perturbations and the appearance of long-term dynamic phenomena in a population of individuals subjected to an evolutionary process seems to be a case where FC tools fit adequately, as shown in (Solteiro Pires et al.; 2003, Solteiro Pires et al., 2006) for genetic algorithms.

4. PSO Swarm Optimization Dynamic analysis

4.1 Problem statement

This section introduces the problem formulation adopted in the study of the PSO dynamic systems. Moreover, the dynamical phenomena involved in the signal propagation within the PSO population is analyzed. For a statistical sample of n independent cases, a particle is randomly initialized, in every experiment, and replaces the corresponding particle of the initial reference population. The experiments reveal a fractional dynamics of the perturbation propagation during the evolution which can be described by system theory tools.

The PSO algorithm, called in this report the 'system', is applied in the optimization of: a quadratic function, the Eason function and the Bohachevsky function.

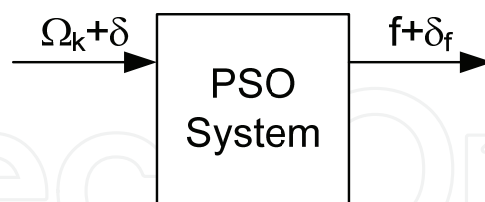


Figure 2. Perturbation of the PSO system

In the first test function case, the objective function consists in minimizing the quadratic function (3) which is adopted as a case study due to its simplicity.

$$f(x) = x^2 \quad (3)$$

This function has only one parameter and its global optimum value is located at $f(x)|_{\text{opt}} = 0$. The variable interval is $x \in [-100, 100]$ and the algorithm uses an encoding scheme with real numbers to codify the particles. A PSO is executed during a period of $T_m = 10000$ iterations with $\{\varphi_1, \varphi_2\} \sim U[0, 1.5]$.

The influence of several factors can be analyzed in order to study the dynamics of the PSO system, particularly the inertia factor I or the φ_i factors, $i = \{1, 2\}$. This effect can vary according to the population size, fitness function and iteration number used. As mentioned previously, one particle of the initial population is changed randomly. The inertia parameter influence is studied to analyze the effect of the perturbation for the values of $I = \{0.50, 0.55, \dots, 0.80\}$ versus the swarm population size $pop = \{6, 8, \dots, 12\}$. The variation of the best global particle fitness evolution is taken as the system output signal as illustrated in Fig. 2.

4.2 The PSO dynamics

Initially, the PSO system is executed without any initial perturbation signal, during $T_m = 10000$ iterations, for a predefined inertia weight value I and swarm population size. The data regarding this test is stored, namely the global particle fitness and the stochastic parameters. This experiment will serve as a reference test. The *optimization* system perturbation consists in replacing the first initial particle of the stored reference swarm population, in every algorithm execution, by another particle randomly generated. Indeed, this stimulus to the system, results in a swarm fitness modification δf which is evaluated. This perturbation test is repeated for $n = 10000$ cases. It is important to state that the remaining test conditions, namely the stochastic reference stored values, remain unchanged along the n experiments. Therefore, the variation of the resulting PSO swarm fitness perturbation, during the evolution, can be viewed as the output signal which varies during the successive iterations.

The output signal consists in the difference between the population fitness with and without the initial perturbation, that is, $\delta f(T) = f_{\text{pert}}(T) - f(T)$. Figure 3a) shows the output signal $\delta f(T)$, for one particle replacement, in the iteration domain. In each experiment the Fourier transform of the signal perturbation, $F[\delta f(T)]$ (see Fig. 3b)) is evaluated in order to analyze the dynamics.

