ARPN Journal of Engineering and Applied Sciences

COMPARATIVE STUDY AND IMPLEMENTATION OF MULTI-OBJECTIVE PSO ALGORITHM USING DIFFERENT INERTIA WEIGHT TECHNIQUES FOR OPTIMAL CONTROL OF A CSTR PROCESS

A. Jayachitra[1] and R. Vinodha[2]

[1]Department of Electronics and Communication Engineering, V. R. S College of Engineering and Technology, Villupuram, Tamil Nadu, India

[2]Department of Electronics and Instrumentation Engineering, Annamalai University, Annamalai Nagar, Tamil Nadu, India

E-Mail: jayachitra98@gmail.com

ABSTRACT

In this paper, different Inertia weight techniques in Particle Swarm Optimization algorithm (PSO) have been compared to search for the optimal PID controller gains for a Continuous Stirred Tank Reactor (CSTR) process. The optimization problem considered is highly nonlinear, complex with multiple objectives, wide operating range and constraints. The efficiency of PSO algorithm lies in the suitable selection of Inertia weight (w) to provide a balance between global exploration and local exploitation which in turn ensures the convergence behaviour of particles. The standard PSO algorithm has premature and local convergence phenomenon when solving complex optimization problem. The proposed approach is efficient in achieving stable convergence characteristics, good computational efficiency and capability to avoid from local optima. In the present study an attempt has been made to review some of the inertia weight techniques. Simulation results demonstrate that Adaptive Inertia weight Particle Swarm Optimization (AWPSO) technique is superior to all PSOs considered with various Inertia weight methods for both single objective and multi-objective functions.

Keywords: CSTR process, PID controller, particle swarm optimization algorithm, inertia weight techniques.

INTRODUCTION

Despite of vast development in control theory, the majority of industrial processes is controlled by the well-established PID controller [1, 2]. The popularity of PID control has been appreciated for its simplicity in terms of design, parameter tuning and to its good performance in a narrow range of operating conditions [3]. Designing controller for unstable system is complicated, since most of the chemical loops exhibit stable and unstable steady states. Also, it exhibits unexpected overshoot or inverse response. Conventional controller tuning methods proposed by most of the researchers are model dependent and require numerical computations in order to get the best possible controller parameters [4-6]. For most of the metaheuristic algorithms there is no need to provide the detailed mathematical description of the process to obtain the optimal controller tuned parameters with proper guidance of the objective function considered for a particular process. Therefore metaheuristic optimization algorithms based controller tuning has become the best choice for solving complex and intricate problems like CSTR which involves optimal control in spite of various constraints [7, 8].

The Particle Swarm Optimization algorithm (PSO) is a novel population based stochastic search algorithm introduced by Dr. Kennedy and Dr. Eberhart in 1995 [9] and its basic idea was originally inspired by simulation of the social behaviour of animals such as bird flocking, fish schooling and so on. In PSO technique, a lot of work has been done by researchers to prove its efficiency in handling complex optimization problem, especially for nonlinearity and non differentiability, multiple optimum and high dimensionality [10-12]. Inertia weight is an important parameter in PSO, which significantly affects the convergence, exploration and exploitation trade-off. Inertia weight is used to control the velocity, which is responsible for balancing of exploration and exploitation process of a swarm. Also, it determines the contribution rate of a particle's previous velocity to its velocity at the current time step.

Different Inertia Weight strategies for different optimization test problems have been done to demonstrate the importance of Inertia weight in the convergence of particle [13]. The basic PSO, presented by Eberhart and Kennedy in 1995 [9], has no Inertia weight. Shi and Eberhart extended PSO by introducing Constant Inertia weight (Constant-w) and stated that larger value of w leads to slow convergence and smaller value of w to fast convergence of particle [14]. Eberhart and Shi proposed a Random Inertia weight (Random-w) technique and inferred that this technique increases the convergence of PSO in early iterations of the algorithm [15]. In Linearly decreasing or Time Varying Inertia weight (TV-w) strategy, improved efficiency and performance of PSO have been observed for Inertia weight from 0.9 to 0.4 [16]. Linearly decreasing inertia weight PSO (LDIW-PSO) algorithm has the shortcoming of premature convergence in solving complex optimization problems. With a few

ARPN Journal of Engineering and Applied Sciences

modifications in parameters, LDIW-PSO can get better optimum fitness convergence speed, stability and robustness [17]. In Global-Local Best Inertia weight (Global-w), the weight 'w' neither assumes constant value nor a linearly decreasing time-varying value instead, it depends on local best and global best values of the particles in each iteration [18]. Chatterjee and Siarry proposed a new variant of PSO, which employs a nonlinear variation of inertia weight. This nonlinear variation has been adopted to employ aggressive, coarse tuning during initial iterations and mild, fine tuning during later iterations so that the optimum solution can be approached with better accuracy [19]. Agees Kumar and Kesavan Nair have proved that the Adaptive Inertia weight (Adaptive-w) PSO based PID controller can coordinate various performance indices of the system and provide an effective tool for trade-off analysis among convergence, stability and robustness [20].

