CHAPTER 1

INTRODUCTION

1.1 GENERAL

Human desire to understand and control the world have given rise to large number of exploration in the field of science and have provided us the concept that enables us to understand, the various types of phenomena and the means to tackle them effectively. The roots of Artificial Intelligence (AI) term can find its way back in the very beginning of the computer age. Early times researchers tries to link concept of biology with computer and tries to figure out how these concept can be used for solving the problems. Modeling of brain, mimicking human learning and simulating biological evolution were some of the problems on which these concept was applied. These biologically motivated computing tends to be easy in handling and understanding and gives improved result activities therefore theses gained importance and since then not even a single day is passed out without listening any news about such innovation [5].

In the field of computer science, for solving several important problems algorithms have always played an important role . They were developed so that a big continuous problem can be breakdown into understandable short format and implementation of problem can become easy. Algorithm have also drawn there inspiration from field of biology and many approaches in this direction have been successful [63]. Therefore these methods have been implemented for tackling the hard and complex optimization problems. Now a day's evolutionary algorithm have found numerous applications in problem solving [11]. Evolutionary Algorithms have received acclaimed from researchers all over the world because they have provided satisfactory and improved results when solving difficult engineering design problems [26] & [38]. Swarm Intelligence algorithms falls into category of bio inspired and are successfully applied to a variable range of problems. Meta heuristic algorithm is based on the replication of natural phenomena : for example, biological evolution, animal behavior, music improvisation, etc.[16]&[64] for global optimization. Algorithms related to the numerical optimization techniques give results of optimization based on simple and ideal models. But many real-world engineering problems are highly non-linear and complex in nature, often resulting into multiple local optima and quite difficult to solve using these algorithms. The computational shortcomings of these numerical techniques have given researchers alternative technique of Meta heuristic algorithms for solve engineering optimization. The exploration of problem sphere by effectively reducing its size is there main technique on which these algorithm works. Therefore in short Swarm Intelligence optimization serves three main purposes: handling large problems effectively, giving faster response and obtaining robust algorithms [39].

1.2 LITERATURE REVIEW

Learning from naturally abundant biological systems and developing different kinds of optimization algorithms have been widely used in both theoretical study and practical applications [35]. While the study of bionics inspires and improves our modern technologies with the principles found in nature, biological structures and functions. Bionics study inspires us not only with its physical yields, but also with various computation methods that can be applied in different areas. In bioinspired design approaches one technique which is very popular is “Swarm Intelligence” [2]. Techniques which are inspired by the behavior of animal societies as well as other social insect colonies that are naturally able to solve large-scale distributed problems are grouped in the swarm intelligence [1]. The swarm intelligence is defined as an algorithm or distributed problem solving devices design inspired by the collective behavior of many animals’ societies, particularly social insect colonies [6]. In current peer to peer systems swarm availability is proven to be a new topic of attention [51]. It has become a research interest among scientists of related fields in recent years [22].

Particle swarm optimization (PSO) technique was developed by Eberhart and Kenndy [3]. It is an approach that mimics the behavior of social organisms where the behavior of different types of social interaction (e.g. flock of birds) is mimicked in order to create an optimization method that is able to solve continuous optimization problems[1]. PSO is a computational intelligence –based techniques that is not largely affected by the size and nonlinearity of the problem thus converges to the optimal solution is mainly a problems where most analytical methods fail to converge [16]. Therefore, it can be effectively applied to different optimization problems in power system [23] and [25]. Literature tells that it has been successfully applied to various problems viz. Economic Dispatch [18], Generation Expansion Problem (GEP), State estimation etc. It has been successfully applied for parameters tuning of controller used for inverted pendulum [52] and Buck Converter [61]. Large scale and practical problems like financial Credit-Risk Assessment [27] and Maglev Transportation [52]. PSO has also been modified and used for solving problems in desire to obtain better result than conventional PSO. It has been again improved [40] upon to get. Modified PSO [24], Adaptive Particle Swarm optimization (APSO) [30] and Multi-dimensional Particle Swarm Optimization for dynamic environment [47] and [31].

Bacterial Foraging Algorithm (BFA) technique was developed by K.M Passino [35] in 2002. In recent years, bacterial foraging algorithm has been developed as a rich source of potential for solving engineering applications and therefore have attracted more and more attention. Until date, Bacterial Foraging Optimization Algorithm (BFOA) has been applied successfully to some engineering problems, such as optimal control [10], harmonic estimation [62], transmission loss reduction [21] or enhancement of performance [41] etc. Problems which have solved using conventional techniques or others evolutionary algorithms have also been implemented using BFA. Automated tuning of PID controller for UAV was performed using Zigler-nicholas and BFA [42] and result obtained in showed that tuning was better for BFA. For Selection of optimum value of speed regulation parameter of governor and frequency bias setting, BFA has been applied for overcoming the deficiencies of Genetic Algorithm (GA) [32]. For solving large scale, non-convex and non-linear problem also BFA have shown good result [43]. Determining the size and location of component at optimum position have also been achieved using BFA [44]. Recently BFA is also finding its application in tuning of controller parameters as this is one of the laborious task if done using conventional. In finding the optimal control for distribution of static compensator PI controller based on BFA has been proposed [20]. Adaptive tuning of controller has been very successful and found fessible in various fields of optimization problems [65], like adaptive control using BFA have been implemented on two different non-linear system one is for dc motor speed control and second for liquid level control and result obtain for both compared with GA[50].

Original BFOA has also undergone modification for improving result for example the chemotactic step size is made adaptive using gradient descent method [28] or step length is varied exponentially depending upon the position of bacteria as for nutrient rich area it take small step length for searching better and in noxious environment bacteria take longer step length [33]. One of the very common problem of economic load dispatch implemented using modified BFA [49]. Researchers are now trying to hybridize two or more optimization techniques for improving results [7]. BFOA with different other algorithms [53] in order to explore its local and global search properties separately. Hybrid least-square adaptive bacterial foraging strategy was used to estimate the harmonic components present in power systems voltage/current waveforms [15]. The BFOA was used for designing of Controllers, for effective online tuning the fuzzy model [57], which is used to designs the feedback controller for servo motors with voltage regulators. Bacterial Foraging Optimization Algorithm also been utilized in feed forward neural network to enhance the learning process and improve its convergence rate and classification accuracy [54]. BFA oriented PSO algorithm was used for PID tuning [37] and results obtained were found better than BFA and PSO implemented separately. Hybrid PSO-BFA was applied to various complex bench mark function in [36], where from the result obtain it was concluded that hybrid algorithm gives higher accuracy in result in comparison to the other two i.e. PSO and BFA. Hybrid PSO-BFA algorithm are not only found there application in simple tuning or on bench mark function but they have been applied to practical problems like it was used for designing of channel equalizer for effective communication [48], here weights of equalizer is updated by PSO algorithm nested within BFA . Bacterial Foraging Algorithm Based is hybrid with PSO and applied to different problems .Mathematical modeling, adaptation, modification and hybrid with other techniques of the algorithm might be a major part of the research on BFOA in future.

1.3 ORGANIZATION OF THESIS

In this dissertation implementation of adaptive control using BFA, PSO and hybrid PSO-BFA algorithm is performed on following nonlinear system:

1. Buck converter

2. Boost converter
3. Surge tank system

An introduction to the system along with the objective to be controlled is included in later chapters. This dissertation comprises of six chapters.

While the present ongoing introduction and literature review is included in chapter 1, a review on all the three BFA, PSO and Hybrid PSO-BFO algorithms have been given in the chapter 2 which will entail the preliminary basics of each algorithm along with their working principles.

Chapter 3 gives a basic insight to control system and its type, Feedback control and its various configuration and adaptive control techniques. Adaptive control technique and its implementation via intelligent algorithms is also briefly described.

Chapter 4 describes the dynamics of nonlinear systems used to analyze the behavior of adaptive control utilizing BFOA, PSO and Hybrid PSO-BFA.

Chapter 5 shows the simulated results and discussion on them pertaining to each non linear system on implementing adaptive control using BFOA, PSO and Hybrid PSO-BFA.

Chapter 6 Concludes the dissertation with discussion on the performance of all the three algorithms individually and comparatively.

CHAPTER 2

REVIEW ON BFA, PSO AND HYBRID PSO-BFA

2.1 BACTERIAL FORAGING OPTIMIZATION

Escherichia coli (E. coli) bacteria is most simple and best understood microorganism in biology. The structure of E.coli bacteria is as follows: outer covering is made up of plasma membrane, inside it is another covering called as cell wall. This cell wall protects the capsule. Inside capsule we have cytoplasm and nucleoid. The cell of E.coli bacteria is of 1 µm diameter and 2 µm in length and is filled up with 70% of water. It weighs near about one picogram. The Whole outer structure of E.Coli bacteria is consist of tiny structure known as pili and they are used for gene transfer to other E.Coli bacteria. E.Coli bacteria make use of flagellum present over its body for movement. When E.coli bacteria grows up then it divides from the middle to give birth to two other E.coli bacteria. By providing them with sufficient food and favorable environment they can reproduce even faster. For this search of food and avoidance of noxious substance in environment, the E. coli bacterium is having some sort of guidance system. This is evident from instance that it tends to swims away from environment consist of acidic or alkaline substance and toward more neutral environment [48].

2.1.1 BFA Operation

E.coli cell body is covered with flagella. These flagella are generally 8-10 in number, with the help of these flagella bacteria is able to perform locomotion. These all flagella together are responsible for two kind of movement one tumble and run. Whenever these flagella all together make counterclockwise rotation, they form a compact and propel the bacteria in a direction this is known as run. In run distance travel by bacteria is long. The opposite phenomena happens when all the flagella moves in clockwise direction i.e. all flagella pull the cell down and they all move in different directions. This is known as tumble. In tumble distance covered by bacteria is small then covered by run. All this locomotion phenomena of E.coli bacteria is known as Chemotaxis. Figure 2-1 depicts the chemotaxis behavior of the E.coli bacteria [56].

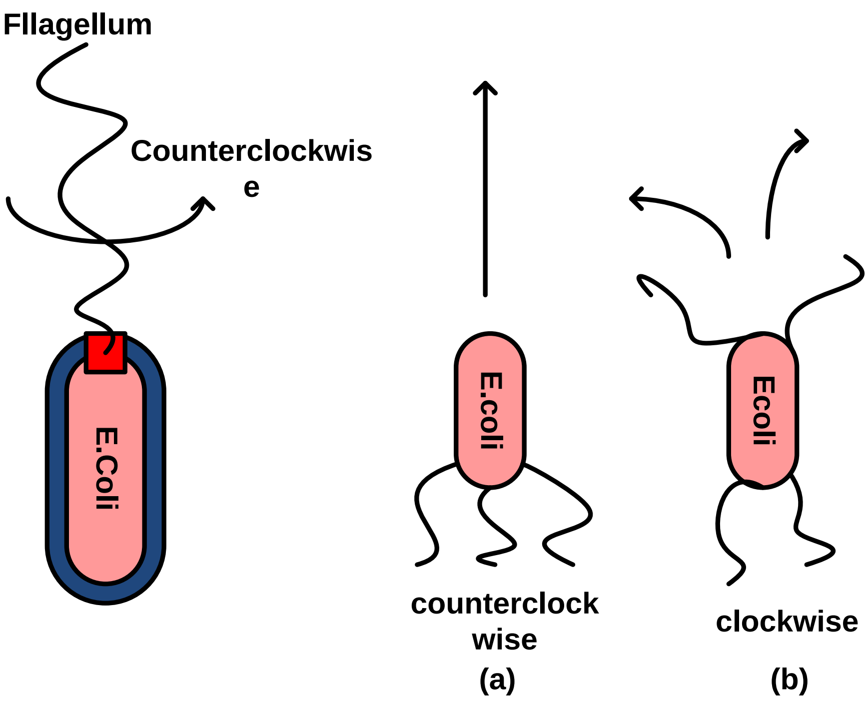

Figure 2-1 Chemotaxis behavior of E. Coli bacteria

E.coli bacteria make use of chemotaxis behavior for searching nutrient rich environment and avoid harmful one. But with chemotaxis it make use of other principles like swarming,re-production and elimination and dispersal. The description of these principles involved is briefly discussed below :

- ***Chemotaxis:*** The counterclockwise or clockwise movement of flagella on bacteria to provide run or tumble respectively are result of the environment in which E.Coli bacteria is present. When it is present in nutrient rich medium then it run and in presence of noxious substance it prefer to tumble so that it can randomly make search of nutrients . During its entire life bacteria make choice between these two modes for searching nutrients. For showing the movement of bacteria let we assume total number of bacteria is S and $\theta_i(j; k; l)$ represent the position of the each i^{th} bacteria in this population at the j^{th} chemotactic step, k^{th} reproduction step, and l^{th} elimination step then movement is depicted as:

$$\theta_i(j+1; k; l) = \theta_i(j; k; l) + C(i)\Phi(j) \tag{2.1}$$

Where C(i) (i = 1; 2;………S) represent the size of the chemotactic step taken in the direction decided by the tumble or run. Φ(j) used to define the random direction of movement after a tumble.