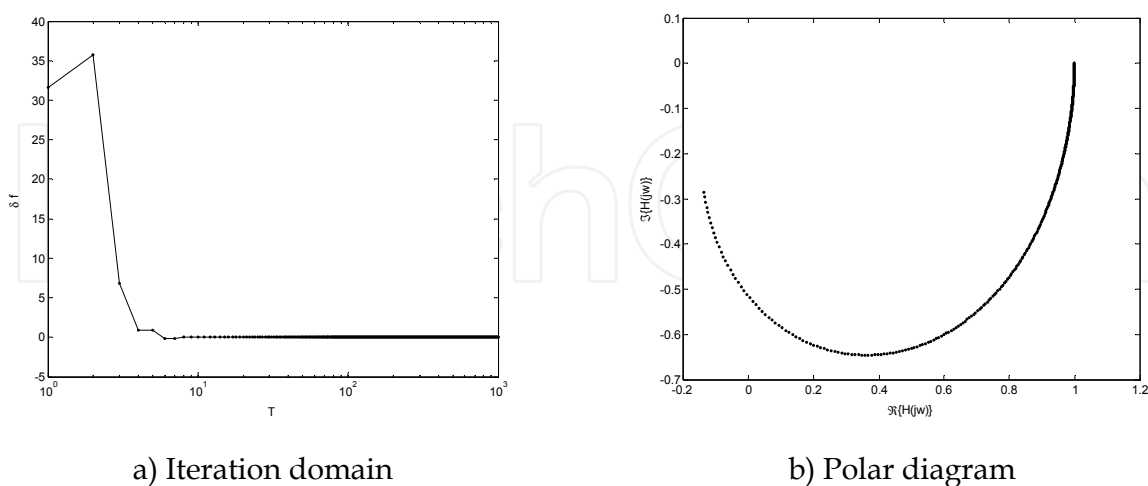


Figure 3. Output signal for an initial perturbation. Experiment with $I = 0.7$ and a swarm population size of $pop = 12$ elements.

With the output signal Fourier description it is possible to evaluate the corresponding normalized transfer function (4):

$$H(j\omega) = \frac{F\{\delta f(T)\}(j\omega)}{F\{\delta f(T)\}(\omega=0)} \quad (4)$$

where ω represents the frequency, T the discrete time evolution (number of iterations used) and $j = \sqrt{-1}$. The transfer function $H(j\omega)$ for this experiment is depicted in Figure 3b). Finally it is obtained a 'representative' transfer function, by using the median of the statistical sample (Tenreiro Machado & Galhano, 1998) of n experiments (see Figure 4). Figure 5 shows the achieved results for inertial values of $I = \{0.50, 0.55, \dots, 0.80\}$. The medians of the transfer functions calculated previously (*i.e.*, for the real and the imaginary parts for each frequency) are taken as the final result of the numerical transfer function $H(j\omega)$.

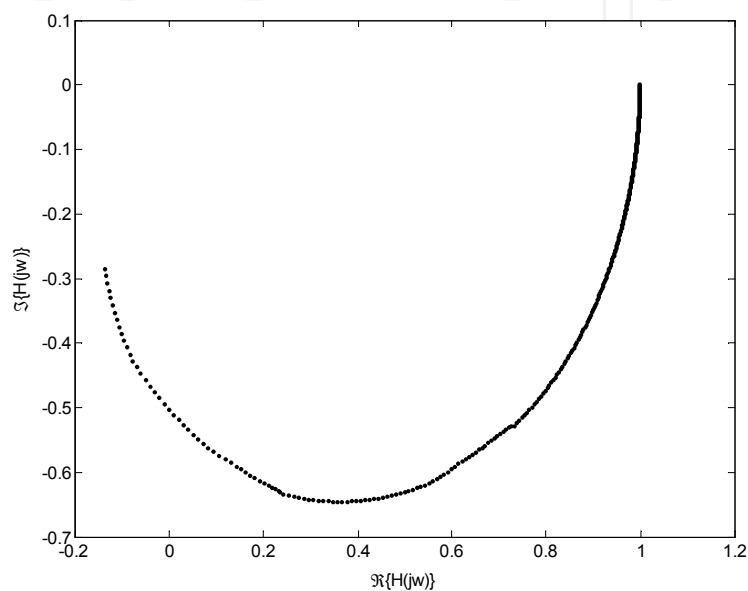


Figure 4. Median transfer function $H(j\omega)$ of $n = 10000$ experiments for an inertial term $I = 0.7$ and $pop = 12$ elements.

