International Journal of Performability Engineering Vol. 11, No. 1, January 2015, pp. 33-42.

Printed in India

A Particle Swarm Algorithm for Optimization of Complex System Reliability

SANGEETA PANT[1*], DHIRAJ ANAND[2] and AMAR KISHOR[3], SURAJ BHAN SINGH[4]

[1]*Department of Mathematics, University of Petroleum and Energy Studies, Dehradun, INDIA:* 248007
[2]*Department of Electrical and Electronics, GEU, Dehradun, INDIA:* 248002
[3]*Department of Electronics and Communication Sciences Unit, Indian Statistical Institute Kolkata , INDIA:* 700108.
[4]*Department of Mathematics, Statistics and Computer Science, G.B.Pant University of Agriculture and Technology, Pantnagar, INDIA:* 236145

(Received on January 28, 2014, revised on May 28, June 07 and July 26, 2014)

Abstract: In recent years, a broad class of stochastic metaheuristics, such as Tabu search, simulated annealing, genetic algorithm, particle swarm optimization, ant colony optimization etc. has been applied for reliability optimization problems. In this paper a particle swarm optimization algorithm is presented. Then, the performance of the proposed algorithm is tested on some complex engineering optimization problems. They are three well-known complex reliability optimization problems. Finally, the results are compared with those given by several well-known methods. Numerical experiments demonstrate that the proposed method is promising and the results obtained by proposed algorithm are either superior or comparable to the previously best known results presented in literature for reliability optimization of complex systems in terms of computation time as well as solution quality.

Keywords: *Reliability optimization, metaheuristics, particle swarm optimization, reliability allocation, redundancy allocation.*

1. Introduction

In the past five decades the problem of reliability optimization and redundancy allocation has been addressed in many studies. Important contributions have been devoted since 1970 [1-2] in order to cope with optimization problem arising in system reliability. The problem of reliability optimization has been widely treated by many authors. To list a few of them, Luss [3] optimized such problems by non linear integer programming procedure, Mohan and Shankar [4] applied random search technique to optimize complex system, Misra and Sharma [5] developed MIP-Technique to solve integer programming problems arising in system reliability design. Tillman *et al.* [6], Lad *et al.* [7], Chaturvedi and Misra [8] provided an excellent survey of earlier approaches applied to solve these problems. Reliability optimization problems are categorized into three typical problems according to the types of their decision variables: (i) reliability allocation, (ii) redundancy allocation, and (iii) reliability-redundancy allocation. From the viewpoint of mathematical programming, reliability allocation is a continuous nonlinear programming problem (NLP), redundancy allocation is a pure integer nonlinear programming problem (INLP), and reliability-redundancy allocation is a mixed integer nonlinear programming problem (MINLP). In general, achieving optimal reliability design is quite difficult because reliability optimization problems are NP-hard [9]. Further, solving such problems using

* Corresponding author's email: pant.sangeet@gmail.com

heuristics or exact algorithms is more difficult. This is because these optimization problems generate a very large search space, and searching for optimal solutions using exact methods or heuristics will necessarily be extremely time consuming. Therefore, metaheuristic algorithms are more suitable for solving reliability optimization problems. Recently, many metaheuristics [10-19] have been employed to solve reliability optimization problems. In this present paper, we applied a PSO to solve reliability optimization and redundancy allocation problems of complex systems. The main concept of PSO is based on the food searching behavior of birds flocking or fish schooling. When PSO is adopted to solve problems, each particle has its own location and velocity, which determine the flying direction and distance respectively. Comparing with other evolutionary approaches PSO has the following advantages [20-22]:

(i) less parameters (ii) easy implementation (iii) fast convergence.

These advantages are good for solving the reliability optimization problems because a population of particles in PSO can operate simultaneously so that the possibility of paralysis in the whole process can be reduced. The results obtained by using PSO approach are compared with the results obtained from other techniques in the literature. As reported, solutions obtained by the proposed method are better than or as well as the previously best-known solutions.

