Optimal Fuzzy Guidance Law Design for Missile against Maneuvering Targets

Labeed Hassan, Seyed Hossein Sadati, Mohamad Ali Ashtiani, and Mohamad Bagher Malaeak

Abstract: In this paper an Optimal Fuzzy Guidance (OFG) law for surface to air missile against maneuvering Targets is introduced. Proportional Navigation Guidance (PNG) law is used to establish the rules of the introduced approach. The OFG attempts to keep miss-distance and control effort as minimum as possible. Based on Time Variant Particle Swarm Optimization (TVPSO), the Membership Functions (MFs) of the proposed design are optimized. To show the relative superiority of the approach, the performance of the new guidance law has been compared with that of PNG. The results confirm the validity of the introduced design and show that OFG performs better than PNG on variety of scenarios; some of which are discussed in the paper.

Index Terms: Fuzzy Logic Controller, Particle Swarm Optimization, Proportional Navigation Guidance

1. INTRODUCTION

GUIDANCE laws are mainly based on classical control techniques such as, sliding mode control [1], adaptive control [2], and linear quadratic based control [3]. The so called "conventional" control approaches, although efficient in most cases, might not be effective for tracking and interception of maneuvering targets. Fuzzy logic controllers (FLCs) have suitable properties that help diminish such difficulties. Most fuzzy guidance laws are, in fact, the fuzzy-logic implementation of existing well-known classical guidance laws; such as PNG, that is because of its simplicity, effectiveness and ease of implementation [4]. FLCs are developed to utilize human expert knowledge in controlling various systems. It is well known that while fuzzy rules are relatively easy to derive from human experts, the fuzzy MFs are difficult to adjust. Tuning of MFs is a time consuming and often frustrating exercise. To overcome these difficulties various techniques have been reported to automate the tuning process of MFs. An adaptive network based fuzzy inference system was introduced [5] and a quantum neural fuzzy network was used to learn the data space of a Tagaki-Sugeno fuzzy controller [6]. In addition; Genetic algorithm has been used in the automatic design of fuzzy controllers in the areas of mobile robotics [7]. In the last decades, Particle Swarm Optimization (PSO) is verified to consider a good technique for tuning because of its simplicity, ease of implementation, and its ability to tackle tough cost functions with many local minima [8]. PSO is a population based stochastic optimization technique developed by Clerc and Kennedy [9]. Particle swarm algorithm imitates human (or insects) social behavior. Individuals interact with one another while learning from their own experience, and gradually the population members move into better regions of the problem space. The swarm of PSO can be envisioned as multiple birds (particles) that search for the best food source (optimum) by using their inertia, their knowledge, and the knowledge of the swarm. Single particles behave similarly because they share the same configuration. While searching for food, the birds are either scattered or go together before they locate the place where they can find the food. While the birds are searching for food from one place to another, there is always a bird that can smell the food very well, that is, the bird is perceptible of the place where the food can be found, having the better food resource information. Because they are transmitting the information, the birds will eventually flock to the place where food can be found.

Time Variant Particle Swarm Optimization TVPSO [10] is used in this paper to optimize the MFs of the proposed design. Through the optimization process, an objective function which includes the terms; miss distance M_D, and control effort C_{EFF} is minimized.

The work is organized as follows: in Section 2 a construction of the proposed Fuzzy Logic Controller is introduced. Then PSO and TVPSO are explained in Section 3. The MFs are optimized in section 4, whereas the results are provided in Section 5. Conclusion is included in section 6.

- *Ph.D. Candidate, Dept of Aerospace Eng, Maleke Ashtar Univ. of Tech., Islamic Republic of Iran. Email: lab.has77@yahoo.com*
- *Assistant Professor, Space Research Institute, Islamic Republic of Iran. Email: hsadati@aut.ac.com*
- *Assistant Professor, Dept of Aerospace Eng, Maleke Ashtar Univ. of Tech., Islamic Republic of Iran. Email: Ma_shahi@yahoo.com*
- *Professor, Dept of Aerospace Eng, Sharif Univ. of Tech., Islamic Republic of Iran. Email: molaek@yahoo.com*

2. CONSTRUCTION OF THE PROPOSED FLC:

Fig. 1 illustrates the construction of the FLC which is composed of five functional blocks.

