Available online at www.sciencedirect.com
ScienceDirect
Electric Power Systems Research 77 (2007) 1689–1698
ELECTRIC POWER SYSTEMS RESEARCH
www.elsevier.com/locate/epsr

Intelligent particle swarm optimized fuzzy PID controller for AVR system

V. Mukherjee [a], S.P. Ghoshal [b,*]

[a] *Department of Electrical Engineering, Asansol Engineering College, Asansol, West Bengal, India*
[b] *Department of Electrical Engineering, National Institute of Technology, Durgapur, West Bengal, India*

Received 18 January 2006; received in revised form 29 November 2006; accepted 1 December 2006
Available online 8 January 2007

Abstract

In process plants like thermal power plants, biomedical instrumentation the popular use of proportional-integral-derivative (PID) controllers can be noted. Proper tuning of such controllers is obviously a prime priority as any other alternative situation will require a high degree of industrial expertise. So in order to get the best results of PID controllers the optimal tuning of PID gains is required. This paper, thus, deals with the determination of off-line, nominal, optimal PID gains of a PID controller of an automatic voltage regulator (AVR) for nominal system parameters and step reference voltage input. Craziness based particle swarm optimization (CRPSO) and binary coded genetic algorithm (GA) are the two props used to get the optimal PID gains. CRPSO proves to be more robust than GA in performing optimal transient performance even under various nominal operating conditions. Computational time required by CRPSO is lesser than that of GA. Factors that have influenced the enhancement of global searching ability of PSO are the incorporation of systematic and intelligent velocity, position updating procedure and introduction of craziness. This modified from of PSO is termed as CRPSO. For on-line off-nominal system parameters Sugeno fuzzy logic (SFL) is applied to get on-line terminal voltage response. The work of SFL is to extrapolate intelligently and linearly, the nominal optimal gains in order to determine off-nominal optimal gains. The on-line computational burden of SFL is noticeably low. Consequently, on-line optimized transient response of incremental change in terminal voltage is obtained.

Keywords: AVR; CRPSO; GA; Optimal control; PID controller; SFL

1. Introduction

Even though several control theories have been developed significantly, we do see the widely popular use of proportional-integral-derivative (PID) controllers in process control, motor drives, flight control, and instrumentation. The reason of this acceptability is for its simple structure which can be easily understood and implemented. Industries too can boast of the extensive use of PID controllers because of its robustness and simplicity. The past decades witnessed many advancing improvements keeping in mind the requirement of the end users. Easy implementation of hardware and software has helped to gain its popularity. Several approaches have been documented in literatures for determining the PID parameters of such controllers. Genetic Algorithm [5], neural network [1], fuzzy based approach [2,4], neuro-fuzzy approach [8], evolutionary computational techniques [10,11] are just a few among these numerous works.

This paper focuses on optimal tuning of PID controller for the AVR using craziness based particle swarm optimization (CRPSO) and binary coded genetic algorithm (GA).

Particle swarm optimization (PSO) [3,6,7] is a population-based evolutionary algorithm. Instead of the survival of the fittest, it is the simulation of the social behavior that motivates PSO. Here each candidate solution is associated with a velocity. Particles or the candidate solutions then *fly* through the search space. The velocity is constantly adjusted with the corresponding particle's and its companions' experience. Resultantly particles move towards better solution areas. A novel velocity, position updating strategy and the introduction of craziness in normal PSO enhances its global searching ability. Thus, sharp tuning of the parameters of PID controller is achieved in AVR system.

* Corresponding author.
E-mail addresses: vivek_agamani@yahoo.com (V. Mukherjee), spghoshalnitdgp@yahoo.com (S.P. Ghoshal).

0378-7796/$ – see front matter © 2006 Elsevier B.V. All rights reserved.
doi:10.1016/j.epsr.2006.12.004

Table 1
Parameters of PID controller and AVR model with transfer function and parameter limits

Item	Transfer function	Parameter limits
PID controller	$G(s) = K_p + \frac{K_i}{s} + K_d s$	$0.2 \le K_p, K_i, K_d \le 2.0$
Amplifier	$\mathrm{TF}_{\text{amplifier}} = \frac{K_a}{1+\tau_a s}$	$10 \le K_a \le 40$; $0.02\,\mathrm{s} \le \tau_a \le 0.1\,\mathrm{s}$
Exciter	$\mathrm{TF}_{\text{exciter}} = \frac{K_e}{1+\tau_e s}$	$1 \le K_e \le 10$; $0.4\,\mathrm{s} \le \tau_e \le 1.0\,\mathrm{s}$
Generator	$\mathrm{TF}_{\text{generator}} = \frac{K_g}{1+\tau_g s}$	K_g depends on load (0.7–1.0); $1.0\,\mathrm{s} \le \tau_g \le 2.0\,\mathrm{s}$
Sensor	$\mathrm{TF}_{\text{sensor}} = \frac{K_s}{1+\tau_s s}$	$0.001\,\mathrm{s} \le \tau_s \le 0.06\,\mathrm{s}$

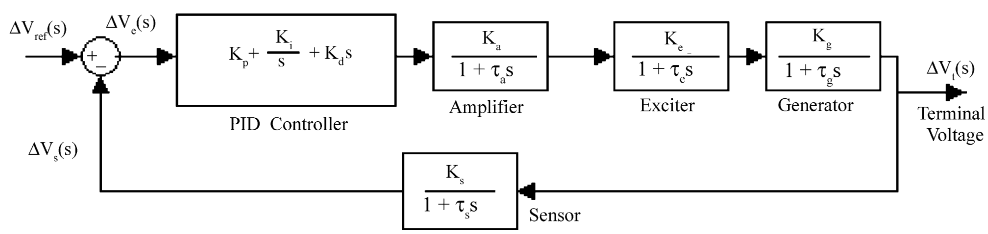

Fig. 1. Block diagram of AVR system along with PID controller.

For varying on-line off-nominal conditions, fast acting, intelligent Sugeno fuzzy logic (SFL) [9,10] is applied to yield incremental change in terminal voltage response.

The rest of the paper is documented in the following headings. Sections 2 and 3 provide PID controller and AVR model and AVR with PID controller, respectively. Section 4 deals with the design of misfitness function. Sections 5–7 narrate an overview of PSO including CRPSO, SFL and input parameters as used to perform simulation study. Results obtained from simulation and observations are described in Section 8. Concluding remarks have been focused in Section 9.

