Swarm and Immune System Based Intelligence Techniques for Optimal Design of Synchronous Generator

V.P. Sakthivel

Assistant Professor
Department of Electrical and Electronics Engineering
Government College of Engineering, Dharmapuri – 636704, India.

Abstract: In this work, swarm and immune system based techniques such as particle swarm optimization (PSO) and artificial immune algorithm (AIA) are applied for optimizing the synchronous generator design (SGD). PSO and AIA have been proven to be effective in solving many real multi constrained optimization problem in different domains. SGD problems are complex and non linear in nature with equality and in equality constraints. Two different SGD problems are used to investigate the effectiveness of the proposal. Comparing with the conventional techniques, the current proposal is found to perform better in terms of computation time. Considering the quality of the solution obtained, AIA seems to be promising alternative approach for solving the SGD problems.

***Index Terms* - Artificial immune algorithm, efficiency optimization, particle swarm optimization, Synchronous generator design.**

I. INTRODUCTION

The designer, aiming at a design which will meet specific requirements laid out by the customers as well as national standards bases his initial design on past experience, analysis the performance of the initial design and makes the necessary changes to meet the specifications. In recent years' digital computers are being widely used in doing the above process. In the optimization problem, the objective is to get a design having minimum material cost, minimum weight or an optimum performance feature like maximum efficiency. The optimization methods were used to make a detailed analysis of design and for obtaining a fairly large number of alternator designs.

The design procedure for the three phase hetero polar type of inductor alternator has been worked out [1] and the stator leakage reactance, which is an important parameter in the design and operation of such alternators have been reviewed. Spooned and Williamson [2] designed and constructed the multi pole radial flux permanent magnet test machines for use as a direct coupled generator in wind turbines. A new technique for the optimal design of the surface permanent magnet synchronous motor considering the parameter correction of synchronous reactance has been presented [3]. The advanced immune algorithm was used in the optimization procedure. The design of outer rotor (the positions of the rotor and stator are exchanged) radial flux permanent magnet multi polar low speed directly coupled wind power convertor for standalone applications has been presented [4]. A 3KVA, 28Vpermanent magnet brushless alternator for light combat aircraft has been designed and analyzed [5]. An analytical algorithm was developed for the permanent magnet brushless alternator and the finite element analysis has been carried out for refining the design and performance [6].

Over the past few years, in order to solve the machine design problem, many salient methods have been developed such as genetic algorithm [7], evolutionary algorithm [8], simulated annealing method [9] and neural network approaches [10].

PSO is a population based swarm intelligence algorithm that shares many similarities with evolutionary computation techniques. The field of swarm intelligence is an emerging research area that presents features of self organization and cooperation principles among group members [11-14].

Recently, a new heuristic algorithm, the so called immune algorithm (IA) has been developed to solve the complex optimization problems [15]. Although the immune algorithm is similar to the genetic algorithm, the IA differs from GA in the memory education and production system. For this reason, some researchers used IA for various applications and obtained a better solution [16, 17].

In this work, the prospect of swarm and immune system based intelligent techniques namely PSO and AIA in solving SGD problems have been studied. Performance comparisons with conventional techniques demonstrate the effectiveness of swarm and immune system based intelligent techniques in solving SGD problems.

The paper is organized as follows. The next section of the paper presents the formulation of SGD problem as a constrained optimization problem. Section 3 contains a brief overview of swarm and immune system based intelligent techniques. Section 4 presents the implementation of intelligent techniques for solving SGD problems. Section 5 reports the simulation results comparing with the conventional methodologies. The paper ends in Section 6 with conclusions.

II. SGD PROBLEM FORMULATION

The design optimization problem of an alternator can be expressed as a general non-linear programming, as follows:

Optimize $F(X) = f(x_1, x_2 \ldots . x_n)$

Subject to $Gj(X) \geq b_j$, j= 1, 2, 3 ……., m

F(X) is the objective function in 'n' variables and there are 'm' inequality constraints, which may be explicit, like bounds on the design variables or implicit constraints like those on performance criteria. A solution set (X) for the equation (1), that satisfies the constraints is a feasible set, and the best feasible set (X) is the optimum.