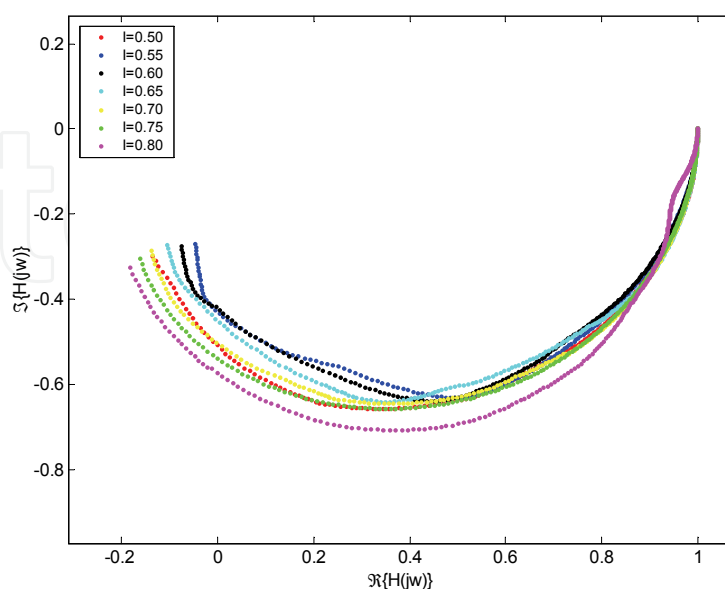


Figure 5. Median transfer function $H(j\omega)$, of the n experiments for $I = \{0.50, 0.55, \dots, 0.80\}$ for a population swarm of $pop = 12$ elements.

Varying the swarm population number of elements in the interval $pop \in [6, 12]$ results in a family of transfer functions. For a swarm size greater than 12 elements there is no difference between the reference test and the perturbation tests. It can be concluded that with large swarms an element has a negligible impact upon the search and, consequently, the performance of the algorithm is independent of the initial swarm. On the other hand, in small swarms, an element has a large impact on the evolution; therefore, it is necessary a large number of perturbation tests to lead to a convergence towards the statistical sample median. From the tests it can be observed that for $I = 0.8$ the median is very irregular because the system is close to the instability region (den Bergh and Engelbrecht, 2006).

4.3 Dynamical analysis

In this section the median of the numerical transfer functions is approximated by analytical expressions with gain $k = 1$ and one pole $a \in \mathbb{R}^+$ of fractional order $\alpha \in \mathbb{R}^+$, given by equation (5):

$$G(j\omega) = \frac{k}{\left(\frac{j\omega}{a} + 1\right)^\alpha} \quad (5)$$

Since the normalized Fourier transform (H) is used, it yields $k = 1$. In order to estimate the transfer function parameters $\{a, \alpha\}$ another PSO algorithm is used, which is named the *identification* PSO. The *identification* PSO is executed during $T_{ide} = 200$ iterations with a 100 particle swarm size. The PSO parameters are: $\{\varphi_1, \varphi_2\} \sim U[0, 1.5]$, $I = 0.6$, and the transfer function parameters intervals are $a \in [4 \times 10^{-3}, 50]$ and $\alpha \in [0, 100]$.

To guide the PSO search, the fitness function f_{ide} is used to measure the distance between the numerical median $H(j\omega_k)$ and the analytical expression $G(j\omega_k)$:

$$f_{ide}(j\omega) = \sum_{k=1}^{nf} \|H(j\omega_k) - G(j\omega_k)\| \quad (6)$$

where nf is the total number of sampling points and ω_k , $k = \{1, \dots, nf\}$, is the corresponding vector of frequencies.

As explained previously, the *optimization* PSO has stochastic dynamics. Therefore, every time the PSO system is executed with a different initial particle replacement, it leads to a slightly different transfer function. Consequently, in order to obtain numerical convergence (Tenreiro Machado & Galhano, 1998) $n = 10000$ perturbation experiments are performed with different replacement particles, while all the other particles remain unchanged. The *optimization* PSO dynamics transfer function is evaluated by computing the normalized signals Fourier transform (FT) (equation 4). The transfer functions medians determined previously (*i.e.*, for the real and the imaginary parts, and for each frequency) are taken as the final result of the numerical transfer function $H(j\omega)$.

Figure 6 and 7 show, superimposed, the normalized median transfer function $H(j\omega)$ and the corresponding identified transfer function $G(j\omega)$, both as polar and amplitude diagrams, respectively. As it can be observed from these figures the fractional order transfer function,

identified by the PSO, captures the *optimization* PSO dynamics quite well, apart from the high frequency range (not represented).