2. PID controller and AVR model

PID controllers are being extensively used by industries today owing to their simplicity. Its main focus here is reduction/elimination of steady state error as well as an improvement in the dynamic response. Reduction/elimination of steady state error is achieved by adding a pole at the origin with the help of integral controller, there by increasing the system type by one. Transient response improvement may be achieved from the action of derivative controller which adds a finite zero to the open loop transfer function. As modeled in this paper, the transfer function of PID controller [11] is

$$G(s) = K_p + \frac{K_i}{s} + K_d s \tag{1}$$

Table 1 depicts parameters of PID controller and AVR model as considered in this paper, transfer function of each item including limits of parameters [11]. In [11], Gaing has taken the generator transfer function as $K_g/(1+\tau_g s)$ where K_g depends on load (0.7–1.0) and $1.0\,\mathrm{s} \le \tau_g \le 2.0\,\mathrm{s}$. The same model has been taken in the present work.

3. AVR with PID controller

Incorporating the above models (Table 1) a composite AVR system along with PID controller is obtained. The block diagram representation is shown in Fig. 1. Equation of the incremental change in terminal voltage ($\Delta V_t(s)$) with an incremental change in reference voltage input ($\Delta V_{\text{ref}}(s)$) is as follows:

$$\left[\Delta V_{\text{ref}}(s) - \left(\frac{K_s}{1+\tau_a s}\right)\Delta V_t(s)\right]\left[\left(\frac{K_a}{1+\tau_a s}\right)\left(\frac{K_e}{1+\tau_e s}\right) \times \left(\frac{K_g}{1+\tau_g s}\right)\left(K_p + \frac{K_i}{s} + K_d s\right)\right] = \Delta V_t(s) \tag{2}$$

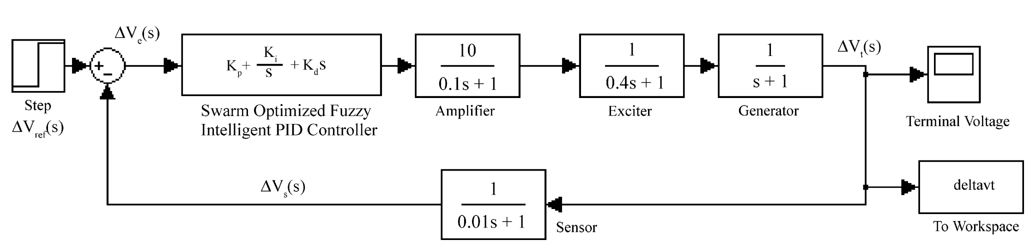

Fig. 2. MATLAB-SIMULINK based block diagram of AVR system along with intelligent PID controller.

i.e.

$$\frac{\Delta V_t(s)}{\Delta V_{ref}(s)} = \frac{(s^2K_d + sK_p + K_i)(K_aK_eK_g)(1+s\tau_s)}{s(1+s\tau_a)(1+s\tau_e)(1+s\tau_g)(1+s\tau_s) + (K_aK_eK_gK_s)(s^2K_d + sK_p + K_i)} \quad (3)$$

Fig. 2 shows the MATLAB-SIMULINK based swarm optimized fuzzy intelligent PID controller block diagram of AVR system.

4. Design of misfitness function

The performance criterion is to be judged based on a misfitness function (MF). Optimization of PID gains by applying any of the optimization techniques corresponds to minimum misfitness function value. The MF is being formulated as follows:

$$\mathrm{MF} = (o_{sh} \times 10,000)^2 + t_{st}^2 + \frac{0.001}{(\max\_\mathrm{dv})^2} \quad (4)$$

Minimization of MF with the help of any optimization technique corresponds to minimum overshoot (o_{sh}), minimum settling time (t_{st}), and maximum max_dv. Repetitive trial run of the optimizing algorithms reveals that o_{sh} is having the minimum and maximum value of 0.0000 and 0.0002, respectively. Thus, the first term in the right hand side of (4) is in the order of 0–4. The numerical value of t_{st} lies from 1.6400 to 5.4194. Thus, the second term in the right hand side of (4) is in the range of 2.6896–29.3699. The value of max_dv lies from 0.0116 to 0.0400, yielding the third term of (4) in the range of 7.4316–0.6250. Thus, incorporation of appropriate weighting factors to the right hand individual terms facilitates to make each term competitive during the optimization process. Any other choice of the weighting factors lead to incompatible numerical values of each term involved in the definition of MF which gives misleading result.

5. Particle swarm optimization

The PSO was first introduced by Kennedy and Eberhart [3]. It is an evolutionary computational model, a stochastic search technique based on swarm intelligence. Social behavioral pattern of organisms such as bird flocking and fish schooling inspired them to look into the effect of collaboration of species when achieving their goals as a group. Dynamics of bird flocking resulted in the possibilities of utilizing this behavior as an optimization tool. These have been used to solve a range of optimization problems.

5.1. Review of PSO algorithm

The PSO [3,6,7] is a population-based optimization technique, where the population is called 'swarm'. In a PSO system, multiple candidate solutions coexist and collaborate simultaneously. Each solution candidate, called a 'particle', flies in the problem space (similar to the search process for food of a bird swarm) looking for the optimal position. A 'particle' with time adjusts its position to its own 'experience', while adjusting to the 'experience' of neighboring particles. If a particle discovers a promising new solution, all the other particles will move closer to it, exploring the region more thoroughly in the process. Based on PSO concept, mathematical equations for the searching process are:

$$\text{Velocity updating equation}: \quad v_i^{k+1} = v_i^k + c1r1(\mathrm{pBest}_i - x_i^k) + c2r2(\mathrm{gBest} - x_i^k) \quad (5)$$

$$\text{Position updating equation}: \quad x_i^{k+1} = x_i^k + v_i^{k+1} \quad (6)$$

The flow chart of PSO algorithm may be outlined as shown in Fig. 3.

