

# 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](mailto:book.department@intechopen.com)

Numbers displayed above are based on latest data collected.  
For more information visit [www.intechopen.com](http://www.intechopen.com)



# Differential Meta-model and Particle Swarm Optimization

Jianchao Zeng<sup>1</sup> and Zhihua Cui<sup>2,1</sup>

<sup>1</sup>*Division of System Simulation and Computer Application, Taiyuan University of Science and Technology*

<sup>2</sup>*State Key Laboratory for Manufacturing Systems Engineering, Xi'an Jiaotong University  
China*

## 1. Introduction

Optimization has been an active area of research for several decades. As many real-world optimization problems become increasingly complex, better optimization algorithms are always needed. Without loss of generality, the unconstrained optimization problem is identified with the discovery of the global minimizer of a real valued objective function

$$f : S \rightarrow R \quad (1)$$

i.e. finding a point  $x^* \in S$  such that

$$f(x^*) \leq f(x), \forall x \in S \quad (2)$$

Where  $S \in R^D$  is a nonempty compact set.

There are two main categories for global optimization methods: deterministic and probabilistic methods. Traditionally, the deterministic methods need some information to determine the optima such as grads etc. However, for many real-world optimization problems, the optimization function may be discontinuous. Furthermore, the deterministic methods only can be applied into the problems with slow dimension because of the huge time-cost for large dimensionality. Therefore, from the 1960's, many researchers pay their attentions to the probabilistic methods. All of these rely on probabilistic judgements to determine whether or not search should depart from the neighbourhood of a local optimum (Forgo, 1988; Hansen, 1992; Rao, 1996).

Different from adaptive stochastic search algorithms, evolutionary computation (EC) is a new kind of probabilistic method. They exploit a set of potential solutions (called population), and detect the optimal problem solution through cooperation and competition among the individuals of the population. The well-known EC methods are all inspired from the evolution of nature such as: genetic algorithm (GA) (Goldberg, 1989; Michalewicz, 1994), evolution strategy (ES) (Back, 1996; Schwefel, 1975), evolutionary programming (EP) (Fogel, 1996), and artificial life methods.

Recently, particle swarm optimization (PSO) method (Kennedy and Eberhart, 1995) is proposed known as a member of the wide category of swarm intelligence methods

Source: Swarm Intelligence: Focus on Ant and Particle Swarm Optimization, Book edited by: Felix T. S. Chan and Manoj Kumar Tiwari, ISBN 978-3-902613-09-7, pp. 532, December 2007, Itech Education and Publishing, Vienna, Austria

(Kennedy and Eberhart, 2001). It simulates the animal social behaviour such as birds flocking, fish schooling, and animal herding.

In this paper, the basic concept of particle swarm optimization is explained. Then, a new framework for PSO - differential meta-model is proposed. Thirdly, a new modified variants – differential evolutionary PSO with PID controller is designed. Finally, we provide a conclusion and some future search topic.

## 2. Particle Swarm Optimization

In a PSO system, multiple candidate solutions coexist and collaborate simultaneously. Each solution called a "particle", flies in the problem search space looking for the optimal position to land. A particle, as time passes through its quest, adjusts its position according to its own "experience" as well as the experience of neighbouring particles. Tracking and memorizing the best position encountered build particle's experience. For that reason, PSO possesses a memory (i.e. every particle remembers the best position it reached during the past). PSO system combines local search method (through self experience) with global search methods (through neighbouring experience), attempting to balance exploration and exploitation.

Each particle maintains two character items: velocity and position. Both of them are updated as follows:

$$v_j(t+1) = v_j(t) + c_1 r_1 (p_j - x_j(t)) + c_2 r_2 (p_g - x_j(t)) \quad (3)$$

$$x_j(t+1) = x_j(t) + v_j(t+1) \quad (4)$$

where  $v_j(t)$  denotes the velocity vector of particle  $j$  at time  $t$ .  $x_j(t)$  represents the position vector of particle  $j$  at time  $t$ . Vector  $p_j$  is the memory of particle  $j$  at current generation, and vector  $p_g$  is the best location found by the whole swarm. Cognitive coefficient  $c_1$  and social coefficient  $c_2$  are known as acceleration coefficients.  $r_1$  and  $r_2$  are two random number with uniform distribution. To reduce the likelihood, a threshold is introduced to limit the value of  $v_{jk}(t+1)$  ( $k^{th}$  value of velocity vector) so that

$$|v_{jk}(t+1)| \leq v_{\max} \quad (5)$$

The well-known earliest modification is the introduction of inertia weight  $w$  (Shi and Eberhart, 1998). The inertia weight is a scaling factor associated with the velocity during the previous time step, resulting a new velocity update equation, so that

$$v_j(t+1) = wv_j(t) + c_1 r_1 (p_j - x_j(t)) + c_2 r_2 (p_g - x_j(t)) \quad (6)$$

