ECONOMIC LOAD DISPATCH STUDIES BASED ON PSO

DISSERTATION/THESIS

SUBMITTED IN PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR THE AWARD OF THE DEGREE
OF

MASTER OF TECHNOLOGY

IN

POWER SYSTEM

Submitted by:

ANURAG KUMAR

2K18/PSY/11

Under the supervision of

PROF. N.K. JAIN & PROF. UMA NANGIA

DEPARTMENT OF ELECTRICAL ENGINEERING

DELHI TECHNOLOGICAL UNIVERSITY
(Formerly Delhi College of Engineering)
Bawana Road, Delhi-110042

2020

CERTIFICATE

This is to certify that the thesis titled " ECONOMIC LOAD DISPATCH STUDIES BASED ON PSO " submitted by ANURAG KUMAR, Roll No. 2K18/PSY/11, student of Master of Technology (Power system) in ELECTRICAL ENGINEERING DEPARTMENT from Delhi Technological University is a original dissertation carried out under our guidance towards partial fulfilment of the requirements for the award of the degree of Master of Technology in Power system. This thesis proposes a new algorithm to optimize economic load dispatch problem.

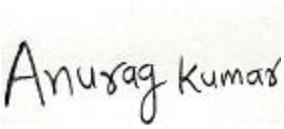
PROF.UMA NANGIA
(Supervisor)

PROF. N.K. JAIN
(Supervisor)

CANDIDATE DECLARATION

I, ANURAG KUMAR, Roll No. 2K18/PSY/11 student of M.Tech (Power System), hereby declare that the dissertation/project titled " ECONOMIC LOAD DISPATCH STUDIES BASED ON PSO " under the supervision of Prof. N.K.Jain and Prof. Uma Nangia is submitted in partial fulfillment for the award of the degree of Master in technology (Power system) and original work carried out under the able guidance of Prof. N.K.Jain and Prof.Uma Nangia. The matter embodied in the dissertation is not plagiarized from anywhere and is not submitted anywhere else for the award of any other degree and diploma.

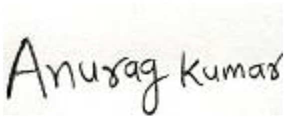
Submitted by-



ANURAG KUMAR
2K18/PSY/11
Power System
EED

ACKNOWLEDGEMENT

For the development of the project " ECONOMIC LOAD DISPATCH STUDIES BASED ON PSO ". I would like to extend my gratitude to Prof. N.K. Jain and Prof. Uma Nangia, Department of electrical engineering for their constant motivation and support during the course of my work. I truly appreciate and value their esteemed guidance and encouragement of the project.



ANURAG KUMAR
2K18/PSY/11
Power System
EED

ABSTRACT

In this project, an Improved PSO algorithm has been developed to solve economic load dispatch problem. In the Proposed PSO algorithm, Retardation factor has been introduced to damp out the oscillations as the particle reaches near the global optimum point. This results in faster convergence as well as lesser cost of generation. The proposed algorithm has been implemented on unconstrained mathematical test functions to check the accuracy and convergence of the algorithm. The Proposed PSO algorithm is implemented on IEEE three and six generator thermal power plants. In the case of mathematical test functions, the comparison is done in terms of the number of iterations performed and the number of function evaluations. In case of an economic load dispatch problem, the comparison is done in terms of fuel cost of generation also. After comparing results in both cases, it is found that the proposed PSO algorithm gives more accurate results in less number of iterations. Number of iterations, number of function evaluations, and time consumed have been measured for different values of retardation factors. Best retardation is the one for which function gets optimized in minimum number of iterations. MATLAB simulation is done to solve the economic load dispatch problem and mathematical test function using Proposed algorithm and Basic particle swarm optimization algorithm.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF SYMBOLS

PSO: Particle swarm optimization

BPSO: Basic particle swarm optimization

NFEV: Number of function evaluations

P_{best} : Personal best

G_{best}: Global best

r.f. : Retardation factor

P_D : Power demand

P_L : Transmission losses

W_{max} : Maximum value of inertia weight

W_{min} : Minimum value of inertia weight

i: number of iterations after which retardation factor is included

P_i^{min} and P_i^{max} : the power limits of ith generator

Iter : Iteration

CHAPTER 1: INTRODUCTION

1.1 OVERVIEW

This work done proposes a new algorithm of particle swarm optimization. The proposed PSO algorithm is implemented on mathematical benchmark functions to check the convergence and speed. Further, it is implemented on the economic load dispatch problem aiming reduction of fuel cost and the computational effort.

1.2 MOTIVATION

In a world of limited non-renewable resources and increasing rates of fuels, we need to have a power system that will have the least running costs. The main objective of a modern interconnected power system is to provide high quality and reliable power supply to consumers at the lowest cost considering constraints applied to the generating units as well as environmental. If we consider load flow study, for any particular load demand generations made by the generators are kept fixed except the slack bus. But in case of economic load dispatch, the generators are free to take the values again to meet load demand in such a way that the fuel consumption cost is minimum. Therefore, the economic load dispatch problem solves two different problems. First is to meet the expected load demand within the specified reserve limits of themselves, and second is to distribute the load between the generators in such a manner that the cost of generation is minimum. Fuel cost curve in case of economic load dispatch is non-linear and very difficult to be optimized using a traditional optimization technique. Therefore, in this project work, we will optimize the fuel cost curve using a particle swarm optimization technique.

1.3 OBJECTIVES

Following are the broad objectives of the research work presented in this thesis

1) To study the behaviour of particles- After plotting the graph between the two variables, we can observe that different particles follow different paths to reach the optimum point.
2) To implement PSO on the economic load dispatch problem- We have attempted to implement the improved PSO over the economic load problem and checked the convergence of the problem using particles.

3) To compare the results of basic PSO with proposed PSO-We have performed economic load dispatch using proposed PSO as well as basic PSO. The results of the two are compared.
4) To check the accuracy of proposed PSO-The results of proposed PSO are checked with basic PSO to check that program of proposed PSO gives results as accurate as basic PSO or even better.

1.4 OUTLINE OF THESIS

Following is the chapter wise details of the material presented in thesis

CHAPTER-1 is an introduction chapter which deals with a small overview of thesis work. The chapter discusses the thesis objectives.

CHAPTER-2 provides a literature review of the PSO algorithm. It discusses the economic load dispatch and optimization of economic load dispatch using the PSO algorithm.

CHAPTER-3 discusses the basic PSO algorithm. It also discusses the flowchart of the basic PSO algorithm. The advantages of basic PSO have been discussed.

CHAPTER-4 discusses the proposed PSO algorithm. Flowchart of the proposed PSO algorithm and the significance of the retardation factor are also discussed.

CHAPTER-5 discusses the implementation of proposed PSO to mathematical test functions. Results of Rosenbrock, Booth, Easom, and Three hump camel function using basic PSO and proposed PSO are discussed. A comparison of basic PSO and proposed PSO results is also discussed.

CHAPTER-6 discusses the implementation of proposed PSO to economic load dispatch. THE proposed PSO is implemented on the IEEE-3 generator and IEEE-6 generator system. Fuel costs of generations using basic PSO, and proposed PSO are compared.

CHAPTER-7 discusses the conclusion of project work. The efficiency of the proposed PSO algorithm in comparison to BPSO is discussed. Future scope of project work is also discussed.

CHAPTER 2: LITERATURE REVIEW

2.1 ECONOMIC LOAD DISPATCH-

The main objective of the economic load dispatch problem to minimize the cost of fuel in such a manner that generations made by different generators don't violate the power limits.There are different methods to optimize fuel cost function. Adaptive Hopfield neural networks are used for optimizing the economic load dispatch. Algorithm of economic load dispatch using an adaptive neural network has been developed in [1] paper. The fuel cost characteristics are quadratic and difficult to optimize. In the paper [2], economic load dispatch is optimized using a biogeography inspired algorithm. It has improved the speed to reach the optimum point as well as time consumed also has improved. In the paper [3], economic load dispatch is an optimized hybrid bacterial foraging optimization technique. This technique doesn't require finding the derivative of any function, unlike the conventional lambda iteration method. Therefore it has produced results better than other artificial intelligent techniques of optimizing any function. lambda iteration method is used to optimize the economic load dispatch problem. It is one of the best convenient methods to optimize the objective function. In the paper, it has been implemented on a small power system network of 15 generators as well as a large modern interconnected power system network of 140 generators. There is a prohibited zone of operation of generators which is because of the physical limits of generators, the paper [4] have considered this aspect while optimizing the fuel cost function. In the era of the energy crisis, 75% of our power demand is fulfilled by the generation made by the thermal power plant. Also, when the power demand is large, then the size of the power system network also increases. One of the solutions is to interconnect the power system network. In the case of a modern interconnected power system network, there is a need of optimizing the economic load dispatch problem. It can help us supply the power to consumers in the most economical manner. To help this objective, a new method of particle swarm optimization technique has been introduced in this thesis work.

2.2 PSO METHOD-

PSO is the artificial intelligent technique developed by Eberhart and Kennedy inspired by bird flocking. Bird's behavior while searching the food is observed and implemented over optimization problems[5]. Algorithm of optimizing mathematical functions is discussed in this paper. Adaptive PSO reported in[6] have introduced a new method of optimizing any benchmark test functions. The particles could be trapped at a local minimum point while searching the optimum point. Adaptive PSO provides an efficient method to search the global optimum point. Adaptive PSO has a better convergence rate. In paper [7], a review of particle swarm optimization is done. The Basic PSO algorithm can be understood as well as different parameters of PSO. Different applications of PSO are discussed here. Modified and improved PSO algorithms are also discussed and can be implemented on various mathematical functions as well as real-life applications. Particle swarm optimization has various applications in mechanical engineering, reported in [8]. It also discusses improved PSO algorithm. Particles move around in multidimensional space in search of an optimum position. The particles change their position and their velocity according to their experiences and their neighboring particle's position. At each iteration, particle updates their position according to the personal best position of themselves as well as the global best position of overall particles. With the help of velocity and position update equations, the PSO algorithm leads us to the optimum point.

2.3 ECONOMIC LOAD DISPATCH USING PSO

Economic load dispatch can be solved using the particle swarm optimization technique. The proposed PSO reported in [9] has considered the practical aspects of the power system. The cost function is not smooth, the prohibited zone of operation of generators is also considered. In paper [10], the Basic PSO algorithm and teaching-learning optimization technique is implemented on IEEE 3,6,15 and 20 generator power system. The results of these two optimization techniques are compared with the conventional method of lambda iteration and these two methods are found to be giving better results in comparison to conventional ones. In paper [11], adaptive PSO has been implemented on economic load dispatch and is found to be converging faster and giving better results. In paper [12], the non smooth fuel cost function is optimized using the particle swarm optimization and is found to be giving better results compared to conventional and other programming methods. In paper [13], the economic load dispatch problem is optimized using the PSO technique, hybrid PSO, Real coded genetic algorithm technique. Fuel cost calculated using the proposed PSO method is the lowest, which is the main objective. Multiobjective economic load dispatch is done in [14], in this paper two objectives are achieved. The first one is to optimize the fuel cost and the second one is to control the emission made by the thermal power plant. These two objectives are contradictory to each other so priority is set up to optimize the objective function. In the paper [15], economic load dispatch is done using three proposed algorithms and results are found to be better than the basic PSO algorithm.

PSO can be implemented over economic load dispatch objective function to get the optimal solution. We can do it using the following steps-

1. Initialize the population within the power limits.
2. Set the number of particles in a swarm, the maximum number of iterations, C_1, C_2, W_{max}, W_{min}, and a pre-specified error value.
3. Set the B coefficients and Power demand.
4. Set iteration count, iter=0
5. For each value of randomly allocated active powers, the fitness function is calculated.
6. All values of the objective function obtained using these active powers are compared for each particle. For each particle, there will be a personal best position. Comparison done over all the population will yield the global best position.
7. The global best will give minimum cost and active power made using this global best position will be the optimal solution.
8. This process will be repeated for several iterations until stopping criteria are met.
   If stopping criteria is satisfied then we will display the cost of generation as well as transmission losses of all the generators.

2.4 CONCLUSION

This chapter discusses about the basic idea of economic load dispatch, PSO method and economic load dispatch using PSO. The literature review of different methods of optimizing fuel cost function of economic load dispatch, different methods of particle swarm optimization and different ways of optimizing economic load dispatch using PSO along with their benefits and drawbacks are presented here. Different PSO algorithms are studied and their convergence and speed are analysed. PSO algorithm is also discussed along with the steps.

CHAPTER 3: BASIC PARTICLE SWARM OPTIMIZATION (BPSO) ALGORITHM

3.1 INTRODUCTION

BPSO is an artificially intelligent algorithm inspired by the movement of birds in search of foods. This method is developed by Eberhart and Kennedy in 1995. Birds are considered as particles in this algorithm. BPSO algorithm gives the way to calculate the optimum point of the objective function. It is a simple algorithm that can be used to optimize any mathematical test functions. In the early era of optimizing the non-smooth and nonlinear objective function, conventional methods like lambda iteration were used. But in the case of optimizing highly nonlinear and complex objective function like economic load dispatch, solving it using these methods become difficult. However, the BPSO algorithm can be used to optimize these types of problems. In any PSO algorithm, a population of particles is generated randomly. These particles are given some random velocities to begin with. Later on, the updated value of particle position and velocity can be calculated using the following equations-

Velocity update equation

$$V_j(i)=w * V_j(i-1)+C_1 r_1\left[(P_{best})j - x_j(i-1)\right] + C_2 r_2\left[G_{best} - x_j(i-1)\right] \quad (1)$$

Position update equation

$$x_j(i) = x_j(i-1) + V_j(i) \quad (2)$$

Where,

j=1,2,3,……..,n

C_1,C_2 =Acceleration constants

w=Inertia weight factor

r_1,r_2= random values between 0 to 1

$x_j(i-1)$= current position of a particle

$V_j(i-1)$=current velocity of a particle

Inertia weight factor can be calculated using the following equation-

$w = w_{max}-[(w_{max}-w_{min})*iter]/itermax$

w_{max}= maximum value of inertia weight factor

w_{min}=minimum value of inertia weight factor

itermax=maximum number of iterations

iter=present count of iteration

$V_j(i)$= Velocity of ith iteration for jth particle

$x_j(i)$= Position of ith iteration for jth particle

3.2 BPSO ALGORITHM

The various steps of Basic particle swarm optimization (BPSO) algorithm are given below

1. First of all, the objective function to be optimized is defined.
2. Various parameters which are maximum number of iterations, swarm size, social acceleration coefficients C_1 & C_2, tolerance value, r_1&r_2 are defined.
3. Initialize the variables like position, the velocity of particles as well as the NFEV.
4. Define the search space for the variables. Random initial values are selected using unifrnd and rand functions. These two functions generate uniformly distributed and randomly distributed numbers respectively.
5. The position and velocity of each particle are updated using the position and velocity update equations.