A. Design Variables

The following were chosen as the principle design variables for the optimization:
Diameter of air gap (x_1)
Length of core (x_2)
Width of stator slot (x_3)
Depth of stator slot(x_4)
Cross sectional area of stator conductor (x_5)
Number of turns per phase (x_6)
Number of stator slots (x_7)
Depth of yoke (x_8)
Depth of core (x_9)
Length of air gap (x_{10})

The remaining design parameters can be expressed either in terms of these variables or assigned values for a particular frame size. The upper and lower limits of design parameters are given in Appendix 1.

B. Objective Function

The efficiency of the synchronous generator is considered as the objective function for design optimization. The design optimization aimed at energy conservation in synchronous generator is invariably related to efficiency as the objective function. The synchronous generator efficiency is improved by minimizing the losses. This efficiency increase will be beneficial to consumers. The objective function for efficiency maximization is as follows:

$$F(x) = \text{Efficiency} + (\text{Sum of the violated constrants}) \times \text{penality factor} \quad (1)$$

C. Constraints

The constraints for an alternator are:
Length of air gap
Regulation
Efficiency
Temperature rise
Maximum magnetic flux density in teeth

The constraints limits are given in Appendix A1.

III. OVERVIEW OF SWARM AND IMMUNE SYSTEM BASED INTELLIGENT TECHNIQUES

A. Particle swarm Optimization

PSO is a stochastic global optimization technique which uses swarming behaviors observed in flock of birds, school of fishes or swarm of bees, in which the intelligence is emerged. It was developed in 1995, by James Kennedy and Russell Eberhart and uses a number of particles that constitute a swarm moving around in an N- dimensional search space looking for the best solution. Each particle in PSO keep track of its co-ordinates in the problem space, which are associated with the best solution (best fitness) it has achieved so far. This value is called pbest. Another "best" value that is tracked by the global version of the particle swarm optimizer is the overall best value and its location obtained so far by any particle in the swarm. This location is called gbest.

Each particle tries to modify its position using the following information.

- The current positions
- The current velocities
- The distance between the current position and pbest
- The distance between the current position and gbest

The PSO concept consists of changing the velocity of each particle toward its pbest and gbest locations at each iteration. Acceleration is weighted by a random term, with separate random number is generated for acceleration toward pbest and gbest locations. The general flowchart of PSO is shown in Fig. 1. Some of the principles and advantages of PSO are as follows,

- PSO is based on the principle that the probability of finding a better minimum near the so far found minimum is more than other places. The particles (solutions) are therefore diverted toward searching around the found minimum.
- PSO is a history based algorithm such that in each step, particles use their own behaviour associated with the previous iterations.
- Compared to the other evolutionary optimization algorithms, such as GA, PSO is easy to implement and, only few parameters to be adjusted. Therefore, the computation time is less and it requires less memory.
- PSO is applicable to mixed integer non linear optimization problems with both continuous and discrete variables.

Let X and V denote the particle's position and its corresponding velocity in search space respectively. At iteration K, each particle i has its position defined by $X_i^K = [X_{i,1}, X_{i,2} \ldots X_{i,N}]$ and a velocity is defined as $V_i^K = [V_{i,1}, V_{i,2} \ldots V_{i,N}]$ in search space N. Velocity and position of each particle in the next iteration can be calculated as

$$V_{i,n}^{k+1} = W \times V_{i,n}^{k} + C_1 \times rand_1 \times (pbest_{i,n} - X_{i,n}^{k}) + C_2 \times rand_2 \times (gbest_n - X_{i,n}^{k}) \quad (2)$$
$$i = 1, 2 \ldots\ldots m$$

$n = 1, 2 \ldots\ldots N$

$$X_{i,n}^{k+1} = X_{i,n}^{k} + V_{i,n}^{k+1} \quad \text{if } X_{min,i,n} \le X_{i}^{k+1} \le X_{max\,i,n}$$
$$= X_{min\,i,n} \quad \text{if } X_{i,n}^{k+1} < X_{min\,i,n}$$
$$= X_{max\,i,n} \quad \text{if } X_{i,n}^{k+1} > X_{max\,i,n} \tag{3}$$