Therefore, the original PSO can be obtained by setting  $w = 1$ . Empirical experiments have been performed with an inertia weight set to decrease linearly from 0.9 to 0.4 during the course of a simulation. There are still many other variants to improve the convergence speed, such as PSO with constriction factor (Clerc, 1999), Spatial neighbourhood (Suganthan, 1999), etc. The details please refer to the corresponding references.

## 3. Differential Meta-model Particle Swarm Optimization

In this section, a uniform model -differential meta-model (Zeng and Cui, 2005) is presented based on the analysis of the standard PSO and its several variants. Then, the convergence

performance is analyzed with linear control theory, and the upper bound estimate of the convergence speed is given using Lyapunov function. Finally, an adaptive parameters adjusted algorithms is given to improve the global optimality of convergence.

### 3.1 Differential Meta-model

Consider the following differential equations:

$$\begin{cases} \frac{dv_i(t)}{dt} = \chi \left[ \left( w - \frac{1}{\chi} \right) v_i(t) + c_1 r_1 (p_i - x_i(t)) + c_2 r_2 (p_g - x_i(t)) \right] \\ \frac{dx_i(t)}{dt} = v_i(t+1) \end{cases} \quad (7)$$

If Euler numeral integral method is used with step length one, we can obtain the following results:

1. When  $w = 1$  and  $\chi = 1$ , the original PSO is obtained;
2. The standard PSO with inertia weight  $w$  is obtained with  $w \neq 1$  and  $\chi = 1$ .

$$\chi = \frac{2}{|2 - \varphi_{12} - \sqrt{\varphi_{12}^2 - 4\varphi_{12}}|} (\varphi_{12} = c_1 r_1 + c_2 r_2)$$

3. When  $w = 1$  and  $\chi = \frac{2}{|2 - \varphi_{12} - \sqrt{\varphi_{12}^2 - 4\varphi_{12}}|} (\varphi_{12} = c_1 r_1 + c_2 r_2)$ , the PSO with constriction factor is obtained.
4. When  $w = 0$  and  $\chi = 1$ , the stochastic PSO is obtained (Cui and Zeng, 2004).

Therefore, based on the choice of the parameters  $w$  and  $\chi$ , the different PSO evolutionary model is represented by equation (7). In other words, equation (7) can be used to express the meta-model for PSO, called differential meta-model of PSO. For convenience, the following symbols are defined as:

$$a_0 = \chi \left( w - \frac{1}{\chi} \right) \quad (8)$$

$$\varphi_1 = \chi c_1 r_1, \varphi_2 = \chi c_2 r_2 \quad (9)$$

$$\varphi_{12} = \varphi_1 + \varphi_2 \quad (10)$$

$$\varphi_p = \varphi_1 p_i + \varphi_2 p_g \quad (11)$$

Then, equation (7) can be expressed as

$$\begin{cases} \frac{dv(t)}{dt} = a_0 v(t) - \varphi_{12} x(t) + \varphi_p \\ \frac{dx(t)}{dt} = v(t+1) \end{cases} \quad (12)$$

which is the standard form of differentia meta-model of PSO.

### 3.2 Analysis of PSO Evolutionary Behavior

The first-order approximate for  $v(t+1)$  is used to replace  $v(t+1)$ , i.e.,

$$v(t+1) = v(t) + \frac{dv(t)}{dt},$$

then resulting the following equation (12):

$$\begin{cases} \frac{dv(t)}{dt} = a_0 v(t) - \varphi'_{12} x(t) + \varphi'_p \\ \frac{dx(t)}{dt} = (a_0 + 1)v(t) - \varphi'_{12} x(t) + \varphi'_p \end{cases} \quad (13)$$

Define:

$$Y(t) = \begin{bmatrix} v(t) \\ x(t) \end{bmatrix}, U(t) = \begin{bmatrix} P_i \\ P_g \end{bmatrix}, A = \begin{bmatrix} a_0 & -\varphi'_{12} \\ 1+a_0 & -\varphi'_{12} \end{bmatrix}, B = \begin{bmatrix} \varphi'_1 & \varphi'_2 \\ \varphi'_1 & \varphi'_2 \end{bmatrix} \quad (14)$$