- ***Swarming:*** Swarming is performed by E.coli to attract other bacteria's and converge at a optimum solution by moving in concentric patterns with high bacterial density. Swarming helps in converging at desired solution more rapidly by forming groups. The one bacteria also repels nearby bacteria in the sense that one might consumes all the nearby nutrients and it is not physically possible to have two cells at the same location.

- ***Reproduction:*** Bacteria when grow up completely they reaches the stage of reproduction. From the total population bacteria best one are chosen and reproduction is performed on them. The chosen bacteria's get divided from the middle in two parts.

$$S = \frac{S}{2} \tag{2.2}$$

  The healthier half replaces the other half of bacteria, which gets eliminated, owing to their poorer foraging abilities. This makes the population of bacteria constant in the evolution process.

- ***Elimination and dispersal:*** In this phenomena sudden unpredicted event can occur which can eliminate a set of bacteria's or completely disperse then into new environment. This can help by placing the newer set of bacteria near the food location. It also helps by preventing being trapped in a premature solution point or local optima [45].

2.1.2 Pseudo Code for BFA

Begin

1. Initialize input parameters
2. Create a random initial swarm of bacteria θi(j, k, l)
3. Evaluate f (θ_i(j, k, l)) : i, i = 1, . . . , S

   For (1) l=1 to number of elimination & dispersal loop

   For (2) k=1 to number of reproduction loop

   For (3) j=1 to number of chemotactic step

For (4) i=1 to number of bacteria

$\theta_i(j+1, k, l) >$ or $< \theta_i(j, k, l)$ Perform the chemotaxis step (tumble-swim or tumble-tumble) for bacteria $\theta_i(j, k, l)$

End For (4)

End For (3)

Perform the reproduction step by eliminating the half worst bacteria and duplicating the other half

End For (2)

Perform the elimination-dispersal step for all bacteria $\theta_i(j, k, l)$ i, $i = 1, \ldots, S$ with probability $0 \leq P_{ed} \leq 1$

End For (1)

Parameters needed to be initialized are: number of chemo-tactic steps (N_c), number of reproduction steps (N_{re}), number of elimination and dispersal steps (N_{ed}), dispersal probability (P_{ed}), number of bacteria (S) & swim length (N_s). Figure 2-2 show flow diagram of bacterial foraging algorithm.

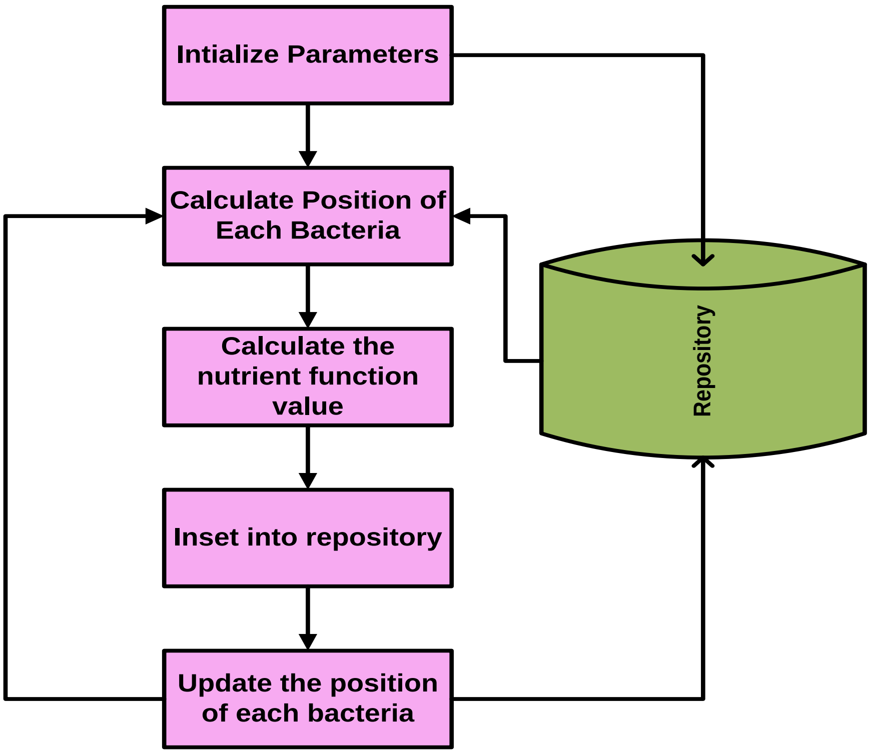

Figure 2-2 Flow chart for Bacterial Foraging Algorithm

In the first step parameter initialization is made after that position of each bacteria is calculated. On that position nutrient value is calculated this nutrient value is stored in repository. Then again position of bacteria is updated based on the movement based on the nutrient value available. Then again cycle repeat from calculating the current position of bacteria.

2.2 PARTICLE SWARM OPTIMIZATION

PSO initiates the swarm of particles to search in a space of possible solutions for a problem. Each particle has a position vector representing a candidate solution to the problem and a velocity vector and also contains a small memory that stores its own best position seen so far and a global test position obtained by communication with particles at the neighborhood. The advancement towards the location (xi*) and global best (g*) by the particle swarm optimizer is ideologically similar to the crossover operation utilized by genetic algorithm .similar to all evolutionary computations paradigms; it uses the concept of fitness. The swarm works through the interactions of members of the populations, even though the exact methods for moving the Particle are quite flexible [59]. Best assigning shortest path and by studying trajectories of particles the optimization behavior of standard PSO can be made invariant to the rotations of the optimizations functions. There are two major components in the movement of a swarming particle: a deterministic component and a stochastic component.

2.2.1 PSO operation

Consider flying through the parameters space swarm of particle searching for optimum. Each particle is attributed with position vector x1(t) and velocity vector y1(t). The velocity vector is calculated by the following:

$$V_i^{t+1} = V_i^t + C_1 e_1 [g^* - x_i^t] + C_2 e_2 [x_t^* - x_i^t] \tag{2.3}$$

Where e1 and e2 are two random vectors taking the value between 1. C1 and C2 are the acceleration constants reflecting the weighting of stochastic acceleration terms, which can typically be taken as C1≈C2≈2and g* is the fitness value .The initial velocity if the particle can be taken as zero, $V1^{t=0}$= 0. The new position can be calculated by

$$x_i(t+1) = x_i(t) + V_i^{t+1} \tag{2.4}$$

Although v1 could be any value, usually it is bounded by some range [0,V_{max}]. Figure 2-3 gives the update of a particle in swarm at the next time in the instant where g (t) is the best value of the particle fitness.

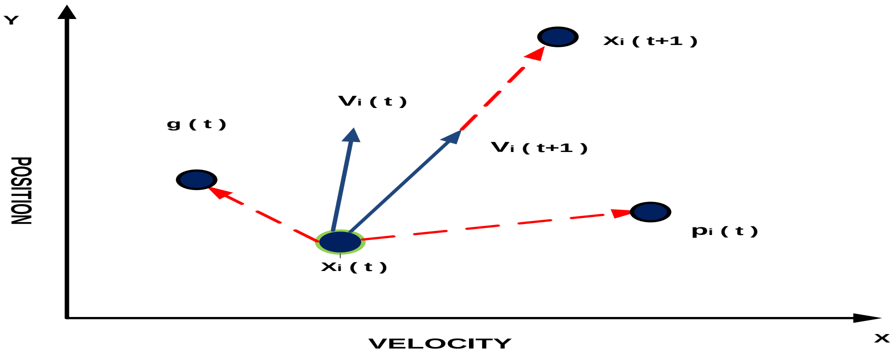

Figure 2-3 Position(x) & velocity(y) update of i^{th} particle in swarm

2.2.2 Pseudo code PSO

For implementation of PSO algorithm, following procedure can be used

1) Initialize the particle by assigning the random position to each particle in the problem hyperspace.
2) Evaluate the fitness function of every particle.
3) Compare the particle's fitness value with x_1* value, then set this value as x_1* and the current particle 's position x_1.
4) The particle that has the best fitness value is identified. The value of its fitness function is identified as g* and its position as g.
5) Calculate the velocities and positions of all the particle using (2.3) and (2.4)
6) Repeat steps 2-5 till a stopping criterion is created (e.g. a sufficiently good fitness value or maximum numbers of iterations).

PSO is attractive for the reason that there are few parameters to adjust to get a proper response and thus it can be used for specific applications focused on specific requirements or it can also be used for approaches for wide range of applications. Figure 2-4 below describes the flowchart for the PSO algorithm for case of understanding.

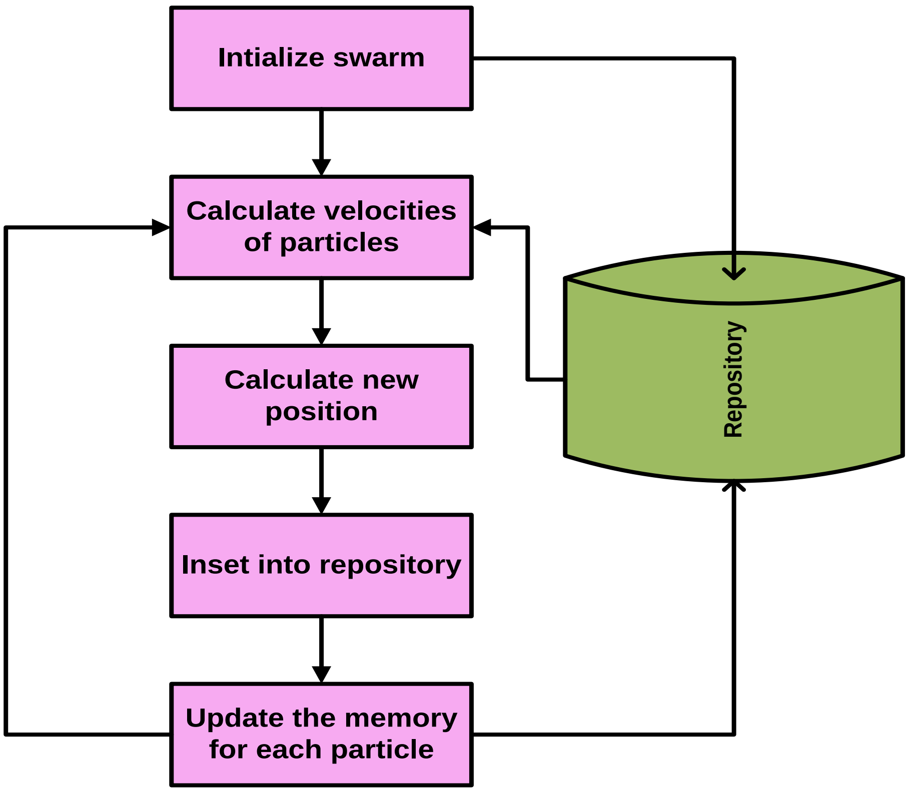

Figure 2-4 Flow chart for Particle Swarm Optimization

In the flow diagram first step is initialization of parameters. Now velocity for each particle is calculated after this new position of particles are calculated. Newly calculated values are then fed into repository and memory of each particle is updated. Then again new velocity for particle is calculated from the updated memory and previously stored parameters in the repository and again all the steps repeat until ending criteria is met.

2.3 HYBRID PSO-BFOA

In chemotactic behavior of bacteria in BFOA a randomly generated tumbling or run is made in appropriate direction Random direction make the algorithm slow and may be responsible for delay or not reaching global solution. Therefore by using a technique to update the position of foragers in BFA. As we are aware that in PSO, the particle stores it local best position and global best position and according to it update its position. Therefore by adapting this procedure of updating in BFA, Hybrid algorithm contain combination of both PSO and BFA. The main aim here is to aims are to make use of PSO property of updating and adjusting the particle position for global optimization and BFA property to disperse into new set of solution by elimination and dispersal event.