2. Particle Swarm Optimization

Particle swarm optimization (PSO) is a population-based search algorithm based on the simulation of the social behavior of birds within a flock [23-24]. PSO is initialized with randomly generated population of particles (initial swarm) and a random velocity is assigned to each particle that propagates the particle in search space towards optima over a number of iterations. Each particle has a memory remembering best position attained by it in the past, which is called personal best position (P_{best}). Each particle has its P_{best} and the particle with the best value (maximum or minimum according to the problem) of fitness is called global best particle (G_{best}). Suppose that the search space is D dimensional, the i^{th} particle of the population can be represented by a D-dimensional vector $X_i = (x_i^1, x_i^2, ..., x_i^D)^T$. The velocity of this particle can be represented by another D-dimensional vector $V_i = (v_i^1, v_i^2, ..., v_i^D)^T$ .The previously best visited position of i^{th} particle is denoted by $P_i = (p_i^1, p_i^2, ..., p_i^D)^T$ and the best particle in the swarm is denoted by $P_g = (p_g^1, p_g^2, ..., p_g^D)^T$. Particle changes its position and velocity according to the following equations:

$$V_{id}^{k+1} = wV_{id}^k + c_1 r_1 [p_{id}^k(t) - x_{id}(t)] + c_2 r_2 [p_g^k(t) - x_{id}^k(t)] \quad (1)$$

$$x_{id}^{k+1}(t+1) = x_{id}^k(t) + v_{id}^{k+1}(t+1) \quad (2)$$

where k = iteration number, d=1,2,3……, D; i= 1,2,3….N; N= swarm size, w = inertia weight, which controls the momentum of particle by weighing the contribution of previous velocity, c_1 and c_2 are positive constants called acceleration coefficients; r_1 and r_2 random numbers uniformly distributed between [0,1]. The particle swarm optimization algorithm which is applied presently to solve reliability optimization problem is similar to the one used by [25] for optimization of nonlinear taper but here we adopted dynamic changing acceleration coefficients as in [26]. These modifications can be mathematically represented by equations (3) and (4) respectively.

$$c_1(t) = c_{1i} + (c_{1f} - c_{1i}) * ITER / ITERMAX \quad (3)$$

$$c_2(t) = c_{2i} + (c_{2f} - c_{2i}) * ITER / ITERMAX \quad (4)$$

where, c_{1i} and c_{2i} are initial values and c_{1f} and c_{2f} are final values of c_1 and c_2 respectively, ITERMAX is maximum number of iterations and ITER is the current iteration. The large value of c_1 and small value of c_2 at beginning the particles are allowed to explore new regions in search space instead of chasing the global best particle. On the contrary with small value of c_1 and a large value c_2 allow particles to move towards the population best. Also we set the constant value of inertia weight instead of linearly decreasing it and vary c_1 and c_2 according to equations (3) and (4). Maximum velocity in each dimension is restricted to $V_{max} = \gamma * X_{max}$, where γ is a constant and can take any value between 0 and 1 (problem dependent). Also here we have restricted velocity and position of the particle according to equations (5) and (6) respectively, i. e.

$$V_{id}^{k+1} = \begin{cases} V_{max}, & \text{if } V_{id}^{k+1} > V_{max} \\ -V_{max}, & \text{if } V_{id}^{k+1} < -V_{max} \\ V_{id}^{k+1}, & \text{otherwise} \end{cases} \quad (5)$$

$$x_{id}^{k+1} = \begin{cases} X_{max}, & \text{if } x_{id}^{k+1} > X_{max} \\ X_{min}, & \text{if } x_{id}^{k+1} < X_{min} \\ x_{id}^{k+1}, & \text{otherwise} \end{cases} \quad (6)$$

3. Problem Description

Reliability engineers often need to work not only on series or parallel systems but also on the systems which are neither purely connected in series nor purely in parallel but may have mixed configuration. Such systems are called complex systems. To evaluate the performance of the proposed approach two reliability optimization problems, (i) reliability allocation and (ii) redundancy allocation, are considered.

3.1 Reliability Allocation Problems

Under this prism two case studies are considered. They are a life support system in space capsule and a complex bridge system.