The rule base contains a number of if-then rules, a database that defines the MFs, a decision making interface which operates the given rules, a fuzzification interface that converts the crisp inputs into "degree of match" with the linguistic values like small or large etc., and a defuzzification interface which reconverts to a crisp output.

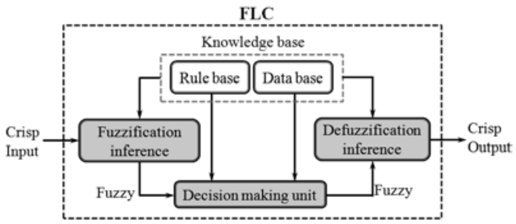


Fig. 1: The structure of FLC.

Minimum Mamdani (AND) method, the most popular inference engine, is used to obtain the best possible conclusion, this type of inference engine allows easy and effective computation and it is appropriate for the real time control application [11]. Furthermore, the Center of Aria (CoA) method, which supplies defuzzified output with better continuity and affectivity [12], is chosen for defuzzification.
To establish the rules of the proposed OFG, the conception of PNG law is used, in which the acceleration command A_C is mathematically expressed as

$$A_C = N.V_C.\dot{\lambda} \quad (1)$$

Where; N is the navigation ratio, V_C is the closing velocity and $\dot{\lambda}$ is the (LOS) angle rate. Similarly to PNG controller the OFG controller uses $V_C, \dot{\lambda}$ as inputs, and A_C as output. The inputs and the output are quantized and normalized within [-1, 1]. The (input/output) data are normalized according to max method normalization [13]. This method divides the performance ratings of each attribute by its maximum performance rating. In the current design the maximum values are obtained based on the knowledge available about missiles dynamic in addition to previous experiences about other classical guidance laws in the PNG's class. The maximum values are shown in Table 1:

Table 1: Maximum values for normalization.

Closing Velocity (V_C)	LOS Angle Rate ($\dot{\lambda}$)	Acceleration Command (A_C)
1300 [m/sec]	0.05 [rad/sec]	200 [m/sec²]

2.1. Number and shape of the MFs:

In this work; three groups of MFs with triangular shape are investigated for the controller. Each variable V_C, $\dot{\lambda}$ and A_C has its own group. In turn, each group has seven MFs, and each MF is described by a linguistic value. The linguistic values can be represented as :{ LN, MN, SN, ZE, SP, MP, LP}, where "L", "M", and "S" represent "Large", "Medium", and "Small" respectively. Similarly; "N", "ZE", and "P" denote "Negative", "Zero", and "Positive" respectively.

Each triangular MF is determined by three parameters such as, (**a**, **b** and **c**). The parameter (**a**) locates the left foot of the MF whiles (**c**) locates the right foot, and (**b**) locates the peak as plotted in Fig. 2.

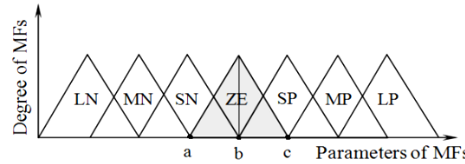


Fig. 2: A typical set of MFs.

The triangular MF has a form declared as:

$$f(x,\mathrm{a},\mathrm{b},\mathrm{c}) = \begin{cases} 0, & x \le \mathrm{a} \\ \frac{x-\mathrm{a}}{\mathrm{b}-\mathrm{a}} & \mathrm{a} \le x \le \mathrm{b} \\ \frac{\mathrm{c}-x}{\mathrm{c}-\mathrm{b}} & \mathrm{b} \le x \le \mathrm{c} \\ 0, & \mathrm{c} \le x \end{cases} \quad (2)$$

The parameters (**a**, **b** and **c**) have to satisfy ($\mathrm{a} < \mathrm{b} < c$). This condition has to be considered throughout the optimizing process.