Where,	m	number of particles in the swarm
	N	number of dimensions in a particle
	K	pointer of iterations (generations)
	$V_{i,n}^{k}$	velocity of particle i at iteration k
	W	weighting factor
	C_j	acceleration factor
	$rand_j$	random number between 0 and 1
	$X_{i,n}^{k}$	current position of particle i at iteration k
	$pbest_i$	personal best of particle i
	gbest	global best of the group

In the above procedures, the convergence speed of each particle could be influenced by the parameters of acceleration factors C_1 and C_2. The optimization process will modify the position slowly, if the value of C_j is chosen to be very low. On the other hand, the optimization process can become unstable, if the value of C_j is chosen to be very high. The first term of formula (2) the initial velocity of particle which reflects the memory behavior of particle; the second term "cognition part" which represents the private thinking of the particle itself; the third part is the "social" part which shows the particles behavior stem from the experience of other particles in the population.

The following weighting function is usually used in (2)

$$W = W_{max} - (W_{max} - W_{min}) \times Iter / Iter_{max} \tag{4}$$

Where W_{max} and W_{min} are initial and final weight respectively, Iter is current iteration number and $Iter_{max}$ is maximum iteration number.

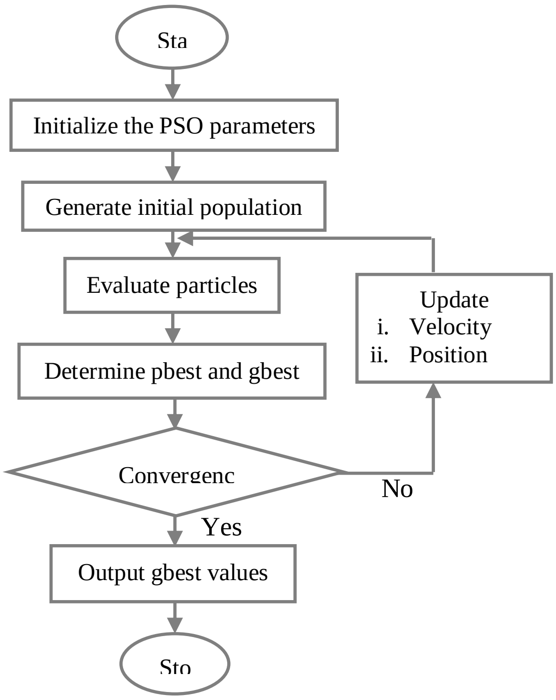

Fig. 1. General flow chart of PSO

The model using (4) is called 'inertia weights approach (IWA)'. The inertia weight is employed to control the impact of the previous history of velocities on the current velocity. Thus the parameter W regulates the trade-off between the global and the local exploration abilities of the swarm. A large inertia weight facilitates exploration, while a small one tends to facilitate exploitation.

B. Artificial Immune Algorithm

Artificial immune algorithm (AIA) has the desirable characteristics as an optimization tool and offers significant advantages over the traditional methods. They are inherently robust and have been shown to efficiently search the large solution space containing discrete and continuous parameters and non-linear constraints, without being trapped in local minima [15]. The AIA may be used to solve a combinatorial optimization problem. In the AIA, *antigen* represents the problem to be solved. An *antibody* set is generated where each member represents a candidate solution. Also, *affinity* is the fit of an antibody to the antigen. In the AIA, the role of antibody lies in eliminating the antigen, while the *lymphocyte* helps to produce the antibody.

In the immune system, there are two kind of lymphocyte: T and B; where each of them has its own function. The T lymphocytes develop in bone marrow and travel to *thymus* to mature. The B lymphocytes develop and mature within the bone marrow. The main purpose of the immune system is to recognize all cells within the body and categorize those cells as self or non-self. Self or non-self antigens are those cells that originally belong to the organism and are harmless to its functioning. The disease-causing elements are known as non-self. Both B-cells and T-cells have receptors that are responsible for recognizing antigenic patterns by different functions. The attraction between an antigen and a receptor cell (or degree of binding) is known as affinity. To handle the infection successfully and effectively, both B-cells and T-cells may be required. After successful recognition, cells capable of binding with non-self antigens are cloned.

In the AIA, the elements of the population undergo mutations resulting in a subpopulation of cells that are slightly different. Since the mutation rate is high, this mutation is called hyper mutation. The principle of AIA can be summarized in Fig. 2.