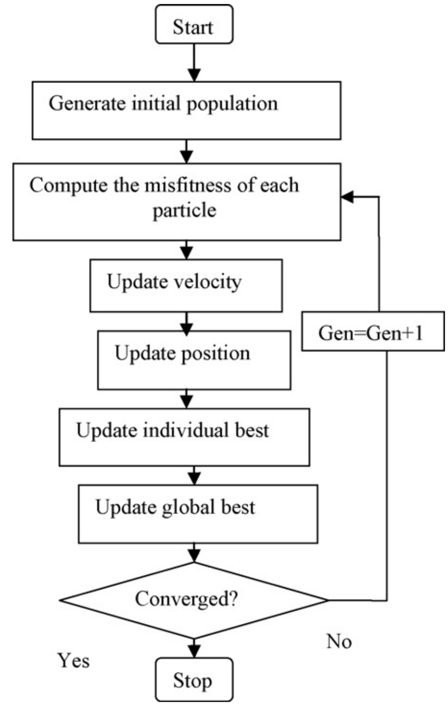

Fig. 3. Flowchart of PSO.

5.2. Craziness based particle swarm optimization

The following modifications help to enhance the global search ability of PSO algorithm.

5.2.1. Position and velocity updating

In (5), the second term on the right hand side represents the personal behavior whereas the third term represents the social behavior of the particles. As numbers $r1$ and $r2$ are generated randomly, they could be too large or too small. If these random numbers are too large, personal and social experiences will be over used driving the particles too far away from the local optimum. While on the other hand, if these two random numbers are too small, the two experiences are not fully utilized reducing the convergence speed. This behavioral pattern is not in harmony with the natural human behavior when at work. In fact, they try to strike a balance between the best personal as well as the group behavioral pattern. It needs to be noted here that the two random weighing patterns in (5) are not completely independent. If one is too small the other should be too large and vice versa. Thus, introduction of only one random number ($r1$) and noting that the sum of two interrelated random numbers are unity, as proposed in [12] may be stated as in (7).

$$v_i^{k+1} = r2v_i^k + (1 - r2)c1r1(\text{pBest}_i - x_i^k) + (1 - r2)c2(1 - r1)(\text{gBest} - x_i^k) \quad (7)$$

Local and global searches are balanced by random number $r2$ as stated in (7). Change in the direction in velocity [12] may be modeled as in (8).

$$v_i^{k+1} = r2\,\text{sign}(r3)v_i^k + (1 - r2)c1r1(\text{pBest}_i - x_i^k) + (1 - r2)c2(1 - r1)(\text{gBest} - x_i^k) \quad (8)$$

In (8), sign($r3$) may be defined as

$$\text{sign}(r3) = \begin{cases} -1, & r3 \le 0.05 \\ 1, & r3 > 0.05 \end{cases} \quad (9)$$

5.2.2. Inclusion of craziness

Diversity in the direction of birds flocking or fish schooling may be handled in traditional PSO with a predefined craziness probability [12]. The particles may be crazed in accordance with (10) before updating its position.

$$v_i^{k+1} = v_i^{k+1} + \text{Pr}(r4)\,\text{sign}(r4)v_i^{\text{craziness}} \quad (10)$$

where, Pr($r4$) and sign($r4$) are defined, respectively, as

$$\text{Pr}(r4) = \begin{cases} 1, & r4 \le P_{\text{craz}} \\ 0, & r4 > P_{\text{craz}} \end{cases} \quad (11)$$

$$\text{sign}(r4) = \begin{cases} 1, & r4 \ge 0.5 \\ -1, & r4 < 0.5 \end{cases} \quad (12)$$

5.3. Selection of CRPSO parameters

The main highlighting features that characterize this novel class of PSO are

(a) balancing between over use and under use of social and personal experiences by random number $r1$ as shown in (7);
(b) balancing between global and local searches using random number $r2$ as given in (7);
(c) change in the direction in velocity of the particles by sign($r3$) as stated in (8);
(d) diversity in the direction by inclusion of craziness by $v_i^{\text{craziness}}$ and P_{craz} as noted in (10).

The authors have investigated the impact of $v_i^{\text{craziness}}$ and P_{craz} on the performance of CRPSO. It has been revealed that a rise in the value of these parameters leads to a rise in fluctuation of final convergent value of MF. If P_{craz} becomes less than 0.2, then, often, $v_i^{\text{craziness}}$ term in (10) is introduced which is not also desirable. The definition of sign($r3$) as in (9) shows that the chance of sign($r3$) to be as -1 is very low due to rare chance of $r3$ becoming less than 0.05 leading to very often chance in change of direction of the particles. Eq. (11) determines whether the velocity of the particle is to be crazed or not. The direction of $v_i^{\text{craziness}}$ is determined by sign($r4$). Depending upon the value of sign($r4$) which on the other hand depends on the chance occurrence of $r4$ to become less than or greater than 0.5, $v_i^{\text{craziness}}$ is added or subtracted in velocity updating Eq. (10). Moderate inclusion of craziness is always recommended as the chance of occurrence of $r4$ becoming less than 0.5 is usually 50%. Thus, the choice of $v_i^{\text{craziness}} = 0.1–0.4$, $P_{\text{craz}} = 0.2$, sign($r3$) as in (9) and sign($r4$) as in (12) may be accepted values and definitions.

6. Review of Sugeno fuzzy logic

The whole process [9,10] can be categorized into three steps. They are:

(a) *Fuzzification of input operating conditions*: Fuzzify the input operating conditions, generator gain (K_g) and generator time constant (τ_g) in terms of fuzzy subsets. The fuzzy subsets for input (K_g) are "Low (L)", "Medium (M)", "Medium High (MH)", and "High (H)". They are associated with overlapping triangular membership functions. The respective nominal central values of the subsets of K_g are (0.7, 0.8, 0.9, and 1.0) at which membership values are unities. Similarly, the overlapping fuzzy subsets for input (τ_g) are "Low (L)", "Medium Low (ML)", "Medium (M)", "High Medium (HM)", "Low High (LH)", and "High (H)". They are also associated with overlapping triangular membership functions. The respective nominal central values of the subsets of τ_g are (1.0, 1.2, 1.4, 1.6, 1.8, and 2.0) at which membership values are unities. These are nominal system operating conditions. *Sugeno fuzzy rule base table* consists of 24 (=4 × 6) logical operating conditions. Each input corresponds to nominal optimal PID gain as output.
(b) *Sugeno fuzzy inference*: For on-line real time imprecise values of operating conditions, firstly their subsets in which the values lie are determined with the help of "IF", "THEN" logic and correspondingly, membership values are determined from the membership functions of the subsets. From Sugeno fuzzy rule base table, corresponding input sets and nominal PID gains are adopted. Now for each satisfying input set number, two membership values like μ_{K_g} and μ_{τ_g} and their minimum $\mu_{\min}$ are determined, along with their corresponding PID gains. For the unsatisfying input sets, as conditioned are not fulfilled in the corresponding fuzzy subsets, $\mu_{\min}$ will be zero.
(c) *Sugeno defuzzification*: Sugeno defuzzification will yield crisp output for each PID gain as:

$$\text{Final crisp output, } G_{\text{crisp}} = \frac{\sum_i \mu_{\min}^{(i)} G_i}{\sum_i \mu_{\min}^{(i)}} \quad (13)$$

where i corresponds to input sets satisfied among 24 input sets, and G_i is any of the PID gains. For example, G_{crisp}/K_p, K_i, K_d are the crisp parameters of PID controller. $\mu_{\min}^{(i)}$ is the minimum membership value corresponding to ith input set being satisfied.