The standard state equation of linear system is obtained:

$$\dot{Y}(t) = AY(t) + BU(t) \quad (15)$$

and its solution is solved as follows:

$$Y(t) = e^{A(t-t_0)}Y(t_0) + \int_0^t e^{A(t-\tau)}BU(\tau)d\tau \quad (16)$$

From linear system theory, if all eigenvalues of matrix A have negative real part, equation (16) is convergence. The eigen-equation of matrix A is listed as follows:

$$|\lambda I - A| = \lambda^2 + (\varphi'_{12} - a_0)\lambda + \varphi'_{12} = 0 \quad (17)$$

and its eigenvalues are solved:

$$\lambda_{12} = \frac{a_0 - \varphi'_{12} \pm \sqrt{(\varphi'_{12} - a_0)^2 - \varphi'_{12}}}{2} \quad (18)$$

Therefore, both eigenvalues of matrix A have negative real part if  $a_0 - \varphi'_{12} < 0$  is true.

In other words, if  $a_0 - \varphi'_{12} < 0$  is satisfied, the differential PSO evolutionary model described by equation (13) is convergent, and the limit is obtained by the follows.

$$\lim_{t \rightarrow +\infty} Y(t) = \lim_{t \rightarrow +\infty} e^{A(t-t_0)}Y(t_0) + \lim_{t \rightarrow +\infty} \int_0^t e^{A(t-\tau)}BU(\tau)d\tau = \begin{bmatrix} 0 \\ \varphi'_p \\ \varphi'_{12} \end{bmatrix} \quad (19)$$

that is:

$$\lim_{t \rightarrow +\infty} v(t) = 0 \quad (20)$$

$$\lim_{t \rightarrow +\infty} x(t) = \frac{\varphi_p'}{\varphi_{12}'} \quad (21)$$

Equation (21) implies that following conclusion is true:

$$\lim_{t \rightarrow +\infty} (\varphi_1' + \varphi_2')x(t) = \varphi_1'P_i + \varphi_2'P_g \quad (22)$$

Because of the randomness of  $\varphi_1'$  and  $\varphi_2'$ , the above equation is satisfied if and only if  $\lim_{t \rightarrow +\infty} x(t) = P_i = P_g$  is true.

It can be seen that all the particles according to the evolutionary equation (13) is convergent to the best position  $p_g$  discovered by any of the particles when  $a_0 - \varphi_{12}' < 0$  is satisfied. In other words, if the best position  $p_g$  keeps constant within the test generation, the positions of all particles will converge to  $p_g$ .

Meanwhile, from equation (15) and  $U(t) = \begin{bmatrix} P_i \\ P_g \end{bmatrix}$ , the behavior of PSO is actually identical

to the trajectory of a linear system under the step signal inputs with stochastic disturbance, and the amplitude of step signal inputs is increased or not decreased within the evolution of PSO.

In following section, the upper bound estimate of convergence speed for equation (15) is deduced. A Lyapunov function is defined as follows

$$V(Y) = Y^T P Y \quad (23)$$

and

$$\dot{V}(Y) = -Y^T Q Y \quad (24)$$

where  $P$  is a positive definite matrix and  $Q$  is a positive definite symmetrical matrix, and following Lyapunov equation is satisfied:

$$A^T P + P A = -Q \quad (25)$$

From the view of linear system, the convergence performance of system can be evaluated by:

$$\eta = -\frac{\dot{V}(Y)}{V(Y)} \quad (26)$$

Evidently, the less is  $V(Y)$  and the larger for absolute  $\dot{V}(Y)$ , the larger value  $\eta$ , the faster convergence speed, and vice versa.

Integrate equation (26) with  $t$  from 0 to  $t$ , resulting:

$$-\int \eta dt = \int \frac{\dot{V}(Y)}{V(Y)} dt = \ln \frac{V(Y)}{V(Y_0)} \quad (27)$$

Further,

$$V(Y) = V(Y_0)e^{-\int_0^t \eta dt} \quad (28)$$

Equation (28) is hard to solve, since then, we suppose

$$\eta_{\min} = \min_Y \left\{ -\frac{V(Y)}{V(Y)} \right\} = \text{constant} \quad (29)$$

$\eta$  in equation (28) is replaced by  $\eta_{\min}$  may result

$$V(Y) \leq V(Y_0)e^{-\int_0^t \eta_{\min} dt} = V(Y_0)e^{-\eta_{\min} t} \quad (30)$$

It can be seen that when  $\eta_{\min}$  is determined, the upper bound estimate of  $V(Y)$  convergent time can be evaluated. From equations (23) and (24), we have

$$\eta_{\min} = \min_Y \left\{ -\frac{V(Y)}{V(Y)} \right\} = \min_Y \left\{ \frac{Y^T Q Y}{Y^T P Y} \right\} = \min \{ Y^T Q Y, Y^T P Y = 1 \} \quad (31)$$

Then, for linear constant system,

$$\eta_{\min} = \lambda_{\min} (QP^{-1}) \quad (32)$$

where,  $\lambda_{\min}(\bullet)$  is the minimum eigenvalue of  $(\bullet)$ .