   $V_j(i)=w * V_j(i-1)+C_1r_1\left[(P_{best})j - x_j(i-1)\right] + C_2r_2\left[G_{best} - x_j(i-1)\right]$

   $x_j(i) = x_j(i-1) + V_j(i)$
6. Initially, Calculate the value of the function for each particle in swarm using the objective function.
7. After calculating the value of the objective function for each particle, P_{best} and G_{best} of particles are found out.
8. Best position of a particle for which the value of the objective function is optimized, known as the personal best position.
9. The best value among all the personal best positions is known as the global best position.

10. The position and velocity of each particle are updated using the position and velocity update equations.
11. This process is repeated for several iterations until the stopping criteria are met.
12. If the personal best position at present iteration is better than the personal best position at the previous iteration then the present value will be set as a new P_{best} position.
13. Similarly, if the global best position at present iteration is better than the global best at present iteration then the present value will be set as a new G_{best} position.
14. The New G_{best} position of the particle will give the optimum point of the objective function.
15. At this G_{best} value of particles, the value of any objective function will be optimum.
16. In the case of optimizing mathematical test function, this point will be defined by the variables of objective function.
17. In the case of optimizing the economic load dispatch problem, the global best position will be active power generated by various generators.
18. If stopping criteria is satisfied, display the value of the number of function evaluations, the minimum value of the function at the global optimum point.
19. Display the time consumed to optimize the function and number of iterations performed.
20. The plot of each particle is traced in search of a global optimum point.
21. The plot between the function value of the objective function and the number of iterations is also traced. This and previous step are carried out to facilitate the understanding and analysis of the process of PSO.

Let us assume

Function value at Personal best at previous iteration= $F_p(i)$

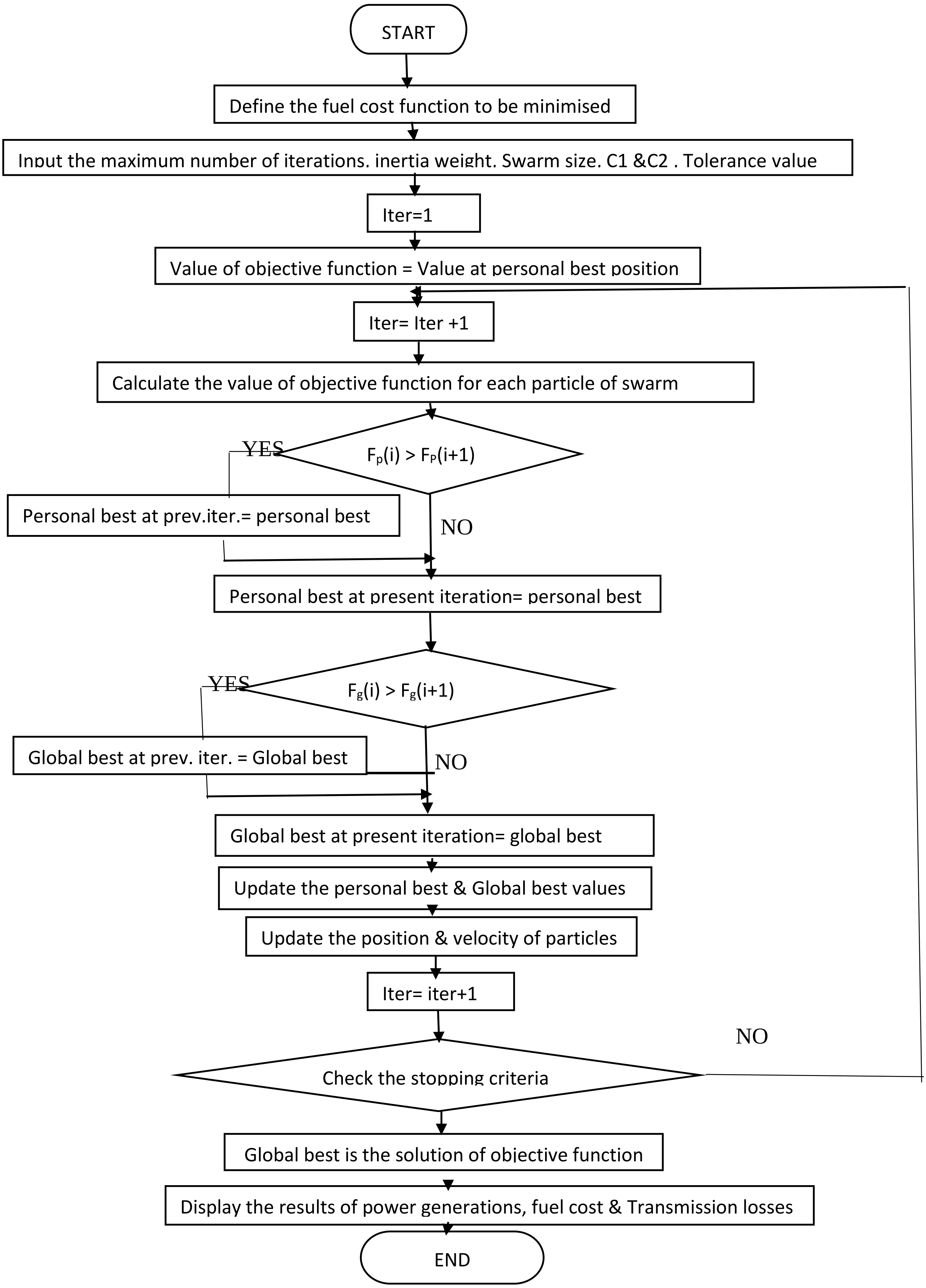
Function value at Personal best at present iteration= $F_p(i+1)$

Function value at Global best at previous iteration=$F_g(i)$

Function value at Global best at present iteration=$F_g(i+1)$

C_1 and C_2= social acceleration coefficients

3.3 FLOWCHART OF BPSO ALGORITHM-



3.4 ADVANTAGES OF BPSO ALGORITHM

1. **No need of mutation and overlapping** Other Optimization technique like genetic algorithm requires genetic diversity in their population. Particle swarm optimization doesn't need mutation to maintain the diversity in the swarm. It has velocity and position update equations which update their position according to personal and global best positions.
2. **Limited algorithm parameters** To optimize any problem , PSO requires limited algorithm parameters to be controlled. It includes swarm size, acceleration coefficients, inertia weight , maximum number of iterations and tolerance value.
3. **Easy implementation** BPSO algorithm can be easily implemented over complex engineering problems. The only thing that should be followed are the steps in the algorithm and any objective function can be optimized. BPSO algorithm doesn't require a crossover and mutation like other artificial intelligence techniques to reach the global optimum point which makes it easy to be implemented.

3.5 CONCLUSION

Following are the conclusion of chapter 3-

1. The basic particle swarm optimization algorithm is discussed in chapter-3.
2. Any objective function can be optimized using the algorithm discussed above.
3. Steps needed to reach the optimum point of any objective function are discussed.
4. MATLAB program can be developed using the flowchart.
5. Every parameter of PSO is explained.
6. Velocity and position update equations of PSO have been discussed as well as the introduction of PSO.
7. It briefly discusses how particles achieve optimum points.
8. BPSO algorithm can be implemented on mathematical test functions as well as problems like economic load dispatch. It is discussed in upcoming chapters.

CHAPTER 4: PROPOSED PSO ALGORITHM

4.1 INTRODUCTION

The proposed PSO algorithm is a new method discussed in the project work. Taking the flaws of basic PSO into consideration, Proposed PSO is developed and implemented on mathematical test functions as well as on economic load dispatch. Though basic PSO is efficient and can be implemented to optimize any objective function. But the BPSO algorithm has a small probability to get stuck at the local minimum point in place of the global minimum point. Also, it has been seen that particles converge faster and reaches an optimum point but have higher velocity even when it is about to reach the optimum point. It leads the algorithm to perform more number of iterations than required to reach the optimum point. Also, the time consumed will be more. Therefore, a retardation factor has been introduced which is to be multiplied with the velocity after a certain number of iterations. Concept of NFEV is introduced in our thesis work, which gives us information about how many times a function is evaluated. This process includes the evaluation of function for each particle's every iteration, personal best, and global best position. It has played a significant role in proving how the proposed PSO algorithm is better than the basic PSO algorithm. The proposed PSO algorithm is implemented on mathematical test functions to check the convergence and accuracy of the algorithm. After getting better results than the BPSO, it is implemented on the economic load dispatch problem. In the case of economic load dispatch, the main objective is to optimize the fuel cost function. Therefore using the proposed PSO algorithm we have optimized the fuel cost function.

4.2 PROPOSED PSO ALGORITHM

1. The objective function to be minimized is defined.
2. Input the maximum number of iterations, swarm size, social acceleration coefficients C_1 & C_2, r_1& r_2, tolerance value and retardation factor.
3. Initialize the variables like position, the velocity of particles as well as the NFEV.
4. Define the search space for the variables. Random initial values are selected using unifrnd and rand. These two functions generate uniformly distributed and randomly distributed numbers respectively.
5. The function value for each particle in swarm using the objective function is calculated.
6. After calculating the value of the objective function for each particle, P_{best} and G_{best} of particles are found out.

7. Best position of a particle for which the value of the objective function is optimized, known as the personal best position.
8. The best value among all the personal best positions is known as the global best position.
9. The position and velocity of each particle are updated using the position and velocity update equations.

$$V_j(i)=w * V_j(i-1)+C_1r_1\left[(P_{best})j - x_j(i-1)\right] + C_2r_2\left[G_{best} - x_j(i-1)\right]$$

$$x_j(i) = x_j(i-1) + V_j(i)$$

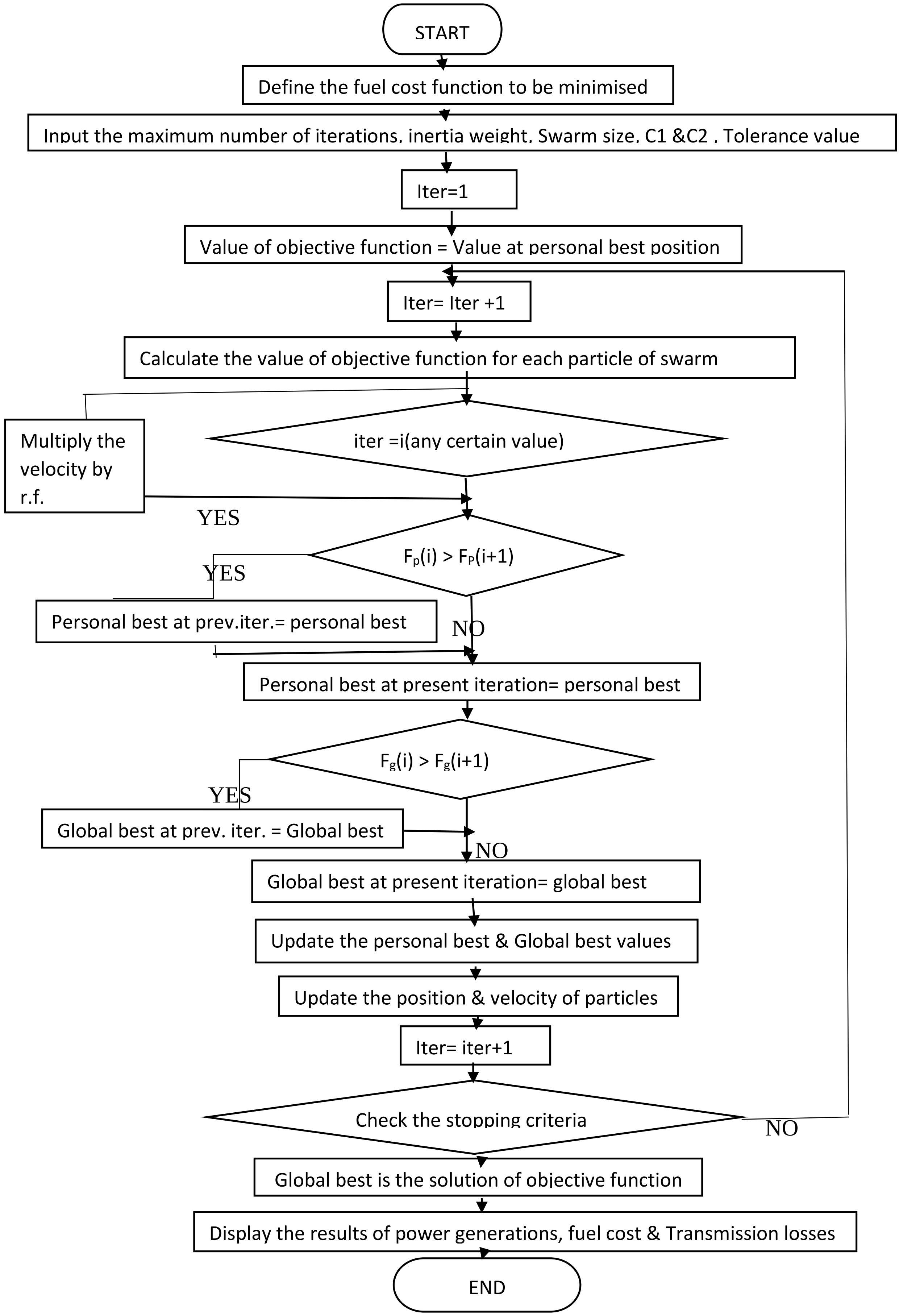
10. A retardation factor is multiplied with the velocity of the particle for the next iteration. It will reduce the velocity of each variable's particle and reduce the number of iterations to reach the optimum point.
11. $V_j(i)$ = retardation factor * $V_j(i-1)$.

    It means that the velocity of any jth particle gets multiplied with the retardation factor. After multiplying with the retardation factor, the velocity of the particle gets reduced for the next iteration. The velocity of particles gets reduced after a certain number of iterations.
12. This process is repeated for several iterations until the stopping criteria are met.
13. If the personal best position at present iteration is better than the personal best position at the previous iteration then the present value will be set as a new P_{best} position.
14. Similarly, if the global best position at present iteration is better than the global best at present iteration then the present value will be set as a new G_{best} position.
15. The New G_{best} position of the particle will give the optimum point of the objective function.
16. If stopping criteria is satisfied, display the value of the number of function evaluations, the minimum value of the function at the global optimum point.
17. Display the time consumed to optimize the function and number of iterations performed.
18. The plot of each particle is traced in search of a global optimum point.
19. The plot between the function value of the objective function and the number of iterations is also traced.