2.3.1 Hybrid PSO-BFA operation

All the steps of BFA are performed in this algorithm but the step in which direction of tumbling should performed is decided by PSO . In the Hybrid PSO-BFA algorithm the random direction generated by tumbling action of bacteria is decided by the PSO algorithm. Therefore this tumble behavior of bacteria is based on the global best position which is decided by PSO algorithm and based on this further best position of each bacteria is determined. The Hybrid PSO-BFA has great advantages in jumping out of local optimal solution and finding the better global optimal solution. Hybrid PSO-BFA shows better convergence and its solution is more stable and has higher quality than PSO and BFOA . The brief pseudo-code of the Hybrid PSO-BFA has been provided below in the next section.

2.3.2 Pseudo Code Hybrid PSO-BFA

Begin

1. Intialize parameters for both PSO and BFA.
2. Create
    - θi(j,k,l) and V_i^t : Randomly initialize the bacterium positions θi(j,k,l) and its velocity V_i^t(i=1,……..,S) in the domain;
    - θpbesti and θpbest : the global best positions and the best position of each bacteria
    - Evaluate f (θ_i(j, k, l)) : i, i = 1, . . . , S
3. Evaluation

For (1) (1=1: number of elimination-dispersal loop)

For (2) (k=1: number of reproduction loop)

For (3) (j=1 : number of chemotactic loop)

For (4) (i=1: number of bacterias)

Update bacteria position θ_i (j+1,k,l) ;

Get fitness of bacterium f (θ_i(j+1,k,l));

End For (4);

Update the HYBRID PSO-BFO inertia weight w;

Update velocity V_i^t ;

Update θpbesti and θpbest ;

End For (3);

Reproduction: computer the health of bacterium i:

$$J^{i}_{health} = \sum_{J=1}^{N_c+1} J^{i}\,(j,k,l), i = 1,2, \ldots s \quad (2.5)$$

End For (2)

End For (1)

End

Figure 2-5 below describes the flowchart for the Hybrid PSO-BFO algorithm for case of understanding.

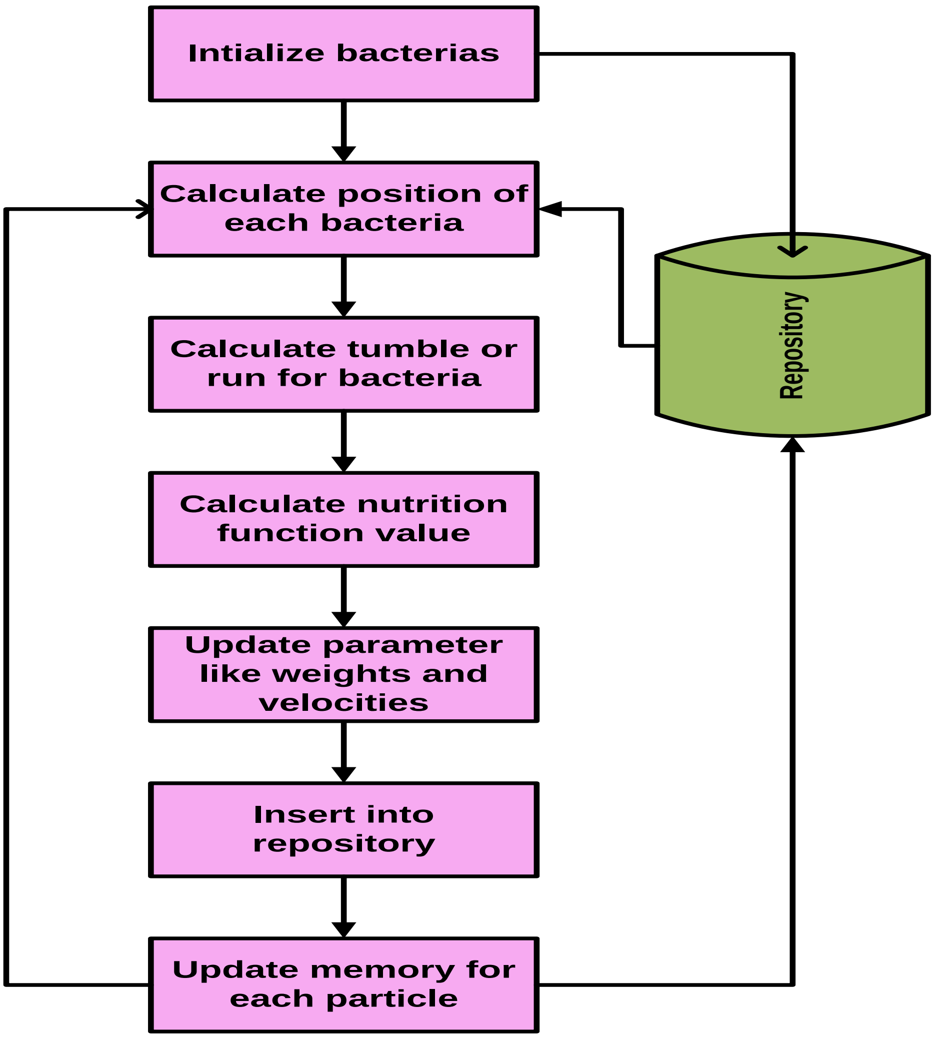

Figure 2-5 Flow chart forHybrid PSO-BFO

In the above flow diagram the first three steps are for BFA as current position of bacterias and nutrient function is calculated . After these steps chemotactic steps for bacteria are calculated by updating the weights and velocity of the particles. These new value get store in repository and memory of each particle is updated then again above steps repeat for new sets of bacteria and particle PSO.

CHAPTER 3

ADAPTIVE CONTROL

3.1 INTRODUCTION

When we use word control in everyday life, we are referring to the act of producing a desired result. By this broad definition, control is seen to cover all artificial processes. Whether the control is automatic, or caused by a human being, it is integral part of our daily existence. However control is not confined to artificial process alone. Imagine living. In a world where temperature is unbearable hot, without the life-supporting oxygen, water or sunlight. We often don’t realize how to control the natural environment we reside. The composition, temperature and pressure of the earth are kept stable in their livable state by an intricate set of natural process. Hence, control is everywhere we look and is crucial for existence of life itself.

A study of control involves a mathematical model of each component of the control system. We have twice used the word system without defining it. A system is a set of self-contained process under study. A control system by definition consists of the system to be controlled-called the plant- as well as the system which exercise control over the plant, called the controller. The controller is said to supply a signal to the plant, called the input to the plant, in order to produce a desired response from the plant, called the output from the plant. When referring to an isolated system, the term input and output are used to describe the signal that goes into a system, and the signal that comes out of a system, respectively [13].

3.1.1 Open-loop and closed-loop control system

Control system are broadly classified as open loop and close loop control. In open loop control system control action is decided without having prior knowledge of plant

Output. Figure 3-1 shows a block diagram of an open loop control system where the controller and plant are shown as rectangular blocks.

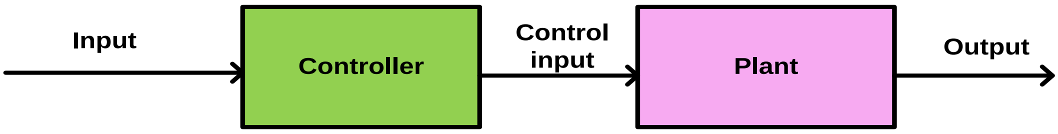

Figure 3-1 An open loop control system

Open loop control is like riffle shooter who gets only one shot at the target. Open loop control gives satisfactory result when controller is having prior knowledge about the plant behavior. But in the presence of incomplete or poor knowledge about plant behavior, or due to noise, open loop control system does not give satisfactory result. Therefore instead of applying a pre-determined control input as in the open loop case, adjusting the control input according to the actual observed output tends to be more successful. In close loop control system plant output obtain is feedback and desired control action is provided based on the error generated by the comparing desired output and actual output obtain. Therefore close loop control seems to be more successful in eliminating noise due to parameter variation. The mechanism by which the information about the actual output is conveyed to the controller is called feedback. On the block diagram, the path from the plant output to the controller input is called a feedback loop [9]. A block diagram example of a possible closed loop system is given in figure 3-2 below:

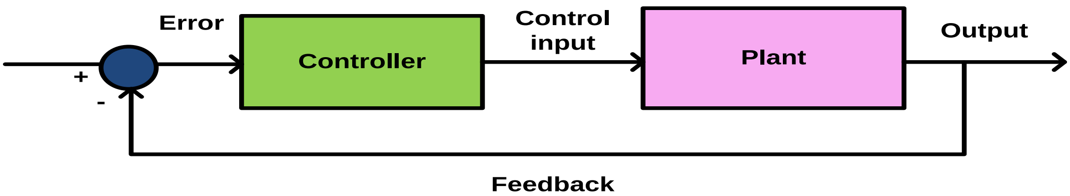

Figure 3-2 Closed loop control system

3.1.2 Controller Configurations

In general, the dynamics of linear controlled process can be represented by the block diagram shown in figure 3-3. The design objective is to have the controlled variables, represented by the output vector y (t) to, behave in certain desirable ways. The problem essentially involves the determination of the control design u (t) over the prescribed time interval so that the design objectives are all satisfied.

Most of the conventional design methods in control system rely on the so called fixed configuration design in that the designer at the outset decides the basic configuration of the overall designed system and the place where the controller is to be positioned relative to the controlled process. The problem then involves the design of the elements of the controller. Because most control efforts involve the modification or compensation of the system-performance characteristics, the general design using fixed configuration is also called compensation.

Figure 3-3 to 3-5 illustrates several commonly used system configurations with controller compensation. These are described briefly as follows.

- Series (cascade) compensation: In Figure 3-3 shows the most commonly used system configuration with the controller placed in series with the controlled process, and the configuration is referred to as series or cascade compensation.

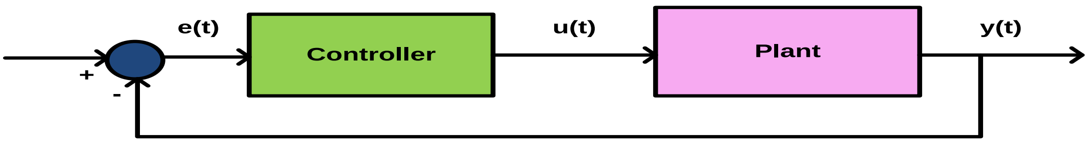

Figure 3-3 Series compensation

- Feedback compensation: In figure 3-4 the controller is placed within the minor feedback path, and the scheme is called feedback compensation.

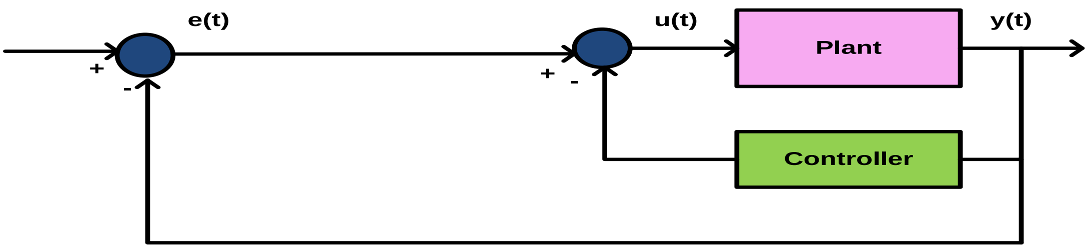

Figure 3-4 Feedback compensation

- State-feedback compensation: In figure 3-5 shows a system that generates the control circuit by feeding back the state variable s through constant real gains, and the scheme is known as state feedback. The problem with the state-feedback. The problem with the state feedback is that for high-order system, the large number of state variables for feedback. Thus, the actual implementation of the state feedback control scheme may be costly or impractical. Even for low order system, often not all the state variables are directly accessible, and an observer or estimator may be necessary to create the estimated state variables from measurements of the output variables.

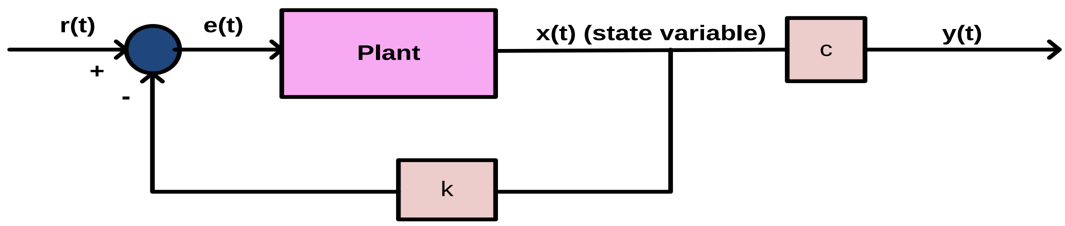

Figure 3-5 State feedback compensation

3.1.3 Fundamental principle of Controller design

After the controller configuration is chosen, the designer must choose a controller type that, with proper selection of its elements values, will satisfy all the design specifications. The types of controllers available for control-system design are bounded only by one’s imagination. Engineering practice usually dictates that one choose the simplest controller that meets all the design specifications. In most cases, the more complex a controller is, the more it costs, the less reliable it is, and more difficult it is to design. Choosing a specific controller for a specific applications is often based on the designer’s past experience. After a controller is chosen, the next task is to choose controller parameters values. These parameter values are the coefficients of one or more transfer functions. Making up the controller. The basic design approach is to use the analysis tool discussed to determine how individual parameter value influence design specifications in conflicting ways. For example a particular parameter value may be chosen so that the maximum overshoot is satisfied, but in process of varying another parameter value in an attempt to meet the rise time requirement, the maximum overshot specification may no longer be met. The more number of design specifications and controller parameters make the design process complicated.