### 3.3 Adaptive Parameter Adjustment Strategy

A differential form of PSO is given in equation (12), and the convergence condition is easy to satisfy. In order to improve the global convergence performance, an adaptive parameter adjustment strategy is discussed in this part.

From the conducted convergence condition  $a_0 - \dot{\varphi}_{12} < 0$ , we have the less value  $a_0 - \dot{\varphi}_{12} < 0$ , the faster convergence speed it is. By this way, we select the parameter values so that the convergence condition  $a_0 - \dot{\varphi}_{12} < 0$  is always true in the evolutionary iteration.

Further, we let  $a_0 - \dot{\varphi}_{12} < 0$  has little absolute value in the earlier stage, while in the later stage of evolution,  $a_0 - \dot{\varphi}_{12} < 0$  has a larger absolute value. In other words, there is a slow convergence speed and a powerful global exploration capacity in the earlier period of evolution, as well as in the later period of evolution, the local exploitation capacity is addressed and convergence speed is faster. In order to balance the ability of the global exploration and the local exploitation, an adaptive parameter adjustment strategy is proposed.

With the definition of  $\dot{\varphi}_{12}$ ,

$$\dot{\varphi}_{12} = \dot{\varphi}_1 + \dot{\varphi}_2 = r_1 c_1 + r_2 c_2 \quad (33)$$

The convergence condition implies  $a_0 < c_1' + c_2'$ . Evidently, the differential PSO algorithm has three parameters  $a_0$ ,  $c_1'$  and  $c_2'$ . From the experiment results,  $c_1' + c_2' = 4$  can make a better performance. Therefore, the parameters are adjusted as follows:

$$c_1' = 1 + at \quad (34)$$

$$c_2' = 3 - at \quad (35)$$

$$a_0 = c_1' + c_2' - \beta \quad (36)$$

Where  $t$  is evolutionary generation,  $a$  and  $\beta$  are two adjusted parameters to control the change rate. Generally,  $a = 0.01 \sim 0.001$ ,  $\beta = 0.01 \sim 0.005$ .

#### 4. Differential Evolutionary Particle Swarm Optimization with Controller (Zeng & Cui, 2005)

With the same model (7), we suppose  $\varphi_0 = \chi(w - \frac{1}{\chi})$ ,  $\varphi_1 = \chi c_1 r_1$ ,  $\varphi_2 = \chi c_2 r_2$ , then equation (7) can be changed as follows:

$$\begin{cases} \frac{dv_i(t)}{dt} = \varphi_0 v_i(t) + \varphi_1 (p_i - x_i(t)) + \varphi_2 (p_g - x_i(t)) \\ \frac{dx_i(t)}{dt} = v_i(t+1) \end{cases} \quad (37)$$

The PSO algorithm described by differential evolutionary equations (37) is called differential evolutionary PSO (DEPSO). The analysis of the evolutionary behavior of DEPSO is made by transfer function as follows. The first order difference approximation of  $v_i(t+1)$  is  $v_i(t+1)$

$= v_i(t) + \frac{dv_i(t)}{dt}$ , then equation (37) will be

$$\begin{cases} \frac{dv_i(t)}{dt} = \varphi_0 v_i(t) + \varphi_1 (p_i - x_i(t)) + \varphi_2 (p_g - x_i(t)) \\ \frac{dv_i(t)}{dt} = (\varphi_0 + 1)v_i(t) + \varphi_1 (p_i - x_i(t)) + \varphi_2 (p_g - x_i(t)) \end{cases} \quad (38)$$

Laplace transformation is made on equation (38), and suppose the initial values of  $v_i(t)$  and  $x_i(t)$  are zero, we have

$$sV_i(s) = \varphi_0 V_i(s) + \varphi_1 (P_i(s) - X_i(s)) + \varphi_2 (P_g(s) - X_i(s)) \quad (39)$$

$$sX_i(s) = (\varphi_0 + 1)V_i(s) + \varphi_1 (P_i(s) - X_i(s)) + \varphi_2 (P_g(s) - X_i(s)) \quad (40)$$