4.2.1 SIGNIFICANCE OF RETARDATION FACTOR-

In this project work, we have proposed a new type of particle swarm optimization. We have proposed to include a retardation factor after a certain number of iterations. The retardation factor is nothing but a certain value which will be multiplied with the velocity of the particle. To achieve the optimum point particle moves in multidimensional space. We have also observed that in the process of achieving the optimum point even when particles are close to the global optimum point, the particles are still moving. It makes the algorithm to reach the optimum point late even when being close to it. Now, if we include the retardation factor the search for optimum point is done using comparatively smaller steps and the number of iterations, the number of function evaluations gets reduced. This whole phenomenon will be a hit and trial method in which we have tested the retardation factor of several values. Hence the retardation factor will help us finding the optimum point quickly. Mathematical test functions are used to observe the characteristics of an algorithm as well as the convergence and accuracy of it. Therefore, first of all, we have implemented the particle swarm optimization over mathematical test functions. Also, the retardation factor is included after a certain number of iterations. After finding desirable results in the case of mathematical test function we have attempted it over the ELD problem. ELD problem requires optimization to reach the optimal point. Therefore, we have implemented particle swarm optimization over the economic load dispatch problem. First, basic PSO is implemented over the ELD problem, and the value of fuel cost is noted down. Then, the retardation factor is included and multiplied with the velocity of particles. After the inclusion of the retardation factor, the fuel cost of generators gets reduced by a small percentage. The objective of the project work is also the same which is to develop an algorithm of PSO which gives better results than the basic PSO in terms of fuel cost and in terms of computational effort.

4.3 FLOWCHART OF PROPOSED PSO ALGORITHM



4.4 CONCLUSION

Following are the conclusion of chapter (4)

1. The proposed particle swarm optimization algorithm is discussed in chapter-4.
2. Any objective function can be optimized using the algorithm discussed above.
3. Steps needed to reach the optimum point of any objective function are discussed.
4. MATLAB program can be developed using the flowchart discussed above.
5. Every parameter of PSO is explained.
6. Velocity and position update equations of PSO have been discussed as well as the introduction of PSO.
7. It briefly discusses how particles achieve optimum points.
8. The proposed PSO algorithm can be implemented on mathematical test functions as well as problems like economic load dispatch. It is discussed in upcoming chapters.
9. The new concept of the retardation factor is introduced and included in the velocity update equation for the next iteration.

CHAPTER 5: IMPLEMENTATION OF PROPOSED PSO TO MATHEMATICAL TEST FUNCTIONS

5.1 Rosenbrock function

$f(x_1, x_2)$=100$(x_1^2 - x_2)^2 + (1 - x_1)^2$

It has a global minimum at (x_1, x_2)=(1,1),where $f(x_1, x_2)$=0 ,Rosenbrock function is a mathematical test function used as a performance test problem for optimization. We have performed basic PSO and improved PSO algorithm on these mathematical test functions.

5. 1. 1 Results of Basic PSO-

no. of particles =20

maximum no. of iterations =200, tolerance=6 & no. of iterations=113

no. of function evalulations =4621

min. value of function=1.275733e-14

x1=1.000000e+00 & x2=1.000000e+00

time=1.481414e-01s

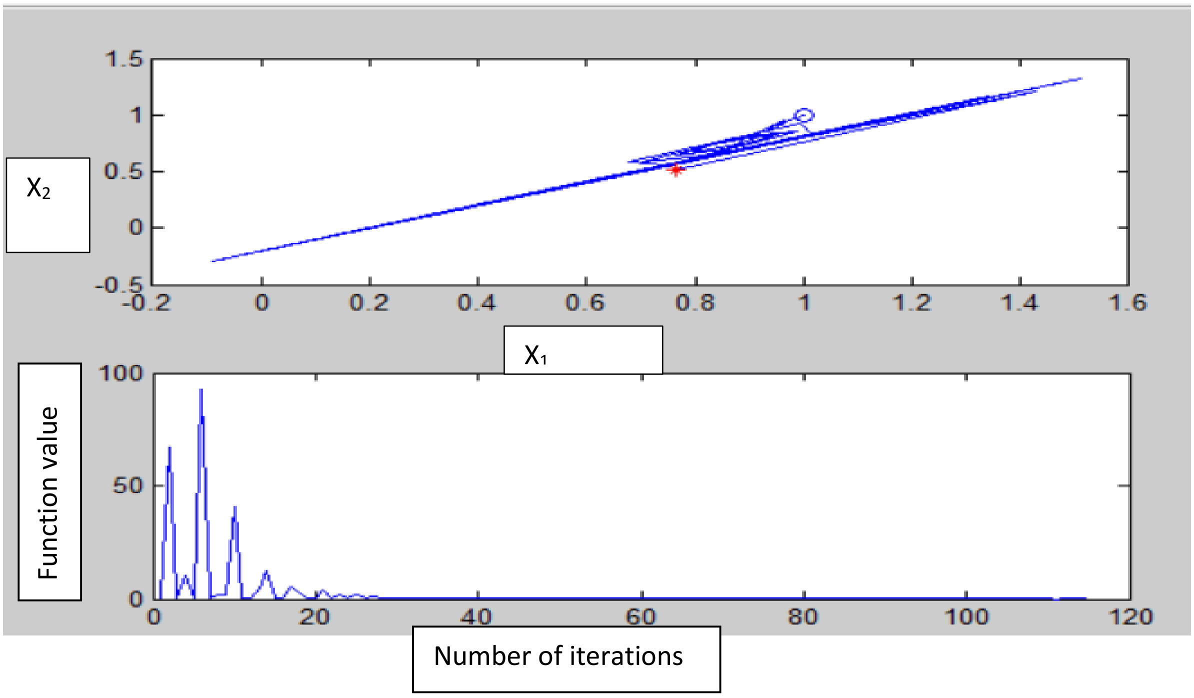


Fig-1(a) movement of first particle to reach the optimum point of rosenbrock function

Fig-1(b) function value vs number of iterations of first particle of rosenbrock function

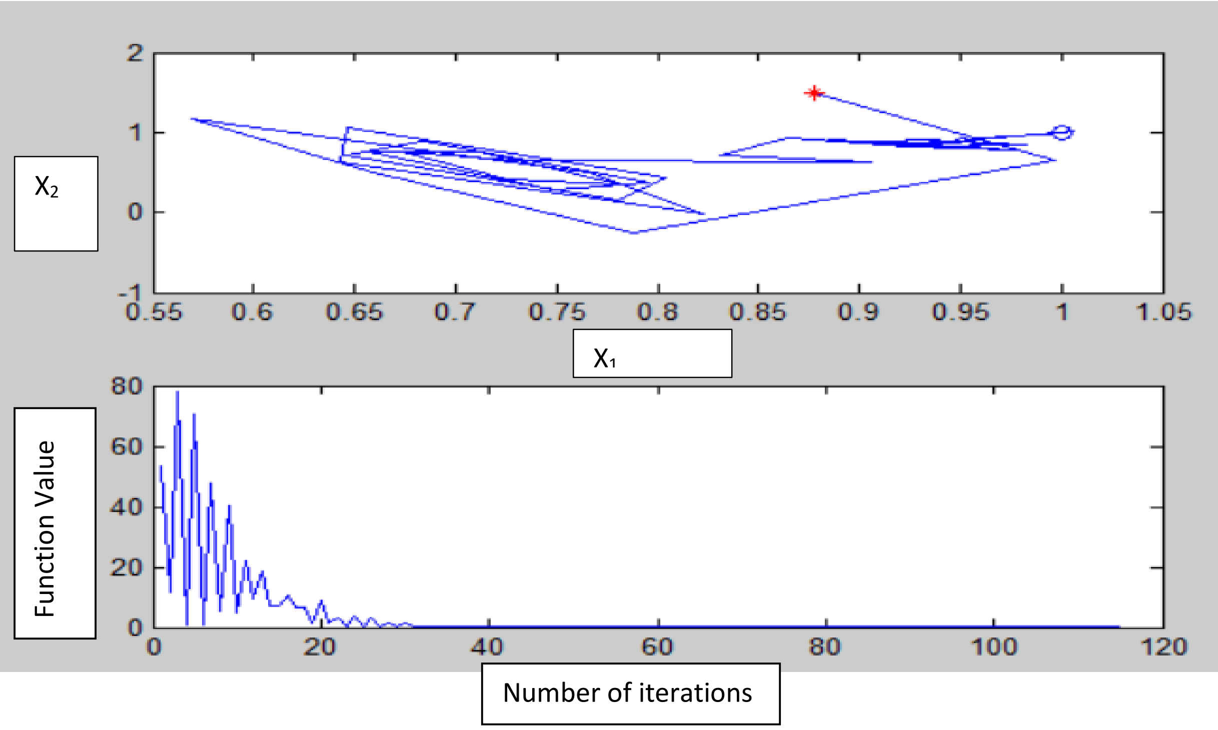


Fig-2(a) movement of second particle to reach optimum point of rosenbrock function

Fig-2(b) function value vs number of iterations of second particle of rosenbrock function

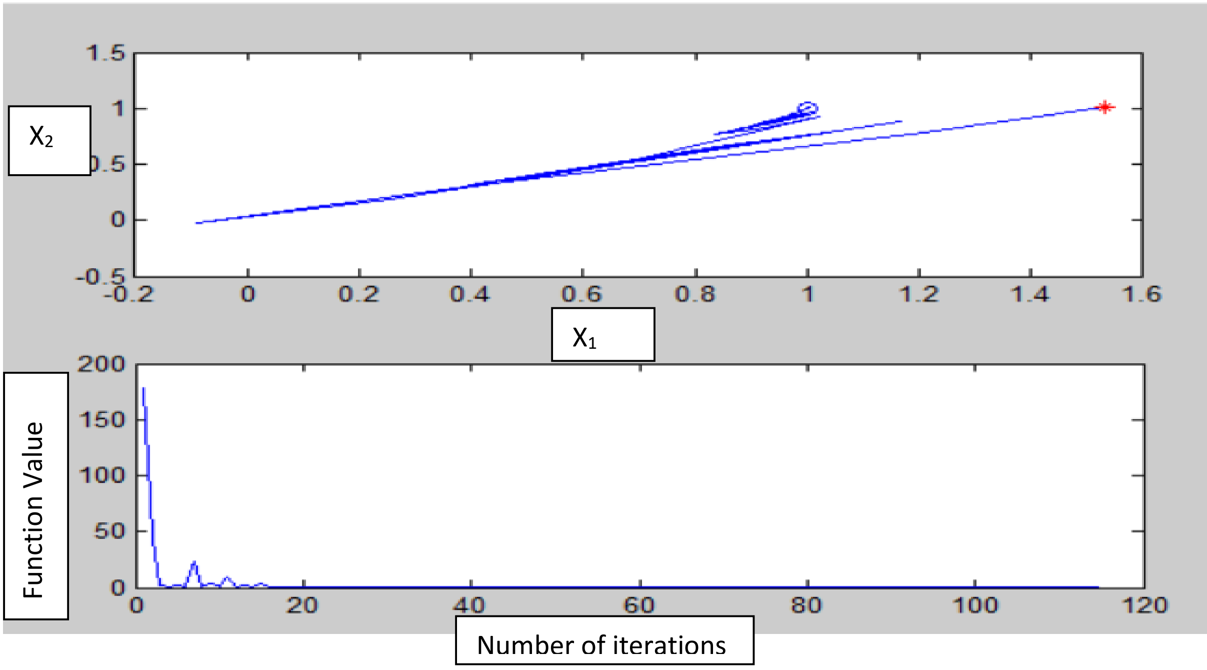


Fig-3(a) movement of the third particle to reach the optimum point of rosenbrock function

Fig-3(b) function value vs the number of iterations of the third particle of rosenbrock function

TABLE NO. -1 OPTIMUM VALUE OF ROSENBROCK FUNCTION W.R.T. TOLERANCE

S.no.	Max. no. Of iterations	Tolerance	Swarm size	No. of iterations	Min. Value of function	X_1	X_2
1	200	6	200	125	1.31809e-16	1.000e+00	1.000e+00
2	200	8	200	136	9.65645e-16	1.000e+00	1.000e+00
3	200	10	200	151	3.65661e-22	1.000e+00	1.000e+00
4	200	12	200	165	3.20474e-28	1.000e+00	1.000e+00
5	200	14	200	170	3.57452e-28	1.000e+00	1.000e+00
6	200	16	200	181	0	1	1

From the table, it is observed that-

- Table shows results of Rosenbrock function with the change of tolerance value.
- Results are tabulated in terms of maximum number of iterations, tolerance value, swarm size, Number of iterations, minimum value of function , X1 and X2.
- When we decrease the value of tolerance, the no. of iterations performed to reach the optimum point is also increased.
- With the decrease in the value of tolerance, the minimum value of the function gets close to zero. This means that a high value of tolerance will help us achieving with the better accuracy.
- Swarm size is kept at 200 for all the cases.
- Minimum value of function is 0 at x1= 1 and x2= 1.
- Maximum number of iterations is kept 200.
- Number of iterations required to optimize the Rosenbrock function is 181 for most accurate value.
- Minimum tolerance value is 10^(-16).