In order to reject the effect of disturbances which are incident on the controlled variables feedback loop is used. It brings variables back to their desired values according to a certain performance index. The output obtained are first measured and then compared with the desired value. The difference generated is fed into the controller which will generate the appropriate control. The output quantity can be any desired plant variable which we need to control.

One of the commonly used controllers in the compensation schemes is a PID controller, which applies a signal to the process that is proportional of the actuating the signal in addition to adding integral and derivative of the actuating signal. Since the signal components are easily realizes and visualized in the time domain, PID controllers are commonly described using time domain methods [14].

3.2 PID CONTROLLER

PID controllers are the most widely-used type of controller for industrial applications. They are structurally simple and exhibit robust performance over a wide range of operating conditions. In the absence of the complete knowledge of the process these types of controllers are the most efficient of choices. The three main parameters involved are Proportional (P), Integral (I) and Derivative (D). The proportional part is responsible for following the desired set-point, while the integral and derivative part account for the accumulation of past errors and the rate of change of error in the process respectively [9]. For the PID controller presented in Fig. 3-6.

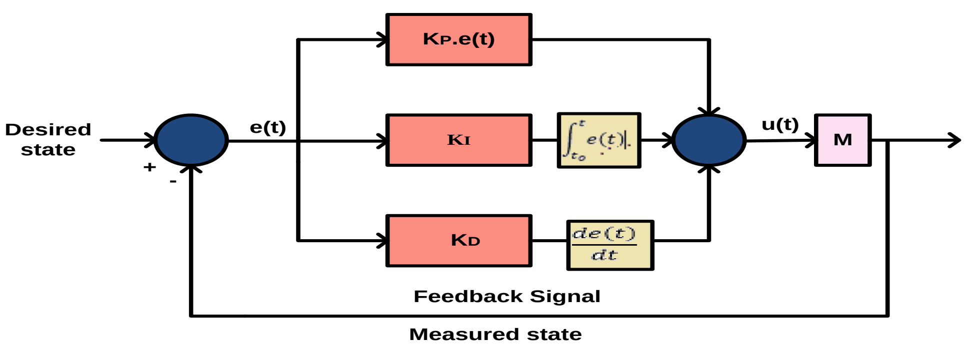

Figure 3-6 PID controller

Output of the PID controller,

$$u(t) = K_P e(t) + K_I \int_{t_o}^{t} e(t) + K_D \frac{d}{dt} e(t) \tag{3.1}$$

Where, Error is e(t) =Set point- Plant output

K_P= proportional gain, K_I = integral gain, K_D = derivative gain

3.2.1 Tuning of PID parameters

Tuning of a PID controller refers to the tuning of its various parameters ($K_P, K_I \& K_D$), so that we can achieve their optimized value for the desired response. The basic requirements of the output will be the stability, desired delay time, rise time, peak time, settling time and overshoot. Different processes have different requirements of these parameters which can be achieved by meaningful tuning of the PID parameters. If the system can be taken offline, the tuning method involves analysis of the step input response of the system to obtain different PID parameters. But in most of the industrial applications, the system must be online and tuning is achieved manually which requires very experienced personnel and there is always uncertainty due to human error. Another method of tuning can be Ziegler-Nichols method. While this method is good for online calculations, it involves some trial-and-error which is not very desirable [4].

3.2.2 PID controller tuning using various evolutionary techniques

By using conventional PID control [19], it is very difficult to obtain accurate control of the process because most of the industrial processes exhibits nonlinear behavior, the variability of parameters and the uncertainty of model are very high. The common methods known for tuning require the process model to be reduced if it's too complicated originally. The above problems can be well addressed by the application of evolutionary techniques for tuning of the PID controller [58]. Biological evolution has emerged as an appealing source of inspiration for addressing optimization problems. But now a days Swarm optimization i.e. part of Evolution is in effect emerging, as a new technique among other available methods. In Swarming highly favorable organisms are able to survive and reproduce in their environments. It can be used for designing innovative solutions to complex problems. The fitness criteria continually change as creatures evolve, so evolution is searching a constantly changing set of possibilities. Searching for solutions in the face of changing conditions is precisely what is required for adaptive computer programs. Furthermore, swarming is also a massively parallel search method rather than a work on one species at a time. It simultaneously perform test on large number species involved and takes decision accordingly based on the performance of species operating in parallel.

These are especially useful for solving problems of computationally complicated and mathematically untraceable. This is due to the convenience of combining natural systems with intelligent machines effectively with the help of soft-computing methods. Neural network, fuzzy logic and genetic algorithm are the soft computing techniques previously used for this purpose. Now recently evolutionary algorithms like Particle swarm optimization and bacterial foraging algorithm are also being used for PID tuning. These methods are found to be very useful for the process of random search from a huge pool of data. The system response of the plant is found to be improved. The dynamic performance of the system can also be improved by improving the overshoot and the rise time of the response can be decreased.

3.3 ADAPTIVE CONTROL

From a past few decades Adaptive control has been existing as an active research field and successfully applied to various applications. In order to achieve or to maintain a desired level of control system performance when the parameters of the plant dynamic model are unknown and/or change in time. Adaptive control covers a set of techniques which provides a systematic approach for automatic adjustment of controllers in real time. If a situation arises where the parameters of the dynamic model of the plant to be controlled are constant but are unknown in certain region of operation. Although the model of the control or the controller structure will not depend upon the particular values of the parameters of the plant model without knowledge of their values correct tuning of the controller parameters cannot be done. An automatic tuning procedure in closed loop for the controller parameters can be provided by the adaptive control techniques.

The various adaptation techniques are characterized by the way in which information is processed in real time to tune the controller for achieving the desired performances. Since the parameters of the controller will depend upon the measurement of system variable through the adaptation loop, the adaptive control system behaves nonlinearly [66].

Parameters can be adjusted directly or indirectly via estimation of process parameters. There is a large number of both direct and indirect methods available. Adaptation can be applied both to feedback and feed forward control parameters. In the indirect Adaptive control parameter estimator that determines the parameters of the model based on observations of process inputs and outputs. There is also a design block that computes controller parameters from the model parameters. The parameters can either be estimated recursively or batch wise. In the direct methods, the key issues are to find suitable features that characterize relevant properties of the closed-loop system and appropriate ways of changing the controller parameters so that the desired properties are obtained. The majority of the PID controllers in industry are tuned manually by instrument engineers. The tuning is done based on past experience and heuristics. By observing the pattern of the closed-loop response to a set point change, the instrument engineers use heuristics to directly adjust the controller parameters. The heuristics have been captured in tuning charts that show the responses of the system for different parameter values. A considerable insight into controller tuning can be developed by studying such charts and performing simulations. When set point changes or major load disturbances occur, properties such as damping, overshoot, period of oscillations and static gains are estimated. Based on these properties, rules for changing the controller parameters to meet desired specifications are executed.

3.4 ADAPTATION USING BFA, PSO AND HYBRID PSO - BFA

Optimization methods such as gradient algorithms and least squares are used to implement estimation methods, which are used to estimate models or controllers in adaptive control. We can also use intelligent algorithms as the basis for adaptive control. To see this, suppose that we consider adaptive control where we seek to learn a plant model during the operation of a system. Suppose that we view learning as foraging for good model information. Suppose that we use an “identifier model,” which is a parameterized model of the plant, and think of the foraging algorithm as searching in the parameter space of that model. By moving parameters, it searches for regions in the parameter space that correspond to finding optimum solution. Suppose that we use the standard definition of identifier error as the error between the model output and the plant output, but squared and summed over the past several samples to define J. As a part of this work adaptive control strategy is used for non-linear systems and it can be represented by a block diagram as shown in figure 3-7.

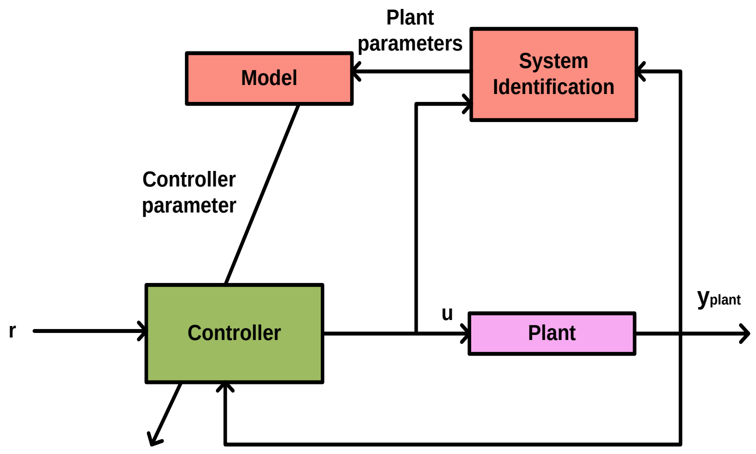

Figure 3-7 Adaptive Control

It can be seen from the block diagram that the plant to be controlled is analyzed by a system identification block. This identification block utilizes plant input and output to generate a signal varying with the plant conditions at a particular instant. This information accessed by the system identification block is then used to generate a signal which characterizes the plant parameters. This signal is then given to the controller designer block. The controller designer block generates a signal in response to the signal received by it pertaining to the plant parameters by the system identification block. Controller derives an input which is nothing but the output of the plant to be controlled. Apart from that, it also receives the signal from the controller designer block. These signals along with the references signal makes the controller to generate a control signal efficient enough to control the plant and thus, by each passing instant it moves the output of the plant close to the desired reference trajectory [55].

The system identification block plays a major role here in controlling the plant as it estimates the plant parameters and accordingly adjusts the controller output indirectly while the direct adaptive control just gives an adaptation depending only on the input and output of the plant. The model that is being tuned is a nonlinear function. The nonlinear mapping it implements is unknown since the plant is assumed to be unknown.

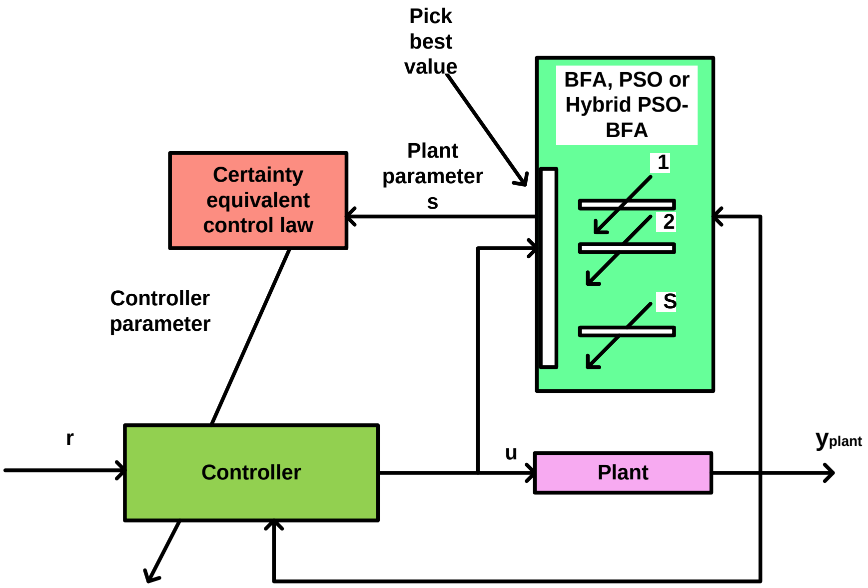

Figure 3-8 System identification block replaced by algorithm

In order to adjust the nonlinear mapping implemented by the BFA or PSO to match the unknown nonlinear mapping of the plant an online function approximation is carried out. Here system identification block is replaced by BFA or PSO algorithm. Figure 3.8 shows System identification block replaced by algorithm [34]. For equation relating to adaptive control refer to Appendix 1.