From equation (39), it is known that



$$V_i(s) = \frac{\varphi_1}{s - \varphi_0} (P_i(s) - X_i(s)) + \frac{\varphi_2}{s - \varphi_0} (P_g(s) - X_i(s)) \quad (41)$$

Substituting (41) into (40) yields

$$X_i(s) = \frac{\varphi_1(s+1)}{s(s-\varphi_0)} (P_i(s) - X_i(s)) + \frac{\varphi_2(s+1)}{s(s-\varphi_0)} (P_g(s) - X_i(s)) \quad (42)$$

Suppose  $P_i(s)$  and  $P_g(s)$  are two input variables,  $X_i(s)$  is output variable, then the system structure reflecting by equation (42) can be shown as in Fig.1.

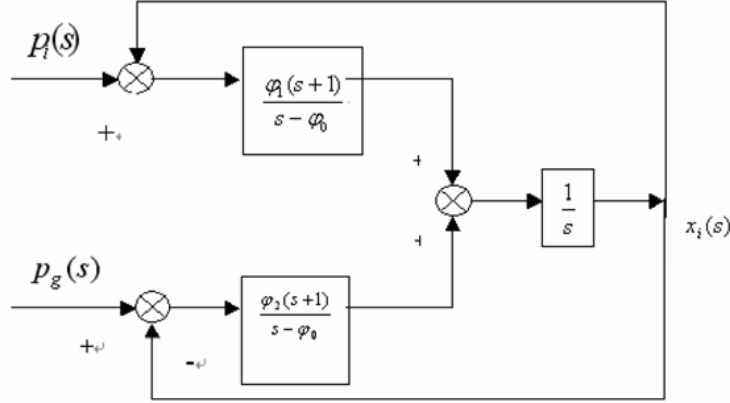


Figure 1. The System Diagram of DEPSO

The open-loop transfer function from  $P_i(s)$  to  $X_i(s)$  is

$$G_{K1}(s) = \frac{\varphi_1(s+1)}{s(s-\varphi_0)} \quad (43)$$

And the eigenequation is  $1 + G_K(s) = 0$ , thus results in

$$s^2 + (\varphi_1 - \varphi_0)s + \varphi_1 = 0 \quad (44)$$

the two eigenvalues are

$$\lambda_{1,2} = \frac{\varphi_0 - \varphi_1 \pm \sqrt{(\varphi_1 - \varphi_0)^2 - 4\varphi_1}}{2} \quad (45)$$

DEPSO will converge when  $\lambda_1$  and  $\lambda_2$  have negative real parts. This is obtained with

$$(\varphi_1 - \varphi_0)^2 - 4\varphi_1 > 0 \text{ .and. } \sqrt{(\varphi_1 - \varphi_0)^2 - 4\varphi_1} < \varphi_1 - \varphi_0 \quad (46)$$

So, the convergence of DEPSO with  $P_i(s)$  as input can be guaranteed if

$$\varphi_1 - \varphi_0 > 0 \quad (47)$$

By the same way, the convergence of DEPSO with  $P_g(s)$  and  $X_i(s)$  as input and output respectively can be guaranteed if  $\varphi_2 - \varphi_0 > 0$ . Therefore, the convergence condition of DEPSO is

$$\varphi_0 < \min(\varphi_1, \varphi_2) \tag{48}$$

From Fig.1, it is obvious that

$$X_i(s) = \frac{\varphi_1(s+1)P_i(s) + \varphi_2(s+1)P_g(s)}{s(s-\varphi_0) + \varphi_1(s+1) + \varphi_2(s+1)} \tag{49}$$

From (48), we have

$$\begin{aligned} \lim_{t \rightarrow \infty} x_i(t) &= \lim_{s \rightarrow 0} sX_i(s) = \lim_{s \rightarrow 0} \frac{\varphi_1(s+1)P_i(s) + \varphi_2(s+1)P_g(s)}{s(s-\varphi_0) + \varphi_1(s+1) + \varphi_2(s+1)} \\ &= \frac{\varphi_1 P_i + \varphi_2 P_g}{\varphi_1 + \varphi_2} \end{aligned} \tag{50}$$

it means

$$-(\varphi_1 + \varphi_2) \lim_{t \rightarrow \infty} x_i(t) + \varphi_1 p_i + \varphi_2 p_g = 0 \tag{51}$$

Since  $\varphi_1$  and  $\varphi_2$  are stochastic variables, it is obviously that the above equation is satisfied only if

$$\lim_{t \rightarrow \infty} x_i(t) = p_i = p_g \tag{52}$$

To improve the dynamic evolutionary behaviour of DEPSO, the evolutionary function of DEPSO is considered as a control plant and PID controller is introduced. The parameter of PID controller can be dynamically adjusted in the evolutionary process, and the new algorithm is called PID-DEPSO.