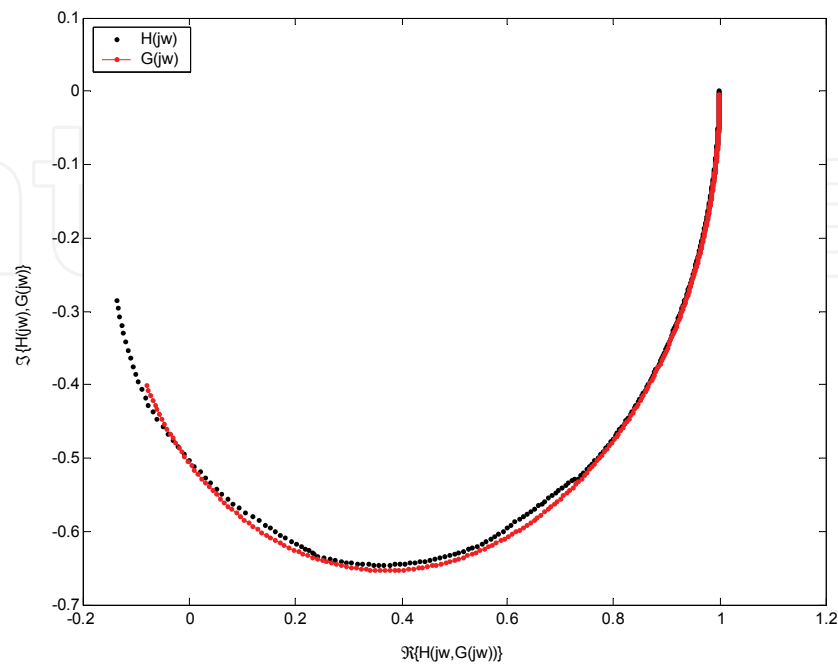


Figure 6. Polar Diagram of $H(j\omega)$ and $G(j\omega)$ for $I = 0.70$ and a swarm size of $pop = 12$ elements

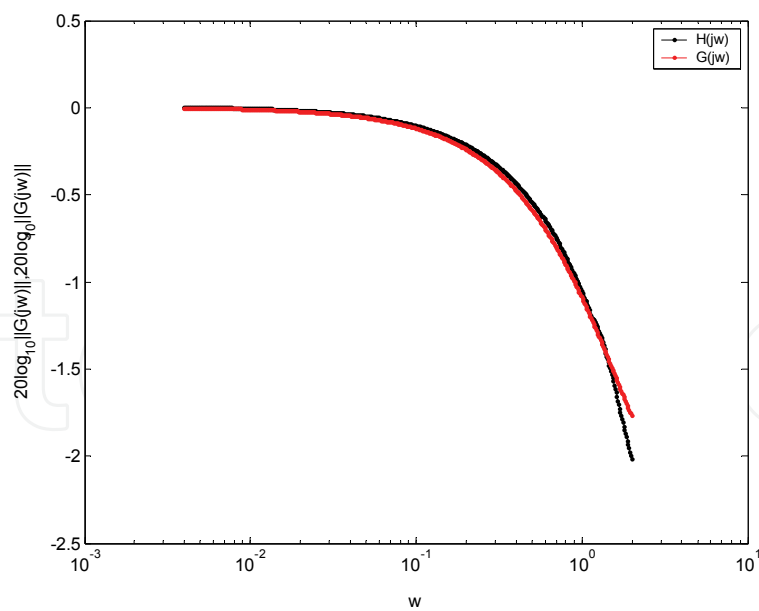


Figure 7. Amplitude Diagram of $H(j\omega)$ and $G(j\omega)$ for $I = 0.7$ and $pop = 12$ elements

For evaluating the influence of the inertia parameter I and the swarm size, several simulations are performed ranging from $I = 0.50$ up to $I = 0.80$ and the number of swarm elements from $pop = 6$ up to $pop = 12$, respectively. The estimated parameters for $\{a, \alpha\}$ are depicted in Figure 8 and 9, respectively.