5.1.2 RESULTS OF PROPOSED PSO

Let us assume, number of iterations after which retardation factor is included=i

no. of particles =20

Retardation factor=0.2 & i= 10

maximum no. of iterations =200

tolerance=6 & no. of iterations=113

no. of function evalulations =4621

min. value of function=1.275733e-14 at x1=1.000000e+00 & x2=1.000000e+00

time = 1.481414e-01 s

TABLE NO. 2 RESULTS OF PROPOSED PSO IMPLEMENTED OVER ROSENBROCK FUNCTION -

S.no.	R.f.	i	No. of iterations performed	NFEV	Min. value of function	X_1	X_2	T (s)
1	0.6	10	121	4861	1.630679e-13	1.000000e+00	9.999999e-01	6.8462e-01
2	0.9	10	121	4781	2.494119e-12	9.999998e-01	9.999997e-01	2.4960e-01
3	0.2	10	94	3781	7.369479e-10	9.999975e-01	9.999940e-01	1.541633e-01
4	0.4	10	113	4541	5.148629e-01	1.071753e+00	1.148356e+00	2.632495e-01
5	0.5	10	107	4301	2.022258e-01	1.071753e+00	1.044969e+00	1.831072e-01
6	0.9	5	110	4421	8.720535e-14	1.000000e+00	1.000000e+00	3.507254e-01
7	0.1	5	114	4581	6.240301e-12	9.999998e-01	9.999998e-01	2.684040e-01
8	0.8	5	110	4421	2.831444e-10	1.000002e+00	1.000003e+00	2.736237e-01
9	0.6	5	97	3901	3.252178e-07	1.000057e+00	1.000115e+00	2.365150e-01

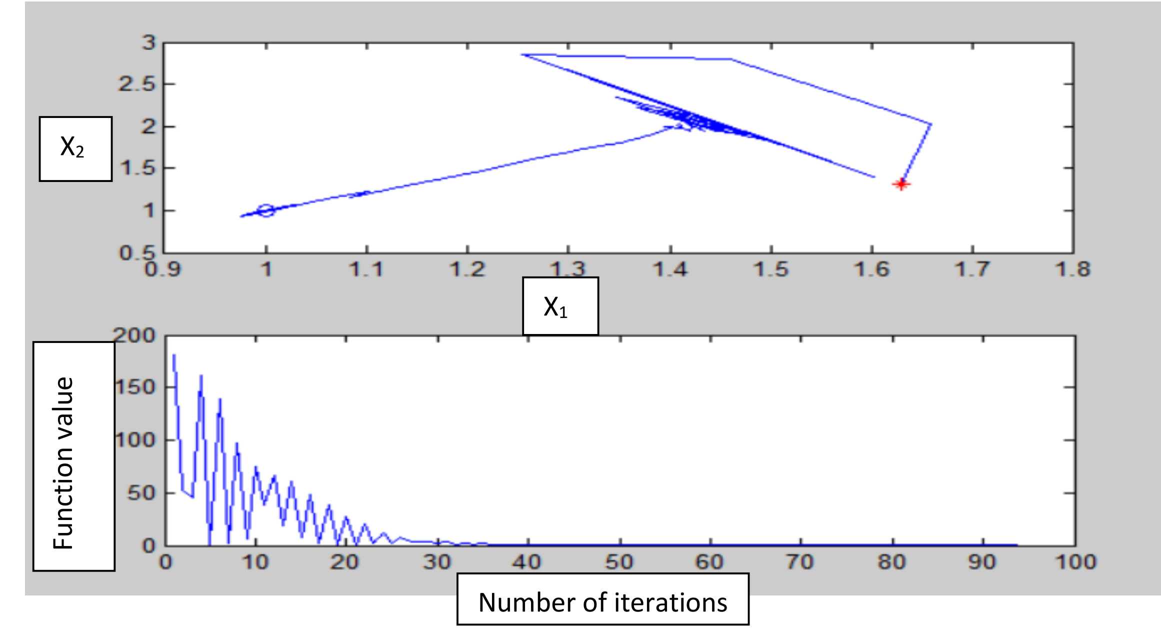


Fig-4(a) movement of first particle to reach the optimum point with r.f. of rosenbrock function

Fig-4(b) function value vs number of iterations of first particle of rosenbrock function with retardation factor

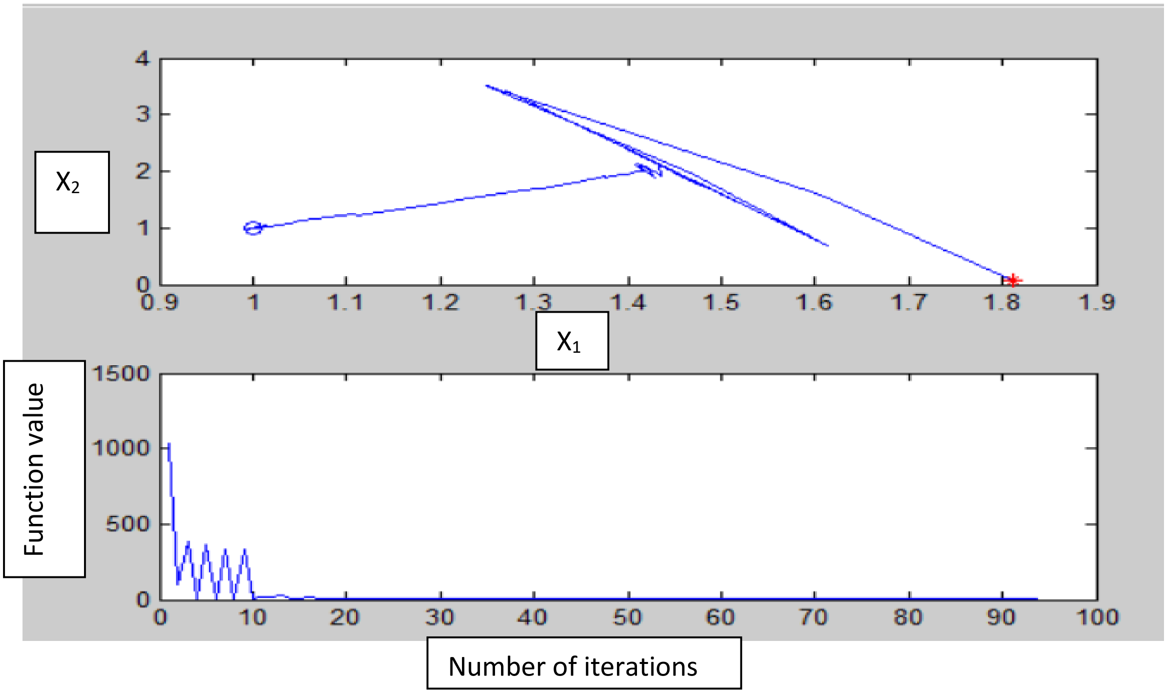


Fig-5(a) movement of second particle to reach the optimum point with r.f. of rosenbrock function

Fig-5(b) function value vs the number of iterations of rosenbrock function with retardation factor

5.1.3 DISCUSSION-

From table no.(2), it is found that we get the best results in terms of iterations required to reach optimum point and NFEV when we use the retardation factor as 0.2 and I as 10.Therefore, the plot of particles to reach the optimum point is drawn using these two values. No. of particles is 20 but we have taken the paths of 2 particles to understand the movement of particles in multidimensional space.Fig. 4(a) is the plot between variables x_1 and x_2, Fig.4(b) is between function value and a number of iterations. The starting point is denoted by + and the ending point is denoted by o.In fig.4(a), it is found that the starting point of a particle is (1.62,1.1). The first particle starts from this point and starts moving randomly in multidimensional space in search of an optimum point. Movement of the first particle stops at point (1,1).If fig.4(a) is compared with fig.1(a),2(a)&3(a), it is found that randomness in fig.4(a) is lesser than the other three.Fig.4(b) shows the plot between function value and a number of iterations. The function value starts from some random value and goes on decreasing till it reaches to zero .From fig.4(b) and 5(b) we can see that number of iterations required to reach function value equal to zero, is reduced in comparison with 1(b),2(b) &3(b).The number of iterations required to reach the optimum point is lowest in case of retardation factor(r.f.)=0.2 and i=10.The retardation factor is around 0.2, which shows that particles need deceleration to reach the optimum point. It is because when we multiply 0.2 with the velocity of the particle, the velocity of the particle gets reduced to a large extent. If the best results of the table no. (2) and table no. (1) are compared, it is found out that r.f.=0.2 and i=10 gives better results.Time consumed to reach the optimum point is also tabulated and compared. It has been found that time consumed will be least in case of r.f.=0.2 and i=10, which is 1.541633e-01 seconds. Thus, Minimum value of function is 7.369479e-10 at x_1=9.999975e-01 & x_2=9.999940e-01.The concept of NFEV (no. of function evaluations) is introduced. NFEV gives us knowledge about the number of times a function is calculated that includes, For each particle's every iteration, personal best position, and global best position.The number of times the function is evaluated (NFEV) is directly linked with the time consumed to reach the optimum point. It is because if the program needs to calculate the personal best and global best position for function evaluation for more number of times, it is obvious that the time consumed will be more. Therefore, for the low value of NFEV, the time consumed will also be less. Therefore, for r.f. =0.2 and i=10, NFEV is the lowest which is 3781.

5.2BOOTH FUNCTION-

$f(x_1, x_2)$=$(x_1 + 2 * x_2 - 7)^2 + (2 * x_1 + x_2 - 5)^2$

It has a global minimum at (x_1, x_2)=(1,3), where $f(x_1, x_2)$=0

5.2.1 RESULTS OF BASIC PSO-

no. of particles in a swarm= 20

maximum no. of iterations to be performed= 100

Number of iterations performed =79

tolerance value= 6

number of function evaluations= 3181

min value of function is 7.903305e-12 and at values of x1=9.999991e-01 and x2=3.000000e+00 , time = 6.086687e-01 s

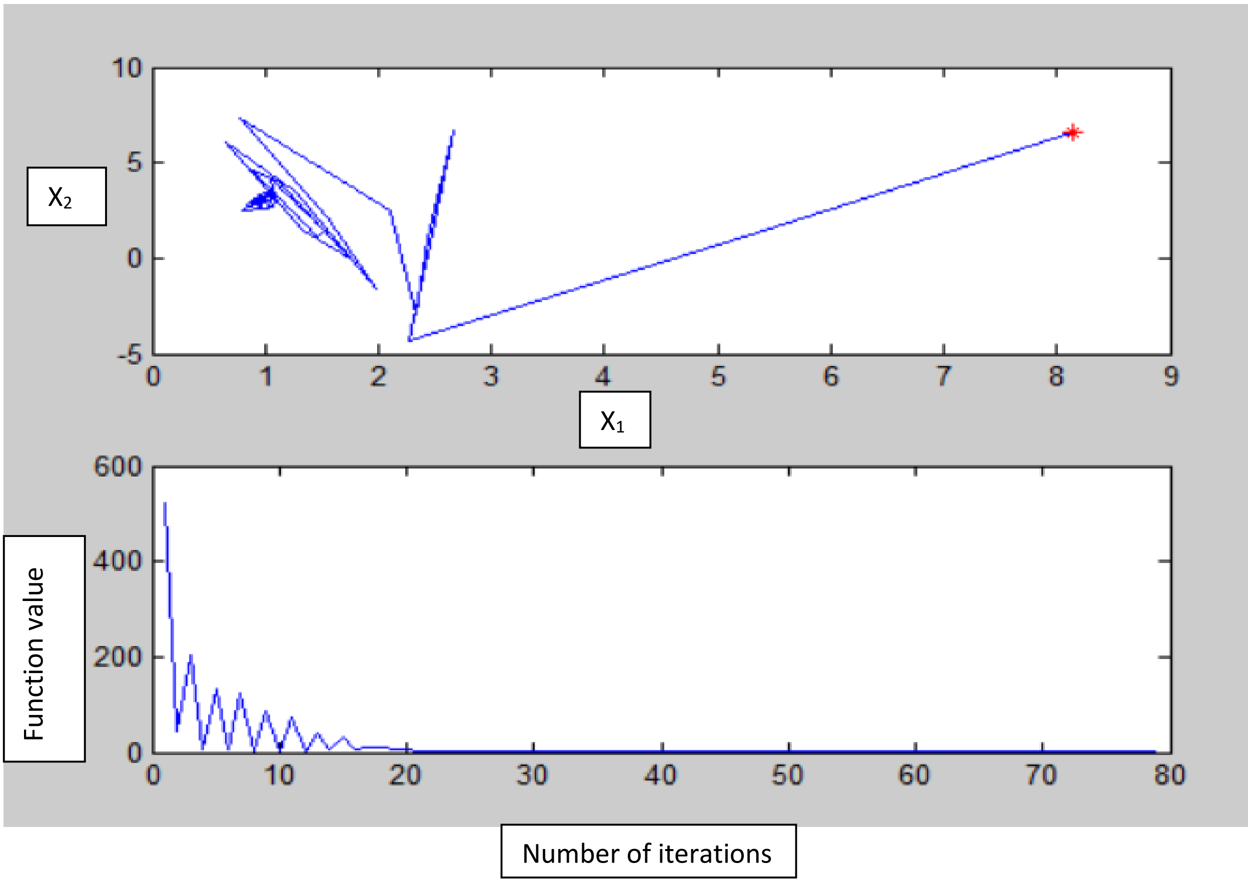


Fig-8(a) movement of the first particle to reach the optimum point of Booth function

Fig-8(b) function value vs the number of iterations of Booth function

5.2.2 RESULTS OF PROPOSED PSO

Function to minimize f = $f(x_1, x_2)$=$(x_1 + 2 * x_2 - 7)^2 + (2 * x_1 + x_2 - 5)^2$

no. of particles in a swarm= 20

maximum no. of iterations to be performed= 100

tolerance value= 6

Number of iterations performed=68

retardation factor= 0.9

number of function evaluations= 2741

min value of function is 1.149921e-10

at values of x1=9.999968e-01

x2= 3.000007e+00

time = 1.350434e-01 s

TABLE NO. 3 -- RESULTS OF PROPOSED PSO IMPLEMENTED OVER BOOTH FUNCTION

S.no.	r.f.	i	No. of Iterations performed	NFEV	Min. Value of function	X_1	X_2	Time Consumed (s)
1.	0.6	10	69	2781	2.661455e-12	1.000001e+00	2.999999e+00	1.515190e-01
2.	0.2	10	82	3301	5.476515e-13	1.000000e+00	3.000000e+00	2.186316e-01
3.	0.9	10	68	2741	1.149921e-10	9.999968e-01	3.000007e+00	1.350434e-01
4.	0.95	10	84	3381	6.858260e-15	9.999999e-01	3.000000e+00	1.692252e-01
5.	0.88	10	70	2821	8.418411e-10	1.000005e+00	2.999983e+00	1.495839e-01
6.	0.1	15	76	3061	6.292171e-13	1.000000e+00	3.000000e+00	1.443330e-01
7.	0.9	15	71	2861	9.028979e-11	1.000006e+00	2.999993e+00	1.395791e-01