2.2. Searching for proper rules:

Based on the concept of PNG law; A_C is proportional to multiplication of the two variables $\dot{\lambda}$ and V_C. So that; it is trivial that the sign of A_C will be negative "N" if one of ($\dot{\lambda}$ or V_C)'s signs were negative, otherwise it will be positive "P", therefore; the sign of A_C could simply define as following:

Table 2: Defining the sign of A_C.

if	V_C	is	N	and	$\dot{\lambda}$	is	P	then	A_C	is	N
if	V_C	is	N	and	$\dot{\lambda}$	is	N	then	A_C	is	P
if	V_C	is	P	and	$\dot{\lambda}$	is	P	then	A_C	is	P
if	V_C	is	P	and	$\dot{\lambda}$	is	N	then	A_C	is	N

The data are normalized within the interval [-1, 1] before feeding to the input of the controller.
It is trivial that; a multiplication of two values (val_1,val_2) in the interval [-1, 1] results a value (val) that is smaller than the smallest of them. In addition, (val) will be Zero "ZE" if any of them were Zero. Adopting this concept the linguistic values of A_C can be defined as described in Table 3:

ISSN 2229-5518

Table 3: Defining the values of A_C.

if	V_c	is	L	and	λ	is	L	then	A_c	is	L
if	V_c	is	L	and	λ	is	M	then	A_c	is	M
if	V_c	is	L	and	λ	is	S	then	A_c	is	S
if	V_c	is	L	and	λ	is	ZE	then	A_c	is	ZE
if	V_c	is	M	and	λ	is	L	then	A_c	is	M
if	V_c	is	M	and	λ	is	M	then	A_c	is	M
if	V_c	is	M	and	λ	is	S	then	A_c	is	S
if	V_c	is	M	and	λ	is	ZE	then	A_c	is	ZE
if	V_c	is	S	and	λ	is	L	then	A_c	is	S
if	V_c	is	S	and	λ	is	M	then	A_c	is	S
if	V_c	is	S	and	λ	is	S	then	A_c	is	S
if	V_c	is	S	and	λ	is	ZE	then	A_c	is	ZE
if	V_c	is	ZE	and	λ	is	L	then	A_c	is	ZE
if	V_c	is	ZE	and	λ	is	M	then	A_c	is	ZE
if	V_c	is	ZE	and	λ	is	S	then	A_c	is	ZE
if	V_c	is	ZE	and	λ	is	ZE	then	A_c	is	ZE

Taken into account the two previous conceptions, (Table 2, Table 3), the entire rules can be obtained as follows:

Table 4: Rules of the OFG.

A_c		λ						
		LP	**MP**	**SP**	**ZE**	**SN**	**MN**	**LN**
V_c	**LP**	LP	MP	SP	ZE	SN	MN	LN
	MP	MP	MP	SP	ZE	SN	MN	MN
	SP	SP	SP	SP	ZE	SN	SN	SN
	ZE	ZE	ZE	ZE	ZE	ZE	ZE	ZE
	SN	SN	SN	SN	ZE	SP	SP	SP
	MN	MN	MN	SN	ZE	SP	MP	MP
	LN	LN	MN	SN	ZE	SP	MP	LP

3. PSO AND TVPSO:

The PSO is a population based stochastic optimization technique consists of a swarm of particles flying through the search space. Every individual in the swarm contains parameters for position and velocity. The position of each particle represents a potential solution to the optimisation problem. The dynamic of the swarm is governed by a set of rules that modify the velocity of each particle according to the experience of the particle itself and that of its neighbors depending on the social network structure within the swarm. By adding a velocity to the current position, the position of each particle is modified.

As the particles move around the space, different fitness values are given to the particles at different locations according to how the current positions of particles satisfy the objective. In a single iteration, each particle tracks its personal best position. Depending on the social network structure of the swarm, the global best position, and/or the local best position, is used to influence the swarm dynamic. After a number of iterations, the particles will eventually cluster around the area where fittest solutions are.