CHAPTER 4

DYNAMICS OF NON-LINEAR SYSTEM

4.1 DC-DC CONVERTER

Many industrial applications require power from dc voltage source. Several of these applications, however, perform better in case these are fed from variable dc voltage sources. From ac system, variable dc output voltage can be obtained through the use of phase controlled converters or motor-generator sets. The conversion of fixed dc voltage to an adjustable DC output voltage, through the use of semiconductor devices, is carried out by the use of DC to DC converters. Buck, Boost and Buck-Boost are three different kind of dc-dc convertor which can operated at different voltage conversion ratios. When operated under open loop condition, these convertor tend to exhibits poor voltage regulation and unsatisfactory dynamic response. [67].

In past, Linear PID and PI controllers are usually used for DC-DC converters. Previously control strategies such as Ziegler-Nichol's method, bode plot, root locus technique, hysteresis method, etc. are also used and they have given good performance around the operating point. However these convertor performance degrade when the operating point varies or system is subjected of large load variation. It is difficult for the controller to adapt changes in operating point, and they exhibit poor performance when the system is subjected of large load variation. The transient response can be even worse if the plant dynamic changes. Due to external/environmental causes, e.g. temperature and pressure, many plants tend to have time-varying dynamics. To assure an environmentally independent good performance, the controller must be able to adapt the changes of plant dynamic characteristics.

4.1.1 Boost Converter

The boost converter produce a DC output voltage greater in magnitude than the magnitude of DC input voltage. The circuit diagram for a boost converter is shown in figure 4-1.

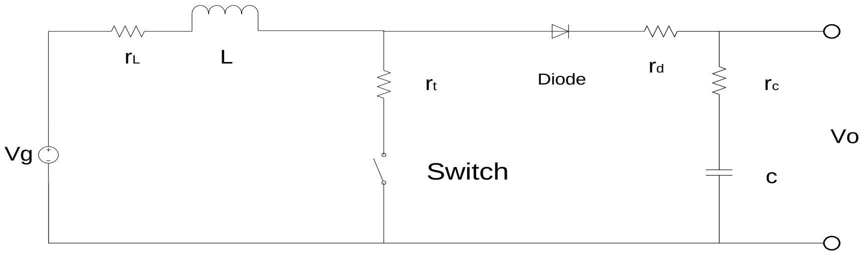

Figure 4-1 Boost Converter

When the switch is on the current in inductor L, rises linearly and at this time capacitor C, supplies the load current, and it is partially discharged. During the second interval when transistor switch is off, the diode is on and the inductor L, supplies the load and, additionally, recharges the capacitor C. As the current tends to decrease, emf induced in L but with the reversepolarity. As a result voltage across the output, given by $V_O = V_g + L\frac{di}{dt}$, exceeds the source voltage V_g. The steady state inductor current and voltage waveform is shown in figure 4-2 [8].

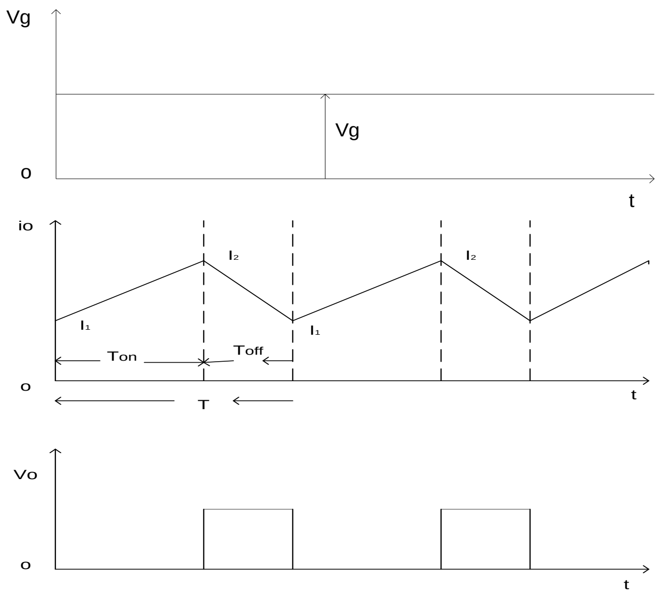

Figure 4-2 Input and output voltage waveform of boost converter

From the energy input to inductor from the source, during the period T_{on}, is

$$W_{in} = \text{(voltage across L) (average current through L)} \, T_{on}$$

$$= V_g \cdot \left(\frac{I_1+I_2}{2}\right) T_{on} \quad (4.1)$$

During the time Toff when the switch is off the energy released by inductor to the load is

$$W_{off} = \text{(voltage across L) (average current through L)} \, T_{off}$$

$$= (V_o - V_g)\left(\frac{I_1+I_2}{2}\right) T_{off} \quad (4.2)$$

Consider the system to be lossless, these two energies given by above two equation will be equal.

$$V_g.\left(\frac{I_1+I_2}{2}\right) T_{on} = (V_o - V_g).\left(\frac{I_1+I_2}{2}\right) T_{off} \quad (4.3)$$

$$V_g.\,T_{on} = V_o.\,T_{off} - V_g.\,T_{off} \quad (4.4)$$

$$V_o\,T_{off} = Vg\,(T_{on}+T_{off}) = V_g\,T \quad (4.5)$$

$$V_o = V_g\,\frac{T}{T_{off}} = V_g\,\frac{T}{T - T_{On}} = V_g\,\frac{1}{1-d} \quad (4.6)$$

$$\frac{V_o}{V_g} = \frac{T_{sw}}{T_{off}} = \frac{1}{1-d} \quad (4.7)$$

In above equations d is duty ratio. Since the converter output voltage is greater than the input voltage [29].The boost converter state equations are presented below. Equations for switch on mode and switch off mode are presented separately:

Switch on:

$$V_g = r_L\, i_L + L\,\frac{di_L}{dt} \quad (4.8)$$

$$R_L\, C\,\frac{dv_c}{dt} + r_c\, C\,\frac{dv_c}{dt} = -\,v_c \quad (4.9)$$

$$X = \begin{bmatrix} i_L \\ v_c \end{bmatrix} \quad (4.10)$$

$$\dot{X} = A_1\, X + B_1\, V_g \quad (4.11)$$

$$A_1 = \begin{bmatrix} a_{11} & a_{12} \\ a_{13} & a_{14} \end{bmatrix} \tag{4.12}$$

$$B_1 = \begin{bmatrix} b_{\ 11} \\ b_{21} \end{bmatrix} \tag{4.13}$$

$$a_{11} = \frac{r_L}{L} \tag{4.14}$$

$$a_{12} = 0 \tag{4.15}$$

$$a_{13} = 0 \tag{4.16}$$

$$a_{14} = \frac{-1}{[C(R_L + r_c)]} \tag{4.17}$$

$$b_{11} = \frac{1}{L} \tag{4.18}$$

$$b_{12} = 0 \tag{4.19}$$

$$V_{out} = C_1 X + D_1 V_{in} \tag{4.20}$$

$$C_1 = [c_{11} \quad c_{12}] \tag{4.21}$$

$$c_{11} = 0 \tag{4.22}$$

$$c_{12} = \frac{R_L}{(R_L + r_C)} \tag{4.23}$$

$$D_1 = 0 \tag{4.24}$$

Switch OFF:

$$(r_d + r_L)\, i_L + L\left(\frac{di_L}{dt}\right) + v_c + r_C\left(\frac{dv_C}{dt}\right) = 0 \tag{4.25}$$

$$i_L = C\left(\frac{dv_C}{dt}\right) + \left(v_C + r_C\, C\left(\frac{dv_C}{dt}\right)\right)\frac{1}{R_L} \tag{4.26}$$

$$\dot{X} = A_2\ X + B_2\, V_g \tag{4.27}$$

$$A_2 = \begin{bmatrix} a_{21} & a_{22} \\ a_{23} & a_{24} \end{bmatrix} \tag{4.28}$$

$$B_2 = \begin{bmatrix} b_{\ 21} \\ b_{22} \end{bmatrix} \tag{4.29}$$

$$a_{21} = -\left(\frac{r_L + R_C \| r_C}{L}\right) \tag{4.30}$$

$$a_{22} = -\frac{R_C}{L(R_L + r_C)} \tag{4.31}$$

$$a_{23} = \frac{R_L}{(R_L + r_c)C} \tag{4.32}$$

$$a_{24} = -\frac{1}{C(R_L + r_C)} \tag{4.33}$$

$$b_{21} = \frac{1}{L} \tag{4.34}$$

$$b_{22} = 0 \tag{4.35}$$

$$V_{out} = v_c \tag{4.36}$$

$$C_2 = [c_{21} \quad c_{22}] \tag{4.37}$$

$$c_{21} = r_c \parallel RL \tag{4.38}$$

$$c_{22} = \frac{R_L}{(R_L + r_C)} \tag{4.39}$$

$$D = 0 \tag{4.40}$$

Mean Values showing effect of on and off duration of switch [60].

$$\dot{X} = A_{avg} X + B_{avg} V_g \tag{4.41}$$

$$A_{avg} = d\, A_1 + (1 - d)\, A_2 \tag{4.42}$$

$$B_{avg} = d\, B_1 + (1 - d)\, B_2 \tag{4.43}$$

$$C_{avg} = d\, C_1 + (1 - d)\, C_2 \tag{4.44}$$

$$d = \frac{t_{on}}{T} \tag{4.45}$$

4.1.2 BUCK CONVERTOR

The buck converter is used for step down operation. A buck converter is as shown in figure 4-3

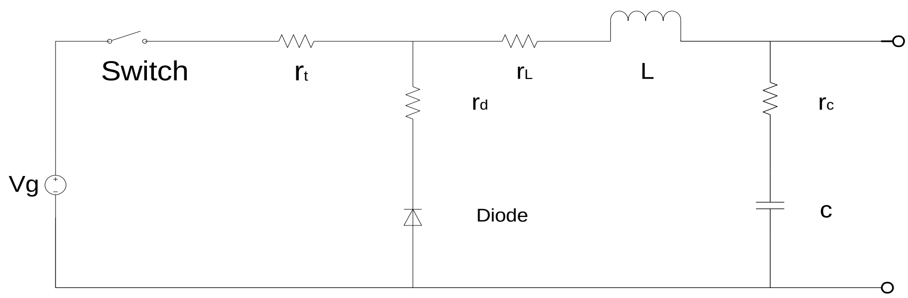

Figure 4-3Buck Converter

The switch represented in figure 4.3 connects source to load and disconnects the load from source at a fast speed [8]. In this manner chopped load voltage is obtained from a constant dc supply of magnitude V_g. Switch can be turned off and on as desired. During the period T_{on} , switch is on and load voltage is equal to source voltage V_g. During the interval T_{off}, switch is off , load current flows through the diode. When the diode is on and switch is off, the voltage across the inductor is reversed. However, current in the inductor cannot change instantaneously and the current starts decreasing linearly. In this cycle also the capacitor is also charged with the energy stored in the inductor.

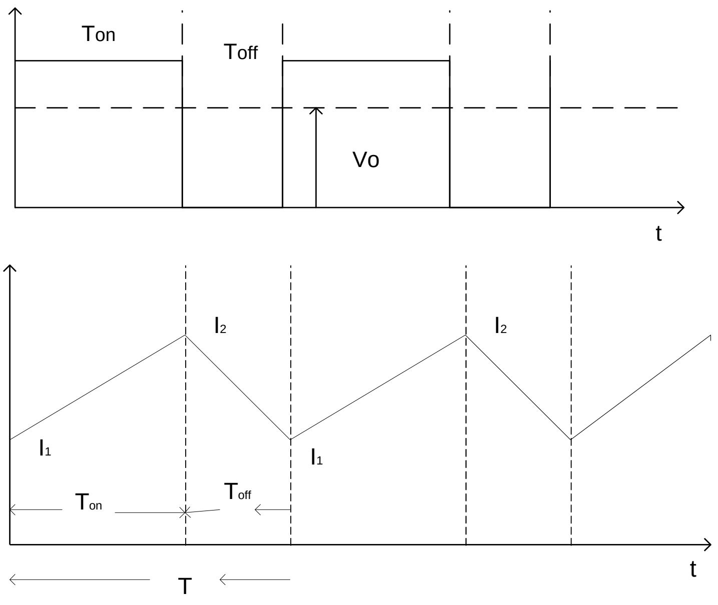

Figure 4-4 Inductor voltage and current waveform

Analyzing the inductor current waveform determines the relationship between output and input voltage in terms of duty cycle. Inductor voltage and current waveforms for a buck converter are as shown in figure 4-4.