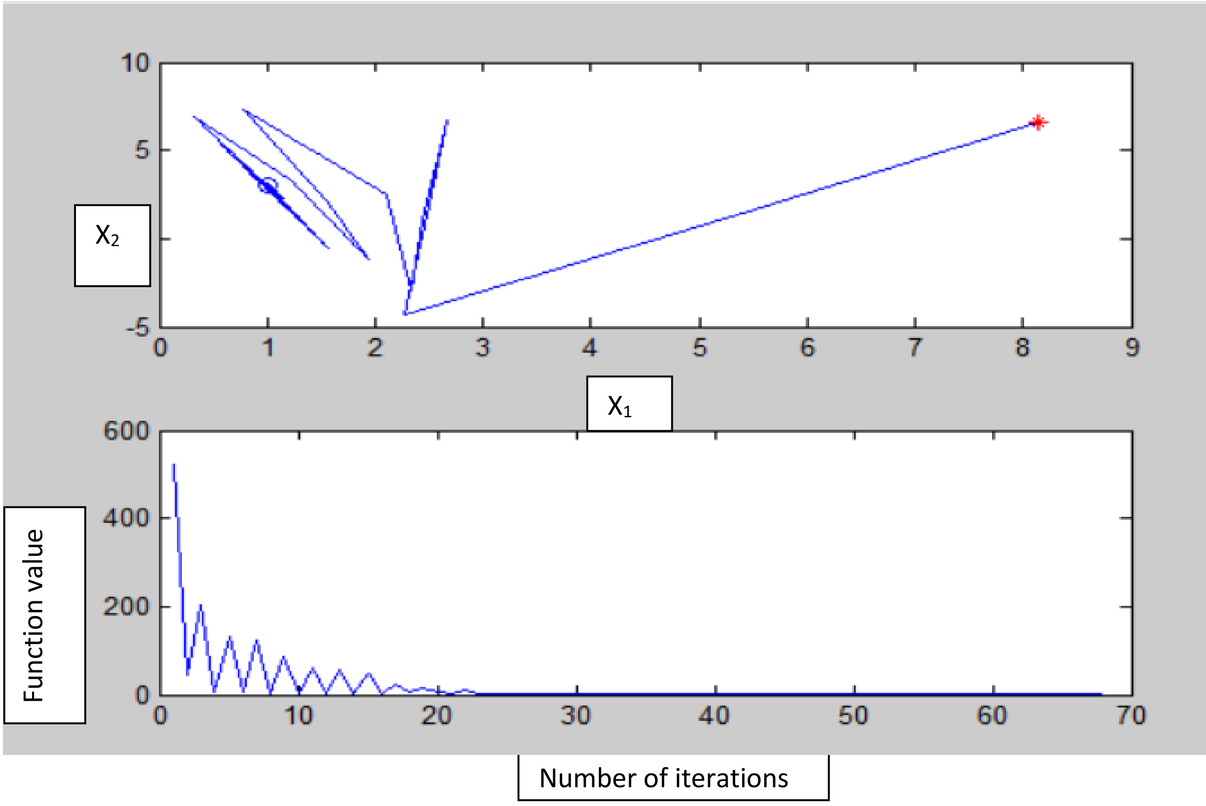


Fig-7(a) movement of the first particle to reach the optimum point with retardation factor of booth function

Fig-7(b) function value vs the number of iterations of booth function with retardation factor

5.2.3 DISCUSSION-

The table shows the results of the Booth function which includes retardation factor, Number of iterations performed, NFEV,x_1,x_2, and time consumed(T).PSO Algorithm is performed over Booth function with swarm size equal to 20 and the maximum number of iterations equal to 100.The number of iterations performed is lowest in the case of r.f.=0.9 & i=10.The retardation factor is around 0.9, which shows that particles don't need much deceleration to reach the optimum point in less number of iterations. It is because 0.9 is nearer to 1 and when we multiply with the velocity of the particle, the velocity of the particle gets reduced but not to a large extent .If compared with the results without the retardation factor, it gives better results in terms of the number of iterations performed. At r.f.=0.9 & i=10, Number of iterations will be 68, No. of function evaluations (NFEV) is 2741, Minimum value of function is 1.149921e-10 at x_1=9.999968e-01 & x_2=3.000007e+00.Time consumed will also be lowest in case of r.f.=0.9 & i=10 , which is 1.350434e-01 s. Fig. 6(a),6(b) shows the plot of particles of Booth function to reach the optimum point as well as the plot of function value vs a number of iterations without retardation factor.Fig. 7(a),7(b) shows the plot of particles of Booth function to reach the optimum point as well as the plot of function value vs the number of iterations with the retardation factor. The starting point of Fig.11(a) is approximately (8,7) and then particles start moving randomly in the 2-D space to reach the optimum point which is (1,3) in this case.Now, If we compare the fig. 6(a) & 7(a) it is found that Fig.6(a) has some randomness while reaching the optimum point.Now, if we see fig 7(a) randomness gets reduced which lets the particle reach the optimum point in less time.Fig.6(b) &7(b) shows the plot of function value vs the number of iterations without and with retardation factor respectively.

5.3 EASOM FUNCTION-

$f(x_1,x_2) = -\cos(x_1) * \cos(x_2) * e^{(-((x_1-3.14)^2+(x_2-3.14)^2))}$

It has a global minimum at (x_1,x_2)=(3.14,3.14) where $f(x_1,x_2)$=0

5.3.1 RESULTS OF BASIC PSO

no. of particles in a swarm= 20maximum no. of iterations to be performed= 100number of iterations performed=64 tolerance value= 6, number of function evaluations= 2581 min value of function is -9.999983e-01 and at values of x1=3.140522e+00 and x2=3.140548e+00, time = 1.841650e-01 s

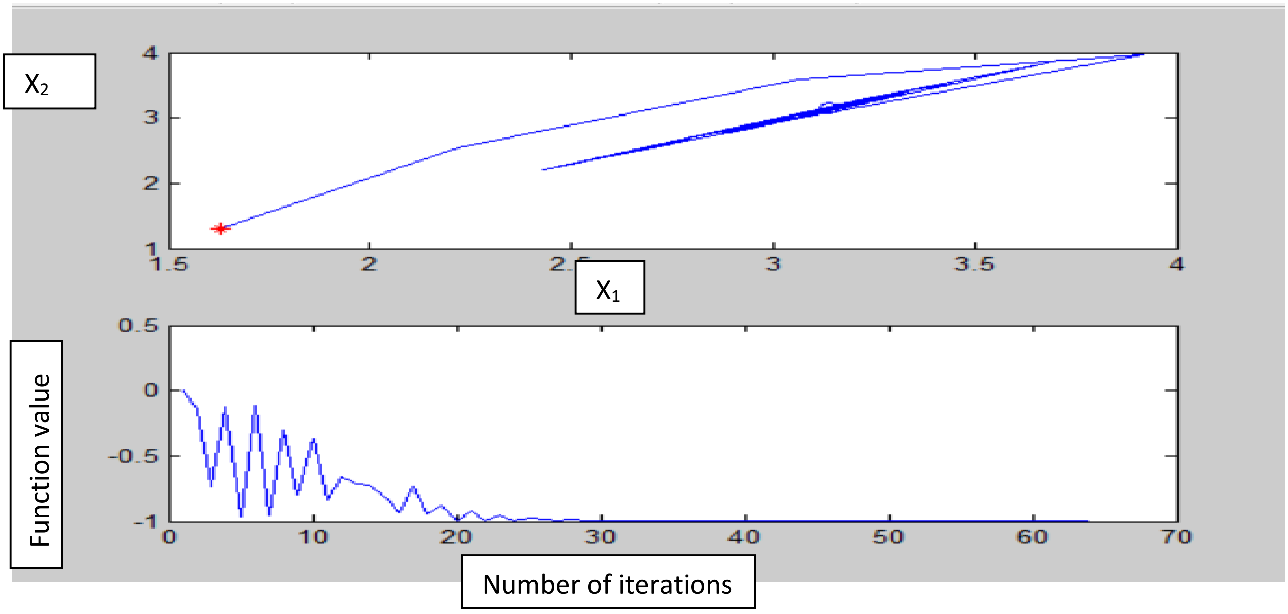


Fig-8(a) movement of the first particle to reach the optimum point of Easom function

Fig-8(b) function value vs the number of iterations of easom function

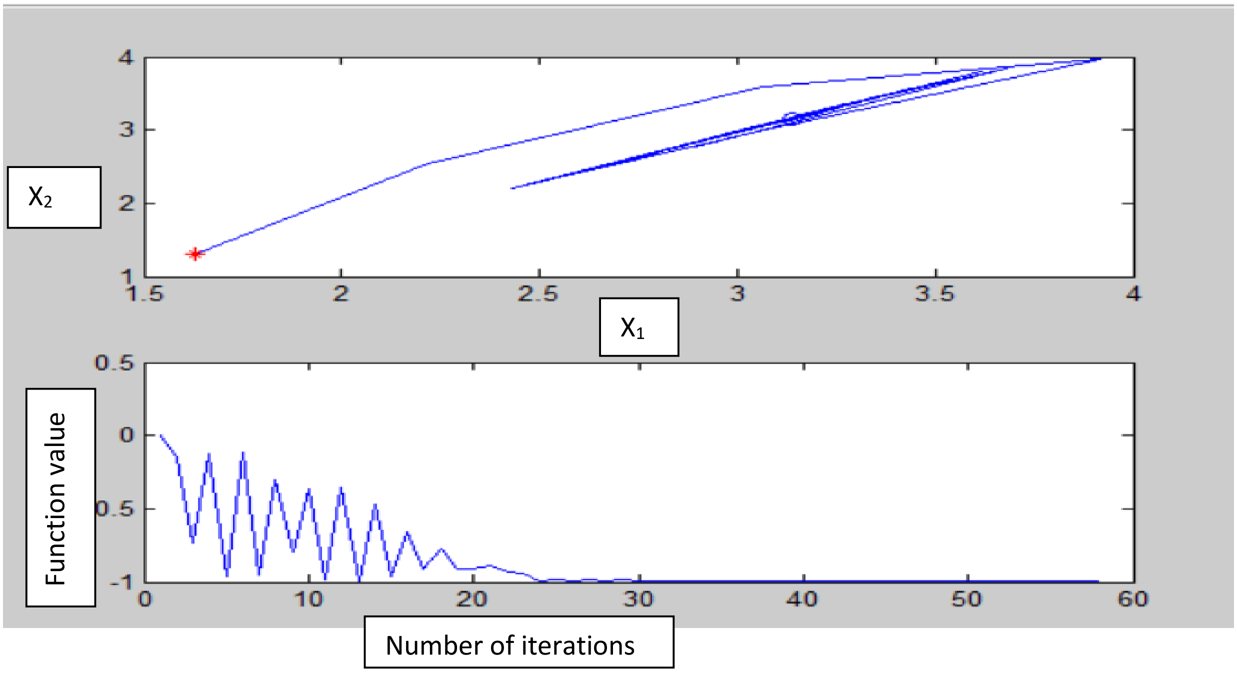


Fig-9(a) movement of first particle to reach the optimum point with retardation factor of Easom function

Fig-9(b) function value vs number of iterations of easom function with retardation factor

5.3.2 RESULTS OF PROPOSED PSO

no. of particles in a swarm= 20

maximum no. of iterations to be performed= 100

Number of iterations performed =58

tolerance value= 6

retardation factor= 0.8

number of function evaluations= 2341

min value of function is -9.999983e-01

at values of x1=3.140628e+00

x2=3.140508e+00

time = 1.088167e-01 s.

TABLE NO 4 RESULTS OF PROPOSED PSO IMPLEMENTED OVER EASOM FUNCTION

S.no.	r.f.	i	No. of iterations performed	NFEV	Min. Value of function	X_1	X_2	Time Consumed (sec)
1	0.1	10	66	2661	-9.999983e-01	3.140534e+00	3.140523e+00	1.726781e-01
2	0.8	10	58	2341	-9.999983e-01	3.140628e+00	3.140508e+00	1.371823e-01
3	0.9	10	60	2421	-9.999983e-01	3.140507e+00	3.140540e+00	1.492741e-01
4	0.7	10	64	2581	-9.999983e-01	3.140528e+00	3.140539e+00	1.567039e-01
5	0.8	5	70	2821	-9.999983e-01	3.140530e+00	3.140528e+00	1.411229e-01
6	0.9	5	59	2381	-9.999983e-01	3.140543e+00	3.140571e+00	1.396203e-01
7	0.1	5	66	2661	-9.999983e-01	3.140534e+00	3.140531e+00	1.521112e-01

5.3.3 DISCUSSION-

Table shows the results of Easom function which includes retardation factor, Number of iterations performed,NFEV,x_1,x_2 and time consumed(T) .PSO Algorithm is performed over Easom function with swarm size equal to 20 and the maximum number of iterations equal to 100.The number of iterations taken to reach the optimum point in the basic PSO algorithm is equal to 64.Therefore, using the retardation factor we will try to optimize the function with a lesser number of iterations and lesser value of NFEV. At r.f. =0.8 and i=10 , function gets optimised in 58 iterations & NFEV is also the lowest which is 2341.The retardation factor is around 0.8, which shows that particles don’t need much deceleration to reach the optimum point in less number of iterations. It is because 0.8 is nearer to 1 and when we multiply with the velocity of the particle, the velocity of the particle gets reduced but not to a large extent. Therefore, Minimum value of the function is equal to -9.999983e-01 at x_1=3.140628e+00 & x_2=3.140508e+00.Time consumed is to reach the optimum point is also the lowest which is 1.371823e-01 s.Fig. 8(a),8(b) shows the plot of particles of Easom function to reach the optimum point as well the plot of function value vs the number of iterations without retardation factor.Fig. 9(a),9(b) shows the plot of particles of Easom function to reach the optimum point as well the plot of function value vs the number of iterations without retardation factor.Best results are given by r.f. =0.8 & i= 10, so fig 9(a) &9(b) are based on these two values. Now, these two plots are compared, it is found that randomness in particles reduced slightly in case of 9(a).Reduction in randomness makes the particle to reach the particle in less amount of iterations and time.Fig 8(b) and 9(b) shows the plot between function value and the number of iterations.If these two plots are compared, we see that fig 9(b) reaches a minimum value of a function, which is -9.999983e-01 in this case, in a lesser number of iterations. The starting point is denoted by + and the ending point is denoted by o. In fig.8(a), it is found that the starting point of a particle is (3,3).