Table 2
Sugeno fuzzy rule base table, optimized PID gains and transient response parameters

K_g	τ_g	Type of controller	K_p	K_i	K_d	o_{sh}	t_{st} (s)	max_dv	MF	Time of execution (s)
0.7	1.0	CRPSO-PID	0.6752	0.4822	0.2052	0.0000	2.0782	0.0235	6.1297	68.55
		GA-PID	0.9453	0.6248	0.3750	0.0001	2.8310	0.0215	11.1779	73.97
	1.2	CRPSO-PID	0.8903	0.5522	0.3089	0.0000	2.3337	0.0284	6.6860	69.03
		GA-PID	0.9687	0.5928	0.3750	0.0001	2.5597	0.0270	8.9238	75.09
	1.4	CRPSO-PID	0.9657	0.5392	0.3266	0.0000	2.1804	0.0262	6.2109	67.01
		GA-PID	0.9199	0.4370	0.5400	0.0001	4.4390	0.0210	22.9723	70.81
	1.6	CRPSO-PID	1.0989	0.4480	0.3067	0.0000	1.8844	0.0224	5.5439	67.95
		GA-PID	0.5996	0.2942	0.1806	0.0001	2.0305	0.0155	9.2853	94.78
	1.8	CRPSO-PID	1.8811	0.8766	0.8057	0.0000	2.1309	0.0317	5.5359	69.05
		GA-PID	0.7773	0.3437	0.2504	0.0001	2.8079	0.0183	11.8704	100.25
	2.0	CRPSO-PID	1.9950	0.8687	0.8465	0.0000	2.0478	0.0294	5.3504	66.47
		GA-PID	0.7656	0.3114	0.2502	0.0001	2.2857	0.0167	9.8101	95.08
0.8	1.0	CRPSO-PID	0.5884	0.4005	0.2261	0.0000	2.0000	0.0237	5.7803	68.52
		GA-PID	0.6871	0.5006	0.2028	0.0002	2.5854	0.0213	12.8884	74.99
	1.2	CRPSO-PID	0.8343	0.5240	0.2802	0.0001	2.0452	0.0295	6.3319	68.48
		GA-PID	0.9062	0.4995	0.5010	0.0002	3.7037	0.0131	23.5446	70.98
	1.4	CRPSO-PID	0.7964	0.4395	0.2801	0.0000	2.4946	0.0259	7.7138	68.39
		GA-PID	0.7510	0.4044	0.2617	0.0001	2.8257	0.0243	10.6781	81.53
	1.6	CRPSO-PID	1.0913	0.5483	0.4183	0.0001	2.1605	0.0317	6.6629	67.14
		GA-PID	0.7504	0.3730	0.2500	0.0002	2.3413	0.0218	11.5859	82.86
	1.8	CRPSO-PID	1.6184	0.7572	0.6788	0.0001	2.1472	0.0311	6.6444	63.47
		GA-PID	0.8906	0.3740	0.5005	0.0002	4.1290	0.0245	22.7146	69.63
	2.0	CRPSO-PID	1.6242	0.7005	0.6845	0.0000	2.0870	0.0280	5.6311	67.16
		GA-PID	0.7497	0.3125	0.2500	0.0001	2.9612	0.0186	12.6592	111.03
0.9	1.0	CRPSO-PID	0.5264	0.3785	0.1546	0.0000	1.7833	0.0238	4.9456	68.59
		GA-PID	0.7656	0.3740	0.5322	0.0001	5.4194	0.0185	33.2917	69.94
	1.2	CRPSO-PID	0.7341	0.4597	0.2378	0.0001	1.9560	0.0282	6.0834	67.98
		GA-PID	0.9149	0.5307	0.4154	0.0002	3.0332	0.0116	20.6319	70.23
	1.4	CRPSO-PID	0.6755	0.3683	0.2050	0.0001	1.8400	0.0295	5.5347	68.25
		GA-PID	0.8736	0.4684	0.3774	0.0002	2.7957	0.0267	13.2187	72.95
	1.6	CRPSO-PID	0.6357	0.3148	0.1963	0.0001	2.2177	0.0203	8.3448	67.38
		GA-PID	0.8279	0.4057	0.3128	0.0002	2.7444	0.0178	14.6879	73.11
	1.8	CRPSO-PID	1.5402	0.7248	0.7109	0.0000	2.5035	0.0349	7.0885	66.80
		GA-PID	0.9984	0.5034	0.3515	0.0001	4.4379	0.0225	22.6703	69.56
	2.0	CRPSO-PID	0.8838	0.3768	0.2992	0.0001	1.6400	0.0237	5.4699	67.45
		GA-PID	0.6543	0.2793	0.2500	0.0002	2.8933	0.0193	15.0558	79.41
1.0	1.0	CRPSO-PID	0.3741	0.2685	0.1000	0.0001	2.4621	0.0190	9.8320	68.74
		GA-PID	0.5781	0.3745	0.2502	0.0002	3.2229	0.0156	18.4962	73.55
	1.2	CRPSO-PID	0.5672	0.3482	0.2041	0.0000	2.3076	0.0265	6.7490	68.41
		GA-PID	0.7529	0.4365	0.3008	0.0001	2.9873	0.0208	12.2354	75.55
	1.4	CRPSO-PID	0.7032	0.3978	0.2452	0.0000	2.1448	0.0280	5.8757	67.69
		GA-PID	0.8737	0.5009	0.3209	0.0001	3.5336	0.0267	14.8891	71.52
	1.6	CRPSO-PID	0.6796	0.3408	0.2183	0.0001	1.7200	0.0240	5.6945	68.52
		GA-PID	0.7616	0.3867	0.2660	0.0002	2.0854	0.0216	10.4922	84.66
	1.8	CRPSO-PID	1.2730	0.5952	0.5671	0.0000	2.3358	0.0321	6.4264	66.42
		GA-PID	0.7969	0.3434	0.5056	0.0001	4.0161	0.0286	18.3516	69.63
	2.0	CRPSO-PID	0.8367	0.3540	0.2886	0.0001	2.0400	0.0250	6.7616	68.16
		GA-PID	0.7049	0.2964	0.2500	0.0002	2.3901	0.0213	11.9167	82.98