The system structure is showed in Fig.2.

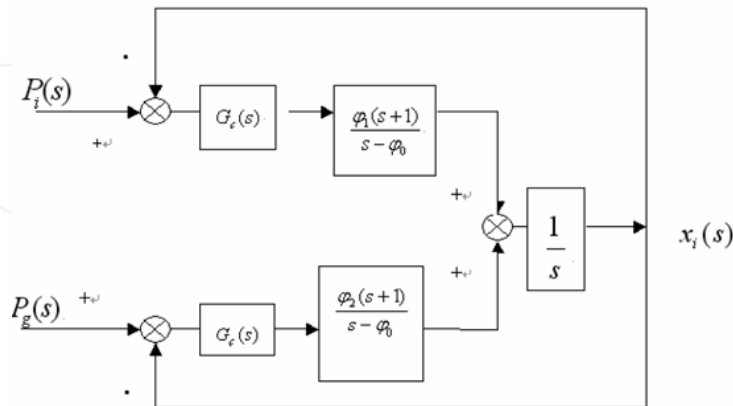


Figure 2. The System Diagram of PID-DEPSO

From Fig.2, we have

$$G_c(s) = K_p \left(1 + \frac{1}{T_I s} + T_D s\right) = K_p' \frac{T_D T_I s^2 + T_I s + 1}{s} \quad (53)$$

Where  $K_p' = \frac{K_p}{T_I}$ .

The open-loop transfer function taking  $P_i(s)$  as input is :

$$G_{Kc_1}(s) = G_c(s) \frac{\varphi_1(s+1)}{s(s-\varphi_0)} = \frac{\varphi_1 K_p' (s+1)(T_D T_I s^2 + T_I s + 1)}{s^2(s-\varphi_0)} \quad (54)$$

its eigenequation is  $1 + G_{Kc_1}(s) = 0$ , then

$$\begin{aligned} & (K_p' \varphi_1 T_D T_I + 1)s^3 \\ & = (K_p' \varphi_1 T_I + K_p' \varphi_1 T_D T_I - \varphi_0)s^2 + K_p' \varphi_1 (1 + T_I)s + K_p' \varphi_1 = 0 \end{aligned} \quad (55)$$

According to Routh's stability criteria, the stability condition of the system with  $P_i(s)$  and  $X_i(s)$  as input and output respectively are

$$K_p' T_D (T_I + 1) > \frac{\varphi_0}{\varphi_1} \quad (56)$$

As a result, the stability of PID-DEPSO is stability if

$$K_p' T_D (T_I + 1) > \max\left(\frac{\varphi_0}{\varphi_1}, \frac{\varphi_0}{\varphi_2}\right) \quad (57)$$

Similarly, if (57) is satisfied,  $\lim_{t \rightarrow \infty} x_i(t) = \lim_{s \rightarrow 0} sX_i(s)$ . From Fig.2, we have

$$X_i(s) = \frac{K_p' (s+1)(T_D T_I s^2 + T_I s + 1)(\varphi_1 P_i(s) + \varphi_2 P_g(s))}{s^2(s-\varphi_0) + K_p' (s+1)(T_D T_I s^2 + T_I s + 1)(\varphi_1 + \varphi_2)} \quad (58)$$

Thus

$$\lim_{t \rightarrow \infty} x_i(t) = \lim_{s \rightarrow 0} sX_i(s) = \frac{\varphi_1 p_i + \varphi_2 p_g}{\varphi_1 + \varphi_2} \quad (59)$$

The evolutionary equation of PID-DEPSO is deduced as follow:

$$\begin{aligned} X_i(s) &= G_{kc1}(s)(P_i(s) - X_i(s)) + G_{kc2}(s)(P_g(s) - X_i(s)) = \\ & \frac{(s+1)(T_D T_I s^2 + T_I s + 1)}{s^2(s-\varphi_0)} [K_p' \varphi_1 (P_i(s) - X_i(s)) + K_p' \varphi_2 (P_g(s) - X_i(s))] \end{aligned} \quad (60)$$

it means:

$$\frac{s^2(s-\varphi_0)}{(s+1)(T_D T_i s^2 + T_i s + 1)} X_i(s) = K_p' \varphi_1 (P_i(s) - X_i(s)) + K_p' \varphi_2 (P_g(s) - X_i(s)) \quad (61)$$