5.4 THREE HUMP CAMEL FUNCTION-

Function to minimize f = $2 * (x_1^2) - 1.05 * (x_1^4) + \frac{x_1^6}{6} + (x_1 * x_2) + (x_2^2)$

It has a global minimum at (x_1, x_2)=(0,0)where $f(x_1, x_2)$=0

5.4.1 RESULTS OF BASIC PSO

no. of particles in a swarm= 20

maximum no. of iterations to be performed= 100

No. of iterations performed=63

tolerance value= 6

number of function evaluations= 2541

min value of function is 1.515304e-09 and at values of x1=2.154484e-05 and x2=-3.728636e-05

time = 4.783367e-01 s

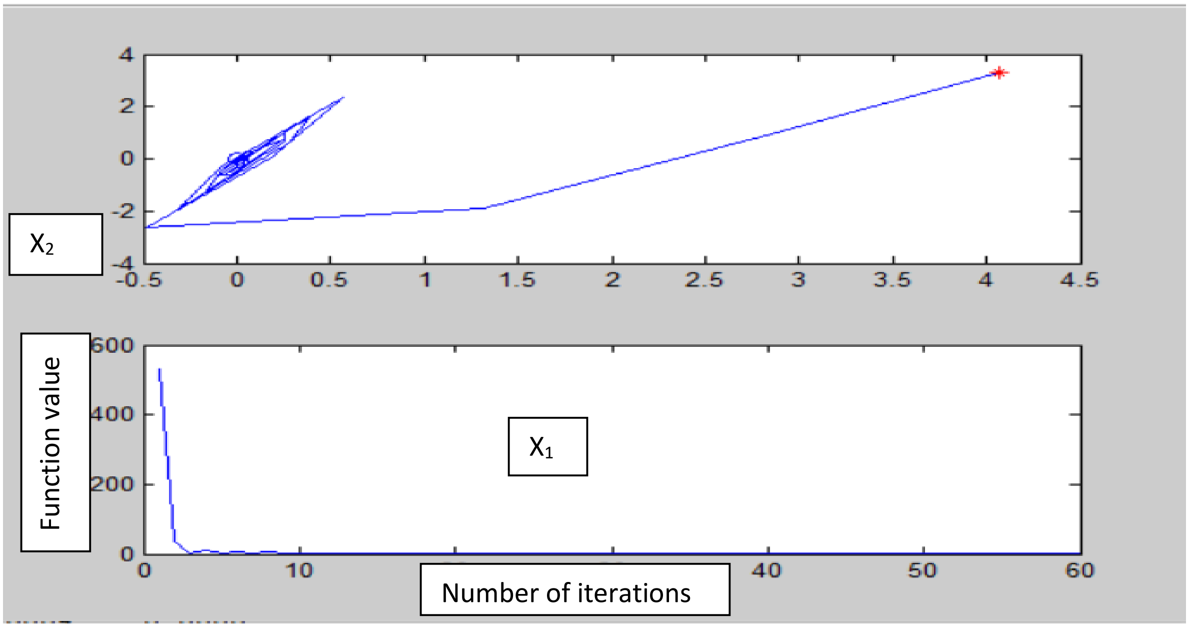


Fig-10(a) movement of the first particle to reach the optimum point of three hump camel function

Fig-10(b) function value vs the number of iterations of three hump camel function

5.4.2 RESULTS OF PROPOSED PSO

No.of particles in a swarm= 20,maximum no. of iterations to be performed= 100,No. of iterations performed=60,tolerance value= 6,retardation factor= 0.9,number of function evaluations= 2421,min value of function is 8.370086e-10 at values of x1=6.570970e-06 and x2=2.430886e-05 time = 1.163674e-01 s

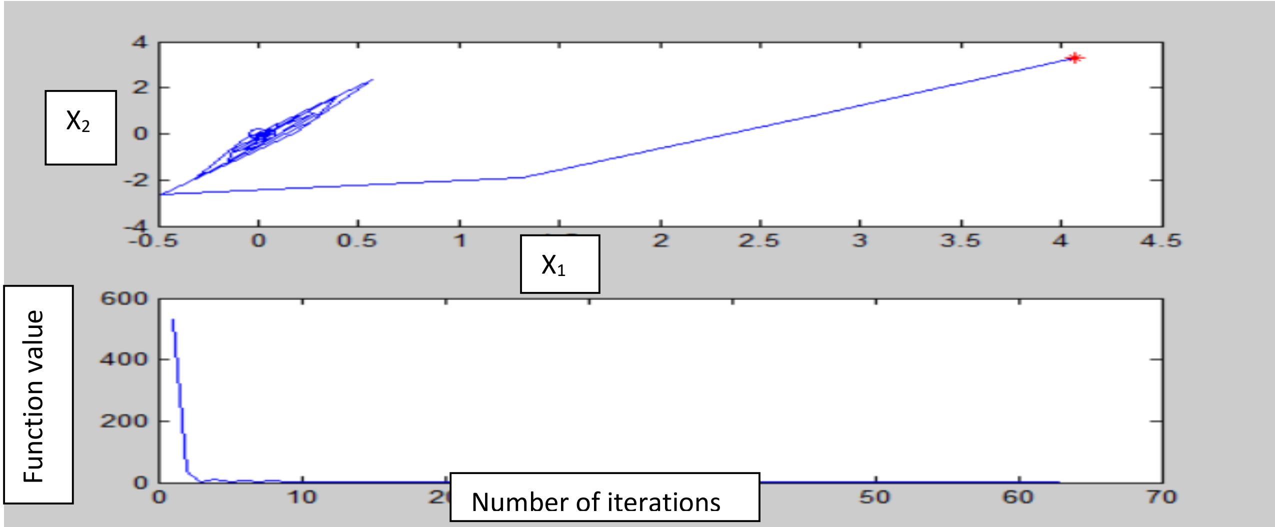


Fig-11(a) movement of the first particle to reach the optimum point with retardation factor of three hump camel function

Fig-11(b) function value vs the number of iterations of three hump camel function with retardation factor

TABLE NO 5- RESULTS OF PROPOSED PSO IMPLEMENTED OVER THREE HUMP CAMEL FUNCTION

S.no.	r.f.	i	No.of iterations performed	NFEV	Min value Of function	X_1	X_2	Time Consumed (sec)
1	0.9	10	60	2421	8.370086e-10	6.570970e-06	2.430886e-05	1.317127e-01
2	0.95	10	62	2501	1.087664e-09	-2.440909e-05	1.891326e-05	1.392860e-01
3	0.8	10	74	2981	5.516790e-12	4.930763e-07	-2.502934e-06	1.694553e-01
4	0.1	10	67	2701	1.330152e-11	-1.191616e-06	3.884669e-06	1.603564e-01
5	0.2	10	66	2661	6.180816e-10	1.870724e-05	-1.173067e-05	1.511186e-01
6	0.9	5	78	3141	2.545495e-14	2.873941e-08	-1.693198e-07	1.554695e-01
7	0.95	5	67	2701	2.806864e-10	8.582395e-07	-1.714430e-05	1.313246e-01

5.4.3 DISCUSSION

Fig. 10(a),10(b) shows the plot of particles of Three hump camel function to reach the optimum point as well the plot of function value vs the number of iterations without retardation factor. Fig. 11(a),11(b) shows the plot of particles of Three hump camel function to reach the optimum point as well the plot of function value vs the number of iterations without retardation factor. Best results are given by r.f. =0.9 & i= 10, so fig 11(a) &11(b) are based on these two values. The starting point is denoted by + and the ending point is denoted by o. The starting point of Fig.11(a) is approximately (4.2,3) and then particles start moving randomly in the 2-D space to reach the optimum point which is (0,0) in this case. If these two plots are compared, we see that fig 11(b) reaches a minimum value of a function, which is 0 in this case, in a lesser number of iterations. The table shows the results of Three hump camel function which includes retardation factor, Number of iterations performed, NFEV,x_1,x_2, and time consumed(T).PSO Algorithm is performed over Three hump camel functions with swarm size equal to 20 and the maximum number of iterations equal to 100.The number of iterations taken to reach the optimum point in the basic PSO algorithm is equal to 63.At r.f. =0.9 and i=10 , function gets optimised in 60 iterations & NFEV is also the lowest which is 2421.The retardation factor is around 0.9, which shows that particles don't need much deceleration to reach the optimum point in less number of iterations. It is because 0.9 is nearer to 1 and when we multiply with the velocity of a particle, the velocity of the particle gets reduced but not to to a large extent. Therefore, Minimum value of the function is equal to 8.370086e-10 at x_1=6.570970e-06 & x_2=2.430886e-05.Time consumed is to reach the optimum point is also the lowest which is 1.317127e-01 s.

5.5 COMPARISON OF RESULTS OF BPSO AND PROPOSED PSO

After implementing the basic PSO and proposed PSO on different mathematical test functions.The results of both the cases are compared in terms of number of iterations and number of function evaluations. The retardation factors giving best results are taken and compared with the BPSO results.

TABLE NO.6- COMPARISON OF ITERATION PERFORMED AND NFEV

BASIC PSO					PROPOSED PSO			
S.no.	**Function**	**Maximum Number of iterations**	**Iterations performed**	**NFEV**	**Maximum No. of Iterations**	**NFEV**	**Retardation Factor**	**Iterations performed**
1	Rosenbrock Function	200	113	4621	200	3781	0.2	94
2	Booth Function	100	79	3181	100	2741	0.9	68
3	Easom function	100	64	2581	100	2341	0.8	58
4	Three hump Camel func.	100	63	2541	100	2421	0.9	60

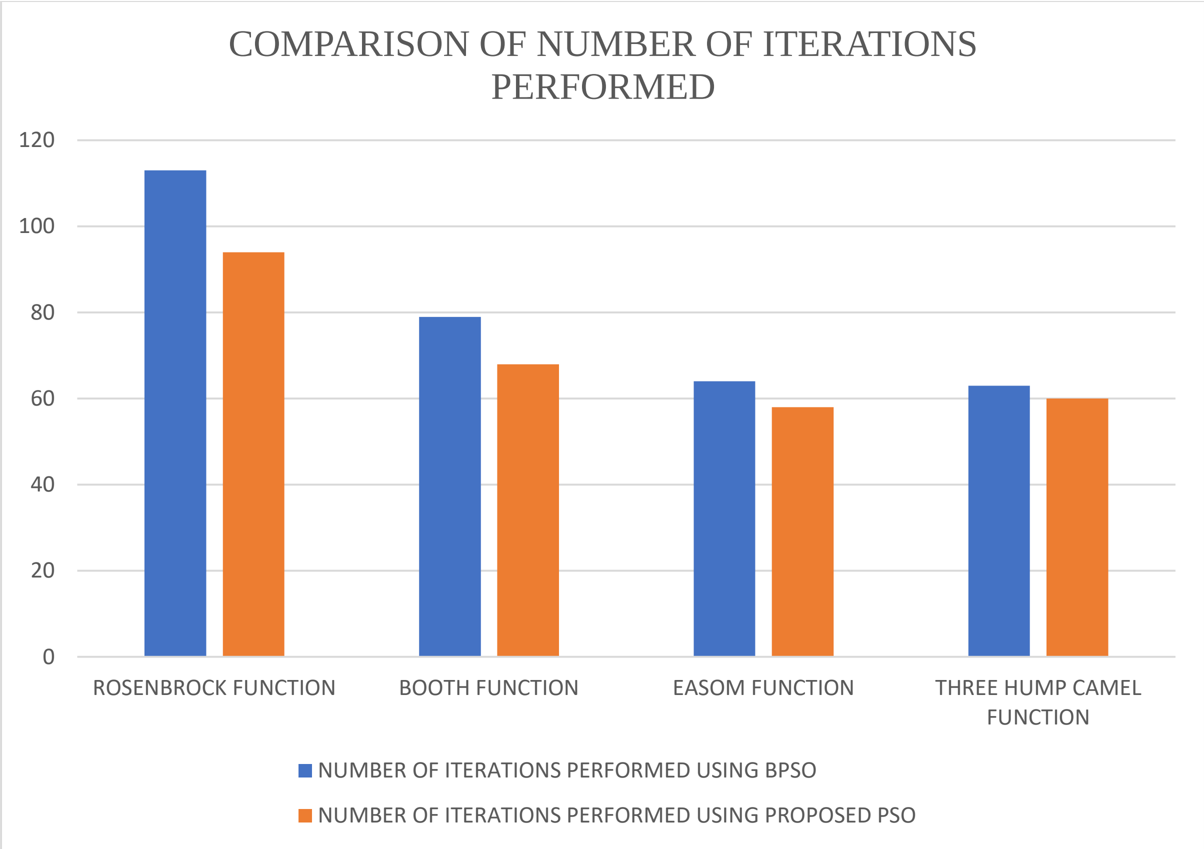


Fig no 12

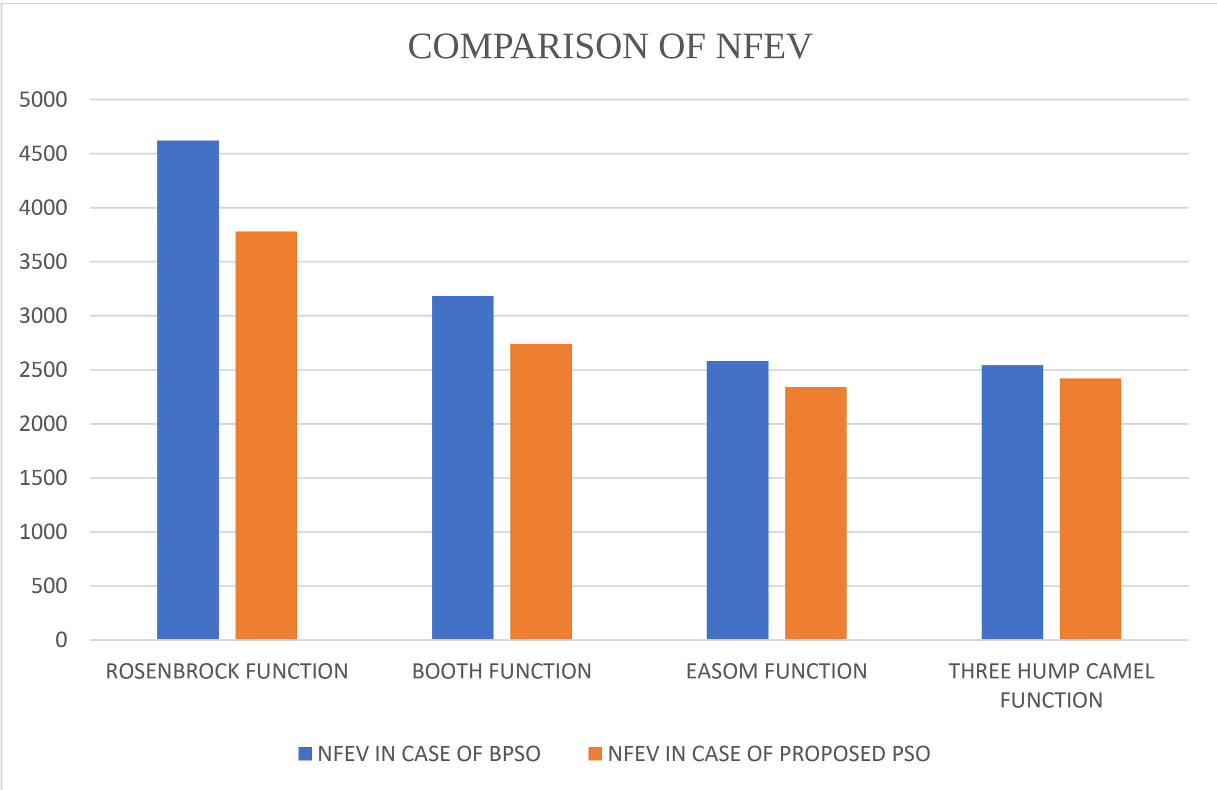


Fig no 13

5.6 CONCLUSION

Following are the conclusions of this chapter 5:

1. To check the authenticity of the proposed PSO, it is implemented over four mathematical test functions.

2. Proposed PSO has been successfully implemented and we have got desirable results in terms of iterations performed & NFEV.

