

"HARDWARE-IN-THE-LOOP" CONTROL USING THE PARTICLE SWARM OPTIMISATION ALGORITHM

J. P. Coelho^a, P.B. De Moura Oliveira^{b,c} and J. Boaventura Cunha^{b,c}

a. Instituto Politécnico de Bragança, Escola Superior de Tecnologia e Gestão
5301-854 Bragança, Portugal.

b. Universidade de Trás-os-Montes e Alto Douro, Dep. Engenharias
5001-911 Vila Real, Portugal

c. CETAV - UTAD- Centro de Estudos e Tecnologias do Ambiente e da Vida da UTAD
5001-911 Vila Real, Portugal
e-mail: jpcoelho@pol.pt

Abstract: In the last two decades, evolutionary based algorithms have proved to be an important tool in solving optimisation problems in many disciplinary areas namely in control system design. However one of its limitations, for some type of applications, is the usually high computational load required, which restricts its use for on-line control. This paper proposes the use of a stochastic search algorithm, known as particle swarm, as an optimisation tool for an on-line predictive control of a custom made thermodynamic system. Preliminary results are presented. Copyright © Controllo 2006

Keywords: Evolutionary algorithms, Model based control, Particle swarm optimization, Predictive Control.

1. INTRODUCTION

A model predictive controller (MPC) is applied to regulate the air temperature of a thermodynamic laboratory process. This type of control strategy has become very popular since the 80's (Camacho and Bordons, 1994). Compared with other control algorithms, MPC has the advantage of providing the system with the ability to react before any deviations in the controlled variable take place. This anticipatory behaviour of the controller is achieved by using a model of the process in order to predict the system response in view of a given set of future control actions. The control values to be injected in the system for a specific time horizon are usually computed by minimising a quadratic cost function of the form (Clark, *et. al.*, 1987):

$$J(k) = \lambda_1 \sum_{j=a}^b \varepsilon^2(k+j|k) + \lambda_2 \sum_{j=1}^c \Delta u^2(k+j-1) \quad (1)$$

with,

$$\varepsilon(k+j|k) = \hat{y}(k+j|k) - w(k+j) \quad (2)$$

In the above formulae, $\Delta u(k+j-1)$ represents the control effort, λ_1 and λ_2 are weights for each expression component, c represents the control horizon and constants a and b represent the instant limits in which it is desirable that the output follows the reference. $\varepsilon(k+j|k)$ is the prediction error between the future trajectory $w(k+j)$ and the predicted output $\hat{y}(k+j|k)$.

Usually the cost function J must be minimised regarding a set of design and physical constraints. It is common to consider magnitude and rate constraints for the control actions and level constraints for the output signal.

Discarding the problems associated to variable forecast, the predictive control resumes to the minimisation of a function subject to constraints. Due to the nature of this function, the controller states are obtained by iterative numerical procedures. These function minimisation strategies can be based in either deterministic or stochastic search methods. The former usually takes information provided by the gradient of the search space in order to make a decision concerning the path to follow. In general, a random kernel search procedure evolves a set of probable solutions taking into consideration only the potential of each solution and a set of transition rules.

The choice of one of the algorithms in detriment of the other depends on several factors, such as: the analyticity of the objective function and convexity of the search space. In cases where multimodal functions must be spanned or gradients can't be computed an evolutionary search of the optimum is, by norm, the best option.

In the current case the objective function presented could be minimized using regular gradient based techniques like the Levenberg-Marquardt optimisation procedure but an evolutionary based technique will be used. Please note that the purpose of this paper is not to prove that an evolutionary algorithm would outperform a more classical optimisation approach. In stead the objective is to show that nowadays, and due to the increasing computational power of the general purpose hardware, evolutionary techniques can be used in applications where the real-time constraint is factor to keep in mind.

2. THE PARTICLE SWARM ALGORITHM

In real-time control, any algorithm must be fast enough to run completely between samples. In this context, the application of a given control strategy depends on the sampling frequency, on the computational power of the hardware and on the complexity of the control algorithm. The model predictive control is, by itself, a computational heavy algorithm. The computational load depends on several factors like the prediction horizon, the complexity of the model and the performance of the search algorithm.

Among the fastest stochastic search algorithms is the particle swarm optimisation algorithm (PSO). This search strategy has more than a decade of existence and was firstly proposed by Kennedy and Eberhart (1995). Since then, several works have been published on this subject concerning his mathematical proprieties (Clerc, 1999; Kennedy and Eberhart, 2001; Shi and Eberhart, 1998; Shi and Eberhart, 1999) or applications to solve practical problems, such as the greenhouse environment control (Coelho, *et al.*, 2005).