3.1.1 Life Support System in Space Capsule

Figure1 shows the first complex system of the present study [6]. The system reliability R_s and system cost C_s of life support system in a space capsule are given by:

$$R_S = 1 - r_3\left[(1-r_1)(1-r_4)\right]^2 - (1-r_3)\left[1 - r_2\{1-(1-r_1)(1-r_4)\}\right]^2 \quad (7)$$

$$C_S = 2K_1 r_1^{\alpha_1} + 2K_2 r_2^{\alpha_2} + K_3 r_3^{\alpha_3} + 2K_4 r_4^{\alpha_4} \quad (8)$$

where, K_1=100, K_2=100, K_3=200, K_4=150and $\alpha_i = 0.6, \quad i = 1,2,3,4$.

This problem is continuous nonlinear optimization problem, where objective is to determine the minimum cost of system subject to the constraints on reliability of the system. The problem formulation is as follows:

$$\text{Minimize } C_S$$

subject to

$$0.5 \le r_i \le 1 \qquad i = 1, 2, 3, 4$$

$$0.9 \le R_S \le 1$$

where, R_i is i^{th} component's reliability.

3.1.2 Complex Bridge System

This problem considers the reliability optimization of complex bridge network [1,4] as shown in the Figure 2 with nonlinear constraint. The system reliability (R_s) and system cost (C_s) are given by

$$R_S = r_1 r_4 + r_2 r_5 + r_2 r_3 r_4 + r_1 r_3 r_5 + 2 r_1 r_2 r_3 r_4 r_5 - r_1 r_2 r_4 r_5 - r_1 r_2 r_3 r_4 - r_2 r_3 r_4 r_5 - r_1 r_2 r_3 r_5 - r_1 r_3 r_4 r_5 \quad (9)$$

$$C_S = \sum_{i=1}^{5} a_i \exp\left[\frac{b_i}{(1 - r_i)}\right] \quad (10)$$

The mathematical expression of the problem is

$$\text{Minimize } C_S$$

subject to

$$0 \le r_i \le 1, \qquad i = 1,2,3,4,5$$

$$0.99 \le R_s \le 1,$$

$a_i = 1$ and $b_i = 0.0003$, for $i = 1,2,3,4,5$.

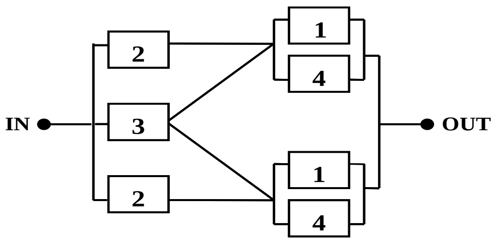

Figure 1: Life-Support System in a Space Capsule

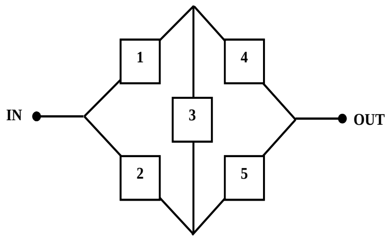

Figure 2: Complex Bridge System

3.2 Redundancy Allocation Problems

Here a mixed series parallel system [11] is considered. The block diagram of a *n*-stage series parallel system is presented in Figure 3.

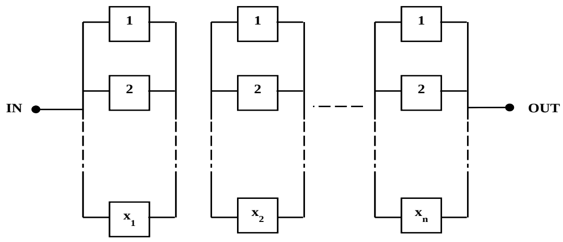

Figure 3: Block Diagram of n-Stage Series Parallel System

This problem is an integer non linear programming problem where objective is to find the optimal number of redundancies $x_i, i = 1,2,\ldots,n$ in a multistage mixed system in order to achieve the maximum reliability under the restriction on cost, weight and volume.

Two different cases for two different values of *n (n= 5 and 15)* are considered. As the value of *n* (no. of stages) increased the complexity of the system will increase as well.