7. Input data and parameters

Maximum population size = 50, maximum allowed iteration cycles = 100, these two parameters are common for both GA and CRPSO algorithms. For GA, number of parameters = 6; number of bits = (number of parameters) × 8; mutation probability = 0.04; crossover rate = 100%; selection ratio, $s_r = 0.3$. For CRPSO, craziness probability $(P_{craz}) = 0.2$, $c1 = c2 = 2.05$. The value of $v_i^{craziness}$ lies between 0.1 and 0.4. The parameters of the block diagram are chosen as $K_a = 10$, $K_e = K_g = K_s = 1.0$, $\tau_a = 0.1$ s, $\tau_e = 0.4$ s, $\tau_s = 0.01$ s, $\tau_g = 1.0$ s. Only K_g is load dependent. Simulation results are obtained by MATLAB 6.5 software on a 3.0 GHz P4 computer.

8. Simulation results and discussions

Step perturbation of 0.01 p.u. of reference voltage has been applied to get the transient response of incremental change in terminal voltage in the present work. The major observations of the present work are as documented:

(i) *Optimized nominal transient terminal voltage response profile and Sugeno fuzzy rule base table*: Table 2 (based on nominal input parameters given in Sugeno fuzzy rule base table) has been computed to illustrate the comparative performance characteristics of CRPSO-PID controller/GA-PID controller. For the present work K_a has been assumed as 10. MATLAB-SIMULINK based simulation shows instability in the performance of AVR system for higher value of K_a. K_g has been varied from 0.7 to 1.0 in steps of 0.1. τ_g has been varied from 1.0 to 2.0 in steps of 0.2. Thus, Table 2 includes 24 different sets of input conditions of AVR system. From Table 2 it may be noted that CRPSO based optimization technique offers (a) lesser overshoot of change in terminal voltage (o_{sh}), (b) lesser settling time of change in terminal voltage ($t_{st} \ll 2.5$ s for CRPSO and $t_{st} > 2.0$ s for GA), and (c) more maximum derivative of change in terminal voltage (max_dv). The value of MF of terminal response profiles are less for CRPSO based optimization than those of GA based optimization technique. It is reflected in Figs. 4–7. Figs. 4–7 depict CRPSO based PID controller exhibits better optimal transient response characteristic in respect of step response of incremental change in terminal voltage. CRPSO based PID-controller settles the response quickly with (a) lesser overshoot, (b) lesser settling time, and (c) more maximum derivative of change in terminal voltage. Sugeno fuzzy intelligent CRPSO-PID-controller performs better than Sugeno fuzzy intelligent GA based PID controller. Sugeno fuzzy rule based off-nominal, on-line optimal gains and incremental change in terminal voltage transient response (using CRPSO based optimal gains of Table 2) for off-nominal parameters is shown in Table 3.

(ii) *Validation of the results obtained*: Analytical transient voltage step response result obtained has been validated with the help of SIMULINK. The PID gains obtained analytically both for CRPSO and GA have been put in SIMULINK diagram and thus Fig. 8 is obtained. This validates analytically obtained Fig. 6.

(iii) *SFL based response*: For on-line, off-nominal input sets of parameters, SFL has been successfully applied to get on-line, optimal PID gains and these PID gains also yield optimal terminal voltage response (Fig. 9).

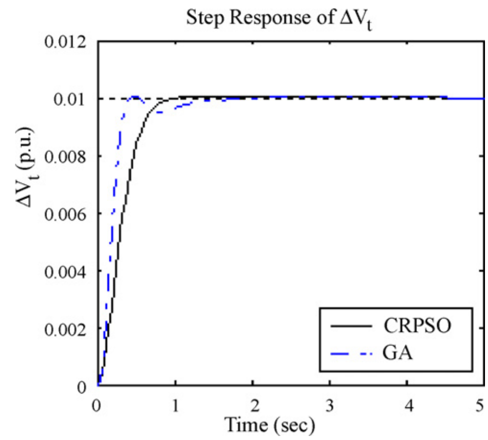

Fig. 4. Step response of incremental change in terminal voltage of PID controller based AVR system ($K_g = 0.7$, $\tau_g = 1.0$).

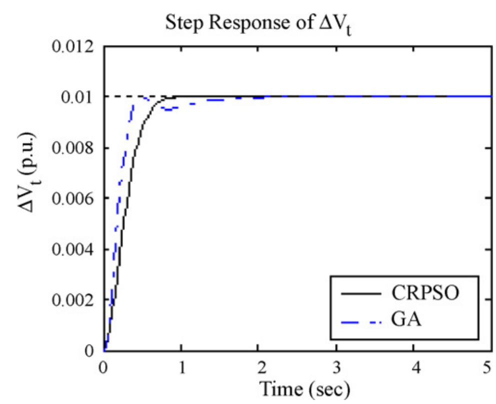

Fig. 5. Step response of incremental change in terminal voltage of PID controller based AVR system ($K_g = 0.8$, $\tau_g = 1.2$).

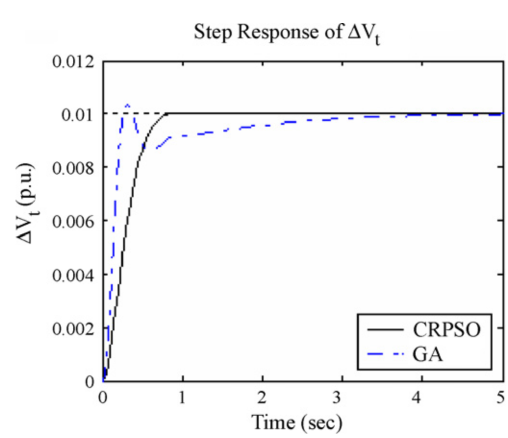

Fig. 6. Step response of incremental change in terminal voltage of PID controller based AVR system ($K_g = 0.9$, $\tau_g = 1.0$).