The swarm behavior is influenced by; the number of particles (N), the neighbourhood population (P), the inertia weight (w), the maximum velocity (v_{max}), and the acceleration calculation ($c.r$) that modifies the velocity. The larger the number of particles in the swarm, the more likely the swarm will converge on the global optimum, because the social information exchange is increased. The influence of the current velocity on the new velocity can be controlled by the inertia weight. A large inertia weight compels large exploration through the search space; a smaller inertia weight causes reduced exploration. The influence of the particle's experience and that of its neighbor is governed by the acceleration calculation.

The further away the particle is from the best position from its own experience and its neighbor, the larger a change in velocity that is made in order to return to that best position. The acceleration limits the trajectory of the particle oscillation. The smaller the acceleration, the smoother the trajectory of the particle is. However, too small an acceleration may lead to slow convergence, whereas too large an acceleration drives the particles towards infinity. The new velocity is limited by the given maximum velocity to prevent particles from moving too fast in the space.

In particular, the velocity associated with each particle in PSO is calculated as the following [14]:

$$v_i(k+1) = wv_i(k) + c_1.r_1(k)\left(x_g - x_i(k)\right) + c_2.r_2(k)\left(x_i^p - x_i(k)\right) \quad (3)$$

Where; w is the inertia weight of the particle, $v_i(k)$ is the velocity of the particle i at time step k, and x_g is the global best performing particle up to time step k in the entire population. x_i^p is the best experience particle i has had up to time step k, and $x_i(k)$ is the current location of particle i, and c_1, c_2 are the acceleration coefficients. r_1, r_2 are random numbers within [0, 1] those represent random fiction. To limit the searching space $v_i(k)$ is limited to be within a certain range of $v_{imin} \le v_i \le v_{imax}$. The new location of particle i can be calculated as:

$$x_i(k+1) = x_i(k) + v_i(k+1) \quad (4)$$

The evaluation of the particle performance is based on a problem specific objective function that decides the "closeness" of the particle to the optimal solution.

In the TVPSO; the vital parameters; inertia weight w and acceleration coefficients c_1 and c_2, are allowed to change with the iterations, making it capable of effectively handling optimization problems of different characteristics.

- The parameter w controls the influence of the previous velocity on the present velocity.
  Here, adaptation of w is introduced in TVPSO namely w_t. The value of w_t is allowed to decrease linearly with iteration from w_1 to w_2. The value of inertia weight w_t at iteration i, is obtained as:

$$w_t = (w_1 - w_2)\frac{I-i}{i} + w_2 \quad (5)$$

Where; I is the maximum number of iterations and i is the iteration number.

- The other two important parameters are c_1 and c_2, where c_1, is called the cognitive acceleration coefficient and c_2 the social acceleration coefficient. To incorporate better compromise between the exploration and exploitation of the search space in the swarm, c_1 has been allowed to decrease from its initial value of c_{1i} to c_{1f} while c_2 has been increased from c_{2i} to c_{2f}. Here, the parameters c_1, c_2 are replaced by c_{1t}, c_{2t} respectively and calculated as follows:

$$c_{1t} = \left(c_{1i} - c_{1f}\right)\frac{i}{I} + c_{1i} \tag{6}$$

$$c_{2t} = \left(c_{2i} - c_{2f}\right)\frac{i}{I} + c_{2i} \tag{7}$$

As it shown, the values of the acceleration coefficients and the inertia weight are always updated through the iterations. The flowchart of the TVPSO process can be shown in Fig. 3.