3.2.1. Case 1- Mixed Series Parallel System with Five Parallel Units

The mathematical formulation of the problem is:
Find the optimal $x_i, i = 1, 2, 3, 4, 5$ to maximize

$$R_S = \prod_{i=1}^{5} \left(1 - \bar{r}_i^{\,x_i}\right)$$

subject to

$$V_S = \sum_{i=1}^{5} P_i x_i^2 \le 110$$

$$C_S = \sum_{i=1}^{5} C_i \left[x_i + \exp\left(\frac{x_i}{4}\right) \right] \le 175$$

$$W_S = \sum_{i=1}^{5} W_i x_i \left[\exp\left(\frac{1}{4x_i}\right) \right] \le 200$$

The relevant data for this problem is given in Table 1.

3.2.2 Case 2- Mixed Series Parallel System with Fifteen Parallel Units

The problem formulation is as follow:
Find the optimal $x_i, i = 1, 2, 3, ..., 15$ which maximizes

$$R_S = \prod_{i=1}^{15} [1 - (1 - r_i)^{x_i}]$$

subject to

$$\sum_{i=1}^{15} C_i x_i \le 400$$

$$\sum_{i=1}^{15} W_i x_i \le 414$$

Table 1: Data Used for Problem 3.2.1

i	r_i	P_i	C_i	W_i
1	0.80	1	7	7
2	o.85	2	7	8
3	0.90	3	5	8
4	0.65	4	9	6
5	0.75	2	4	9

Table 2: Data Used for Problem 3.2.2

i	r_i	C_i	W_i	i	R_i	C_i	W_i
1	0.90	5	8	9	0.78	4	7
2	0.75	4	9	10	0.91	5	8
3	0.65	9	6	11	0.79	6	9
4	0.80	7	7	12	0.77	7	7
5	0.85	7	8	13	0.67	9	6
6	0.93	5	8	14	0.79	8	5
7	0.78	6	9	15	0.67	6	7
8	0.66	9	6				

4. Simulation Results and Discussion

The PSO is coded in Bloodshed Dev C++. All the programs were run on an Intel (R) Pentium (R), D CPU, 1.86GHz processors with 0.99 GB of Random Access Memory (RAM). Table 3 shows the setup of parameters $c_{1i}, c_{2i}, c_{1f}, c_{2f}$, γ and W used for reliability optimization problems, which affect convergence rate and robustness of the search procedure. Different parameter settings have been checked and tuned for each problem, the parameter setting corresponding to best results are reported in Table 3. Tables 4 to 7 presents a comparison between the best results obtained in this paper with other results in literature. The comparison was made on the basis of the number of function evaluations as previously have been done by [12], [13] and [14].

Table 3: Parameters for PSO

Examples	Pop size	c_{1i}	c_2	c_1	c_2	w	γ
Life support system in space capsule	30	2	1	1	2	0.50	0.25
Complex bridge system	30	2.5	1.5	1.5	2.5	0.70	0.50
Mixed series parallel system consisting five parallel units	30	2	1	1	2	0.50	0.25
Mixed series parallel system consisting fifteen parallel units	40	2.5	0.5	0.5	2.5	0.70	0.75

For life support system in space capsule, as mentioned in Table 4, total 2040 cost function evaluations are made and PSO provided 641.823562 as the optimal system cost and with 0.900000 system reliability. Results obtained due to present study have been listed along with the results obtained for the same in the past as given in Table 4. ACO [12] also obtained same results but with higher (20,100) function evaluations. Whereas MGDA [13] and C-SOMGA [14] obtained 641.823608 and 641.824000 optimal cost with 65,603 and 1,00,000 function evaluations respectively. Thus, for this problem where objective is to minimize system cost subject to the constraints on system reliability, PSO obtained the identical solution with ACO but consumed less function evaluations. Further, comparison indicates that solution obtained by PSO scores over C-SOMGA [14], MGDA [13], RMMM-CES [27], INESA [11] and SA [11] in terms of both accuracy as well as convergence speed.