In this work, comparative study on five inertia weight strategies in Particle Swarm Optimization algorithm has been done for all the inertia weight methods considering cost functions individually and as a weighted sum of the individual objectives for tuning of PID controller parameters to achieve global control of concentration variable in CSTR plant. Simulations have been performed using Integral Squared Error (ISE), Integral Absolute Error (IAE), Integral Time Squared Error (ITSE) and Integral Time Absolute Error (ITAE) as objective functions. The proposed Multi-Objective Adaptive Inertia weight Particle Swarm Optimization (MOAWPSO) based PID controller has been validated by analysing and comparing the performance of other Multi-Objective PSOs (MOPSO) with various inertia weight techniques. The distribution of inertia weight for different weight methods in PSO has been simulated to analyse the impact of *w* on objective functions.

The organisation of this article is as follows: Section II provides a brief description of the optimization process considered and Section III discusses the outline of PID controller tuning as well as the criteria used to evaluate the performance of CSTR process. PSO algorithm overview and its implementation in detail have been presented in section IV. In section V simulated results obtained for different Inertia weight methods are shown and finally conclusions of the present research work have been reported in section VI.

Nonlinear CSTR process description

In the CSTR process considered for this study (as shown in Figure-1), an irreversible, exothermic reaction occurs. Therefore Reactant A of concentration C_{Af} is converted to Product B of Concentration C_A in a constant volume reactor cooled by a single coolant stream. This process is complex with a wide operating range. The first principles model of the CSTR and the operating point data as specified by Pottmann and Seborg [21] has been used in the simulation studies (Table-1).

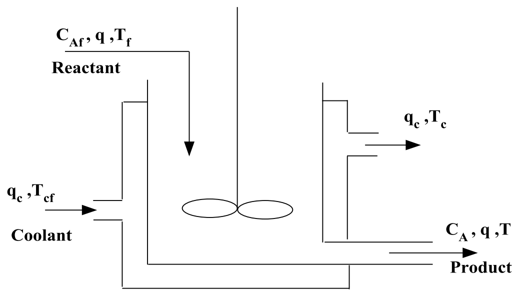


Figure-1. Schematic of CSTR process.

Table-1. Steady state operating data of CSTR process.

Process variable	Normal operating condition
Measured product concentration(C_A)	0.0989 mol/lit
Reactor temperature (T)	438.7763 K
Coolant flow rate (q_c)	103 lit/min
Process flow rate (q)	100.0 lit/min
Feed concentration (C_{Af})	1 mol/lit
Feed temperature (T_f)	350.0 K
Inlet coolant temperature (T_{cf})	350.0 K
CSTR volume (V)	100 lit
Heat transfer term (hA)	$7*10^5$ cal/(min.k)
Reaction rate constant (k_0)	$7.2*10^{10}min^{-1}$
Activation energy term (E/R)	$1*10^4$ K
Heat of reaction (-ΔH)	$2*10^5$ cal/mol
Liquid density (ρ, ρ_c)	$1*10^3$ g/lit
Specific heats (C_p, C_{pc})	1 cal/(g.k)
Fouling coefficient $\varphi_h(t)$	1
Deactivation coefficient $\varphi_c(t)$	1

The first principles model of CSTR process is described by a set of equations. The mass balance on component A is formulated as

$$\frac{dC_A}{dt} = \frac{q}{V}\left(C_{Af} - C_A\right) - k_0 C_A \exp\left(-\frac{E}{RT}\right)\varphi_c(t) \qquad (1)$$

Assuming constant volume, heat capacity C_p, density ρ and neglecting changes in potential and kinetic energy, the reactor energy balance is defined in equation (2).

VOL. 10, NO 22, DECEMBER, 2015 ISSN 1819-6608
ARPN Journal of Engineering and Applied Sciences

$$\frac{dT}{dt} = \frac{q}{V}\left(T_f - T\right) + \frac{(-\Delta H)k_0 C_A}{\rho C_p}\exp\left(-\frac{E}{RT}\right)\varphi_c(t) + \frac{\rho_c C_{pc}}{\rho C_p}\frac{q_c}{V}\left(1-\exp\left(-\frac{hA}{q_c\rho C_p}\varphi_h(t)\right)\right)\left(T_{cf} - T\right) \quad (2)$$

MATERIALS AND METHODS

Design of PID control system using different PSO variants

Tuning of PID controller means adjusting the controller gains to achieve the best possible control for a particular process to satisfy the performance specifications like margin of stability, transient response and bandwidth. Although trial and error can be used, analytical approach is used to compute the gain that can minimize the performance index.