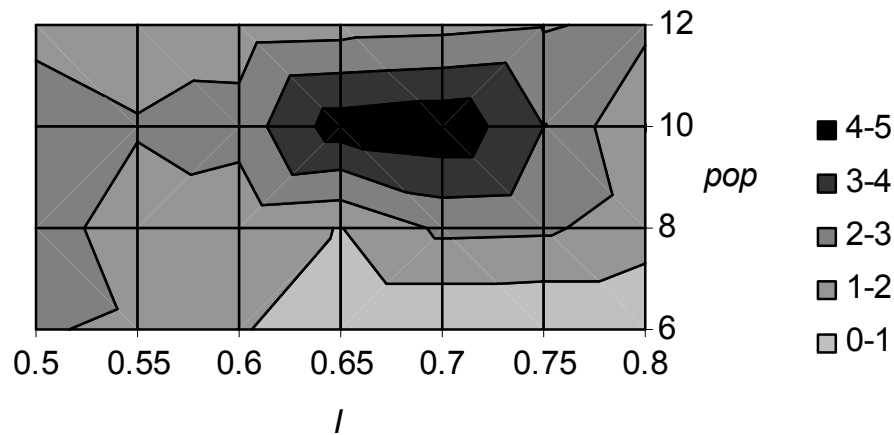


Figure 8. Parameter a versus $\{I, pop\}$

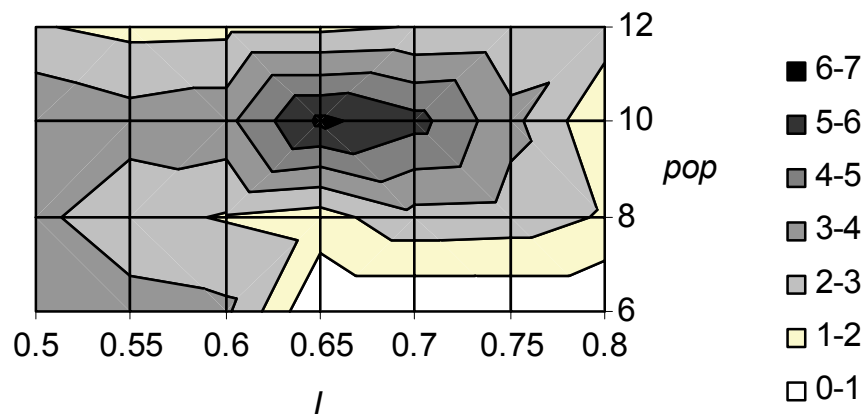


Figure 9. Parameter α versus $\{I, pop\}$

The results reveal that the transfer function parameters $\{a, \alpha\}$ have some dependence with the inertia coefficient I and the swarm size pop . It can be observed that the transfer function parameters have maximum values at $I = 0.65$ and for $pop = 10$ elements. Moreover, it can be seen that there is a correlation between parameters a and α .

In what concerns the transfer function, by enabling the zero/pole order to vary freely we get non-integer values for α . The alternative adoption of integer-order transfer functions would lead to a larger number of zero and poles to get the same quality in the fitting of curves.

5. Other illustrative examples

In this section additional experiments are presented, in which the PSO system is deployed to optimize: the Easom function (7) and the Bohachevsky function (8).

$$f(x_1, x_2) = -\cos(x_1) \cos(x_2) e^{-(x_1-\pi)^2 - (x_2-\pi)^2} \quad (7)$$

$$f(x_1, x_2) = x_1^2 + x_2^2 - 0.3 \cos(3\pi x_1) - 0.4 \cos(4\pi x_1) + 0.7 \quad (8)$$

These functions (7,8) are more complex than the quadratic function used in previous section. In these cases, a swarm of $pop = 20$ elements was used in the experimental tests while varying the inertial parameter in the set $I = \{0.5, 0.6, \dots, 0.8\}$. The polar diagrams illustrated by Figures 10 and 11 were obtained for the Easom and the Bohachevsky fitness functions, respectively.