Conceptually, the PSO is an algorithm based on the

social behaviour of groups of organisms such as herds, schools and flocks. As an evolutionary technique the PSO is a population based algorithm, formed by a set of particles, which represent a potential solution for a given problem. Each particle moves through a n -dimensional search space with an associated position vector $X_i(t) = \{x_{i1}(t), x_{i2}(t), \dots, x_{in}(t)\}$ and velocity vector $V_i(t) = \{v_{i1}(t), v_{i2}(t), \dots, v_{in}(t)\}$ for the current i particle and evolutionary iteration t .

The original PSO model integrates two types of knowledge acquisition by a particle: through its own experience and from social sharing from other population members. The former was termed cognition-only model and the latest social-only model (Kennedy, 1997). The behaviour of each particle is based on these two types of knowledge. In this context, the behaviour of particle i in the search space is governed by the following two equations:

$$v_{id}(t+1) = \omega(t) \cdot v_{id}(t) + \varphi_1 \cdot [p_{id}(t) - x_{id}(t)] + \varphi_2 \cdot [p_{gd}(t) - x_{id}(t)] \quad (3)$$

$$x_{id}(t+1) = x_{id}(t) + v_{id}(t+1) \quad (4)$$

in which d represents the dimension index, $1 \leq d \leq n$, $p_{id}(t)$ represents the best previous position of particle i in the current iteration t , $p_{gd}(t)$ represents the global best in the current iteration for a pre-defined neighbourhood type. Parameter φ_1 is known as the cognitive constant and φ_2 as the social constant, that represent uniformly distributed random numbers generated in a pre-defined interval (usually between 0 and 2). The $\omega(t)$ variable represents the inertia weight and its value affects the type of search. A large ω value will direct the PSO for a global search while a small ω will direct the PSO for a local search. In order to make a global search in the early run and more local in the end, the inertia weight can be made to vary linearly from a larger value to a smaller one. In the present work the following rule has been applied to tune the inertia weight through the iterations:

$$\omega(t) = 0.9 - 0.5 \frac{t}{MaxEpochs} \quad (5)$$

where $MaxEpochs$ refers to the limit, for the iteration number. Please note that, for a particular run, the total number of epochs can be less than the maximum number of epochs. In fact, and for the particular problem addressed in this paper, the stop criteria were, besides the maximum epoch number, the convergence of the best particle. In particular, the algorithm decides that a reasonable solution was found if the best particle doesn't change in thirty epochs.

Additionally, in each epoch, the velocity of the particles is bounded by a maximum value V_{max} . The value of this constraint is intimately connected with the maximum "jump" each particle can make. The

value selected for V_{max} should not be too high to avoid oscillations, or too low to avoid search traps. Additionally the particle's position should be, if necessary, relocated to a point inside the defined search space.

3. EXPERIMENTAL SETUP

The laboratory experimental setup was built using spare parts of a hair dryer merged with some instrumentation and power electronics. The motivation for the construction of this system was a course in digital control that took place at the Polytechnic Institute of Bragança. During lectures, students should build a replica of the system presented here in order to understand some of the building blocks of a "real" control system. They should be aware of the physical meaning of the actuators, sensors and plant transfer function, the impact, on the closed loop dynamic, of the DAC and pre-filters, identification of noise or disturbances hotspots and so on.

Essentially, the system assembled was a thermodynamic process build around an electrical heater, a fan and a pipe. The pipe used was a PVC tube with an inner diameter of 63 mm and a length of 60 cm. Additionally the two actuators, a fan and a heating resistor grid, were embedded in the tube.

While in operation regime, the fan, mechanically linked to a brush type universal electric motor, forces air to circulate through the pipe. The mass of air entering the tube is subsequently heated, at its inlet, by an electric resistor. The kinetic energy of the air molecules are then propagated through the pipe.

The control system purpose is to regulate the air temperature in a specific spot of the tube. Three temperature sensors have been installed to measure: the temperature of the heating element, the environment temperature and the outlet tube temperature (installed at, approximately, ten centimetres away from the tube outlet). A diagram of the system building blocks and its picture are showed in the following two figures.