Table 4: Result Comparison for Life Support System in Space Capsule

	PSO	C-SOMGA	MGDA	ACO	CEA (best)	INESA	SA
r_1	0.500000	0.500023	0.500010	0.500000	0.500000	0.500060	0.500950
r_2	0.838920	0.838900	0.838919	0.838920	0.838920	0.838870	0.837750
r_3	0.500000	0.556000	0.500000	0.500000	0.500000	0.500010	0.500250
r_4	0.500000	0.500000	0.500000	0.500000	0.500000	0.500020	0.500150
R_S	0.900000	0.900001	0.900000	0.900000	0.900000	0.900010	0.900010
C_i	641.823562	641.824000	641.823608	641.823562	641.823577	641.833200	641.903000
FE	2040	100000	65,603	20,100	-	-	-

For the complex bridge system, Table 5 presents the results obtained by PSO along with all previously reported results. Table 5 shows that PSO provides most relevant improvement to the previous six best known solutions. It obtained minimum system cost 5.019918 at 1,20,000 function evaluations. Although, it consumed more function evaluations than C-SOMGA [14], MGDA [13] and ACO [12] but still it is able to give minimum system cost.

Table 5: Result Comparison for Complex Bridge System

	PSO	C – SOMGA	MGDA	ACO	I-NESA	SA	Random Search Techniques[4]
r_1	0.934821	0.935359	0.935400	0.933869	0.937470	0.935660	0.939240
r_2	0.935028	0.934304	0.935403	0.935073	0.932910	0.936740	0.934540
r_3	0.791948	0.790332	0.788027	0.798365	0.784850	0.792990	0.771540
r_4	0.935005	0.935504	0.935060	0.935804	0.936410	0.938730	0.939380
r_5	0.934735	0.934575	0.934111	0.934223	0.933420	0.928160	0.928440
R_S	0.990000	0.990000	0.990000	0.990001	0.990000	0.990010	0.990040
C_S	5.019918	5.019920	5.019919	5.019923	5.019930	5.019970	5.020010
FE	1,20,000	100000	50,942	80160	-	-	-

Same investigation has been done for mixed series parallel system with consisting five parallel units. The results are presented in Table 6. Again PSO is able to obtain best known solutions which were earlier obtained by C-SOMGA [14], ACO [12], SA [11], CEA [27], INESA [11] and generalized Lagrange function approach [6]. PSO only requires 1350 (lowest among all) function evaluations to achieve this value. Thus PSO again yielded superior solution in comparison to all other six methods in terms of consumption of CPU time.

Table 6: Result Comparison for Mixed Series Parallel System Consisting Five Parallel Units

	PSO	C-SOMGA	ACO	CEA (best)	INESA	SA	Generalized Lagrange Function Approach
x_1	3	3	3	3	3	3	3
x_2	2	2	2	2	2	2	2
x_3	2	2	2	2	2	2	2
x_4	3	3	3	3	3	3	3
x_5	3	3	3	3	3	3	3
Rs	0.904500	0.904500	0.904500	0.904500	0.904500	0.904500	0.904500
FE	1350	100000	1800	-	-	-	-

Finally, for mixed series parallel system with consisting fifteen parallel units, Table 7 shows the results obtained by PSO along with other past reported solutions. The best result yielded by PSO is (3, 4, 6, 4, 3, 2, 4, 5, 4, 2, 3, 4, 5, 4, 5). The system reliability and cost have been obtained as 0.945613 and 392.0 respectively. The results are compared with the results obtained by other five optimization methods, which are C-SOMGA [14], MGDA [13], ACO [12], INESA [11], SA [11] and integer programming technique [3]. MGDA and ACO obtained the same system reliability with 217157 and 244000 function evaluations. Hence it can be concluded easily that the accuracy obtained by PSO, MGDA, ACO are exactly matching. while PSO found to be superior to MGDA, ACO when speed is compared. While C-SOMGA obtained 0.9450 system reliability with 100000 function evaluation. Thus PSO achieved higher reliability than C-SOMGA. Further, present study shows that results obtained by PSO is again far better than the earlier reported results corresponding to INESA and SA both in terms of accuracy and speed. Thus, Observation of Table 7 finds that the solution provided by PSO gives better reliability at relatively lower function evaluations than C-SOMGA, MGDA, ACO, INESA and SA.