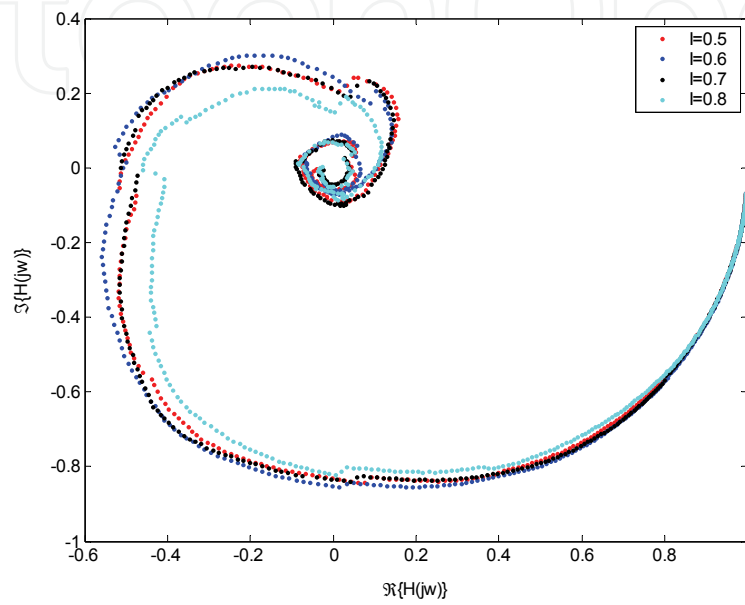


Figure 10. Median transfer function $H(j\omega)$ of the n experiments for the Easom function and $pop = 20$ elements

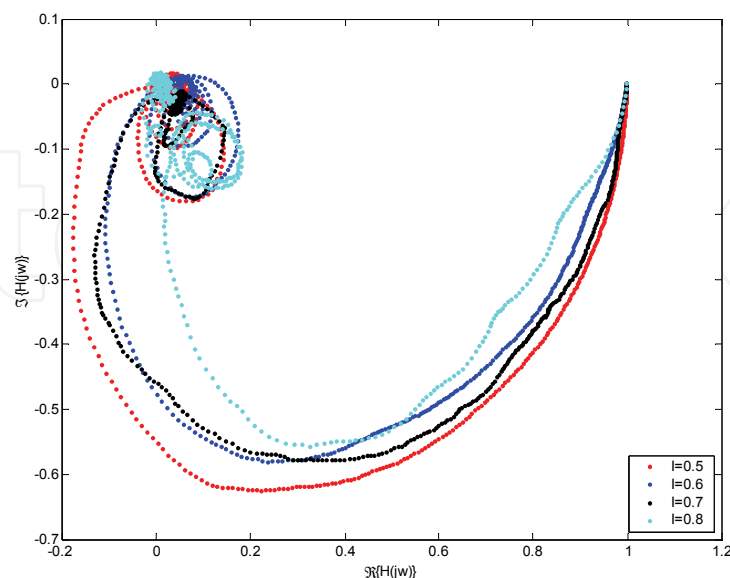
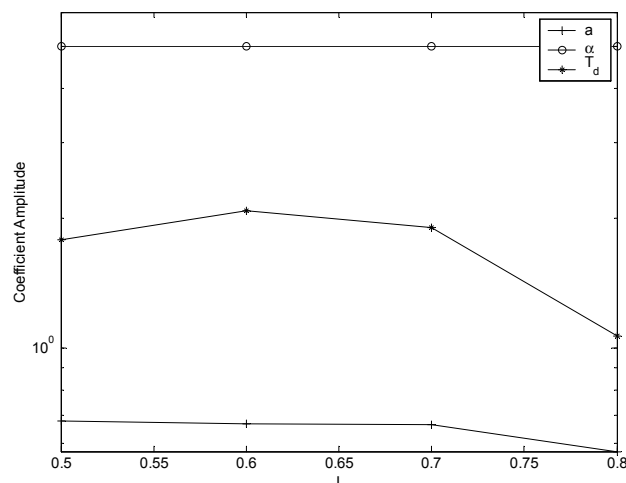


Figure 11. Median transfer function $H(j\omega)$, of the n experiments for the Bohachevsky function and $pop = 20$ elements