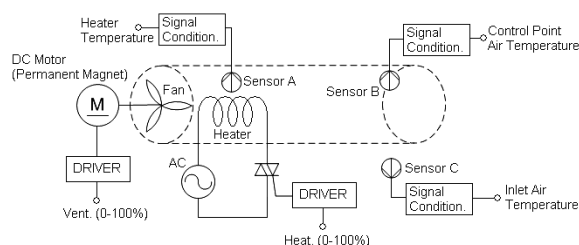


Fig. 1. Block diagram of the system to be controlled.

As one can see, the proposed system has two degrees of freedom, i.e., it is possible to manipulate the mass of air, at room temperature, entering the tube by regulating the fan speed and the heat produced by the resistor by controlling the mean power applied to this element.

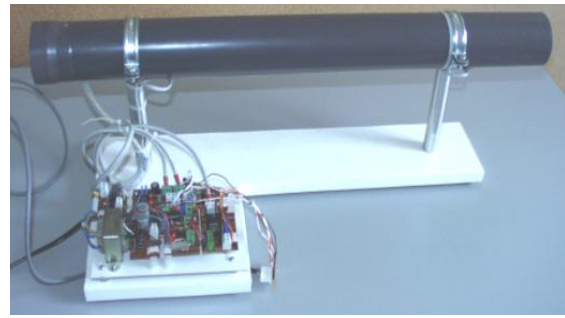


Fig. 2. Photograph of the heating process.

In order to speed-up the construction of the device, an integrated temperature sensor (LM35CZ) was used to measure the temperature. Please note that, due to the time constant of this sensor element, the dynamic added to the overall system is not negligible. In fact, a sensor with wider bandwidth, like a thermocouple, could be a best choice. However, the signal conditioning of this type of sensors is more troublesome when compared to the "plug-and-play" behaviour of the integrated temperature sensor used. Additionally the sensor used has also conditioned the resolution of the analogue to digital conversion (ADC). In fact, between the full range of operation (0-100°C), the sensor accuracy is $\pm 1^\circ\text{C}$. So the ADC resolution does not need to be better than 1%, i.e. 7 bits are enough. The communication between the prototype and the computer was handled by a custom made ISA bus data acquisition card with an 8 bit resolution.

The control and measured signals, via the data acquisition card, was manipulated in a PC compatible digital computer with a Pentium II processor running at 450 MHz. Moreover, due to the time constants involved in the process, a 1Hz sampling frequency was found to be suitable.

4. SYSTEM MODEL

Regardless of the system's two degrees of freedom, in this work the air flow rate was kept constant and the environment temperature was kept more or less constant by means of an air conditioning system. Hence the system input is a voltage that controls the mean power applied to the heater and the output is the outlet air temperature.

From the controlled temperature to the heat energy delivered by the heating element, the system could be approximated to a simple first order system with a pure time lag. However, the relationship between the control signal (a tension between 0 and 5V) and the heating energy is non-linear. Actually the power control on the resistor grid was handled by a phase based switching circuit based on the TCA785 integrated circuit. So, the shape of the wave applied to the actuator is illustrated in figure 3, where $\beta T/2$ refers to the switching angle.

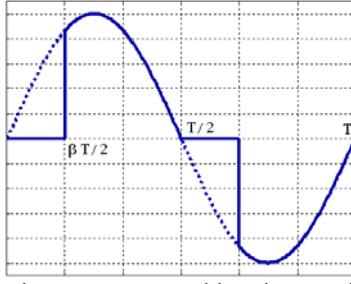


Fig. 3. Heating power control by phase switching.

By using the TCA785, the relationship between the commutation point and the relative control voltage is:

$$\beta = \frac{V_x}{V_{\max}} \quad (6)$$

As one knows, the instantaneous power applied to a static resistor R by the wave in figure 3 is (in case of a voltage),

$$P(t) = \frac{V^2(t)}{R} \quad (7)$$

Due to the low pass behaviour of the heater, the power delivered by the actuator will be, in fact, the mean value of the instantaneous powers applied in the time interval from $\beta T/2$ to $T/2$. In this case,

$$\bar{P}(V_x) = \frac{2}{TR} V_p^2 \int_{\beta \frac{T}{2}}^{\frac{T}{2}} \sin^2(\omega t) dt \quad (8)$$

So, the theoretical relationship between the heating power and the control voltage is:

$$\bar{P}(V_x) = \frac{V_p^2}{2R} \left(\left(1 - \frac{V_x}{V_{\max}}\right) + \frac{\sin(2\pi \cdot V_x / V_{\max})}{2\pi} \right) \quad (9)$$

where V_p refers to the peak value of the input sine wave, $V_x \in [0, V_{\max}]$ is the control voltage and V_{\max} is a fixed reference voltage. As can be seen by the figure below, the relationship between the heating power and the control voltage is highly non-linear.