Table 7: Result Comparison for Mixed Series Parallel System Consisting Fifteen Parallel Units

	PSO	C-SOMGA	MGDA	ACO	INESA	SA	Integer Programming
x_1	3	3	3	3	3	3	3
x_2	4	4	4	4	4	4	4
x_3	6	5	6	6	5	5	5
x_4	4	3	4	4	3	4	3
x_5	3	3	3	3	3	3	3
x_6	2	2	2	2	2	2	2
x_7	4	4	4	4	4	4	4
x_8	5	5	5	5	5	5	5
x_9	4	4	4	4	4	4	4
x_{10}	2	3	2	2	3	3	3
x_{11}	3	3	3	3	3	3	3
x_{12}	4	4	4	4	4	4	4
x_{13}	5	5	5	5	5	5	5
x_{14}	4	5	4	4	5	5	5
x_{15}	5	5	5	5	5	4	5
R_S	0.945613	0.945000	0.945613	0.945613	0.944749	0.943259	0.944749
FE	1.6×10^5	10^5	217157	244000	-	-	-

5. Conclusion

In this paper, a new PSO was proposed for solving complex network reliability. In general, system reliability optimization problems are nonlinear programming problems and proved to be NP-hard from computational point of view. That is, they are more difficult to solve than any general nonlinear programming problem. Moreover, the comparison with studies in the literature involving the same problems demonstrated that PSO has better efficiency in solving the complex network reliability optimization problem as it can provide a solution which is either comparable to or superior to the best available results. The simulation results indicates that our approach is a viable alternative since PSO is able to obtain solutions not only in terms of accuracy and but also in terms of speed than those obtained from previously published in literature.

References

[1] Misra, K. B., and M. D. Ljubojevic. *Optimal Reliability Design of A System: A New Look.* IEEE Transactions on Reliability, 1973; R-22(5): 255-258.

[2] Misra, K. B. *Optimal Reliability Design of A System Containing Mixed Redundancies.* IEEE Transactions on Power Apparatus & Systems, 1975; 94(3): 983-993.

[3] Luus, R. *Optimization of System Reliability by A New Nonlinear Integer Programming Procedure*. IEEE Transactions on Reliability, 1975; R-24(1): 14-16.

[4] Mohan, C. and K. Shanker. *Reliability Optimization of Complex Systems using Random Search Technique*. Microelectronics Reliability, 1988; 28 (4): 513-518.

[5] Misra, K. B. and U. Sharma. *An Efficient Algorithm to Solve Integer-Programming Problems Arising in System-Reliability Design*. IEEE Transactions on Reliability, 1991; 40(1): 81-91.

[6] Tillman, F. A., C. L. Hwang, and W. Kuo. *Optimization of Systems Reliability*. Marcel Dekker Inc., New York, 1980.

[7] Lad, B. K., M. S. Kulkarni, and K. B. Misra. *Optimal Reliability Design of a System.* In: Handbook of Performability Engineering. Misra, K. B. (Ed.) Springer, Verlag, London, 2008, 499–519.

[8] Chaturvedi, S. K., and K. B. Misra. *MIP: A Versatile Tool for Reliability Design of A System*. In: Handbook of Performability Engineering. Misra, K.B. (Ed.) Springer Verlag, London, 2008, 521-531.

[9] Chern, M. S. *On the Computational Complexity of Reliability Redundancy Allocation in a Series System*. Operations Research Letters, 1992; 11(5): 309-315.

[10] Kuo, W., and R. Wan. *Recent Advances in Optimal Reliability Allocation*. IEEE Transactions on System, Man and Cybernetics, Part A: System Humans, 2007; 37(2):143-156.

[11] Ravi, V., B. S. N. Murty and J. Reddy. *Nonequilibrium Simulated-Annealing Algorithm Applied to Reliability Optimization of Complex Systems*. IEEE Transactions on Reliability, 1997; 46(2): 233-239.

[12] Shelokar, P. S., V. K. Jayaraman, and B.D. Kulkarni. *Ant Algorithm for Single and Multiobjective Reliability Optimization Problems*. Quality and Reliability Engineering International, 2002; 18(6): 497-514.

[13] Ravi, V. *Optimization of Complex System Reliability by A Modified Great Deluge Algorithm*. Asia-Pacific Journal of Operational Research, 2004; 21(4): 487–497.

[14] Deep, K., and Deepti. *Reliability Optimization of Complex Systems through C-SOMGA*. Journal of Information and Computing Science, 2009; 4(3): 163-172.

[15] Yeh, W. C. *A Two-Stage Discrete Particle Swarm Optimization for the Problem of Multiple Multi-Level Redundancy Allocation in Series Systems*. Expert Systems with Applications, 2009; 36(5): 9192–9200.