The following performance indices given in equation (3), (4), (5), (6) are considered as objective functions in this work.

$$IAE = \int_0^\infty |e(t)|dt \quad (3)$$

$$ISE = \int_0^\infty [e(t)]^2 dt \quad (4)$$

$$ITAE = \int_0^\infty t|e(t)|dt \quad (5)$$

$$ITSE = \int_0^\infty t[e(t)]^2 dt \quad (6)$$

The transfer function of PID controller is described by equation (7) in the continuous domain with K_p, K_i and K_d as the proportional, integral and derivative gains respectively.

$$G_{PID}(s) = K_p + \frac{K_i}{s} + K_d s \quad (7)$$

The output of the PID controller in time domain is given by

$$u(t) = K_p e(t) + K_i \int_0^t e(t) + K_d \frac{d}{dt} e(t) \quad (8)$$

where, u(t) and e(t) are the control and tracking error signals in time domain respectively. Figure-2 illustrates the implementation of PSO algorithm based PID controller tuning in CSTR process.

Generally PID parameters designed for a particular operating point are not suitable to control any process globally. Therefore, for multimodal problem, global control can be achieved by utilizing population based algorithm [22]. In this work, PSO based optimal PID controller with various Inertia weight methods has been used for global control of CSTR process.

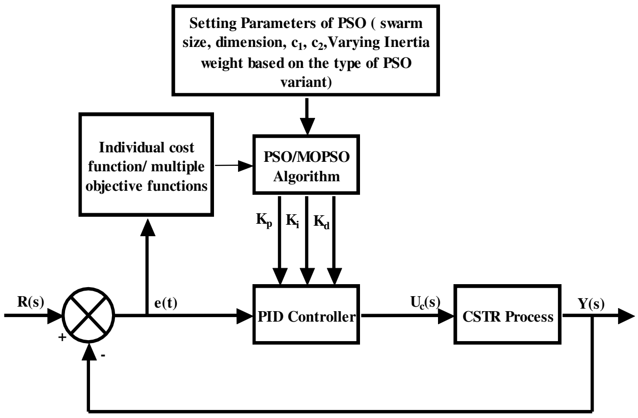


Figure-2. Block diagram of PSO based PID controller tuning.

PSO algorithm overview

The particle swarm optimization algorithm is a population based search algorithm based on the simulation of the social behaviour of birds within a flock. A swarm consists of a set of particles, where each particle represents a potential solution. Particles are then flown through the hyperspace, where the position of each particle is changed according to its own experience and that of its neighbours. An individual particle *i* is composed of three vectors: its position in the *D*-dimensional search space $x_i = (x_{i1}, x_{i2,}............., x_{id})$, best position that it has individually found $p_i = (p_{i1}, p_{i2,}......p_{id})$ and its velocity $v_i = (v_{i1}, v_{i2},.........., v_{id})$.

Particles are initialized in a uniform random manner throughout the search space and velocity is also randomly initialized. These particles then move throughout the search space and updates itself at each time step by updating the velocity and position of each particle in every dimension by following a set of update equations as given in equation (9), (10).

$$v_{id} = v_{id} + c_1 r_1 (p_{id} - x_{id}) + c_2 r_2 (p_{gd} - x_{id}) \quad (9)$$

$$x_{id} = x_{id} + v_{id} \quad (10)$$

where, d = 1, 2… n represents the dimension and i = 1, 2,…, S represents the particle index. *w* is the inertia weight, c_1 and c_2 are cognitive and social acceleration coefficients respectively, and r_1 and r_2 are random numbers obtained from a uniform random distribution function in the interval 0 to 1. The parameters r_1 and r_2 are used to maintain the diversity of the population. p_{id} and p_{gd} represent the best previous position of the i^{th} particle and the position of the best particle among all particles in the population respectively. Equations (9) and (10) define the standard PSO algorithm. Later, the concept of an Inertia weight was developed by Shi and Eberhart [14] in 1998 to provide a balance between exploration and exploitation, eliminating the need for setting of maximum velocity. The resulting velocity update equation becomes

$$v_{id} = w * v_{id} + c_1 r_1 (p_{id} - x_{id}) + c_2 r_2 (p_{gd} - x_{id}) \quad (11)$$

The parameter *w,* helps the particles converge to p_{gd}, rather than oscillating around it. The inertia weight controls the influence of previous velocities on the new velocity. Large inertia weights cause larger exploration of the search space while smaller inertia weights, focus the search on a smaller region. The choice of the PSO parameters especially *w* seems to be very important for the speed and efficiency of the algorithm. Inertia weight plays an important role in the convergence of the optimal solution to a better optimal value as well as the execution time of the simulation run.

Features of PSO

- PSO implementation is easier since the numbers of parameters utilized are very few compared to other nature-inspired algorithms.
- Being a population-based search algorithm, it is less susceptible of getting trapped in local optima.
- PSO makes use of the probabilistic transition rules that can search a complicated and uncertain area, enabling more flexibility and robustness than conventional methods.
- PSO can easily deal with non-differentiable objective functions, this property relieves PSO of assumptions and approximations, which are often required by traditional optimization models.
- PSO has the ability to control the balance between the global and local exploration of the search space.
- PSO only has one operator, velocity calculation, so the computation time is decreased significantly.