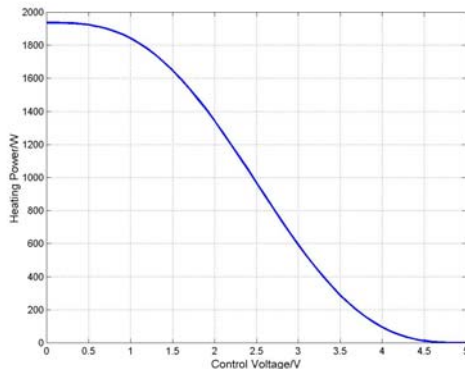


Fig. 4. Relationship between heating power and control voltage (for a 25Ω resistor@220V, 50Hz).

Due to the complexity of the physical model, and considering the fact that, for a MPC control strategy, a forecasting model must be available, a preliminary system identification process was carried out. In this procedure an excitation signal, derived from a PRBS signal (Åström and Wittenmark, 1989), with random amplitude was used. The need of several amplitude bands in the input signal is linked with the non-linear behaviour of the system (at least from the control voltage point-of-view).

Considering that the outdoor temperature and ventilation rate is approximately constant, the model that has been found sufficiently accurate in order to model the dynamic behaviour of the plant is:

$$T_{\text{pipe}}[k] = 0.96 \cdot q^{-1} T_{\text{pipe}}[k] + 0.194 \cdot q^{-1} \text{Heat}^2[k] + 2.168 \cdot q^{-2} \text{Heat}^2[k] + 0.749 \quad (10)$$

in which T_{pipe} is the outlet air temperature, Heat is the relative voltage applied to the phase control hardware that drives the heater. In the above equation q refers to the time shift operator and k refers to the sampling instant kT where T is the sampling period.

As one can see, the model incorporates not the Heat variable but its square. This variable transformation was obtained after several fitting iterations for the particular model structure presented in equation 10. Although the model seems non-linear it is still linear in the parameters. Hence the values of the model coefficients were obtained using the least squares method. In figure 5, the open-loop simulation results for the proposed model under validation data are presented.

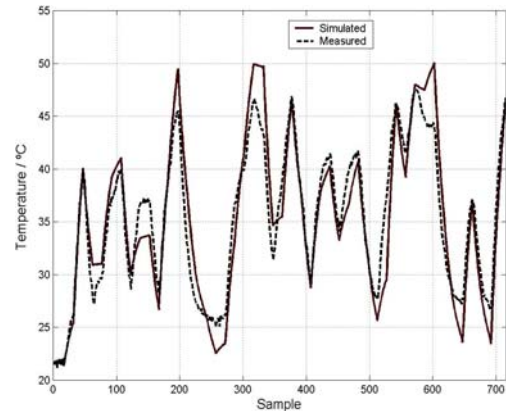


Fig. 5. Open-loop simulation of the ARX plant model under validation data.

5. PI CONTROLLER

In order to compare the performance of the investigated strategy, a PI controller was designed using classical frequency methods. The process's frequency response was obtained using, as the control voltage, a sine wave. The spectrum of this signal was swept for a set of discrete points between 2mHz and 0.3Hz. Afterwards, a continuous time

transfer function was obtained. In concrete the model that has been found to be suitable for data fitting was a first order transfer function with a pure time delay with the following parameters:

$$\frac{T_{pipe}(s)}{Heat(s)} = \frac{e^{-1.4s}}{s + 0.06} \quad (11)$$

The next figure shows the quality of the above approximation.

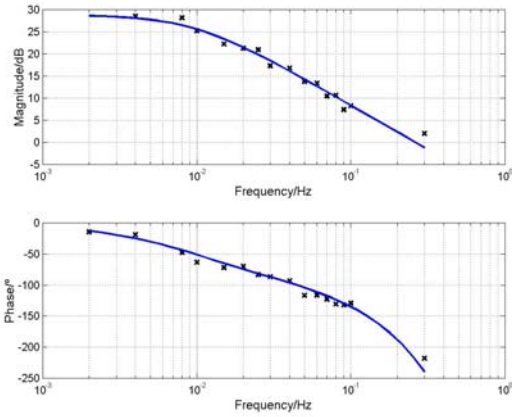


Fig. 6. Measured (dots) and approximated frequency response of the plant.