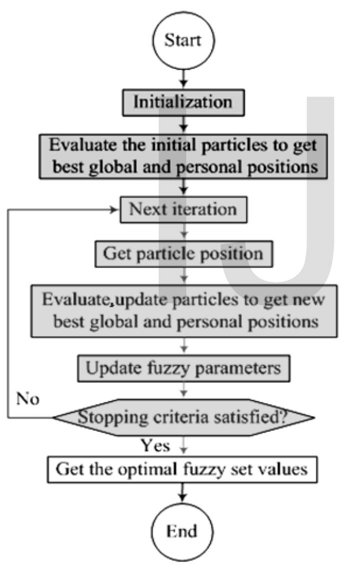


Fig. 3: Flowchart of TVPSO to adjust the MFs.

Throughout the optimization process, TVPSO updates the velocity vector for each particle then adds that velocity to the position of the particle. Velocity updating is influenced by both the best global solution, associated with the lowest cost (objective function) ever found by a particle, and the best local solution, associated with the lowest cost in the present population.

If the best local solution has a cost less than the cost of the current global solution, then the best local solution is replaced by the best global solution. The algorithm continues updating the velocities and adds them to the corresponding positions until a termination criterion, such as a limit on the number of iterations or satisfactory results, is reached, thereupon the process will stop.

4. MFs OPTIMIZATION:

In the current work, we have 14 MFs in the inputs and 7 MFs in the output. Each of the MFs has its own three parameters, as a result, there is a number of 63 parameters have to be optimized. The population is set to be $P = 100$ and the total searching iteration is set to be $I = 500$. The following factors are used through the optimization, $w_1 = 0.7, w_2 = 0.4, c_{1i} = 2.5, c_{1f} = 0.5, c_{2i} = 0.5, c_{1f} = 2.5$, while r_1, r_2 are randomized within [0, 1].

The object function is defined as:

$$F\big(z(k)\big) = k_1.R_{TM}\big(T_f\big) + k_2.\int_0^{T_f} {A_C}^2 dt \tag{8}$$

Where; $z(k)$ is a vector denotes the parameters of the MFs need to be optimized, k_1, k_2 are designed constants refer to preference of the terms, M_D and C_{EFF} respectively.

The changing of the object function value throughout the search of the optimal solution is plotted in Fig. 4. It can be seen that the searching can be terminated after about 320 iterations, when there is no reduction in total value of object function was observed.

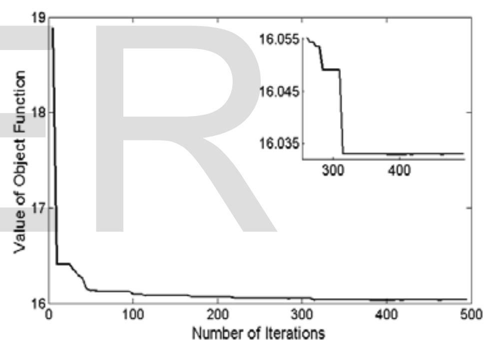
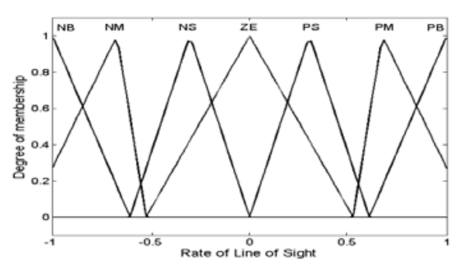
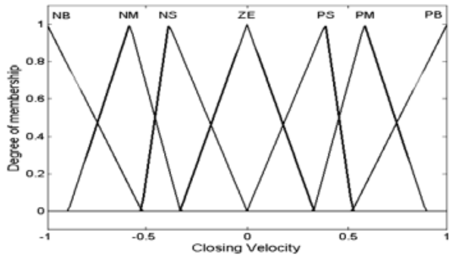


Fig. 4: Object function Reduction by TVPSO process.

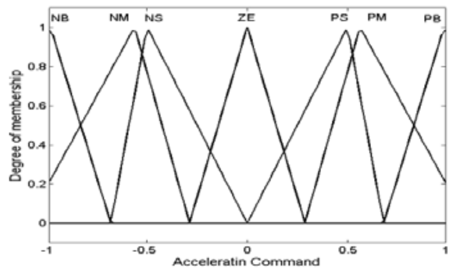
As soon as the process is finished, the optimized parameters can be extracted. Fig. 5 shows the optimized MFs of the inputs and the output.