The following inertia weight methods of PSO given in Table-2 have been used for simulation, to find optimum PID controller parameters for the considered nonlinear system and the Inertia weight algorithm of PSO is illustrated in Figure-3.

Table-2. Formula for different inertia weight techniques.

Different inertia weight techniques	Formula used
Constant inertia weight (Constant-w)	$w = 0.7$, [14]
Random inertia weight (Random-w)	$w = 0.5 + \frac{rand()}{2}$, [15]
Global-Local best inertia weight (Global-w)	$w_i = (1.1 - \frac{gbest_i}{pbest_i})$, [16]
Time varying inertia weight (TV-w)	$w = (w_{max} - w_{min})(\frac{iter_{max} - iter}{iter_{max}})$ $+ w_{max}$,[17]
Adaptive inertia weight (Adaptive-w)	$w = w_o + r(1 - w_o)$, [18]

Begin Algorithm
Input: function to optimize(f)
particle size(ps)
problem dimension(d)
search space range, [min X, max X]
velocity range, [min V, max V]
Output:
x^* : the best value found
Initialize: for all particles in problem space
$x_i = (x_{i1}, x_{i2}, \ldots\ldots\ldots\ldots, x_{id})$
$v_i = (v_{i1}, v_{i2}, \ldots\ldots\ldots\ldots, v_{id})$
Evaluate (x_i), get p_{id},
($i = 1, \ldots, ps$)
p_{gd} ←best of p_{id}
Repeat
Calculate ω
Update V_i for all particles using (9)
Update x_i for all particles using (11)
Evaluate (x_i) in d variables and get p_{id}, ($i = 1, \ldots, ps$)
If (x_i) is better than p_{id}, then $p_{id} \leftarrow x_i$
If the best of p_{id} is better than p_{gd} then
p_{gd} ←best of p_{id}
Until stopping criteria (e.g., maximum iteration or error tolerance is met)
$x^* \leftarrow p_{gd}$
Return x^*
End Algorithm

Figure-3. Inertia weight PSO algorithm.

VOL. 10, NO 22, DECEMBER, 2015 ISSN 1819-6608
ARPN Journal of Engineering and Applied Sciences

Formulation of MOAWPSO

A linearly-decreasing weight would not be adequate to improve the performance of the PSO due to its nonlinear nature as suggested by Shi and Eberhart [23]. Therefore the following formula to change the inertia weight at each generation as proposed [18] has been utilised in this paper.

$$w = w_o + r(1 - w_o) \tag{12}$$

where w_0 is the initial positive constant in the interval [0, 1] and r is random number in the interval [0, 1]. The suggested range for w_0 is [0, 0.5], which make the weight w randomly varying between w_0 and 1. Mahfouf *et al.* proposed [24] an Adaptive Inertia weight PSO (AWPSO) algorithm to improve the performance of the PSO for multi-objective optimization problems in which the velocity in Equation (11) is modified with an acceleration factor α as in equation (13)

$$\begin{aligned} v_{id} = w * v_{id} + \alpha[c_1 r_1 (p_{id} - x_{id}) + \\ c_2 r_2 (p_{gd} - x_{id})] \end{aligned} \tag{13}$$

The acceleration factor α is defined as follows:

$$\alpha = \alpha_o + \frac{iter}{iter_{max}} \tag{14}$$

where, iter is the current generation, $iter_{max}$ denotes the total number of generations and suggested range for α_o is [0.5, 1].

In Equation (13), the acceleration term will increase with the number of iterations, enhancing the global search ability at the end of a run and prevents local minima. The simplest approach to deal with Multi-Objective Problems (MOPs) is to define an aggregate objective function f_i as a weighted sum of the individual objectives.

Multi-objective Adaptive Inertia weight PSO is formulated as a weighted sum of individual objective functions like ISE, IAE and ITSE with adaptive weight measure as a PSO variant. The aggregate objective function f_{agr} (k) has been coined as reported [25] and given in equation (15).

$$f_{agr}(k) = \sum_{i=1}^{n} w_i f_i(k) \tag{15}$$

where, $\sum_{i=1}^{n} w_i = 1$, n is the number of objective functions and k denotes k^{th} particle and weight of each objective function normalized as in equation (16).

$$w_i = \frac{\mu_i}{\sum_{j=1}^{n} \mu_j} \tag{16}$$

where, $\mu_i, \mu_j \in$ U (0, 1), μ_i and μ_j are random numbers obtained from a uniform random distribution function in the interval [0,1]

RESULTS AND DISCUSSIONS

The results have been analysed by implementing five different Inertia weight methods in PSO. The goal of PSO is to tune PID parameters for the global range of CSTR process with minimum error criteria like ISE, ITSE, ITAE and IAE. PSO based optimum PID controller parameters and performance analysis under various inertia weight methods for the four error criteria have been tabulated in Table-3, 4, 5 and 6 respectively. The CSTR process has been simulated using the nonlinear first principles model given by (1) and (2) and the process output concentration has been computed by solving the nonlinear differential equation using Matlab 7.0. The controller saturation limit between 97 and 109 lit/min is considered with initial conditions given by q_c = 103 lit/min, C_A = 0.0989 mol/lit and T = 438.77 K and the sampling time of about 0.083s is selected for all the simulation studies.With nominal and shifted operating point as set points, simulation studies have been carried out to demonstrate the set point tracking capability of the CSTR process as shown in Figure-4, 5, 6 and 7. The parameters assigned for various PSO variants have been reported in Table-7.