The transfer function of the digital PI controller will be obtained using an emulation procedure, i.e. the digital PI will be obtained by discretization of an analogue PI. Using the model derived and taking into consideration the following open and closed loop requirements:

- Zero steady state error to a step input
- Phase margin larger than 50° and gain margin larger than 6dB

a continuous-time PI transfer function was obtained. Subsequently, and using a Tustin approximation, the digital controller found to fulfil the above specifications was:

$$K(z) = \frac{0.06095z - 0.057405}{z(z-1)} \quad (12)$$

as one can see from the Bode plots of the open loop transfer function.

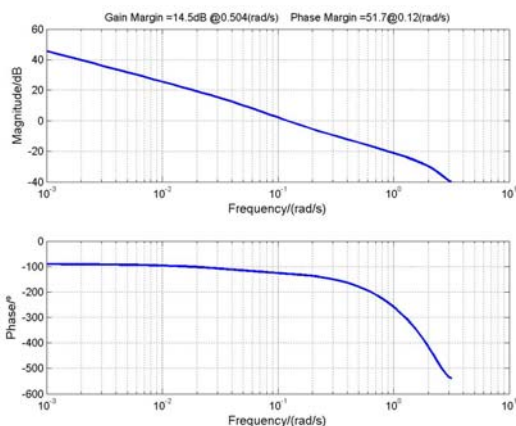


Fig. 7. Simulation of open loop frequency response and relative stability margins.

6. EXPERIMENTAL RESULTS

Although disturbance rejection is a major goal when designing a controller, in this paper only the set-point accuracy and control effort will be addressed. This constraint is made due to the fact that the inlet air temperature is the major system disturbance and can be measured. Hence, a feedforward loop can be inserted in the system in order to cope with disturbance rejection requirement. In fact, the study carried out in this paper belongs to an initial effort to study, implement and physically test a set of soft-computing techniques for multivariable control.

The results obtained are showed in the following four figures. The first set of two images concerns the PI control strategy and the last two the MPC with the PSO kernel.

Regarding the MPC strategy, the results obtained were accomplished using a fifty particle swarm PSO algorithm evolved during two hundred generations. In this experiment the prediction horizon and the control horizon was set to ten steps ahead and the weight factors λ_1 and λ_2 was set to 0,6 and 0,4, respectively.

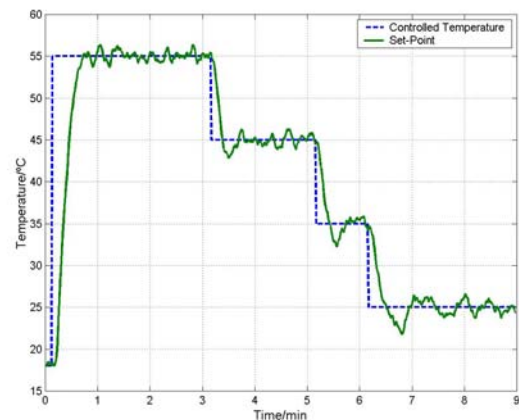


Fig. 8. Set-point tracking accuracy with the PI controller.

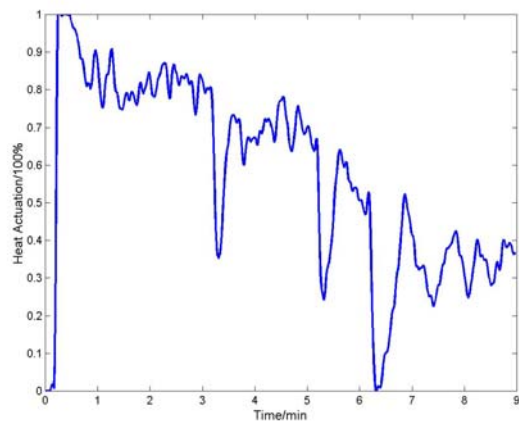


Fig. 9. Control signal computed with the PI controller.

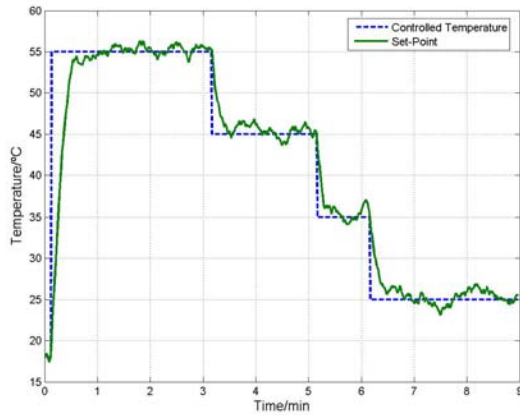


Fig. 10. Set-point tracking accuracy with the MPC controller.