Table 3
Sugeno fuzzy based off-nominal, on-line optimal gains and incremental change in terminal voltage transient response (using CRPSO based optimal gains of Table 2)

Sl. no.	Off-nominal parameters (K_g, τ_g)	K_p	K_i	K_d	o_{sh}	t_{st} (s)	max_dv	MF
1	0.77, 1.33	0.6976	0.3968	0.2170	0.0000	2.4400	0.0223	7.9645
2	0.87, 1.89	0.9864	0.4425	0.3552	0.0000	2.1707	0.0262	6.1687
3	0.95, 1.67	1.5546	0.6884	0.7590	0.0000	2.6746	0.0333	8.0553
4	1.01, 1.96	1.4855	0.6495	0.8319	0.0000	2.8640	0.0400	8.8275
5	0.72, 1.42	0.9630	0.5296	0.3584	0.0000	2.5610	0.0283	7.8073

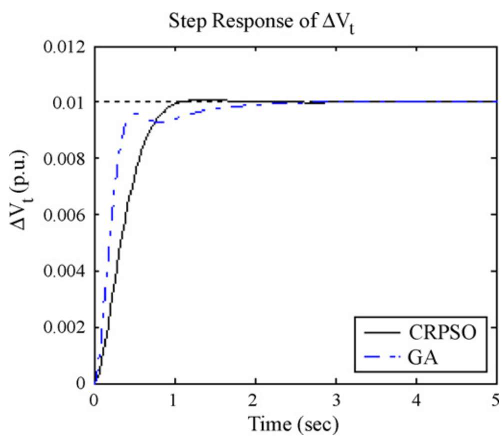

Fig. 7. Step response of incremental change in terminal voltage of PID controller based AVR system ($K_g = 1.0$, $\tau_g = 1.0$).

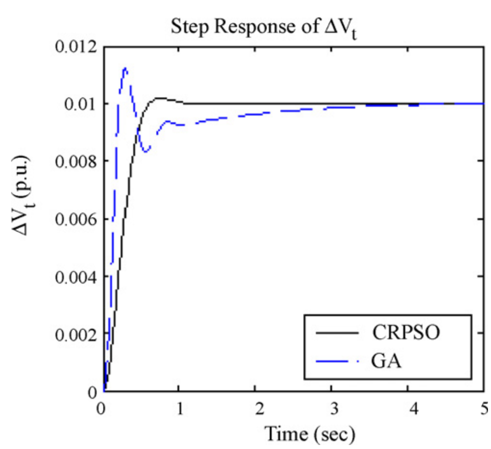

Fig. 8. MATLAB-SIMULINK based step response of incremental change in terminal voltage of PID controller based AVR ($K_g = 0.9$, $\tau_g = 1.0$).

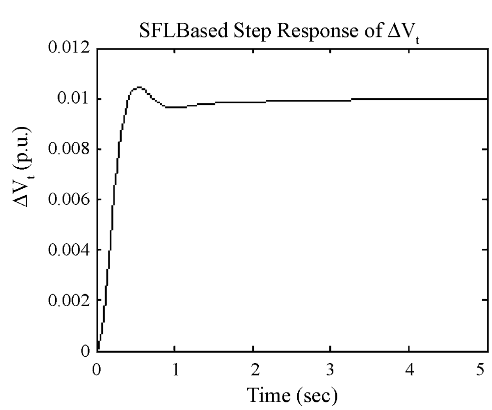

Fig. 9. SFL based response of an AVR system with PID controller (off-nominal input set: $K_g = 0.77$, $\tau_g = 1.33$, CRPSO based Sugeno fuzzy rule base Table 3).

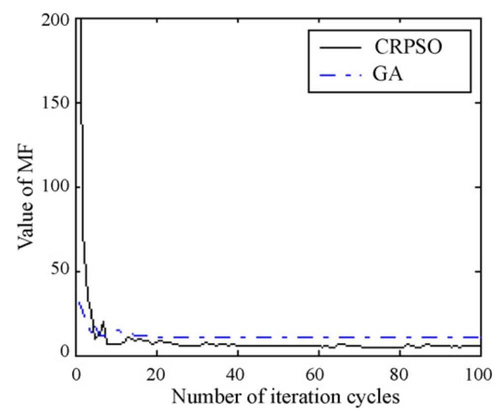

Fig. 10. Convergence profile of CRPSO and GA ($K_g = 0.9$, $\tau_g = 1.2$).

(iv) *Convergence profile*: The minimum misfitness function against number of iteration cycles of the swarm is recorded in each iteration to get the convergence profile of the algorithm. Fig. 10 portrays the convergence profiles of minimum MF of CRPSO and GA. From this figure it is clear that CRPSO converges faster than GA. GA yields suboptimal higher values of MF. It is interesting to note that in [11], GA convergence characteristic exhibits fluctuation due to the low value of crossover rate and mutation probability. In [11], Gaing assumed crossover rate = 60% and mutation probability = 0.01. In the present work, it is observed that low values of crossover rate and mutation probability results in more suboptimal results. So to get better response of GA, this phenomenon dictates that the two requisite factors are high crossover rate (100%) and high value of mutation probability (0.04).

(v) *Comparison with recent work of PSO-PID controlled AVR system*: In [11], Gaing optimized the parameters of PID controller in AVR system using particle swarm optimization technique. The PSO used in [11] has been termed as HPSO1 in [10] except the concept of selection ratio. The detailed algorithm has also been discussed in [10]. With the same input data and parameters as in [11] and selection ratio = 0.3, Table 4 gives the optimized performance of the PID controller. Table 4 contrasts CRPSO-PID controller with HPSO1-PID controller. From this table the superiority of CRPSO in terms of step response profile of incremental change in terminal voltage is clear. CRPSO-PID controller gives (a) much lesser value of o_{sh}, (b) much lesser value of t_{st}, and (c) more value of max_dv. Thereby, it yields lower value of MF. Fig. 11 displays the same superiority of CRPSO-PID controller. Derivative feed-