Table-3. PSO based optimum parameters under different weight variants using ITSE as cost function.

Inertia weight techniques	Error criteria	Optimum parameters of PID controller		
	ITSE	K_p	K_i	K_d
Constant-w	6.958	23.1348	1.8789	4.7422
Global-w	4.604	50.3973	2.9916	1.2689
Random-w	2.664	16.6111	0.5226	0.5969
TV-w	1.576	36.8646	0.7087	1.4457
Adaptive-w	1.286	14.3972	0.2483	1.8896

VOL. 10, NO 22, DECEMBER, 2015 ISSN 1819-6608
ARPN Journal of Engineering and Applied Sciences

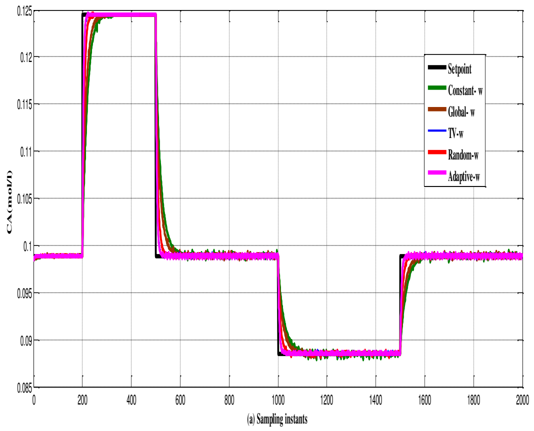


Figure-4. Output of CSTR process for various PSO variant with ITSE as cost function.

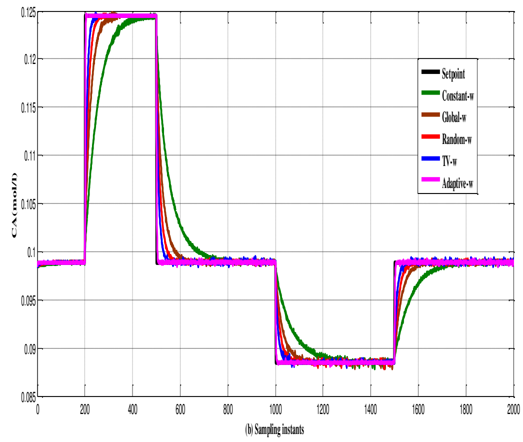


Figure-5. Output of CSTR process for various PSO variant with ISE as cost function.

Table-4. PSO based optimum parameters under different weight variants using ISE as cost function.

Inertia weight techniques	Error Criteria	Optimum parameters of PID controller		
	ISE	K_p	K_i	K_d
Constant-w	0.0374	16.6751	3.6637	1.1515
Global-w	0.0154	31.0409	2.9555	0.5880
Random-w	0.0098	21.8642	1.2784	0.6393
TV-w	0.0054	47.4244	1.6378	0.8773
Adaptive- w	0.0010	57.3681	0.2989	0.7063

Table-5. PSO based optimum parameters under different weight variants using IAE as cost function.

Inertia weight techniques	Error criteria	Optimum parameters of PID controller		
	IAE	K_p	K_i	K_d
Constant-w	4.030	17.8812	4.0203	0.8529
Global-w	3.161	22.1548	3.9013	0.4834
Random-w	0.812	35.4049	1.5903	1.3666
TV-w	0.460	19.4559	0.4901	1.3756
Adaptive- w	0.265	42.9483	0.6172	1.1041

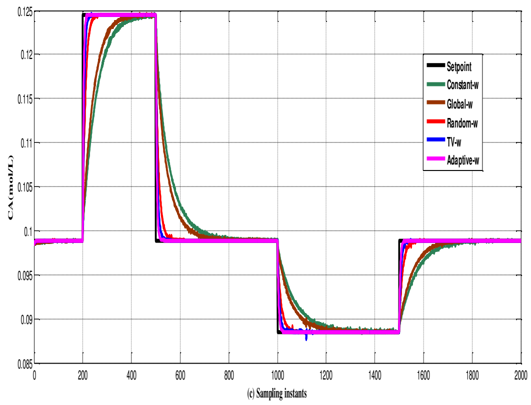


Figure-6. Output of CSTR process for various PSO variant with IAE as cost function.

ISSN 1819-6608
ARPN Journal of Engineering and Applied Sciences

www.arpnjournals.com

Table-6. PSO based optimum parameters under different weight variants using ITAE as cost function.