In the first step, N antibodies are generated randomly and evaluated using a suitable fitness function. While the fitness of all antibodies is known, new population is generated through three steps: replacement, cloning and hyper mutation. These three steps maintain the diversity and help the algorithm to expand the search space. In the replacement step, the low antibodies are replaced. Those with the highest affinity are selected to proliferate by cloning, where the cloning rate of each immune cell is proportional to its affinity. If the high affinity antibody has not been cloned, hyper mutation is applied where the mutation rate for each immune cell is inversely proportional to its affinity [16-17]. When the new population is generated, AIA continues with repeated evaluation of the antibodies through selection, clonal proliferation and hyper mutation until the termination criterion is met.

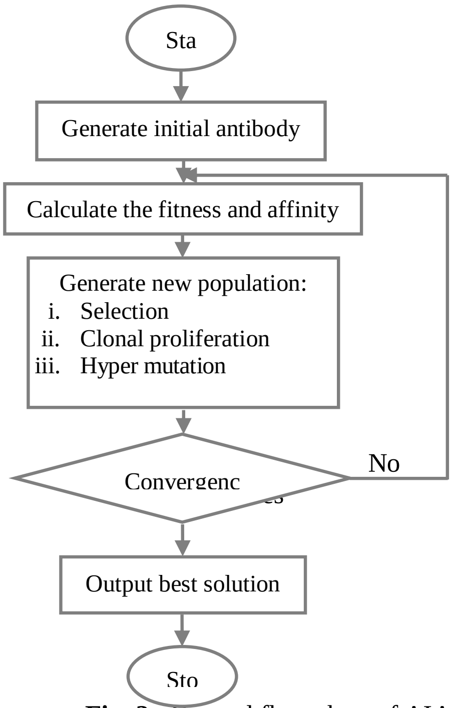

Fig. 2. General flow chart of AIA

Compared with the genetic algorithm and the other evolutionary algorithms, the AIA provides the following advantages:

- The memory cell is maintained without applying operators such as recombination, selection etc. to the population.
- AIA operates on the memory cell and so guarantees the fast convergence.
- The diversity of the immune system is embedded by means of affinity calculation.
- AIA will exercise immunity adjustment according to the antigen concentration, so as to improve the multiplicity of the antigen. As a result the optimal solution search process will not fall into the trap of locally optimal solutions.

IV. Implementation of Intelligent Techniques for Solving SGD Problems

In this paper, the process to solve constrained SGD problems using intelligent techniques was developed to obtain efficiently a high quality solution. The intelligent techniques are used to determine the optimal design variables of synchronous generator, thus maximizing the efficiency of the generator.

A. Implementation of PSO for SGD Problems

The PSO procedure for optimizing the SDD problem can be summarized as follows:

Step 1. Get the specifications, lower and upper limits of the synchronous generator design variables.
Step 2. Initialize parameters W_{max}, W_{min}, C_1, C_2 and $Iter_{max}$.
Step 3. Generate initial population of ‘m’ particles with random positions and velocities.
Step 4. *Calculate fitness*: Evaluate the objective value of current particle Eq. (1).
Step 5. *Update personal best*: Compare the fitness value of each particle with its pbests. If the current value is better than the previous pbest, then set the pbest value to the current value.
Step 6. *Update global best* : Compare the fitness value of each particle with the gbest. If the current value is better than the previous gbest, set gbest to the current particle’s value.
Step 7. *Update weight:* Calculate weight W^{k+1} using Eq. (4).
Step 8. *Update velocities*: Calculate velocities V^{k+1} using Eq. (2).
Step 9. *Update positions*: Calculate positions X^{k+1} using Eq. (3).
Step 10. Return to step (4) until the current iteration reaches the maximum iteration number.
Step 11. Output the optimal solution in the last iteration.

B. Implementation of AIA for SGD Problems

The objective function and the constraints of the SGD problem are regarded as the antigen and the feasible solutions are regarded as the antibodies. The antibody is composed of synchronous generator design variables. The implementation of AIA for the optimization of SGD problems can be described as follows:

Step 1. Input the specifications, lower and upper limits of the synchronous generator design variables.