3. Plots of particles have been drawn and the path of particles have analyzed.

4. Now, the proposed PSO can be implemented over the economic load dispatch problem as we know that it is an effective way of optimizing an objective function.

CHAPTER-6: IMPLEMENTATION OF BPSO AND PROPOSED PSO TO ECONOMIC LOAD DISPATCH

6.1 INTRODUCTION

The fuel cost function of the economic load dispatch problem is nonlinear and complex to optimize. To optimize the economic load dispatch problem, there are different methods like genetic algorithm, tabu search. But in this project work, the Particle swarm optimization method is used to optimize the economic load dispatch. Basic PSO and proposed PSO methods are used to optimize the fuel cost function. Basic PSO follows the conventional method given by Kennedy and Eberhart [5]. Proposed PSO has introduced the new concept of retardation factor and implemented on economic load dispatch problem. This method is inspired by the bird flocking and follows the bird's movement to reach the global optimum point. First of all Basic PSO has been implemented on the IEEE 3 & 6 generator system. In these two systems, three and six generators do generations simultaneously. After optimizing the economic load dispatch problem using basic PSO, we have implemented the proposed PSO to optimize the economic load dispatch problem.

6.2 FORMULATION OF ECONOMIC LOAD DISPATCH PROBLEM

The objective of the economic load dispatch problem to minimize the cost of fuel. Generations made should be fulfilling the load demand in such a manner that the fuel cost should be minimum.

Also, we can represent it like

$F_t = \sum_{i=1}^{n} F_i(P_i)$

Also, $P_D + P_L = \sum_{i=1}^{n} P_i$

F_t is a function of the generation of plant and it is the function to be minimized.

$F_t = \sum_{i=1}^{n} A_i * P_i^2 + B_i * P_i + C_i$

F_t is a function of the generation of plants and is a function to be minimized.

Where F_t is the total fuel cost of the thermal power plant.

A_i, B_i and C_i are the cost coefficients of generators.

Also, using the power balance equation

$$P_D = \sum_{i=1}^{n} P_i$$

Total transmission losses are calculated as

$$P_L = \sum\sum P_m * B_{mn} * P_n$$

B_{mn} are known as transmission loss coefficients

The generators should be satisfying following inequality

$P_i^{min} \leq P_i \leq P_i^{max}$

P_i^{min} and P_i^{max} are the power limits of ith generator

Power generations made should be by each generator should be in the range of maximum power and minimum power limits of themselves.

6.3 ECONOMIC LOAD DISPATCH USING BPSO

In this case, BPSO algorithm is implemented on economic load dispatch problem. Algorithm followed to optimize the fuel cost function is discussed in chapter 4.3, flowchart of proposed PSO algorithm.

Number of particles in swarm =100

Maximum number of iterations= 5000

Tolerance value =10^(-5)

W_{max}=0.9

w_{min}=0.4

In case of IEEE-3 generator system input data are taken as

Pd=812.57 and 585.3 MW

a=[561 310 78]

b=[7.92 7.85 7.97]

c=[0.00156 0.00194 0.00482]

pmax=[600 400 200] MW

pmin=[100 100 50] MW

P_{max} & P_{min} = maximum and minimum power generation limits of three units respectively

In case of IEEE-6 generator system input data are taken as

Pd=700 & 800 MW

Pd= load demand

a=[756.79886 451.32513 1049.9977 1243.5311 1658.5596 1356.6592];

b=[38.53973 46.15916 40.39655 38.30553 36.32782 38.27041];

c=[0.15240 0.10587 0.02803 0.03546 0.02111 0.01799];

a,b &c are the cost coefficients.

These are used to make the fuel cost function.

pmax=[125 150 225 210 325 315]MW

pmin=[10 10 35 35 130 125]MW

P_d = Power demands of load

P_{max} & P_{min} = maximum and minimum power generation limits of six units respectively

6.3.1 RESULTS OF IEEE 3 GENERATOR SYSTEM

In the case of the IEEE 3 generator system basic PSO is implemented over two load demands, the first one is 812.57 MW & the second one is 585.3 MW. Generations made by units, transmission losses, fuel cost, number of iterations performed, time consumed and NFEV are calculated.

Table no.- 7 OPTIMAL SCHEDULING OF UNITS OF IEEE-3 GENERATOR SYSTEM USING BPSO

S.no.	P1 (MW)	P2 (MW)	P3 (MW)	Pd (MW)	Losses (MW)	Fuel cost (R/Hr)	Iterations Performed	Time (Sec)	NFEV
1.	392.872	333.959	121.967	812.57	36.22	8183.04	554	4.35869	110800
2.	221.185	267.502	112.145	585.3	15.4959	5970.23	574	5.49043	114800

6.3.2 RESULTS OF IEEE 6 GENERATOR SYSTEM

In the case of the IEEE 6 generator system basic PSO and proposed PSO is implemented on the system having two load demands. The first one is 700 MW & the other one is 800 MW. Generations made by units, transmission losses, fuel cost, number of iterations performed, time consumed and NFEV are calculated.

.TABLE NO.- 8 OPTIMAL SCHEDULING OF UNITS OF IEEE-6 GENERATOR SYSTEM USING BPSO

S. No	P1 (MW)	P2 (MW)	P3 (MW)	P4 (MW)	P5 (MW)	P6 (MW)	Pd (MW)	Losses (MW)	Fuel cost (R/Hr)	iter	Time	NFEV
1	28.67	10	116.57	120.30	231.65	212.23	700	19.4563	36912.8	392	5.10299	78400
2	33.07	13.45	138.78	138.07	259.35	242.65	800	25.4056	41898.2	820	11.4038	164000

6.4 ECONOMIC LOAD DISPATCH USING PROPOSED PSO

In this case, proposed PSO algorithm is implemented on economic load dispatch problem. Algorithm followed to optimize the fuel cost function is discussed in chapter 4.3, flowchart of proposed PSO algorithm.

6.4.1 RESULTS OF IEEE-3 GENERATOR SYSTEM-

In the case of the IEEE 3 generator system, proposed PSO is implemented on the system having two load demands. The first one is 812.57 MW & the other one is 585.3 MW. Generations made by units, transmission losses, fuel cost, number of iterations performed, time consumed and NFEV are calculated.

Table no. -9 OPTIMAL SCHEDULING OF UNITS OF IEEE-3 GENERATOR SYSTEM USING PROPOSED PSO

S.No.	P1 (MW)	P2 (MW)	P3 (MW)	Pd (MW)	Losses (MW)	Fuel cost (R/Hr)	Iterations performed	Time (sec)	NFEV	R.f.
1	392.86	333.956	121.966	812.57	36.2242	8182.93	538	3.91295	107600	0.5
2	221.16	267.486	112.138	585.3	15.4938	5969.88	544	3.95074	108800	0.7

6.4.2 RESULTS OF IEEE-6 GENERATOR SYSTEM-

The results of IEEE -6 generator system are found using the proposed PSO algorithm. In this case, proposed PSO algorithm is implemented on economic load dispatch problem. In the case of the IEEE 6 generator system, proposed PSO is implemented on the system having two load demands. The first one is 700 MW & the other one is 800 MW. Generations made by units, transmission losses, fuel cost, number of iterations performed, time consumed and NFEV are calculated.

TABLE NO.-10 OPTIMAL SCHEDULING OF UNITS OF IEEE-6 GENERATOR SYSTEM USING PROPOSED PSO

S. No.	P1 (MW)	P2 (MW)	P3 (MW)	P4 (MW)	P5 (MW)	P6 (MW)	Pd (MW)	Losses (MW)	Fuel cost (R/Hr)	iter	Time (sec)	NFEV	r.f.
1	28.679	10	116.57	120.30	231.65	212.23	700	19.456	36912.7	153	1.93	30600	0.8
2	33.076	13.4	138.78	138.07	259.35	242.65	800	25.405	41897.6	612	7.4318	122400	0.7