Inertia weight techniques	Error criteria	Optimum parameters of PID controller		
	ITAE	K_p	K_i	K_d
Constant-w	1290	23.954	2.5231	1.8540
Global-w	986.1	28.3165	2.2942	1.0423
Random-w	445.9	14.0639	0.5205	1.2737
TV-w	236.3	39.4796	0.7802	0.2347
Adaptive- w	50.67	41.9933	0.1680	0.6392

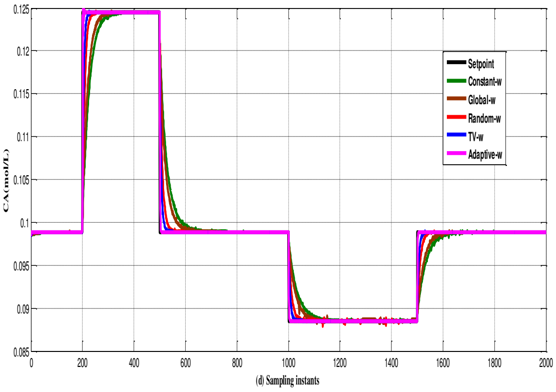
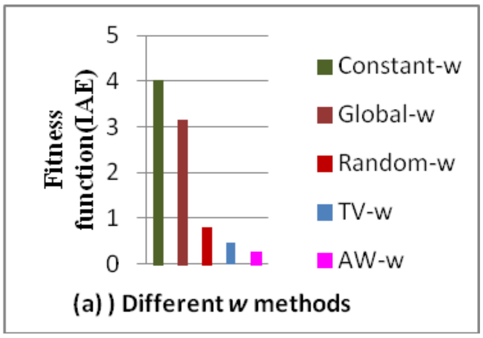
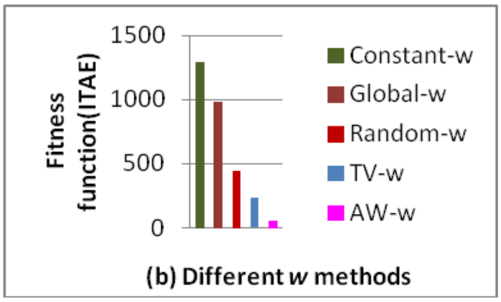


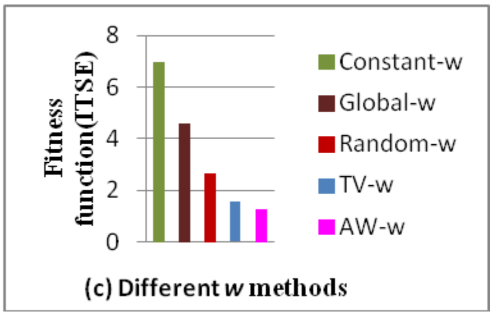
Figure-7. Output of CSTR process for various PSO variant with ITAE as cost function.

Table-7. Parameters assigned for various PSO variants.

Parameters assigned	Constant-w	Global-w	Random-w	TV-w	Adaptive-w
Dimension: Swarm size: C_1, C_2: r_1, r_2:	Dimension = 3 Swarm size = 20 $C_1 = C_2 = 2$ r_1, r_2: are random numbers obtained from a uniform random distribution function in the interval [0, 1].				







VOL. 10, NO 22, DECEMBER, 2015 ISSN 1819-6608
ARPN Journal of Engineering and Applied Sciences

www.arpnjournals.com

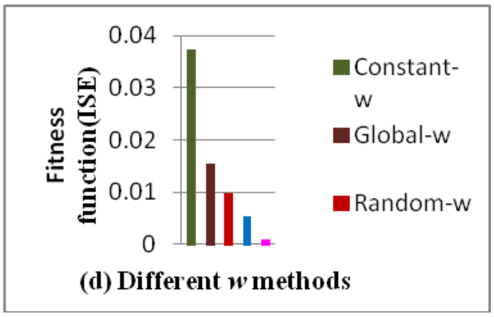


Figure-8 (a), (b), (c), (d). Effect of inertia weight on individual Fitness functions

From Figures-4, 5, 6 and 7, with optimum tuned parameters K_p, K_i and K_d, the PID controller are able to track the set point variations close to the desired set point (concentration of the reactor) considering the cost functions ISE, ITSE, ITAE and IAE individually. Figure-8 shows that the performance of AWPSO method has been found to be the best of all other inertia weight methods in terms of error minimization on almost all the fitness functions.

MOPSO have been considered with an aggregate objective function as a weighted sum of three individual cost functions namely ISE, ITSE and IAE. The weights assigned for the three cost functions are randomly distributed and uniformly followed for all the inertia weight methods. Table-8 shows optimum PID parameters under different weight variants of PSO and respective cost function values. The cost functions treated in MOPSO is weighted as 0.4232, 0.2971 and 0.2797 for ISE, ITSE and IAE respectively. Table-8 and Figure-9 shows AWPSO performs better with minimum ISE, ITSE and IAE on a multi-objective case also, where individual cost functions are made as a weighted sum.