Step 2. *Define antigen and antibody*: Antigen corresponds to the objective function and constraints and antibody corresponds to the feasible solution of the problem.

Step 3. Generate an initial population of strings (antibodies) randomly.

Step 4. Calculate objective value for each individual using Eq. (1).

Step 5. *Selection:* Select the best N individual with lowest fitness value (minimization problem).

Step 6. *Clonal proliferation:* Clone (reproduce) these best N individuals (antibodies). Note that the clone size for each selected individual is an increasing function of the affinity with the antigen. In other words, the number of posterity of each antibody is proportional to their fitness values, i.e., the lowest the fitness, the larger the clone size.

Step 7. *Hyper mutation*: Mutate the population of clones, proportional to the affinity of the antibody with the antigen.

Step 8. Calculate the new fitness values of these new individuals.

Step 9. *Tournament selection:* Select those individuals, who are superior to the individuals in the memory set, and then the superior individuals replace the inferior individuals in the memory set. While the memory set is updated, the individuals will be eliminated while their structures are too similar. So the individuals in the memory set can kept with diversity.

Step 10. Return to step (4) until the current iteration reaches the maximum iteration number.

Step 11. Output the optimal design variables of synchronous generator.

IV. SIMULATION RESULTS AND DISCUSSIONS

In this work, swarm and immune system based intelligent techniques namely PSO and AIA for solving the SGD problems are presented. The purpose of SGD problem is to maximize the cost objective function while satisfying performance indices constraints. PSO and AIA are investigated to determine the optimal design variables of synchronous generator. Simulation results of two different synchronous generators are implemented to indicate the robustness of intelligent techniques.

Intelligent techniques are employed to design 500KVA and 3000KVA synchronous generators to assure its optimization efficiency, where the objective function is limited by the design variables and performance indices constraints. The specifications of the synchronous generators are given in Appendix A2.

The performance of PSO and AIA are compared with conventional technique. Simulations were done under the Matlab environment. Parameters of PSO and AIA are as follows:

- Population size = 20
- Maximum number of iterations = 100
- Acceleration factors C_1 and C_2 = 2
- Initial inertia weight (W_{max}) = 0.9
- Final inertia weight (W_{min})= 0.4

A. Case Study 1

This case designs a 500kVA synchronous generator. Table 1 summarizes the results of designing the synchronous generator using intelligent techniques and compares the conventional approach. As shown from the Table, AIA donates superior result in terms of efficiency. Table 2 lists the statistical comparison between PSO and AIA techniques in terms of the best, worst and average values, percentage deviation and computational (CPU) time through 50 trials. It is clear that the efficiency obtained by the proposed AIA is better than other algorithms. Fig. 3 shows the obtained efficiency for each algorithm. On the other hand, a graph for convergence rate of the objective function is given in Fig. 4. It can be seen that the objective function is stabilized after 9 iterations. Also, the mean CPU time of AIA is the shortest one.

Table 1
Design Comparison for 500 KVA Generator

Design variables and performance indices	Conventional method	PSO	AIA
Diameter of the stator (m)	0.8	0.82	0.82
Gross length of armature (m)	0.5	0.385	0.424
Width of the stator (m)	0.013	0.01	0.01
Depth of the stator (m)	0.073	0.04	0.047
Area of stator conductors (mm2)	40	67.45	52
Depth of yoke (m)	0.088	0.115	0.126
No. of turns/phase	165	165	164
No. of slots in stator	78	80	80
Length of air gap (m)	0.005	0.005	0.005
Outer diameter of stator (m)	1.14	1.16	1.59
Peak flux density in stator teeth (wb/m2)	0.849	0.84	0.872
Peak flux density in air gap (wb/m2)	1.51	1.35	1.33
Full load field current (A)	31.64	27.81	29.6
Weight of iron (Kg)	2736.74	2199.2	2356.43
Weight of copper (Kg)	1455.72	1169.7	1249.26
Temperature rise(° C)	54.3	50	49.45
Voltage regulation (%)	26.53	26.235	25.23
Efficiency (%)	95.37	97.9	98.02

Table 2
Statistical Comparison between PSO and AIA for 500 KVA Generator

Compared value	PSO	AIA
Best value	97.9	98.02
Worst value	96.2	96.85
Average value	97.05	97.436
PD (%)	1.7	1.2
CPU time (sec)	2.85	2.729