Table 4
Performance judgment of PID controller with proposed PSO and recent work

K_g	τ_g	Type of controller	o_{sh}	t_{st} (s)	max_dv	MF
0.7	1.6	CRPSO-PID	0.0000	1.8844	0.0224	5.5439
		HPSO1-PID	0.0001	2.1807	0.0217	7.8791
0.8	1.4	CRPSO-PID	0.0000	2.4946	0.0259	7.7138
		HPSO1-PID	0.0001	2.0402	0.0225	9.4279
0.9	1.2	CRPSO-PID	0.0001	1.9560	0.0282	6.0834
		HPSO1-PID	0.0002	2.3351	0.0221	11.5002
1.0	1.8	CRPSO-PID	0.0000	2.3358	0.0321	6.4264
		HPSO1-PID	0.0001	2.5035	0.0214	9.4511

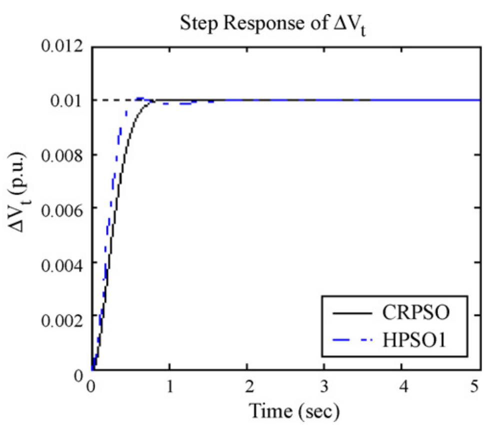

Fig. 11. Comparison of step response of incremental change in terminal voltage of CRPSO-PID controller with HPSO1-PID controller ($K_g = 0.9$, $\tau_g = 1.2$).

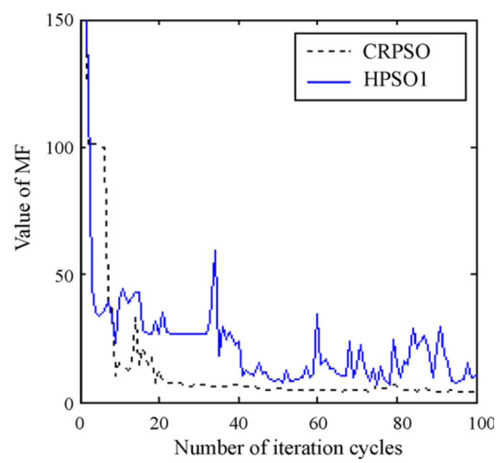

Fig. 12. Convergence profile of CRPSO-PID controller and HPSO1-PID controller ($K_g = 0.9$, $\tau_g = 1.2$).

back reduces the overshoot to a great extent. Comparison of convergence profile of CRPSO with that of HPSO1 is also shown in Fig. 12. CRPSO converges faster than HPSO1.

(vi) *Performance analysis of the proposed method with other optimizing methods*: In [10], hybrid particle swarm optimization with constriction factor approach termed as HPSO2 has been elaborately discussed. Hybrid Taguchi-particle swarm optimization (HTPSO) has received considerable attention and discussion in [13]. These two varieties of particle swarm optimization technique have also been contrasted with the proposed CRPSO algorithm in the present work. A glance on Table 5 focuses on the novelty of the proposed algorithm. In this table, CRPSO-PID controller maintains much lesser values of o_{sh}, much lesser values of t_{st} and more value of max_dv and thereby its lesser value of MF. Fig. 13 displays the same observations. By comparing convergence profiles of CRPSO with those of HPSO2 and HTPSO, it may be inferred that CRPSO exhibits better convergence profile (Fig. 14).

Table 5
Performance analysis of CRPSO-PID controller and other types of PSO-PID controller

K_g	τ_g	Type of controller	o_{sh}	t_{st} (s)	max_dv	MF
0.7	1.8	CRPSO-PID	0.0000	2.1309	0.0317	5.5359
		HPSO2-PID	0.0001	2.2240	0.0305	7.0212
		HTPSO-PID	0.0001	2.5304	0.0297	8.5366
0.8	1.6	CRPSO-PID	0.0001	2.1605	0.0317	6.6629
		HPSO2-PID	0.0001	2.3425	0.0307	7.5483
		HTPSO-PID	0.0001	2.5347	0.0298	8.5508
0.9	1.4	CRPSO-PID	0.0001	1.8400	0.0295	5.5347
		HPSO2-PID	0.0001	2.1450	0.0294	6.7580
		HTPSO-PID	0.0001	2.5147	0.0277	8.6270
1.0	2.0	CRPSO-PID	0.0001	2.0400	0.0250	6.7616
		HPSO2-PID	0.0001	2.4100	0.0247	8.4472
		HTPSO-PID	0.0001	2.5792	0.0237	9.4326

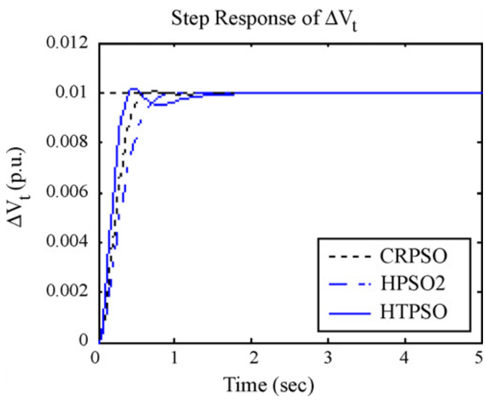

Fig. 13. Comparison of step response of incremental change in terminal voltage for PID controlled AVR system ($K_g = 0.9$, $\tau_g = 1.2$).

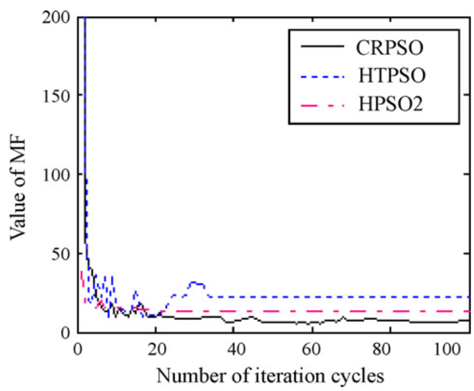

Fig. 14. Comparison of convergence profile of CRPSO with HPSO2 and HTPSO ($K_g = 0.9$, $\tau_g = 1.2$).

(vii) *Time of optimization*: As shown in Table 2 due to less computational burden CRPSO based optimization takes lesser time of optimization as compared to GA for same number of maximum iteration cycles. This is so because CRPSO does not involve any selection, crossover, etc. HPSO1, HPSO2 also involve selection like GA so that their times of execution are also less than those of CRPSO but better than GA. It is so as GA involves 100% crossover and mutation in the whole population. HTPSO takes much more time of optimization because selection of better strings involve very rigorous, extensive Taguchi selection unlike ordinary selection ratio based selection in GA, HPSO1 or HPSO2.