Table-8. Optimum PID parameters of MOPSO under different inertia weight methods.

MOPSO for w techniques	Error criteria			Optimum parameters of PID controller		
	ISE	ITSE	IAE	K_p	K_i	K_d
Constant-w	0.008	3.970	0.795	10.75	0.47	0.94
Global-w	0.004	2.041	0.437	45.30	1.08	1.90
Random-w	0.002	0.895	0.162	32.17	0.27	2.43
TV-w	0.004	1.736	0.381	44.18	0.92	0.60
Adaptive-w	0.001	0.442	0.085	48.27	0.15	0.17

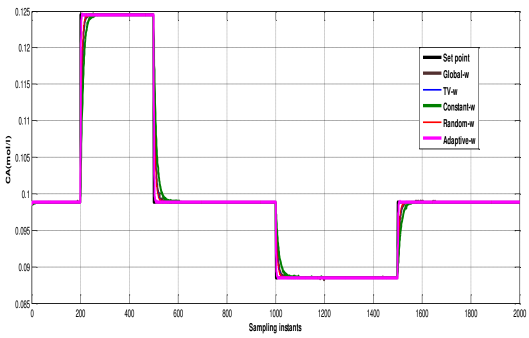


Figure-9. Output of CSTR process for various PSO variant considering cost functions as weighted sum.

The impact of various inertia weight methods in PSO:
To depict the importance of the parameter, inertia weight factor *w*, five samples of swarm size have been plotted for all considered weight variants in Figure-10.

- **Observation on constant inertia weight method**
  In constant inertia weight method, it has been observed that a large inertia weight facilitates a global search and takes much time to converge and a small inertia weight facilitates a local search with minimum convergence time. Also the distribution of inertia weight is constant throughout the iteration process as shown in Figure-10.

- **Observation on Global-Local Best inertia weight**
  In this PSO variant, the inertia weight depends on p_{gd} and p_{id} of the particles in each generation. From Figure-10.it is clear that it neither takes a constant value nor a linearly decreasing time-varying value which may lead to premature convergence to a local minimum.

- **Observation on random inertia weight**
  It assumes random *w* for each iteration. Simulation result from Figure-9 reveals that this strategy increases the convergence of PSO in early iterations of the algorithm.

- **Observation on TVIW**
  This method has improved the efficiency and performance of PSO. It has been found that inertia

weight from 0.9 to 0.4 provides appreciable result and the distribution of the inertia weight in a population size of five has been presented in Figure-10. Even though this strategy has the ability to converge to optimum value as shown in Figure-4, 5, 6 and 7, sometimes there is a possibility of getting trapped into the local optimum.

- **Observation on AWPSO**
In Figure-10, it has been observed that the inertia weight distribution does not linearly decrease with time and hence AWPSO has the capability to avoid from local optima. Adaptive Inertia Weight strategy has been proposed to prevent premature convergence to local minimum and to improve its searching capability. Population diversity is controlled by adaptive adjustment of the inertia weight throughout the iteration process.

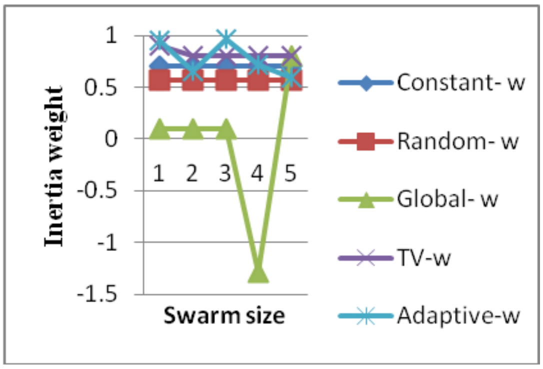


Figure-10. Distribution of inertia weight considering various PSO weight variants.

CONCLUSIONS

This research work presents a comparative study on five inertia weight strategies in Particle Swarm Optimization algorithm. Simulations have been done for all the inertia weight methods in PSO considering cost functions individually and as a weighted sum of the individual objectives. Simulation results prove that AWPSO algorithm implemented for optimal control of CSTR process converges faster with minimum error compared to other inertia weight measures under single and multiobjective targets. The better performance of AWPSO is due to its adaptiveness with respect to its inertia weight and acceleration coefficients enabling a good balance between the exploration and exploitation of the search space. Also adaptive adjustment of inertia weight prevents premature convergence from local minima rooting to good results.

REFERENCES

[1] K.J Åström and T. Hägglund. 1995. PID controllers: theory, design and tuning. Research Triangle Park (NC), Instrument Society of America.

[2] P. Cominos and N. Munro. 2002. PID controllers: recent tuning methods and design to specification. IEEE Proc Control Theory Application. 149(1): 46-53.

[3] Coelho LS and Coelho AAR. 1999. Automatic tuning of PID and gain scheduling PID controllers by a derandomized evolution strategy. Artificial Intelligence Eng Des Anal Manuf. (5): 341–349.