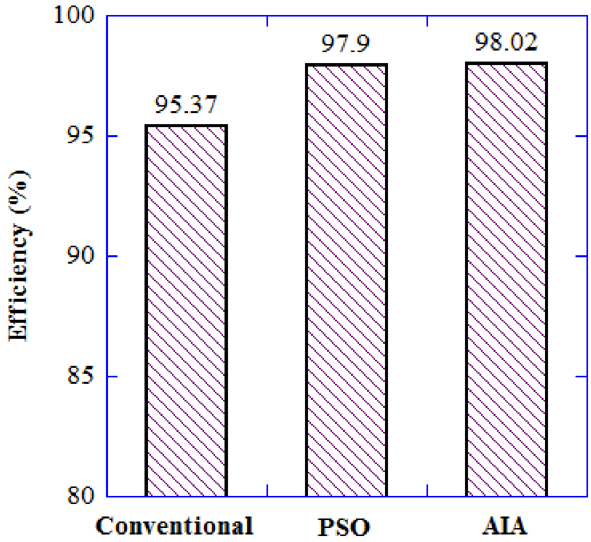

Fig. 3. Efficiencies obtained by various algorithms for Case 1

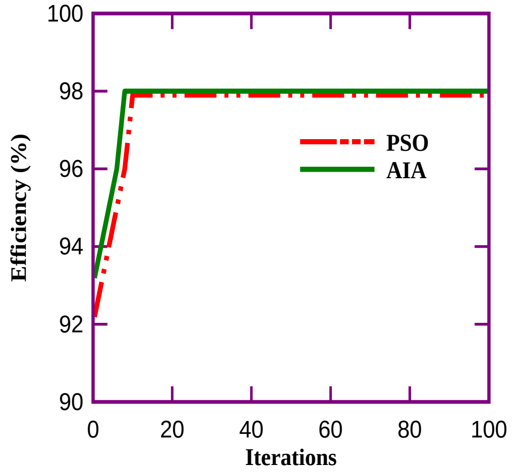

Fig. 4. Convergence behavior of AIA for Case 1

B. Case Study 2

This case considers 3000KVA generators as a higher capacity machine to confirm the superiority of AIA over other algorithms in reaching optimum solution. Table 3 outlines the optimal design variables and their performance indices for each algorithm.

It can be noticed that the suggested AIA achieves higher efficiency compared with other algorithms while satisfying the constraints of generators. Thus, AIA performs better than these algorithms in terms of efficiency even for higher capacity machines. In Table 4, the statistical comparison of PSO and AIA methods are presented. It is evident that the proposed AIA gives better results in terms of maximum efficiency and smaller CPU time than PSO algorithm. Fig. 5 shows the objective values associated with PSO and AIA. The superiority of the proposed algorithm in increasing the objective can be verified as shown in Fig. 6.

Table 3

Design Comparison for 3000 KVA Generator

Design variables and performance indices	Conventional method	PSO	AIA
Diameter of the stator (m)	3.2	2.6	2.6
Gross length of armature (m)	0.44	.399	0.402
Width of the stator (m)	0.013	0.015	0.015
Depth of the stator (m)	0.073	0.05	0.05
Area of stator conductors (mm2)	70.2	80	80
Depth of yoke (m)	0.101	0.111	0.104
No. of turns/phase	218	292	286
No. of slots in stator	314	254	254
Length of air gap (m)	0.0061	0.0083	0.0081
Outer diameter of stator (m)	3.58	2.95	2.94
Peak flux density in stator teeth (wb/m2)	0.853	0.839	0.84
Peak flux density in air gap (wb/m2)	1.52	1.786	1.694
Full load field current (A)	225	213.44	216.45
Weight of iron (Kg)	24331.54	15021.35	15480.54
Weight of copper (Kg)	12942.3	7990.08	8240.94
Temperature rise(° C)	39..77	39.73	39.52
Voltage regulation (%)	28.42	28.74	28.46
Efficiency (%)	95.33	97.4	97.76