(**a**)



(**b**)



(**c**)

Fig. 5: Optimized MFs of; **a**) Line of Sight Angle Rate, **b**) Closing Velocity, **c**) Acceleration Command.

5. RESULT AND ANALYSIS:

To examine the resulted guidance law, performance of OFG law is compared with PNG law. For that, engagement geometry of missile-target is considered with the following assumptions:

- Respect to the missile: initial position (0, 0) km, velocity 1000 m/sec, the missile can accelerate within [-200, +200] g.
- Respect to the target: initial position (15, 3) km, velocity 300 m/sec, the missile can accelerate within [-3.5, 5] g.
- Where g =9.8 m.sec^{-2}, is the gravity constant. The navigation ratio of PNG is *N*=4.

One of the important factors in the simulation process is usually the integration time-step. This is normally chosen based on nature of the problem or experience. Here, we use a time step equal to 0.01 second, mainly because a typical missile-gyro gyrates around 100 cycles per second. Fig. 6 illustrates a simple guidance loop.

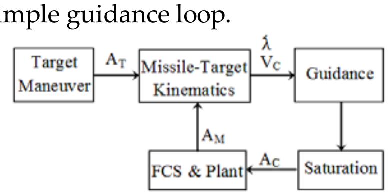


Fig. 6: Simple Guidance Loop.

Engagement accuracy of PNG and OFG for 19 different scenarios, respect to the target accelerations, is fully examined. The examined scenarios are chosen respect to the following target acceleration values (-3.5,-3, -2.5 … 4.5, 5) g. Root Mean Square values (RMS) of the terms, M_D and C_{EFF} for all scenarios are calculated and tabulated as follows:

Table 5: RMS values for the all scenarios.

RMS	M_D[m]	C_{EFF} [m²/sec³] $\times 10^{-4}$
OFG	5.75	4.86
PNG	6.83	6.26

The tabulated results declare that OFG law overweighs PNG law and shows decrements of (16% and 22%) with regard to the terms; M_D and C_{EFF} respectively.

Fig. 7 shows the trajectories for the first scenario (target accelerations is -3.5g), where Fig. 8 shows the missile acceleration commands for the corresponding scenario.

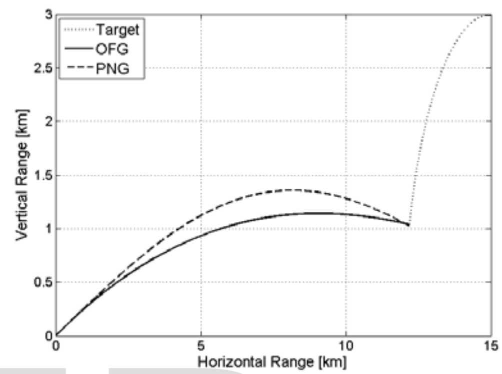


Fig. 7: Trajectories for OFG and PNG.

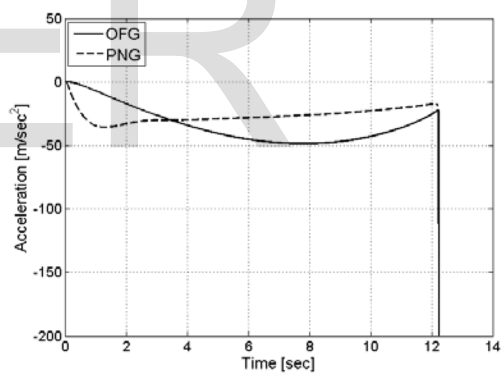


Fig. 8: Acceleration Commands for OFG and PNG.