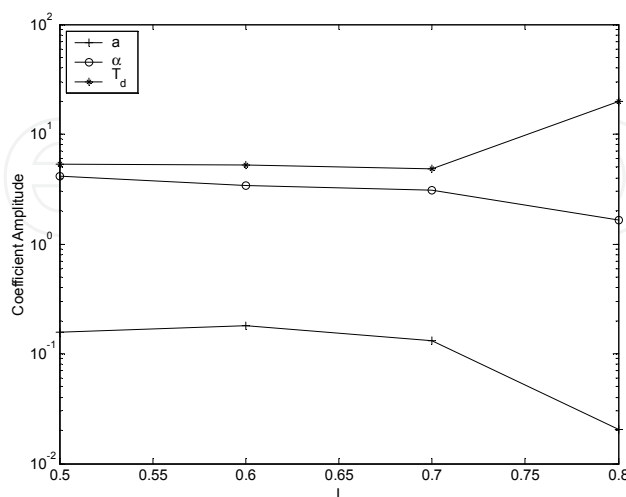
The approximations are carried out by the same *identification* PSO described previously. However, in these experiments, the medians of the numerical transfer functions are approximated by analytical expressions incorporating a time delay T_d (9).

$$G(j\omega) = \frac{e^{-j\omega T_d}}{\left(\frac{j\omega}{a} + 1\right)^\alpha} \quad (9)$$

The polar diagrams confirm the existence of a time delay T_d , which represents the perturbation propagation in the swarm evolution. Moreover, in these experiments the dynamics follows the behavior of a low-pass filter too. The parameters obtained by the *identification* PSO can be observed in Figure 12. The results reveal that the transfer function parameters $\{a, \alpha, T_d\}$ have some dependence with the inertia coefficient I .



a) Easom function



b) Bohachevsky function

Figure 12. Parameters $\{a, \alpha, T_d\}$ of $G(j\omega)$

6. Conclusion

This work analyzed the signal propagation and the phenomena involved in the discrete time evolution of a particle swarm optimization algorithm. The particle swarm algorithm was deployed as an optimization tool using three different functions as tests cases. The *optimization* PSO system was subjected to a statistical sample of tests. In each test a particle of a reference swarm was replaced by a randomly generated particle and the global population fitness perturbation effect measured. A second PSO algorithm was used to identify the parameters of a fractional order transfer function. The results indicate that the fractional calculus provides a good understanding of the effects corresponding to the propagation of the perturbations signals over the operating conditions.

7. Acknowledgment

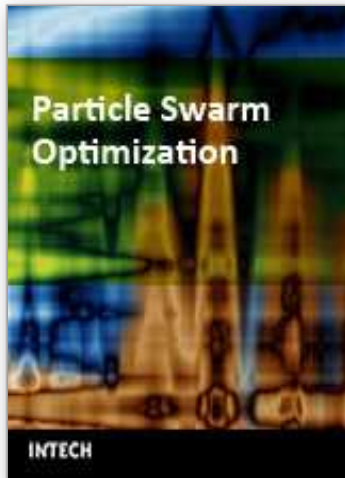
The authors would like to acknowledge the GECAD Unit.

8. References

- Clerc, M.; Kennedy, J. (2002). The particle swarm - explosion, stability, and convergence in a multidimensional complex space. *IEEE Transactions on Evolutionary Computation*, 6, 1, (2002) 58-73.
- den Bergh, F. V.; Engelbrecht, A (2006). P., A study of particle swarm optimization particle trajectories, *Inf. Sci.*, 176, 8, (2006) 937-971.
- Gement, A. (1938). On fractional differentials. *Proc. Philosophical Magazine*, 25, (2008) 540-549.
- Kennedy, J.; Eberhart, R. (1995). Particle swarm optimization, *Proceedings of the 1995 IEEE International Conference on Neural Networks*, pp. 1942-1948, Perth, Australia, IEEE Service Center, Piscataway, NJ, 1995.
- Lovbjerg, M.; Rasmussen, T. K.; Krink, T. (2001). Hybrid particle swarm optimiser with breeding and subpopulations, *Proceedings of the Genetic and Evolutionary Computation Conference (GECCO)*, pp. 469-476, San Francisco, California, USA, July 2001, Morgan Kaufmann.
- Méhauté, A. L. (1991). *Fractal Geometries: Theory and Applications*, Penton Press.
- Oustaloup, A. (1991). *La Commande CRONE: Commande Robuste d'Ordre Non Intier*, Hermes.
- Podlubny, I. (1999) *Fractional Differential Equations*, Academic Press, San Diego.
- Ross, B. (1974), *Fractional Calculus and its applications*, *Lecture Notes in Mathematics*, 457, Springer Berlin, (1974).
- Shi, Y.; Eberhart, R. C. (1999). Empirical study of particle swarm optimization, *Proceedings of the Congress of Evolutionary Computation*, pp. 1945-1950, Mayflower Hotel, Washington D.C., USA, July 1999, IEEE Press, 1999.
- Solteiro Pires, E. J.; Tenreiro Machado, J. A.; de Moura Oliveira, P. B. (2003). Fractional order dynamics in a GA planner. *Signal Processing*, 83, 11, (2003) 2377-2386.
- Solteiro Pires, E. J.; Tenreiro Machado, J. A.; de Moura Oliveira, P. B. (2006). Dynamical modelling of a genetic algorithm. *Signal Processing*, 86, 10, (2006) 2760-2770.
- Tenreiro Machado, J. A. (1997). Analysis and design of fractional-order digital control systems. *Journal System Analysis-Modelling-Simulation*, 27, (1997) 107-122.
- Tenreiro Machado, J. A. (2001) System modelling and control through fractional-order algorithms. *FCAA - Jornal of Fractional Calculus and Ap. Analysis*, 4, (2001) 47-66.