Table 4

Statistical Comparison between PSO and AIA for 3000 KVA Generator

Compared value	PSO	AIA
Best value	97.4	97.76
Worst value	95.9	96.45
Average value	96.65	97.104
PD (%)	1.5	1.42
CPU time (sec)	2.8	2.69

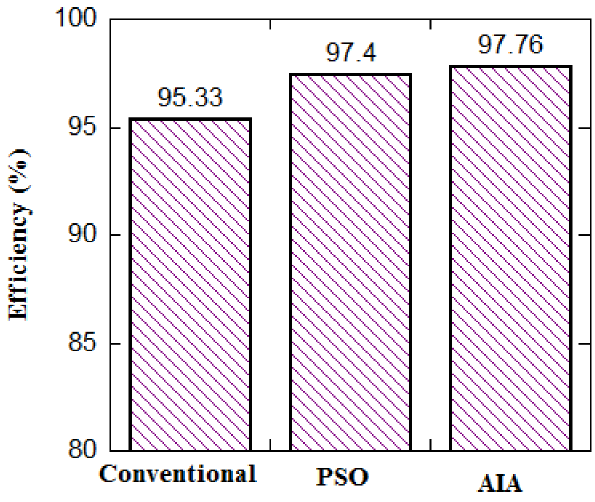

Fig. 5. Efficiencies obtained by various algorithms for Case 2

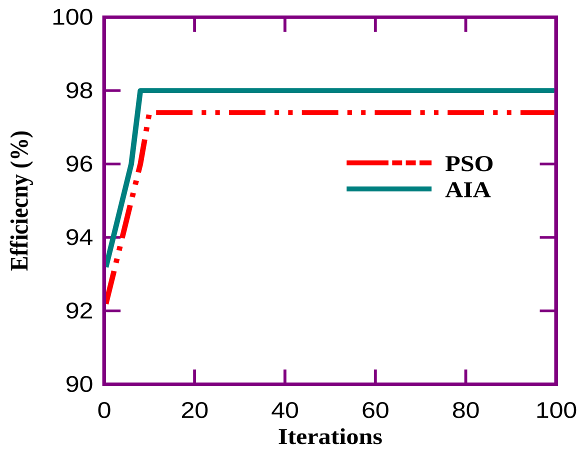

Fig. 6. Convergence behavior of AIA for Case 2

C. Comparison and Discussion

The superiority of the proposed AIA is investigated here by comparison with PSO algorithms in terms of convergence property and computation efficiency.

1. Convergence property

From Figs. 4 and 6, one can get that the descending speeds at the beginning are high; this indicates the high convergence of the proposed algorithm based on evolution search. AIA can be convergent quickly and get the optimum results in very small iteration numbers. It is confirmed to have a good convergence property.

2. Computation efficiency

As seen in Tables 2 and 4, CPU times of the proposed AIA are smaller than PSO algorithm Thus, it can get better computation efficiency than PSO algorithm.

V. CONCLUSIONS

The swarm and immune system based intelligent techniques namely PSO and AIA have been successfully implemented to solve SGD problems. The AIA algorithm has the ability to find the better quality solution and has better convergence characteristics, computational efficiency, and robustness. Many nonlinear performance characteristics of the synchronous generator have been considered. It is clear from the results obtained by different trials that the proposed AIA method has good convergence property and can avoid the shortcoming of premature convergence of other optimization techniques to obtain better quality solution. Due to these properties, the AIA method in the future can be tried for solution of complex power system problems in the search of better quality results.

APPENDICES

A1. Design parameters with their limits

Design parameters	Synchronous Generator 1		Synchronous Generator 2	
	Lower limit	Upper limit	Lower limit	Upper limit
Stator diameter (m)	0.775	1	2.55	3.25
Gross length of armature (m)	0.375	1.45	0.35	0.425
Width of the stator (mm)	10	20	10	15
Depth of the stator (mm)	40	60	50	75
Area of stator conductors (mm^2)	50	75	60	80
Depth of yoke (m)	0.06	0.12	0.08	0.12
No. of turns / phase	98	168	174	330
No. of stator slots	68	90	246	336
Length of air gap (mm)	3	6	5	9
Outer diameter of stator (m)	0.975	1.36	3	5.48

A2. Specifications of Synchronous Generator

Specifications	Synchronous Generator 1	Synchronous Generator 2
Capacity	500 KVA	3000 KVA
Voltage	3.3 KV	6.6KV
Frquency	50	50
Number of poles	10	32

REFERENCES

[1] K. Pradipta, Dash and K. Amulya Das Gupta, "Design Aspects of Three Phase Inductor Alternators," *IEEE Transactions on Power Apparatus and Systems*. Vol. 88, no. 11, pp. 1718-24, 1989.