Suppose

$$\frac{dx_i(t)}{dt} = v_i(t+1) = v_i(t) + \frac{dv_i(t)}{dt} \quad (62)$$

then  $V_i(s) = \frac{s}{s+1} X_i(s)$ , substituting it into (61), we have

$$\frac{s(s-\varphi_0)}{(s+1)(T_D T_i s^2 + T_i s + 1)} V_i(s) = K_p' \varphi_1 (P_i(s) - X_i(s)) + K_p' \varphi_2 (P_g(s) - X_i(s)) \quad (63)$$

Let

$$\frac{dv_i(t)}{dt} = \varphi_0 v_i(t) + a_i(t) \quad (64)$$

then  $A_i(s) = (s - \varphi_0)V_i(s)$ , substituting it into (63), we have

$$\frac{sA_i(s)}{(s+1)(T_D T_i s^2 + T_i s + 1)} = K_p' \varphi_1 (P_i(s) - X_i(s)) + K_p' \varphi_2 (P_g(s) - X_i(s)) \quad (65)$$

Let  $a = 1 + K_p' T_D T_i (\varphi_1 + \varphi_2)$ ,  $\beta = K_p' T_i (\varphi_1 + \varphi_2) (\varphi_0 T_D + T_D + 1)$ , and the Laplace inverse transformation of  $sP_i(s)$ ,  $s^2 P_i(s)$ ,  $sP_g(s)$  are zero, then

$$\frac{da_i(t)}{dt} = -\frac{\beta}{a} V_i(t) - \frac{\beta}{a} a_i(t) + K_p' \cdot \frac{\varphi_1}{a} \cdot (P_i - X_i(t)) + K_p' \cdot \frac{\varphi_2}{a} \cdot (P_g - X_i(t)) \quad (66)$$

Let  $\frac{\beta}{a} = \gamma$ ,  $K_p' \cdot \frac{\varphi_1}{a} = \varphi_1'$ ,  $K_p' \cdot \frac{\varphi_2}{a} = \varphi_2'$  then the evolutionary equations of PID-DEPSO are

$$\frac{da_i(t)}{dt} = -\varphi_0 \gamma v_i(t) - \gamma a_i(t) + \varphi_1' (p_i - x_i(t)) + \varphi_2' (p_g - x_i(t)) \quad (67)$$

$$\frac{dv_i(t)}{dt} = \varphi_0 v_i(t) + a_i(t) \quad (68)$$

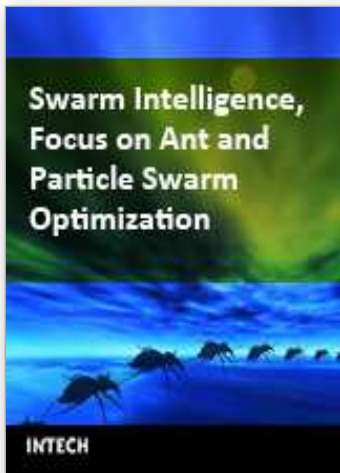
$$\frac{dx_i(t)}{dt} = v_i(t+1) = v_i(t) + \frac{dv_i(t)}{dt} = (\varphi_0 + 1)v_i(t) + a_i(t) \quad (69)$$

## 5. Conclusion

This chapter introduces one uniform differential meta-model, and a new variant development for PSO combined with PID controller is proposed. The current results show it is an interesting area with the control theory to improve the performance. The future research includes incorporating some other controllers into the PSO methodology.