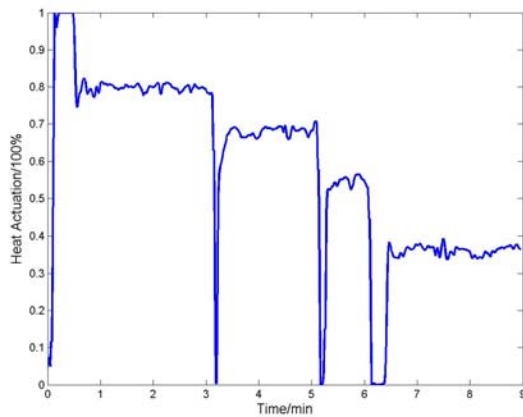


Fig. 11. Control signal computed with the MPC driven by the PSO algorithm.

From the results obtained we can conclude that, for this particular application, there is no significant performance benefit in using a more elaborated control law when comparing to a simple PI controller. But, this article aim is not to make a comparative study about the two control techniques. The objective was to show that satisfactory results can be obtained when using the PSO search algorithm within an on-line control algorithm framework. Moreover, even if the sampling interval is a constraint, this kind of stochastic search algorithm can be easily embedded in a custom chip or DSP. Although the use of this algorithm, in the context presented in this article, was not embossed due to the similar performances obtained with the MPC and the PI controllers, the advantages derived from the application of evolutionary algorithms in control applications can be more relevant when applied, for example, in discontinuous search spaces.

7. CONCLUSION

In this paper, preliminary results of a real-time controller applied to a physical plant using the particle swarm optimisation algorithm have been presented. From the results obtained one is able to conclude that this tool has real practical use outside the simulation environment. Indeed, evolutionary algorithms great potential it's their ability to evolve a

set of possible solutions in a highly sophisticated multimodal search space with large number of discontinuities. Moreover its application in multiobjective problems allied to automatic decision mechanisms can be of great interest in applications outside the computer environment. With an obsolete computer one has shown that it is possible to use this kind of evolutionary search tool in practical real-time control applications.

REFERENCES

- Åström, K. J. and B. Wittenmark (1989). *Adaptive Control*. Addison-Wesley, ISBN: 0-201558-66.
- Camacho, E. F. and C. Bordons (1994). *Model Predictive Control in the Process Industry*, Springer, Sevilla.
- Clarke, D. W., C. Mohtadi and P.S. Tuffs (1987). Generalized Predictive Control – Part I. The Basic Algorithm, *Automatica*, **Vol 23**, No 2, pp.137-148.
- Clerc M., (1999). The Swarm and the Queen Towards a Deterministic and Adaptive Particle Swarm Optimization. *Proceedings of the 1999 Congress of Evolutionary Computation*, IEEE Press, **Vol. 3**, pp. 1951-1957.
- Coelho, J. P., De Moura Oliveira, P. B and Cunha, J. B. (2005), Greenhouse Air Temperature Predictive Control using the Particle Swarm Optimisation Algorithm, *Computers and Electronics in Agriculture*, Vol. 49, pp.330-344.
- Kennedy, J. and R.C. Eberhart (1995). Particle Swarm Optimization, *Proc. Of the 1995 IEEE Int. Conf. On Neural Networks*, pp. 1942-1948. IEEE Service Center, Piscataway, NJ.
- Kennedy J. (1997). The Particle Swarm: Social Adaptation of Knowledge. *Proceedings of the 1997 IEEE Int. Conference on Evolutionary Computation, Indianapolis, US*, pp. 303-308.
- Kennedy, J. and R. C. Eberhart (2001). *Swarm Intelligence*. Academic Press, ISBN: 1-55860-595-9, pp. 344-345.
- Shi Y. and R. C. Eberhart (1998): "Parameter Selection in Particle Swarm Optimisation", *Evolutionary Programming VII, Lecture Notes in Computer Science 1447*, pp. 591-600, Springer Verlag.
- Shi Y. and R. C. Eberhart (1999). Empirical Study of Particle Swarm Optimization, *Proceedings of the 1999 IEEE Congress of Evolutionary Computation*, **Vol 3**, pp. 1945-1950.