6.5 COMPARISON OF RESULTS

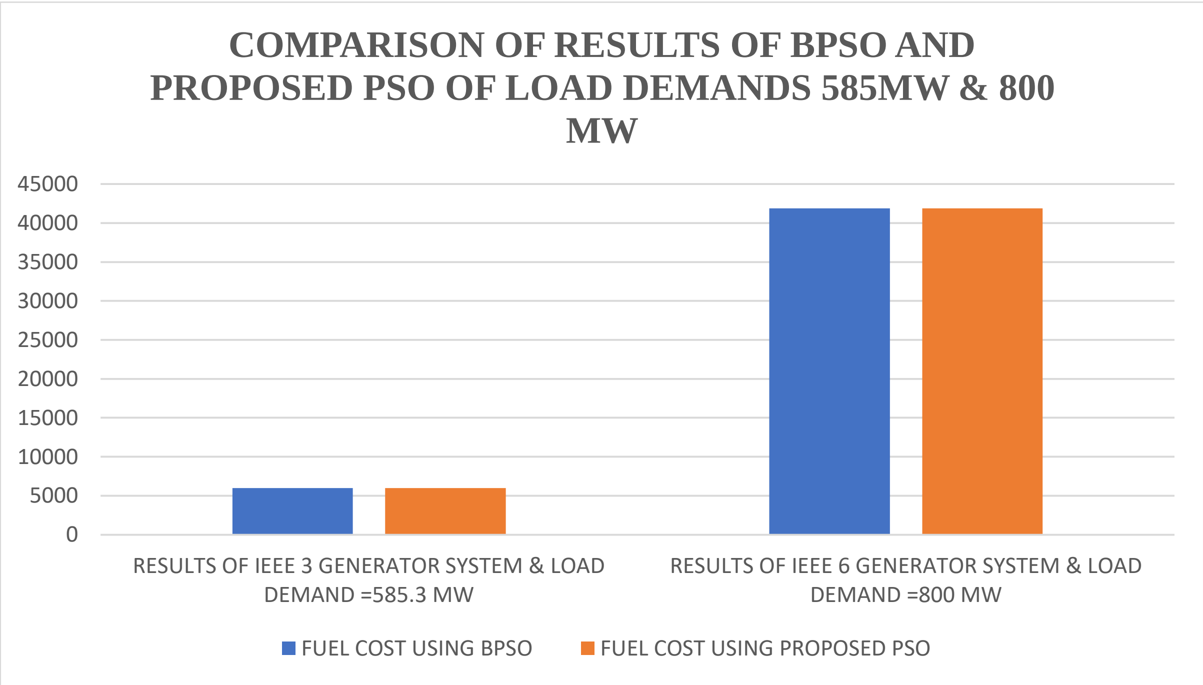


Fig no 14

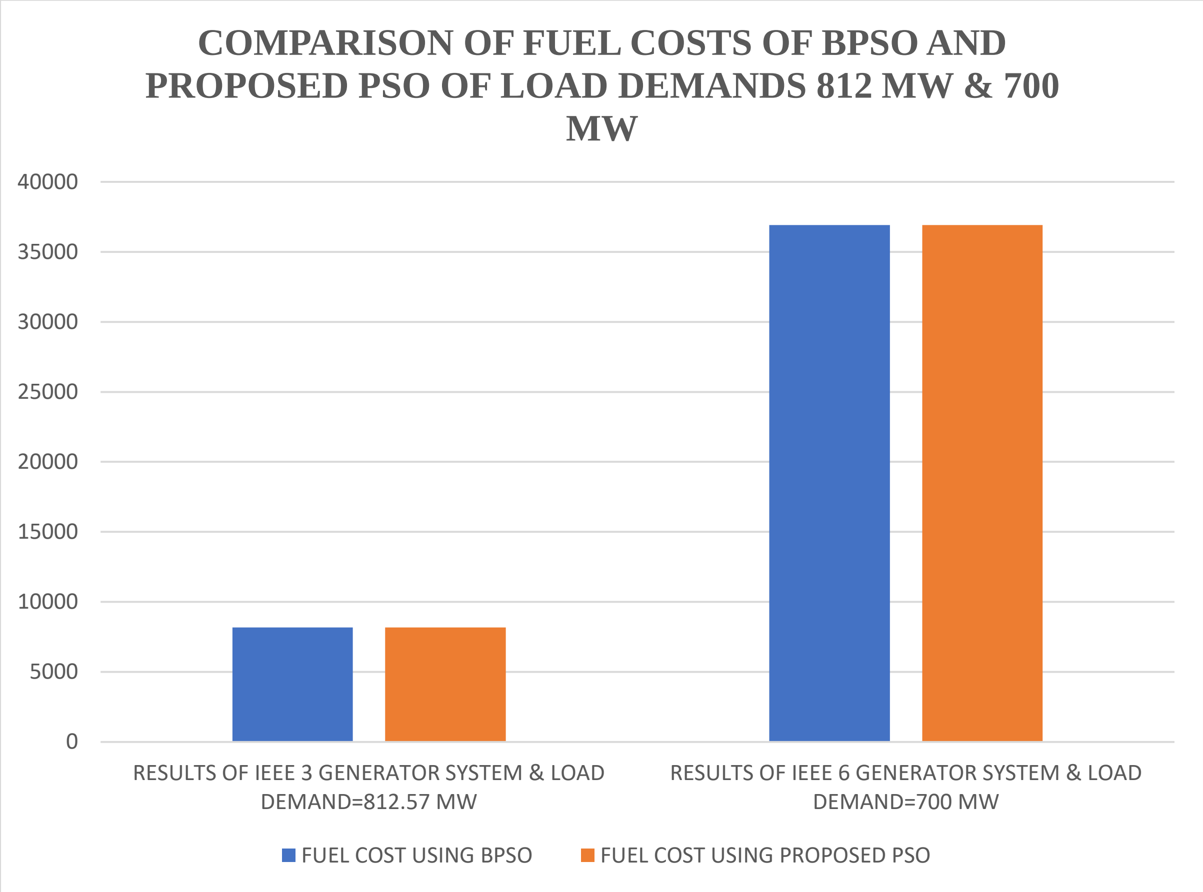


Fig no 15

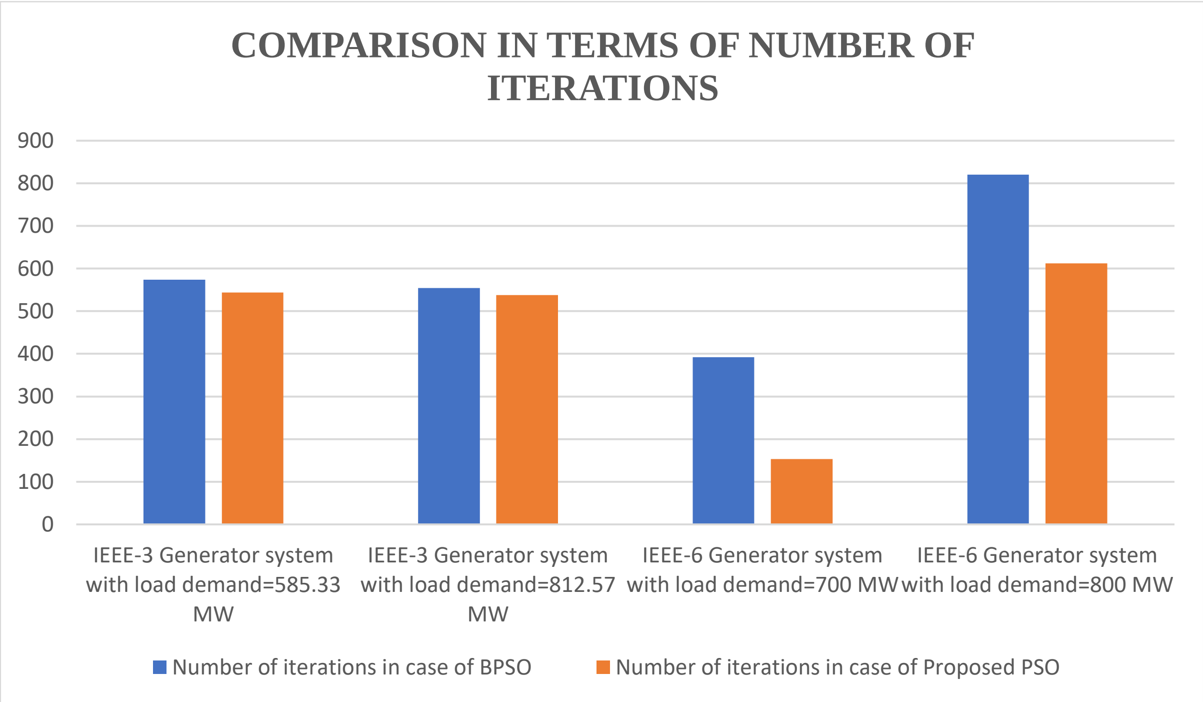


Fig no 16

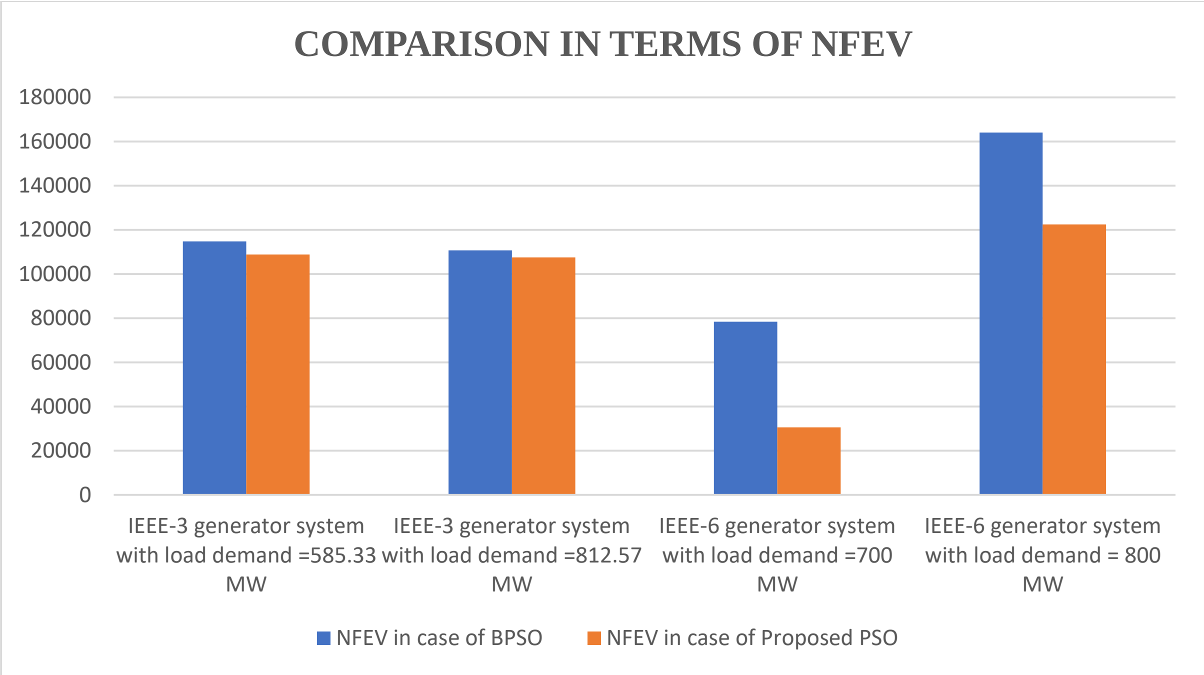


Fig no 17

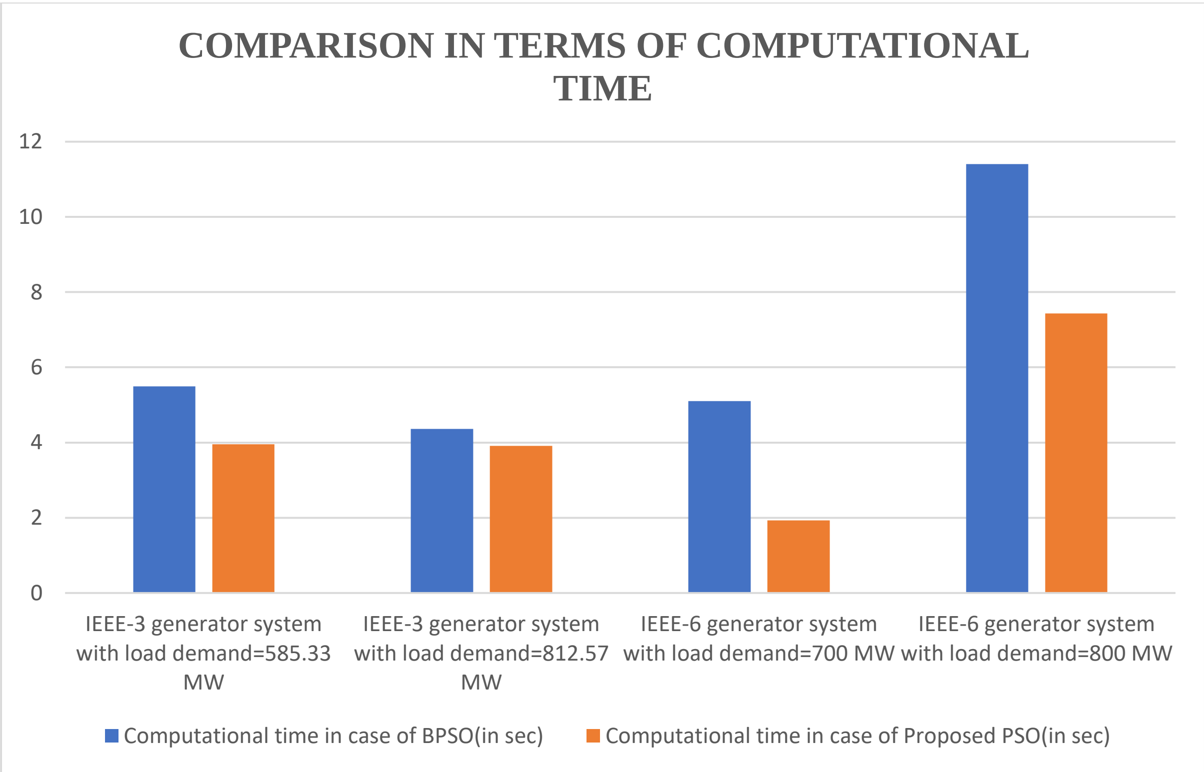


Fig no 18

6.6 CONCLUSION

Following are the conclusions of this chapter 6:

1. The basic PSO method and proposed PSO method are implemented on IEEE 3 generator and 6 generator system.
2. Generations made by IEEE 3 generator and 6 generator system according to the load demand, transmission losses & Fuel costs are tabulated.
3. Value of C_1 and C_2 are 2 &2. Inertia weight is between 0.9 & 0.4.
4. Cost coefficients, B-matrix, and power demand are taken to be the same in both the cases of basic and proposed PSO algorithm.
5. Economic load dispatch is done in such a manner that fuel cost is minimum. Generations made by generators are calculated and found to be in the range of their power limits.
6. Total power generation is greater than the load demand because transmission losses are also here. To fulfill load demand, power generation should be higher as a certain amount is going to be wasted in losses.
7. The algorithm used here is discussed in the flowchart of basic PSO and proposed PSO implemented over economic load dispatch. The retardation factor is included in the case of the proposed PSO algorithm.
8. In the case of the IEEE 6 generator system basic PSO and proposed PSO is implemented on the system having two load demands. The first one is 700 MW & the other one is 800 MW.
9. In the case of the IEEE 3 generator system basic PSO and Proposed PSO is implemented over two load demands, the first one is 812.57 MW & the second one is 585.3 MW.
10. If the results of table (9) and (10) are compared with the results of table(7) and (8), it has found that fuel cost in the Proposed method is less than that of basic PSO.
11. As the fuel cost, in this case, is less, it means that the proposed PSO gives better results in comparison to basic PSO. Also, it can be said that the proposed PSO is as accurate as of the basic PSO algorithm. It can be used to optimize other complex engineering problems. It proves to be a better alternative of the basic PSO algorithm.
12. Comparison of basic and proposed PSO algorithm are made in terms of number of iterations, NFEV and time consumed. It proves that proposed PSO algorithm gives better results and should be preferred for optimization.

CHAPTER 7: CONCLUSION AND FUTURE SCOPE OF WORK

7.1 CONCLUSION

Following are the conclusion of this Project work:

1. Basic as well as proposed PSO is implemented over an economic load dispatch problem.
2. Results of both cases are compared with each other and it is found that the proposed PSO gives better results in comparison with basic PSO.
3. Matlab programs are developed to solve the problem of economic load dispatch using particle swarm optimization.
4. Mathematical test functions are also optimized using PSO and proposed PSO method.
5. The Proposed method uses the retardation factor for faster convergence.
6. The retardation factor is varied between 0 to 1.
7. Using the hit and trial method, we have checked the results for every retardation factor and favourable results are shown.
8. It is found that the number of iterations performed in the case of the proposed PSO is lesser than the number of iterations performed in the case of basic PSO. The number of function evaluations in the case of the proposed PSO is also less than that of basic PSO.
9. But in case of economic load dispatch problem, swarm size is 100 and the maximum number of iterations is 5000. Fuel cost of the IEEE-3 generator and IEEE-6 generator system is also less in the case of proposed PSO in comparison to basic PSO implementation.
10. A comparison is done between the conventional PSO optimization technique and the Proposed PSO optimization technique in terms of NFEV, computational time and number of iterations. It is done over mathematical test functions and economic load dispatch.
11. We can use other programming methods to optimize this problem which could be our future work.

7.2 FUTURE SCOPE

THE proposed PSO algorithm can be implemented for optimum load flow analysis using PSO. Voltage profile at different buses can be improved and line loss reduction can also be done. The operating time of a directional overcurrent relay is also an objective function. Hence the sum of operating time of different directional overcurrent relay can also be optimized using proposed PSO for good coordination of directional overcurrent relay. The proposed PSO algorithm can be implemented on different mechanical engineering optimization problems also.

REFERENCES

[1] Lee, K. Y., Yome, A. S., & Park, J. H. (1998). Adaptive Hopfield Neural Networks for Economic Load Dispatch. *IEEE Transactions on Power Systems*, 519-526.

[2] Bhattacharya, A., & Chattopadhyay, P. k. (2010). Hybrid Differential Evolution With Biogeography-based optimization for solution of economic load dispatch. *IEEE TRANSACTIONS ON POWER SYSTEMS*, 1955-1964.

[3] Panigrahi, B. K., & Pandi, V. R. (2008). Bacterial foraging optimisation: Nelder–Mead hybrid algorithm for economic load dispatch. *IET Generation, Transmission & Distribution*, (pp. 556-565).

[4] Takeang, C., & Aurasopon, A. (2019). Multiple of Hybrid Lambda Iteration and Simulated Annealing Algorithm to Solve Economic Dispatch Problem with Ramp Rate Limit and Prohibited Operating Zones. *Journal of Electrical Engineering & Technology*, 111-120.

[5] Kennedy, J., & Eberhart, R. (1995). Particle Swarm Optimization. *IEEE*, (pp. 1942-1948).

[6] Shi, Y., & Eberhart, R. C. (1999). Empirical Study of Particle Swarm Optimization. *IEEE*, 1945-1950.

[7] JAIN, N. K., NANGIA, U., & JAIN, J. (2018). A Review of Particle Swarm Optimization. *Journal of The Institution of Engineers (India): Series B volume* , 407-411.

[8] Kulkarni, k. N., Patekar, S., Bhoskar, T., Kulkarni, O., Kakandikar, G. M., & Nandedkar, V. M. (2015). Particle Swarm Optimization Applications to Mechanical Engineering- A Review., (pp. 2631 – 2639).

[9] Panigrahi, B. K., Pandi, V. R., & Das, S. (2008). Adaptive particle swarm optimization approach for static and dynamic economic load dispatch. *Energy Conversion and Management 49*, 1407–1415.

[10] DAS, B., & Sengupta, K. T. (2015). ECONOMIC LOAD DISPATCH USING PSO AND TLBO. *Michael Faraday IET International Summit: MFIIS*, 212-219.

[11] JAIN, N. K., NANGIA, U., & JAIN, J. (2018). Economic Load Dispatch Using Adaptive Social Acceleration., (p. 9).

[12] PARK, J. B., LEE, K. S., & SHIN, R. J. (2003). Economic load dispatch for nonsmooth cost functions using particle swarm optimization. 938-943.

[13] Sudhakaran, M., Vimal Raj, P. A.-D., & Palanivelu, T. G. (n.d.). Application of Particle Swarm Optimization for Economic Load Dispatch Problems., (p. 7).

[14]Singh, H. P., Brar, Y. S., & Kothari, D. P. (2013). MULTIOBJECTIVE LOAD DISPATCH USING PARTICLE SWARM OPTIMIZATION. *8th Conference on Industrial Electronics and Applications (ICIEA)* (pp. 272-277). IEEE.

[15] Kumar, N., Nangia , U., & Sahay , K. B. (2014). Economic Load Dispatch Using Improved Particle Swarm Optimization Algorithms. *IEEE*.