## 6. References

- Back, T. (1996). *Evolutionary Algorithms in Theory and Practice*. Oxford University Press, New York.
- Clerc, M. (1999). The Swarm and the Queen: Towards a Deterministic and Adaptive Particle Swarm Optimization. *Proceedings of the Congress on Evolutionary Computation*, pp.1951- 1957, Washington DC, USA, July 1999. IEEE Service Center, Piscataway, NJ.
- Cui, Z.H. & Zeng, J.C. (2004). A New Particle Swarm Optimizer. *Proceedings of the 2004 Congress on Evolutionary Computation*, pp.316-319. IEEE Service Center, Piscataway, NJ.
- Fogel, D. (1996). *Evolutionary Computation: Towards a New Philosophy of Machine Intelligence*. IEEE Press, Piscataway, NJ.
- Forgo, F. (1988). *Nonconvex Programming*. Akademiai Kiado, Budapest.
- Goldberg, D. (1989). *Genetic Algorithms in Search, Optimization, and Machine Learning*. Addison Wesley, Reading, MA.
- Hansen, E.R. (1992). *Global Optimization Using Interval Analysis*. Marcel Dekker, New York.
- Kennedy, J. & Eberhart, R.C. (1995). Particle Swarm Optimization. *Proceedings of IEEE International Conference on Neural Networks*, IV: pp. 1942-1948. IEEE Service Center, Piscataway, NJ.
- Kennedy, J. & Eberhart, R.C. (2001). *Swarm Intelligence*. Morgan Kaufmann Publishers.
- Michalewicz, Z. (1994). *Genetic Algorithms + Data Structures = Evolution Programs*. Springer, Berlin.
- Rao, S.S. (1996). *Engineering Optimization-Theory and Practice*. Wiley.
- Schwefel, H.P. (1975). *Evolutionsstrategie und numerische Optimierung*. Technical University of Berlin, Department of Process Engineering, *Dr.-Ing.Thesis*.
- Shi, Y. & Eberhart, R.C. (1998). A Modified Particle Swarm Optimizer. In: *Porto VW, Saravanan N, Waagen D and Eiben AE (eds) Evolutionary Programming VII*, pp. 611-616, Springer.
- Suganthan, P.N. (1999). Particle Swarm Optimizer with Neighbourhood Operator. *Proceedings of the Congress on Evolutionary Computation*, pp.1958-1961, Washington DC, USA, July 1999. IEEE Service Center, Piscataway, NJ.
- Zeng, J.C. & Cui, Z.H.(2005). A Differential Meta-model for Particle Swarm Optimization. *Progress in Intelligence Computation & Applications*, pp.159-164, Wuhan, China.
- Zeng, J.C. & Cui, Z.H. (2005). A Differential Evolutionary Particle Swarm Optimization with Controller. In: Wang, L., Chen K. and Ong, Y.S. (eds) *Lecture Notes in Computer Science*, Vol.3612, pp.467-476, Changsha, China.



## **Swarm Intelligence, Focus on Ant and Particle Swarm Optimization**

Edited by Felix T.S. Chan and Manoj Kumar Tiwari

ISBN 978-3-902613-09-7

Hard cover, 532 pages

**Publisher** I-Tech Education and Publishing

**Published online** 01, December, 2007

**Published in print edition** December, 2007

In the era of globalisation, the emerging technologies are governing engineering industries to a multifaceted state. The escalating complexity has demanded researchers to find the possible ways of easing the solution of the problems. This has motivated the researchers to grasp ideas from nature and implant them in the engineering sciences. This way of thinking led to the emergence of many biologically inspired algorithms that have proven to be efficient in handling computationally complex problems with competence, such as Genetic Algorithm (GA), Ant Colony Optimization (ACO), Particle Swarm Optimization (PSO), etc. Motivated by the capability of the biologically inspired algorithms, the present book on "Swarm Intelligence: Focus on Ant and Particle Swarm Optimization" aims to present recent developments and applications concerning optimization with swarm intelligence techniques. The papers selected for this book comprise a cross-section of topics that reflect a variety of perspectives and disciplinary backgrounds. In addition to the introduction of new concepts of swarm intelligence, this book also presented some selected representative case studies covering power plant maintenance scheduling; geotechnical engineering; design and machining tolerances; layout problems; manufacturing process plan; job-shop scheduling; structural design; environmental dispatching problems; wireless communication; water distribution systems; multi-plant supply chain; fault diagnosis of airplane engines; and process scheduling. I believe these 27 chapters presented in this book adequately reflect these topics.

### **How to reference**

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

Jianchao Zeng and Zhihua Cui (2007). Differential Meta-model and Particle Swarm Optimization, Swarm Intelligence, Focus on Ant and Particle Swarm Optimization, Felix T.S. Chan and Manoj Kumar Tiwari (Ed.), ISBN: 978-3-902613-09-7, InTech, Available from:

[http://www.intechopen.com/books/swarm\\_intelligence\\_focus\\_on\\_ant\\_and\\_particle\\_swarm\\_optimization/differential\\_meta-model\\_and\\_particle\\_swarm\\_optimization](http://www.intechopen.com/books/swarm_intelligence_focus_on_ant_and_particle_swarm_optimization/differential_meta-model_and_particle_swarm_optimization)

**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](http://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

© 2007 The Author(s). Licensee IntechOpen. This chapter is distributed under the terms of the [Creative Commons Attribution-NonCommercial-ShareAlike-3.0 License](https://creativecommons.org/licenses/by-nc-sa/3.0/), 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