During Ton, load current rises whereas during Toff load current decays . From figure 4-4, average load voltage Vo is given by

$$V_O = \frac{T_{on}}{T_{on}+T_{off}} V_g = \frac{T_{on}}{T} V_g = dV_g \tag{4.46}$$

$$T = T_{on} + T_{off} \tag{4.47}$$

Where d is duty cycle ratio and given as:

$$d = \frac{T_{on}}{T_{sw}} \tag{4.48}$$

Thus load voltage can be controlled by varying duty ratio d. output voltage can be written as

$$V_o = d\,V_g \tag{4.49}$$

The buck converter state equations are presented below. Equations for switch on mode and switch off mode are presented separately

Switch ON:

$$V_g - (v_C + r_C\, C\frac{dv_c}{dt}) = (r_t + r_L)\, i_L + L\frac{di_L}{dt} \tag{4.50}$$

$$i_L = C\frac{dv_c}{dt} + (v_c + r_c\, C\frac{dv_c}{dt})1/R_L \tag{4.51}$$

$$X = \begin{bmatrix} i_L \\ v_C \end{bmatrix} \tag{4.52}$$

$$\dot{X} = A_1\, X + B_1\, V_g \tag{4.53}$$

$$A_1 = \begin{bmatrix} a_{11} & a_{12} \\ a_{13} & a_{14} \end{bmatrix} \tag{4.54}$$

$$B_1 = \begin{bmatrix} b_{\ 11} \\ b_{21} \end{bmatrix} \tag{4.55}$$

$$a_{11} = -\frac{[r_L + r_t + \frac{r_C\, R_L}{R_L + r_C}]}{L} \tag{4.56}$$

$$a_{12} = \frac{-R_L}{[L(R_L + r_c)]} \tag{4.57}$$

$$a_{13} = \frac{-R_L}{[C(R_L + r_c)]} \tag{4.58}$$

$$a_{14} = \frac{-1}{[C(R_L + r_c)]} \tag{4.59}$$

$$b_{11} = \frac{1}{L} \tag{4.60}$$

$$b_{12} = 0 \tag{4.61}$$

Switch OFF:

$$(r_d + r_L)i_L + L\left(\frac{di_L}{dt}\right) + v_c + r_C\left(\frac{dv_C}{dt}\right) = 0 \tag{4.62}$$

$$i_L = C\left(\frac{dv_C}{dt}\right) + \left(v_C + r_C C\left(\frac{dv_C}{dt}\right)\right)\frac{1}{R_L} \tag{4.63}$$

$$\dot{X} = A_2\, X + B_2\, V_g \tag{4.64}$$

$$A_2 = \begin{bmatrix} a_{21} & a_{22} \\ a_{23} & a_{24} \end{bmatrix} \tag{4.65}$$

$$B_2 = \begin{bmatrix} b_{\ 21} \\ b_{22} \end{bmatrix} \tag{4.66}$$

$$a_{21} = -\frac{[r_L + r_d + \frac{r_C R_L}{R_L + r_C}]}{L} \tag{4.67}$$

$$a_{22} = -\frac{R_L}{[L(R_L+r_C)]} \tag{4.68}$$

$$a_{23} = \frac{R_L}{[C(R_L+r_C)]} \tag{4.69}$$

$$a_{24} = \frac{-1}{[C(R_L+r_C)} \tag{4.70}$$

$$b_{21} = 0 \tag{4.71}$$

$$b_{22} = 0 \tag{4.72}$$

$$V_{out} = v_c \tag{4.73}$$

$$C = [0 \quad 1] \tag{4.74}$$

$$D = 0 \tag{4.75}$$

Mean values are obtained from considering both on and off state equation [34].

$$\dot{X} = A_{avg}\, X + B_{avg}\, V_g \tag{4.76}$$

$$A_{avg} = dA_1 + (1-d)A_2 \tag{4.77}$$

$$B_{avg} = dB_1 + (1-d)B_2 \tag{4.78}$$

$$C_{avg} = d\, C \tag{4.79}$$

$$d = \frac{t_{on}}{T} \tag{4.80}$$

4.2 LEVEL CONTROL IN A NONLINEAR SURGE TANK SYSTEM

Consider the surge tank shown below in figure 4-5 with the input u (t), the height of the liquid h(t) and an outlet below the tank characterized by $\bar{d}$ which is related to the outlet pipe diameter.

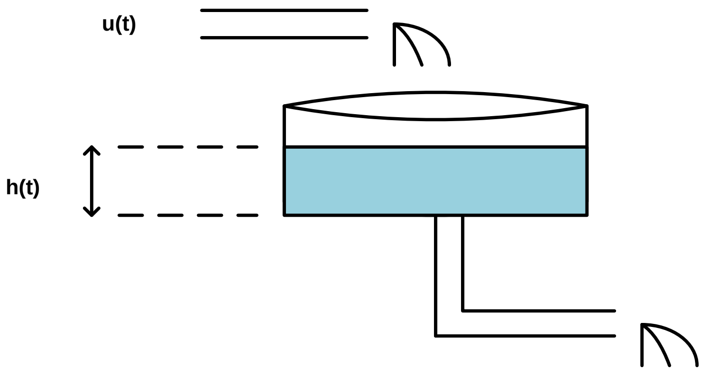

Figure 4-5 Surge Tank System

The input changes dynamically and the objective is to maintain the level of liquid in the surge tank at h while u(t) is varying input to the system. The mathematical model of the system can be given as :

$$h(k+1)=h(k)+T\left(\frac{\bar{d}\sqrt{19.6\,h(k)}}{|\bar{a}h(k)+\bar{b}|}\right)+\frac{\bar{c}}{\bar{a}h(k)+\bar{b}}\,u(k) \qquad (4.81)$$

The end use of the model defines the model requirements. The above model is derived by relating the inputs to measured outputs the needs to be regulated. The volume of liquid in the level vessel varies as e function of the inlet and the outlet flow rates.

The density is assumes to e constant. Balanced equation based on an instantaneous rate of change can be given as

$$\begin{bmatrix}\text{rate of change of}\\ \text{total mass of the}\\ \text{fluid inside the vessel}\end{bmatrix}=\begin{bmatrix}\text{mass flow rate}\\ \text{of fluid}\\ \text{into the vessel}\end{bmatrix}-\begin{bmatrix}\text{mass flow rate}\\ \text{of fluid}\\ \text{out of the vessel}\end{bmatrix} \qquad (4.82)$$

Following notations are used in the modeling equations:

u = inlet volumetric flow rate (volume/time)

o = outlet volumetric flow rate (volume/time)

V = volume of liquid in the vessel

h = height of the liquid in the vessel

ρ = liquid density (mass/volume)

A = cross sectional area of vessel = $\bar{a}h(k)+\bar{b}$ where $\bar{a}>0$ and $\bar{b}>0$

$\bar{c}$ = clogging factor ε (0,1). $\bar{c}$ =1 shows that the filter in the actuator

$\bar{d}$ = area of liquid discharge related to the oulet pipe diameter.

Where the total mass of fluid inside the vessel is denoted by V ρ, the rate of change is $\frac{dV\rho}{dt}$, and the density of the outlet stream is equal to the density oof the vessel contents.

$$\frac{dV\rho}{dx}=\bar{c}u\rho-o\rho \qquad (4.83)$$

The volumetric flow rate Q can be written as Q=A v where A=area and v=velocity. Thus, the outlet volumetric flow rate can be expressed with $\bar{d}$ and oulet velocity v as:

$$o=\overline{dv} \qquad (4.84)$$

If we equate kinetic and potential energies of liquid, we get

$$mgh=\frac{1}{2}mv^2 \qquad (4.85)$$

g = acceleration due to gravity and m= mass of liquid . m get cancelled on both sides of the equation and hence we get

$$v = \sqrt{2gh} \tag{4.86}$$

Using equation 4.83 & 4.84 a modified equation 4.82 after cancelling out ρ on both sides of the equation

$$\frac{dV}{dx} = \bar{c}u - \tilde{d}\sqrt{2gh} \tag{4.87}$$

Volume V of the tank can be expressed as V=Ah. Thus, the equation can be modified as.

$$\frac{dh}{dx} = \frac{\bar{c}u - \tilde{d}\sqrt{2gh}}{A} \tag{4.88}$$

This can be modified further as

$$\frac{dh}{dx} = \frac{\bar{c}u}{\bar{a}h(k)+\bar{b}}u - \frac{\tilde{d}\sqrt{2gh}}{\bar{a}h(k)+\bar{b}} \tag{4.89}$$

Putting the values of g=9.8 and converting into discrete form

$$h(k+1)=h(k) + T\left(\frac{-\bar{d}\sqrt{19.6\,h(k)}}{|\bar{a}h(k)+\bar{b}|}\right) + \frac{\bar{c}}{\bar{a}h(k)+\bar{b}}\,u(k) \tag{4.90}$$

In the simulation model we use $\tilde{a} = 0.01, \tilde{b} = 0.2, \tilde{c} = 1, \tilde{d} = 1$ and $T = 0.1$. We assume that the plant input saturates at ± 50 so that if the controller generates an input u(k), then

$$u(k)=\left\{\begin{matrix} 50 & \text{if } u(k) > 50 \\ u(k) & \text{if} -50 < u(k) < 50 \\ -50 & \text{if } u(k) < -50 \end{matrix}\right\} \tag{4.91}$$

Since the liquid level cannot go negative, the system model is modified as

$$h(k+1)=\max\left\{0.0001, h(k) + T\left(\frac{-\bar{d}\sqrt{19.6\,h(k)}}{|\bar{a}h(k)+\bar{b}|}\right) + \frac{\bar{c}}{\bar{a}h(k)+\bar{b}}\,u(k)\right\} \tag{4.92}$$

The above model described by (4.92) is used for adaptive control of the liquid level [40] using Bacterial Foraging Algorithm, Particle Swarm Optimization and Hybrid PSO-BFOA.

CHAPTER 5

SIMULATION RESULTS AND DISCUSSION

The simulation results for PID tuning, controller design for buck and boost converter and liquid level control of surge tank by Particle Swarm Optimization, Bacterial Foraging Algorithm and Hybrid PSO-BFA are presented and described in the following section.

5.1 PID TUNING

The block diagram given in figure 5-1 outlines the process of PID tuning using algorithms. In evaluation part the plant is analyzed according to the output and the error generated and then this value modify the algorithm used and then parameters of controller is decided.

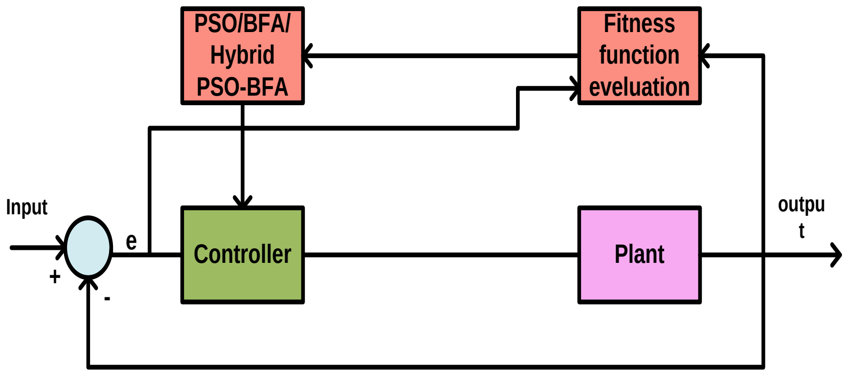

Figure 5-1 Block diagram of plant with controller tuned by algorithm

The transfer functions $\frac{5}{S^4+3S^3+7S^2+5S}$ is used here as a plant and PID tuning using PSO, BFOA, and Hybrid PSO-BFA is performed and value of K_p, K_i and K_d is found. The cost function used here is the integral of the squared error and minimum cost function value is found for all three algorithms. The cascaded feedback strategy is used in which PID controller is in series with the process to be tuned. The values of the parameters used in PSO, BFA and Hybrid PSO-BFA are given in the table 5-1

Table 5-1 Parameter Of algorithms

Parameter	Hybrid PSO-BFA	BFA	PSO
S	100	100	100
P	3	3	3
N_s	4	4	-
N_c	10	50	50
N_{re}	4	4	-
N_{ed}	2	2	-
P_{ed}	0.25	0.25	-
C_1	1.5	-	2
C_2	0.2	-	2
W	0.5	-	0.9

p in the above table denotes the dimension of the search space Figure 5-2 show Step response obtain for PSO.