[16] Pant, S., and S. B. Singh. *Particle Swarm Optimization to Reliability Optimization in Complex System.* IEEE Int. Conf. on Quality and Reliability, Bangkok, Thailand, Sept 14-17, 2011; 211-215.

[17] Rani, M., S. P. Sharma and H. Garg. *A Novel Approach for Analyzing the Behaviour of Industrial Systems under Uncertainty*. International Journal of Performability Engineering, 2013; 9(2): 201–210.

[18] Zio, E., F. Di Maio, and S. Martorell. *Fusion of Artificial Neural Networks and Genetic Algorithms for Multi-Objective System Reliability Design Optimization*. Journal of Risk and Reliability, 2008; 222 (2): 115-126.

[19] Kishore, A., S. P. Yadav, and S. Kumar. A Multi Objective Genetic Algorithm for Reliability Optimization Problem. International Journal of Performability Engineering, 2009; 5(3): 227–234.

[20] Hassan, R., B. Cohanim, O. De Weck, and G. Venter. *A Comparison of Particle Swarm Optimization and the Genetic algorithm.* In 46th AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics, and Materials Conference, Austin, TX, April 18-21, 2005.

[21] Clow, B., and T. White. *An Evolutionary Race: A Comparison of Genetic Algorithms and Particle Swarm Optimization used for Training Neural Networks.* Int. Conf. on Artificial Intelligence, Las Vegas, Nevada, USA, June 21-24, 2004; 582-588.

[22] Hodgson, R. J. W. *Particle Swarm Optimization applied to the Atomic Cluster Optimization Problem*. The Genetic and Evolutionary Computation Conference, New York, USA, July 9-13, 2002; 68-73.

[23] Engelbrecht, A. P. *Fundamentals of Computational Swarm Intelligence*. Jhon Wiley and Sons Ltd., Chichester, England, 2005.

[24] Kennedy, J., and R. Eberhart. *Particle Swarm Optimization*. IEEE Int. Conf. on Neural Networks, Perth, Austrailia, 27 Nov-1 Dec, 1995; 1942-1948.

[25] Chauhan, N., A. Mittal, D. Wagner, M. V. Kartikeyan, and M. K. Thumm. *Design and Optimization of Nonlinear Tapers using Particle Swarm Optimization*. International Journal of Infrared and Millimeter Waves, 2008; 29(8): 792-798.

[26] Ratnaweera, A., S. K. Halgamuge, and H. C. Watson. *Self-Organizing Hierarchical Particle Swarm Optimizer with Time-Varying Acceleration Coefficients*. IEEE Transactions on Evolutionary Computation, 2004; 8(3): 240-255.

[27] Rocco, C. M., A. J. Miller, J. A. Moreno and N. A. Carrasquero. *Cellular Evolutionary Approach Applied to Reliability Optimization of Complex Systems*. Annual Reliability and Maintainability Symposium, Los Angeles, CA, Jan 24-27, 2000; 210-215.

Sangeeta Pant is currently working as Assistant Professor in University of Petroleum and Energy Studies, Dehradun, India. She holds Ph.D. in Mathematics with concentration on Metaheuristics and Reliability Optimization from G. B. Pant University of Agriculture and Technology, Pantnagar, India.

Dheeraj Anand holds B.Tech. in Electronics & Communication Engineering from Graphic Era University, Dehradun, India. His area of interest is computer programming and on self-motivating projects.

Amar Kishor is currently a postdoctoral fellow at Electronics and Communication Sciences Unit, Indian statistical Institute Kolkata, India. Earlier, he worked as Assistant professor, Department of Mathematics, School of Engineering and Technology, Sharda University, Greater Noida, India. He Holds Ph.D. in Mathematics from I. I. T. Roorkee. His current research areas include Aggregation Operators, OWA Operators, Multi-objective Optimization, Reliability Optimization, Evolutionary Algorithms and Fuzzy Logic.

Suraj Bhan Singh is Professor in G. B. Pant University of Agriculture and Technology, Pantnagar, India. He holds Ph.D. in Mathematics. His current research area includes, Reliability Analysis and Operations Research.