6. CONCLUSION:

In the current work, an OFG law for surface to air missile against maneuvering Targets is introduced. Rules of the OFG law are established based on the conception of the classical PNG law. MFs of the proposed design are optimized using TVPSO algorithm. The optimization is achieved under the consideration of minimizing an object function which includes the terms; miss distance, and control effort. Many different scenarios are studied with regard to different values of terms; target's accelerations. Root Mean Square values of the miss distance and control effort are calculated. The results showed that OFG performs better than classical PNG. Nevertheless, further investigation might be required to examine the effect of noise and uncertainties in both missile and target dynamics.

REFERENCES:

[1] Fu Kuang Yeh, Hsiuan Hau Chien, and Li Chen Fu., "Design of optimal midcourse guidance sliding-mode control for missiles with TVC," *IEEE Transaction on Aerospace and Electronic Systems*. vol. 39, no. 3, pp. 824-837, July 2003.

[2] Chih Min Lin, and Ya Fu Peng, "Missile guidance law design using adaptive cerebellar model articulation controller," *IEEE Transaction on Neural Networks*. vol. 16, no. 3, pp. 636-644, May 2005.

[3] Jin Ik Lee, In Soo Jeon, and Min Jea Tahk, "Guidance law to control impact time and angle," *IEEE Transaction on Aerospace and Electronic Systems*. vol. 43, no. 1, pp. 301-310, May 2007.

[4] P. Zarchan, "Tactical and Strategic Missile Guidance," Fifth Edition (Progress in Astronautics and Aeronautics), pp. 185-205, January 2007.

[5] S. Lo, "An Adaptive-Network Based Fuzzy Inference System for Prediction of Work piece Surface Roughness in End Milling," *Journal of Materials Processing Technology*. vol. 142, Issue 3, pp. 665 -675, April 2006.

[6] Lin C.J., Chen C.H., and Lee C. Y., "TSK-type quantum neural fuzzy network for temperature control," *International Mathematical Forum*. vol. 1, no. 18, pp. 853 – 866, 2006.

[7] Mucientes M., Moreno D.L., Bugarin A., and Barro S., "Design of a fuzzy controller in mobile robotics using genetic algorithms," *Applied Soft Computing*. vol. 7, no. 2, pp. 540-546, March 2007.

[8] S. M. S. Boroujeni, R. Hemmati, H. Delafkar and A. S. Boroujeni, "Optimal PID power system stabilizer tuning based on particle swarm optimization,". *Indian Journal of Science and Technology*, vol. 4, no. 4, pp. 379-383, April 2011.

[9] Clerc, M. and Kennedy, J., "The particle swarm explosion, stability, and convergence in a multidimensional complex space," *IEEE Transactions on Evolutionary Computation*. vol. 6, no. 1, pp. 58-73, 2002.

[10] P. K. Tripathi, S. Bandyop and S. K. Pal, "Multi Objective Particle Swarm Optimization with time variant inertia and acceleration coefficients," *Information Sciences*. vol. 177, no. 22, pp. 5033–5049, November 2007.

[11] B. S. U. Mendis, T. D. Gedeon, J. Botzheim, and L. T. Koczy, "Generalized Weighted Relevance Aggregation Operators for Hierarchical Fuzzy Signatures," *International Conference on Computational Intelligence for Modeling Control and Automation*, Sydney, Australia. December 2006.

[12] J. Lindblad and N. Sladoje, "Feature Based Defuzzification at Increased Spatial Resolution," *Proc., of 11th Intern. Workshop on Combinatorial Image Analysis*, Berlin, Germany, Lecture Notes in Computer Science. vol. 4040, pp. 131-143, June 2006.

[13] S. Chakraborty and C. Yeh, "A Simulation Based Comparative Study of Normalization Procedures in Multi-attribute Decision Making," *The 6th WSEAS Int. Conf. on Artificial Intelligence, Knowledge Engineering and Data Bases, Corfu Island, Greece*. vol. 6, pp. 102-109, February 2007.

[14] Qinghai Bai, "Analysis of Particle Swarm Optimization Algorithm. CCSE," *Computer and Information Science*. vol. 3, no. 1, pp. 180-184, February 2010.