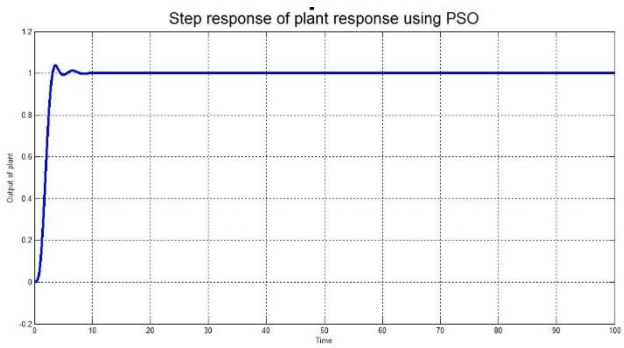

Figure 5-2 Step response using PSO

The simulation was run for 100 sec. In the step response obtain for PSO we can observe that there is an 1^{st} and 3^{rd} overshoot. The overshoots are of less magnitude and do not remain for much time. Later on with time the output response track the desired response. Hence stable response is obtain.

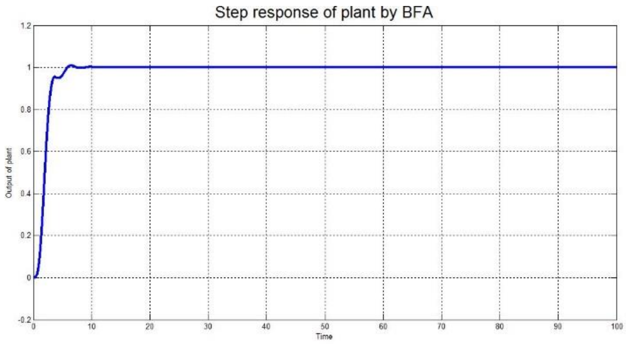

Figure 5-3 Step response using BFA

From the step response obtain for the given transfer function using BFA given in figure 5-3 , an undershoot is seen in the output response but later on a little overshoot is seen and after that output response track the desired response.

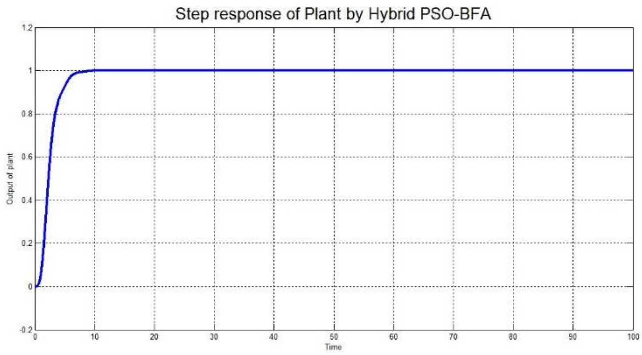

Figure 5-4 Step response using Hybrid PSO-BFA

For the above mentioned Transfer function, the maximum deviation from the desired output is found for PSO. The best graph was obtained for Hybrid PSO-BFA.

Table 5-2 Parameters of Controller

Parameters	PSO	BFA	Hybrid PSO-BFA
K_P	0.7852	0.8685	0.8314
K_i	0.6400	0.2755	0.5564
K_d	3.6063	2.5868	2.1744

Table 5-2 shows the value of K_P, K_i and Kd obtain for PSO, BFA and Hybrid PSO-BFA.

From the above table we can conclude that the values of parameter obtain by the BFA and Hybrid algorithm is nearly same but value for PSO varies.

Table 5-3 Cost Function Value

Algorithm	Minimum Cost Function value
PSO	17.2941
BFA	15.7815
Hybrid PSO-BFA	15.2806

Table 5-3 shows minimum cost function value obtained for PSO, BFA and Hybrid PSO-BFA. From the table 5-3 it is reflected cost function value for PSO has the worst minimum cost function value.

5.2 CONTROLLER DESIGN FOR BOOST AND BUCK CONVERTER

Boost Converter

Mean values of state equations for boost converter is shown below, the detailed explanation is given in chapter 5.

$$\dot{X}=A_{avg}X + B_{avg}V_g \tag{5.1}$$

$$A_{avg}=dA_1 + (1-d)A_2 \tag{5.2}$$

$$B_{avg}=dB_1 + (1-d)B_2 \tag{5.3}$$

$$C_{avg}= dC_1 + (1-d)C_2 \tag{5.4}$$

$$d = \frac{t_{on}}{T} \tag{5.5}$$

Table 5-4 Parameters value for Boost converter

Parameters	**Values**
V_g	12 V
r_L	2 Ω
R	0.2 Ω
r_d	0.7Ω
r_C	0.5Ω
r_{Load}	120 Ω
L	750 Mh
C	100 Mf
D	0.5

The values of A_{avg}, ,B_{avg} and C_{avg} matrix is calculations using values given in table 5-4.

$$A_{avg} = \begin{bmatrix} 0.335 \times 10^3 & -0.6635 \times 10^3 \\ 4.979 \times 10^3 & -82.98 \end{bmatrix} \tag{5.6}$$

$$B_{avg} = \begin{bmatrix} 1.33 \times 10^3 \\ 0 \end{bmatrix} \tag{5.7}$$

$$C_{avg} = [2.4895 \quad 0.995]$$ (5.8)\

Open loop response of boost converter is shown in figure 5-5. As from the figure it can be seen that when boost is operated in open loop configuration it does not give the desired result, from the graph it can be seen that offset remains and there is disturbance in the voltage profile.

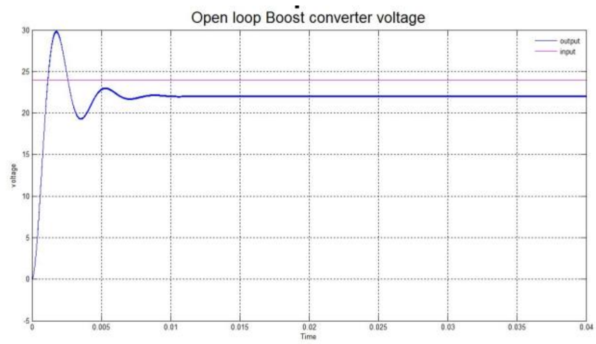

Figure 5-5 Open loop response of Boost converter

To improve the response of boost converter it is operated in close loop and the parameters of controller is determined by the determined by the PSO, BFA and Hybrid PSO-BFA algorithm. Then the controller parameter will adjust the duty ratio so that it can on and off the switch to produce the desired output.

The close loop operation of boost converter is shown in figure 5-6.Here Algorithms will search for the parameters of controller according to the fitness function. For calculating fitness function in our simulation integral square error method is used. In which, error obtained is first squared and then integration is applied on it.

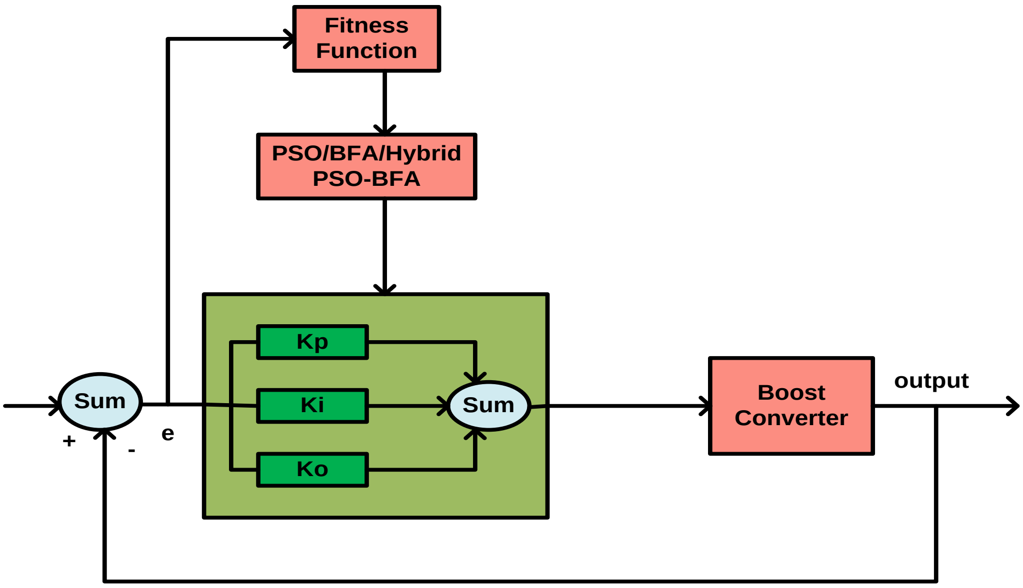

Figure 5-6 Closed loop diagram for Boost converter

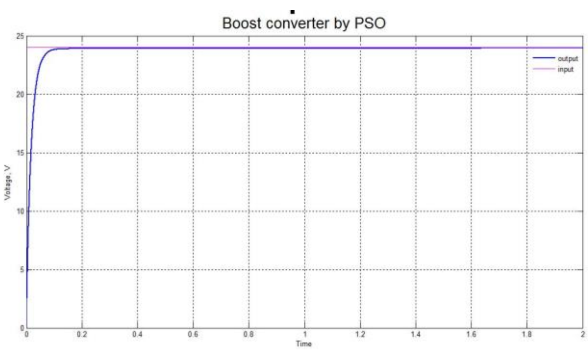

Figure 5-7 Close loop response of Boost converter using PSO

Figure 5-7 is showing close loop response of boost converter obtain by PSO. The graph obtain is better the open loop response as there is less no overshoot, oscillation and offset is eliminated. But as it takes more time to settle and show little offset for till long time, so overall it shows sluggish response.

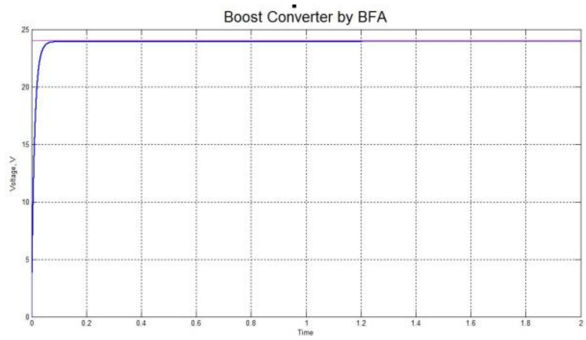

Figure 5-8 Close loop response of Boost converter using BFA

From graph in figure 5-8 shows the close loop response of boost converter obtain using BFA the response obtain. Output response obtains eliminates all undesirable characteristics seen in open loop respons. Settling time obtain is also less.

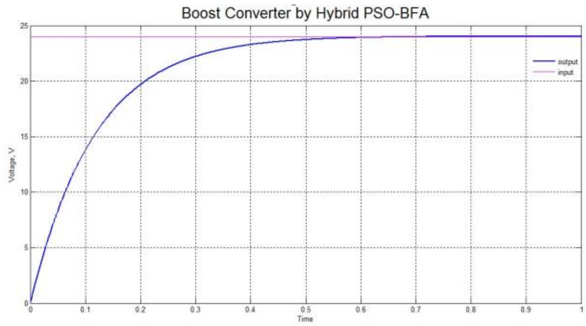

Figure 5-9 Close loop response of Boost converter using Hybrid PSO-BFA

In figure 5-9 we observe that for Hybrid PSO-BFA the closed loop response obtain is tracking the desired output and gives better result than the PSO and BFA.

We observed from the figure 5-9 that response obtain show more delay and rising time but setting time is very much less than what obtain with BFA and PSO.

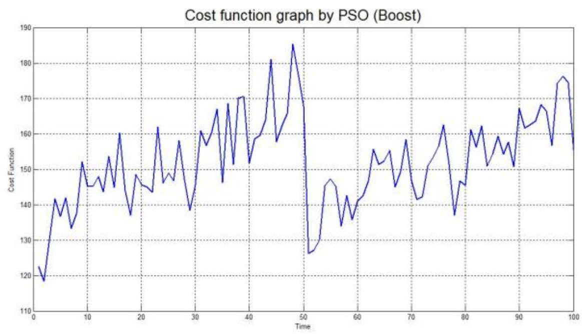

Figure 5-10 Cost function graph for PSO (Boost)

Figure 5-10 shows cost convergence graph for Boost converter Operated with PSO and it is observed that cost convergence is very poor.

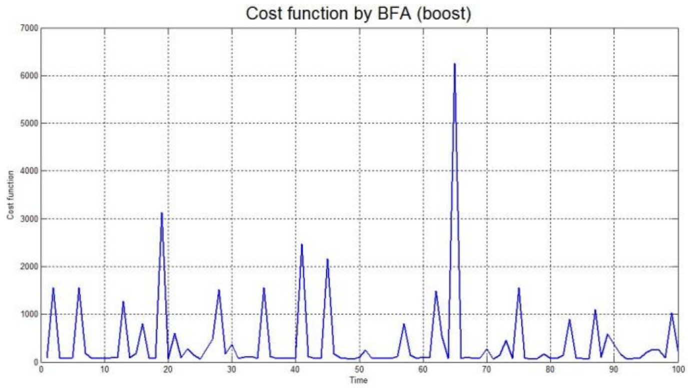

Figure 5-11 Cost function graph for BFA(Boost)

Figure 5-11 shows the cost function convergence graph for BFA. Large number of fluctuation is observed in the cost convergence hence poor cost convergence.