- Tenreiro Machado, J. A.; Galhano, A. M. S. F. (1998). A statistical perspective to the fourier analysis of mechanical manipulators. *Journal Systems Analysis-Modelling-Simulation*, 33 (1998), 373-384.
- Torvik, P. J.; L. Bagley, R. (1984). On the appearance of the fractional derivative in the behaviour of real materials. *ASME Journal of Applied Mechanics*, 51 (june 1984), 294-298.
- Vinagre, B. M.; Petras, I.; Podlubny, I.; Chen, Y. Q. (2002). Using fractional order adjustment rules and fractional order reference models in model-reference adaptive control, *Nonlinear Dynamics*, 29 (July 2002), 269-279.
- Westerlund, S. (2002). *Dead Matter Has Memory! Causal Consulting*. Kalmar, Sweden.

IntechOpen



Particle Swarm Optimization

Edited by Aleksandar Lazinica

ISBN 978-953-7619-48-0

Hard cover, 476 pages

Publisher InTech

Published online 01, January, 2009

Published in print edition January, 2009

Particle swarm optimization (PSO) is a population based stochastic optimization technique influenced by the social behavior of bird flocking or fish schooling. PSO shares many similarities with evolutionary computation techniques such as Genetic Algorithms (GA). The system is initialized with a population of random solutions and searches for optima by updating generations. However, unlike GA, PSO has no evolution operators such as crossover and mutation. In PSO, the potential solutions, called particles, fly through the problem space by following the current optimum particles. This book represents the contributions of the top researchers in this field and will serve as a valuable tool for professionals in this interdisciplinary field.

How to reference

In order to correctly reference this scholarly work, feel free to copy and paste the following:

E. J. Solteiro Pires, J. A. Tenreiro Machado and P. B. de Moura Oliveira (2009). Particle Swarm Optimization: Dynamical Analysis through Fractional Calculus, Particle Swarm Optimization, Aleksandar Lazinica (Ed.), ISBN: 978-953-7619-48-0, InTech, Available from:

http://www.intechopen.com/books/particle_swarm_optimization/particle_swarm_optimization__dynamical_analysis_through_fractional_calculus

INTECH
open science | open minds

InTech Europe

University Campus STeP Ri
Slavka Krautzeka 83/A
51000 Rijeka, Croatia
Phone: +385 (51) 770 447
Fax: +385 (51) 686 166
www.intechopen.com

InTech China

Unit 405, Office Block, Hotel Equatorial Shanghai
No.65, Yan An Road (West), Shanghai, 200040, China
中国上海市延安西路65号上海国际贵都大饭店办公楼405单元
Phone: +86-21-62489820
Fax: +86-21-62489821

© 2009 The Author(s). Licensee IntechOpen. This chapter is distributed under the terms of the [Creative Commons Attribution-NonCommercial-ShareAlike-3.0 License](#), which permits use, distribution and reproduction for non-commercial purposes, provided the original is properly cited and derivative works building on this content are distributed under the same license.

IntechOpen

IntechOpen