[2] E. Spooned, and A. Williamson, "Direct Coupled Permanent Magnet Generators for Wind Turbine," *IEE Proceeding of Electric Power Applications*. Vol.143, no. 1, pp. 1-8, 1996.

[3] Jang Sung Chun and Joong Suk Yoon, "Optimum Design of Synchronous Motor with Parameter Correction Using Immune Algorithm," *IEEE Transactions on Energy Conversion* vol. 14, no. 3, pp. 610-615, 1999.

[4] J. Chen, C. Nayar, and L. Xu, "Design and Finite Element Analysis of an Outer Rotor Permanent Magnet Generator for Directly Coupled Wind Turbine Applications," Proceedings *of the IEEE Transactions on Magnetics*, vol. 36, no. 5, pp. 3802-3809, 2000.

[5] M. Comanescu, A. Keyhani, and Min Dai, "Design and Analysis of Permanent Magnet Generator for Automotive Applications," *IEEE Transactions on Energy Conversion,* vol. 18, no. 1, 2003.

[6] Bhim Singh and Jally Ravi, "Design and Analysis of a 3 KVA, 28V Permanent Magnet Brushless Alternator for Light Combat Aircraft," *IEEE Transactions on Industrial Applications*. 39, 1665 -1669, 2006.

[7] Menmet Cunkas and Ramazan Akkaya, "Design Optimization of Induction Motors By Genetic Algorithm and Comparison with Existing Motor," Mathematical *and computational Applications*, vol. 11, no. 3, 2006.

[8] S. Subramanian, and R. Bhuvaneswari, "Evolutionary Programming Based Determination of Induction Motor Efficiency," Electric *Power Components and Systems*, vol. 34, no. 5, pp. 565-576, 2006.

[9] S. Subramanian, and R. Bhuvaneswari, "Multi-Objective Optimal Design of Three-Phase Induction Motor Using Simulated Annealing Technique," *The International Journal of Computation and Mathematics in Electrical and Electronics Engineering-COMPEL*, vol. 24, no. 4, pp. 1415-1427, 2005.

[10] G.T. Bellarmine, R. Bhuvaneswari, and S. Subramanian, "Radial Basis Function Network Based Design Optimization of Induction Motor," *Proceedings of IEEE SOUTHEASTCON*, Memphis, Tennessee, March 30-April 2006, USA. pp. 75-80.

[11] C. Eberhart and J. Kennedy, "Particle swarm optimization," *IEEE Int. Con. Neural Networks*, vol. 4 , pp. 1942-1947, 1995..

[12] Y. Shi and R.C. Eberhart, "A modified particle swarm optimizer," *IEEE Int. Con. Evolutionary Computations*, pp. 69-73, 1998.

[13] Y.Shi and R.C. Eberhart, "Empirical study of particle swarm optimization," *IEEE Int. Proc. Evolutionary Computations*, vol. 3, pp. 1945-1946, 1999.

[14] Z.L. Gaing, "Particle swarm optimization to solving the economic dispatch considering the generator constraints," *IEEE Transactions on Power Systems*, vol. 18, no. 1, pp. 1187-1195, 2003.

[15] J.D. Farmer, N.H. Packard and A.A. Perelson, "The immune system adaption and machine learning," *Physica D* 22, pp. 187-204, 1986.

[16] C.H. Lin, C.S. Chen, C.J. Wu and M.S. Kang, "Application of immune algorithm to optimal switching operation for distribution loss minimization and loading balance", *IEE proceedings on Generation Transmission and Distribution*, vol. 150, no. 2, pp. 183-189, 2003.

[17] K. Mori, M. Tsukiyama, and T. Fukuda, "Immune algorithm with searching diversity and its application to resource allocation problem", *Trans. Inst. Elect. Eng. Japan,* vol. 10, pp. 872-878, 1993.