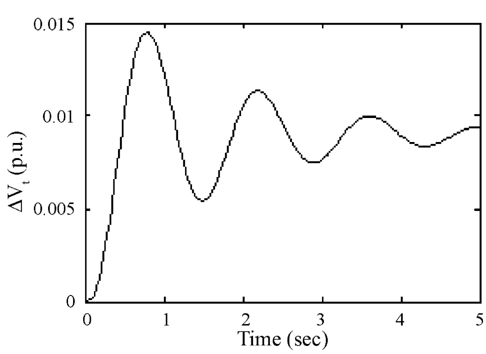

Fig. 15. Step response of incremental change in terminal voltage of an AVR system without PID controller.

(viii) *AVR without PID-controller*: Fig. 15 shows the step response of incremental change in terminal voltage of the system without inclusion of PID-controller. Oscillatory transient response with non-zero settling error is observed.

9. Conclusion

For tuning of PID controller gains with off-line, as well as, nominal input conditions, this paper represents a novel evolutionary search technique. It is the Craziness based Particle Swarm Optimization with lot of random probability based factors. For on-line input conditions, Sugeno fuzzy logic has been used. Binary coded genetic algorithm has also been considered for the sake of comparison. GA is less robust and gives suboptimal higher result. The step response of the incremental change in terminal voltage for the optimal gains (as calculated by adopting craziness based particle swarm optimization/genetic algorithm) of the PID controller has been plotted. Validity of the analytical results obtained has been established with the help of MATLAB-SIMULINK software. Better quality solution of step response of terminal voltage with less computational effort has been obtained in CRPSO-SFL based PID controller. With reference to the authors' previous work on other PSO algorithms, comparison of CRPSO with those algorithms is also done to establish the superiority of optimization as compared to those obtained by other PSO algorithms. Other recent PSO approaches have also been contrasted to highlight the potential benefit of the proposed PSO algorithm. Craziness based particle swarm optimization with on-line application of Sugeno fuzzy logic has been effectively applied for on-line tuning of PID controller for automatic voltage regulator.

Appendix A. List of symbols

$c1$	positive constant representing cognitive learning rate
$c2$	positive constant representing social learning rate
gBest	group best
K_a	amplifier gain
K_d	derivative gain of PID controller
K_e	exciter gain
K_g	generator gain
K_i	integral gain of PID controller
K_p	proportional gain of PID controller
K_s	sensor gain
max_dv	maximum derivative of change in terminal voltage
MF	misfittness function
o_{sh}	maximum overshoot of change in terminal voltage
pBest$_i$	personal best of ith particle
P_{craz}	a predefined craziness probability
$r1$, $r2$, $r3$ and $r4$	random numbers in the interval [0,1]
t_{st}	settling time of change in terminal voltage (s)
$v_i^{craziness}$	a random number chosen uniformly in the interval $[v_i^{min}, v_i^{max}]$

v_i^k current velocity of ith particle at kth iteration
v_i^{k+1} modified velocity of ith particle
$v_i^{\max}$ maximum velocity of the ith particle
$v_i^{\min}$ minimum velocity of the ith particle
ΔV_{ref} incremental change in reference voltage
ΔV_{t} incremental change in terminal voltage
x_i^k current position of ith particle at kth iteration
x_i^{k+1} modified position of ith particle

τ_{a} amplifier time constant (s)
τ_{e} exciter time constant (s)
τ_{g} generator time constant (s)
τ_{s} sensor time constant (s)

References

[1] Q.H. Wu, B.W. Hogg, G.W. Irwin, A neural network regulator for turbogenerators, IEEE Trans. Neural Networks 3 (1992) 95–100.
[2] S.H. Qin, G. Borders, A multi-region fuzzy logic controller of nonlinear process control, IEEE Trans. Fuzzy Syst. 2 (1994) 74–81.
[3] J. Kennedy, R.C. Eberhart, Particle swarm optimization, in: Proceedings of IEEE International Conference on Neural Networks, Perth, Australia, 1995, pp. 1942–1948.
[4] G.K.I. Mann, B.-G. Hu, R.G. Gosine, Analysis of direction fuzzy PID controller structures, IEEE Trans. Syst., Man, Cybern. Part B 29 (3) (1999) 371–388.
[5] R.A. Krohling, J.P. Rey, Design of optimal disturbance rejection PID controllers using genetic algorithm, IEEE Trans. Evol. Comput. 5 (February) (2001) 78–82.
[6] R.C. Eberhart, Y. Shi, Particle swarm optimization: developments, applications and resources, in: Evolutionary Computation Proceedings of the 2001 Congress, vol. 1, May 2001, pp. 81–86.
[7] H. Yoshida, K. Kawata, Y. Fukuyama, A particle swarm optimization for reactive power and voltage control considering voltage security assessment, in: Proceedings of IEEE Power Engineering Society Winter Meeting, vol. 2, 2001, pp. 815–820.
[8] G.K. Venayagamoorthy, R.G. Harley, D.C. Wunsch, Implementation of adaptive critic-based neurocontrollers for turbogenerators in a multimachine power system, IEEE Trans. Neural Networks 14 (September (5)) (2003) 1047–1064.
[9] S.P. Ghoshal, S. Goswami, Application of GA based optimal integral gains in fuzzy based active power-frequency control of non-reheat and reheat thermal generating systems, Electr. Power Syst. Res. 67 (2003) 79–88.
[10] S.P. Ghoshal, Optimization of PID gains by particle swarm optimization in fuzzy based automatic generation control, Electr. Power Syst. Res. 72 (2004) 203–212.
[11] Z.L. Gaing, A particle swarm optimization approach for optimum design of PID controller in AVR system, IEEE Trans. Energy Convers. 19 (June (2)) (2004) 384–391.
[12] S.L. Ho, S. Yang, G. Ni, E.W.C. Lo, H.C. Wong, A particle swarm optimization-based method for multiobjective design optimizations, IEEE Trans. Magnetics 41 (May (5)) (2005) 1756–1759.
[13] R. Roy, S.P. Ghoshal, Evolutionary computation based optimization in fuzzy automatic generation control, in: IEEE-Power Engineering Society Conference, Paper No. 88, New Delhi, India, April, 2006.