[4] G. Ziegler and N. B. Nichols. 1942. Optimum settings for automatic controllers, Trans. ASME. 64: 759-768.

[5] G.H Cohen and G.A Coon. 1953. Theoretical Consideration of Retarded Control, Trans ASME. 75: 827/834.

[6] K. J. Astrom and T. Hagglund. 2001. The future of PID control, Control Eng. Pract. 9(11): 1163-1175.

[7] Yogesh Chaudhari *et al.* 2013. Design and Implementation of Intelligent Controller for a Continuous Stirred Tank Reactor System Using Genetic Algorithm. International Journal of Advances in Engineering and Technology. 6(1): 325-335.

[8] A.Jayachitra and R. Vinodha. 2015 Global Optimization of PID Parameters for a Nonlinear System Using Firefly Algorithm. International Journal of Applied Engineering Research. 10: 21833-21852.

[9] Kennedy J, Eberhart R. 1995. Particle swarm optimization. Proceedings of IEEE International conference neural networks, Perth, Australia. 4: 1942-48.

[10] Wei-Der Chang and Shun-Peng Shih. 2010. PID controller design of nonlinear systems using an improved particle swarm optimization approach. Commun Nonlinear Sci Numer Simulat. 15: 3632-3639.

[11] V. Rajinikanth and K. Latha. 2011. Optimization of PID controller parameters for unstable chemical systems using soft computing technique. International Review of Chemical Engineering. 3(3): 350-358.

[12] S. M. GirirajKumar, Deepak Jayaraj and Anoop. R. Kishan. 2010. PSO based Tuning of a PID Controller for a High Performance Drilling Machine. International Journal of Computer Applications. 1(19): 0975-8887.

[13] J. C. Bansal, P. K. Singh, Mukesh Saraswat, Abhishek Verma, Shimpi Singh Jadon and Ajith Abraham. 2011. Inertia Weight Strategies in Particle Swarm Optimization. Third World Congress on Nature and Biologically Inspired Computing. IEEE. pp. 640–647.

[14] Y. Shi and R. Eberhart. 2002. A modified particle swarm optimizer. In Evolutionary Computation Proceedings, 1998. IEEE World Congress on Computational Intelligence, the 1998 IEEE International Conference. pp. 69-73.

[15] R. C. Eberhart and Y. Shi. IEEE, 2002. Tracking and optimizing dynamic systems with particle swarms", In Evolutionary Computation, 2001. Proceedings of the 2001 Congress on. 1: 94-100.

[16] J. Xin, G. Chen and Y. Hai. IEEE, 2009. A Particle Swarm Optimizer with Multistage Linearly-Decreasing Inertia Weight. In: Computational Sciences and Optimization. International Joint Conference on. 1: 505-508.

[17] Martins Akugbe Arasomwan and Aderemi Oluyinka Adewumi. 2013. On the Performance of Linear Decreasing Inertia Weight Particle Swarm Optimization for Global Optimization. The Scientific World Journal. pp. 1-12. http://dx.doi.org/10.1155/2013/860289.

[18] M. S. Arumugam and MVC Rao. 2006. On the performance of the particle swarm optimization algorithm with various Inertia Weight variants for computing optimal control of a class of hybrid systems. Discrete Dynamics in Nature and Society. pp. 1-7. Doi.10.1155/DDNS2006/79295.

[19] A. Chatterjee and P. Siarry. 2006. Nonlinear inertia weight variation for dynamic adaptation in particle swarm optimization. Computers and Operations Research. 33: 859-871.

[20] C. Agees Kumar and N. Kesavan Nair. 2010. Multi-Objective PID Controller based on Adaptive Weighted PSO with application to Steam Temperature control in Boilers. International Journal of Engineering Science and Technology. 2(7): 3179-3184.

[21] M. Pottman and D. E. Seborg. 1992. Identification of non-linear processes using reciprocal multiquadric functions. Journal of Process Control. 2. 4: 189-203.

[22] A. Jayachitra and R. Vinodha. 2014. Genetic Algorithm Based PID Controller Tuning approach for Continuous Stirred Tank Reactor. Advances in Artificial Intelligence. pp. 1-8. Doi.org/10.1155/2014/791230.

[23] Y. Shi and R. Eberhart. 2001. Fuzzy Adaptive Particle Swarm Optimization. Congress on Evolutionary Computation, Seoul, Korea. pp. 101-106.

[24] M. Mahfouf, Minyou-Chen and D. A. Linkens. 2004. Adaptive Weighted Particle Swarm Optimization (AWPSO) of Mechanical Properties of Alloy Steels. 8th International Conference on Parallel Problem Solving from Nature (PPSN VIII), Birmingham, U.K.

[25] Zhang Xing-hua and Zhu Xiao-rong. 2006. Multi-objective optimization of PID regulators based on adaptive genetic Algorithms. Systems Engineering and Electronics. 28(5): 744-747.