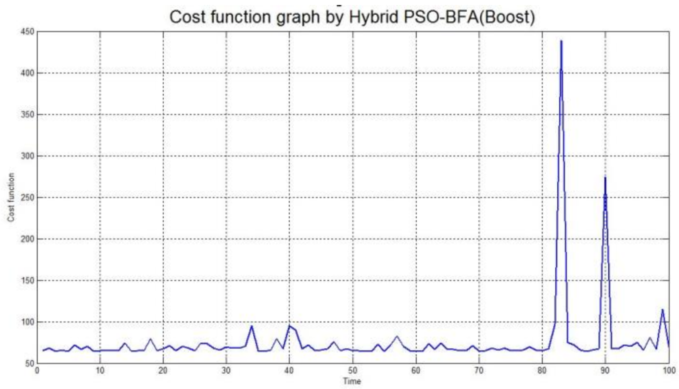

Figure 5-12 Cost function graph for Hybrid PSO-BFA(boost)

Figure 5-12 depicts the cost function convergence graph for Hybrid PSO-BFA and from the graph it can be seen that the convergence graph gives better result as it is continuously for large span of time give the less value for cost function.

When we compare convergence graph of Hybrid with that of BFA and PSO we can find better convergence as less oscillation are observed and even highest deviation is observed only twice in complete graph while in the other two very large oscillation are noticed.

The worst convergence was seen for PSO as for larger span of time it shows very deviation of large magnitude

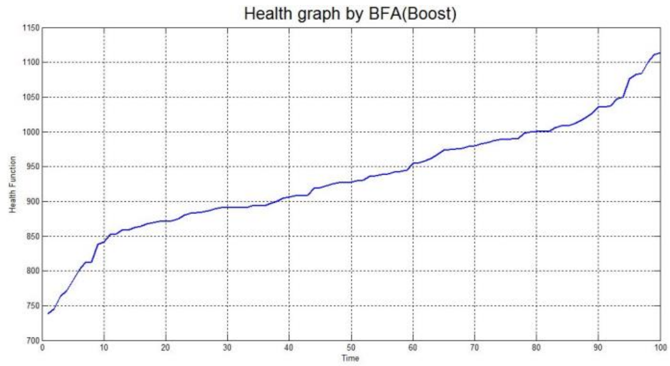

Figure 5-13 Bacteria health graph for BFA (Boost)

Figure 5-13 shows the health of bacteria's during different span of time and the health of bacteria has improved with time.

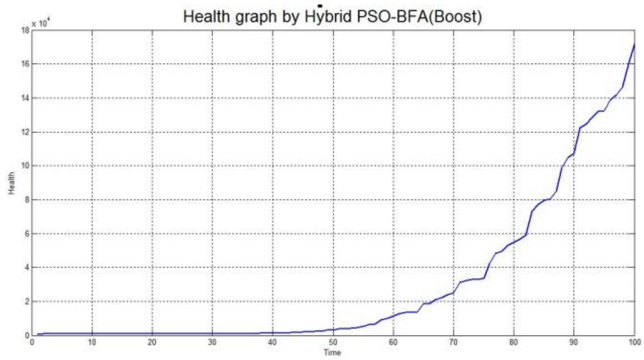

Figure 5-14 Bacteria health graph for Hybrid PSO-BFA (Boost)

Figure 5-14 shows the health of the bacteria's for Hybrid PSO-BFA algorithm and it can be observed that in the starting of the algorithm the health of bacteria's is poor and is improved latter with time.

The values of Controller Gains obtain is summarized in the table 5-5 given below:

Table 5-5 Controller value obtain for boost converter

Parameters	**PSO**	**BFA**	**Hybrid PSO-BFA**
K_P	1.19	2.80	0.277
K_i	0.90	0.67	0.98
K_d	0.030	0.024	0.032

From the values of the parameters obtain for boost controller by all these three algorithms have been summarized in the table 5-6 below:

Table 5-6 Comparison of result obtain from PSO, BFA and Hybrid PSO-BFA

Parameters	**Delay Time (sec)**	**Rise Time (sec)**	**Peak Time (sec)**	**Overshoot (%)**	**Settling Time (sec)**
PSO	0.02	0.16	-	-	1.63
BFA	0.03	0.08	-	-	1.2
Hybrid PSO-BFA	0.08	0.54	-	-	0.71

From the table we conclude that the best delay time is provided by PSO but it tends to sluggish in term of settling time.

From BFA performance we can observe that it improve the its rise time and settling time.

But result obtained by the Hybrid PSO-BFA algorithm shows poor delay and rise time but better settling time response than Hybrid PSO-BFA .

Buck Converter

Mean values of state equations of buck converter are given below, their detailed explanation is given in chapter 5

$$\dot{X} = A_{avg} X + B_{avg} V_g \tag{5.9}$$

$$A_{avg} = dA_1 + (1-d)A_2 \tag{5.10}$$

$$B_{avg} = dB_1 + (1-d)B_2 \tag{5.11}$$

$$C_{avg} = dC \tag{5.12}$$

Value of buck converter parameters are shown in table 5-7. Here input voltage is 100 V and according to duty ratio of d= 0.5we get output 50 V

Table 5-7 Parameter values for buck converter

Parameters	Values
V_g	100 V
r_L	0.7 **Ω**
r_t	0.2 **Ω**
r_d	0.7**Ω**
r_c	1.18**Ω**
r_{Load}	118 **Ω**
L	0.42 **mH**
C	1450 **μF**
D	0.5

The values of A_{avg}, ,B_{avg} and C_{avg} matrix is calculation using values given in table

$$A_{avg} = \begin{bmatrix} -4.6363 & -4.7146 \times 10^3 \\ 13.828 \times 10^2 & -11.572 \end{bmatrix} \tag{5.13}$$

$$B_{avg} = \begin{bmatrix} 2.380 \times 10^3 \\ 0 \end{bmatrix} \tag{5.14}$$

$$C_{avg} = [0 \quad 0.5] \tag{5.15}$$

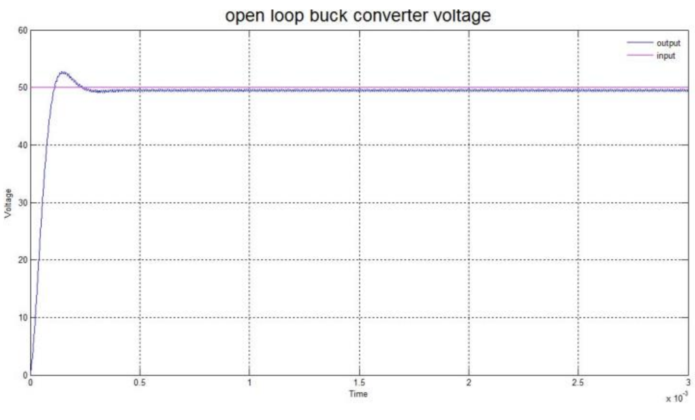

Figure 5-15 Open loop response of Buck converter

Figure 5-15 shows the open loop graph of Buck converter. From figure we can observe that there is an offset and disturbance in the voltage response of converter. To eliminate these problems buck converter is operated in closed loop and parameters of controller is tuned using PSO, BFA and Hybrid PSO-BFA.

Figure 5-16 depicts the diagram for buck converter in closed loop. Here the error generated is used to calculate the fitness function after that according to fitness function parameter of the controller are tuned by the PSO, BFA or Hybrid PSO-BFA. The fitness function is calculated using integral square error method.

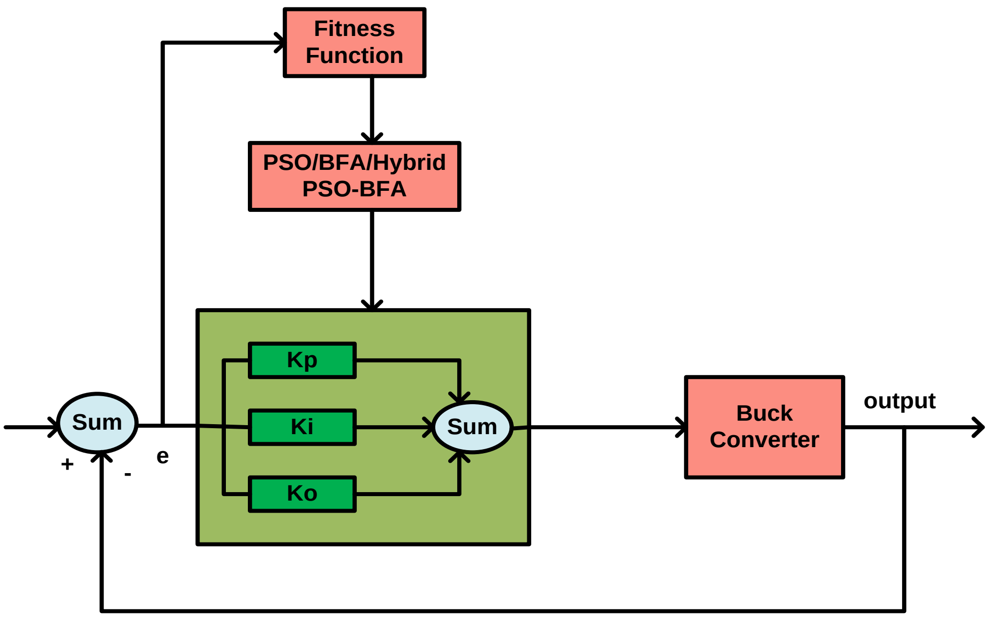

Figure 5-16 Closed loop diagram for Buck converter

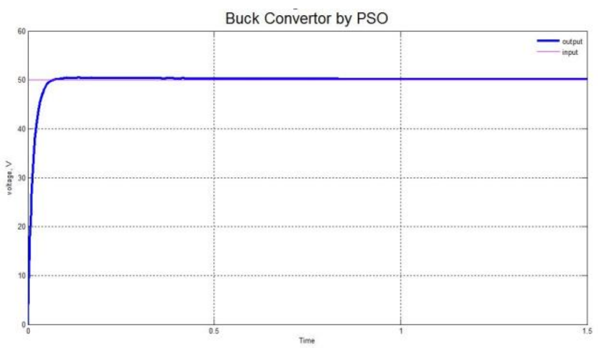

Figure 5-17 Close loop response of Buck converter using PSO

Figure from 5-17 shows the closed loop response of buck converter obtained using PSO. In response we observe no overshoot. Even oscillation during open loop response of buck converter are removed to large extent. But still there is not smooth transition to the desired output and little offset still persist.

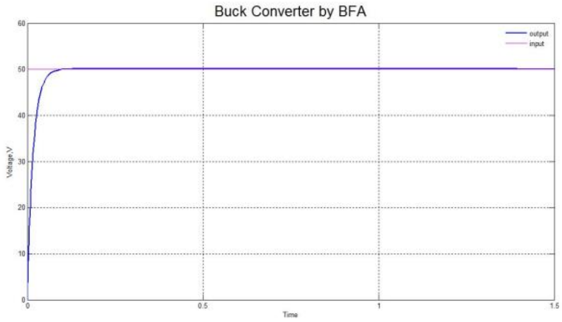

Figure 5-18 Close loop response of Buck converter using BFA

In Figure 5-18 Oscillations are completely vanished and smooth transition is observed. But offset persist for very large span of time which makes response to settle after long delay of time.

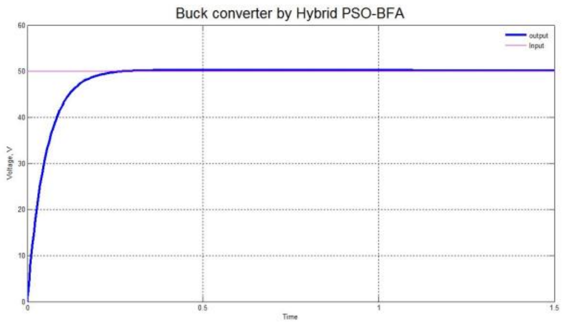

Figure 5-19 Close loop response of Buck converter using Hybrid PSO-BFA

Figure 5-19 shows the closed loop response obtained for the parameters value of controller obtained for Hybrid PSO-BFA. The closed loop response obtain is similar to the response shown by BFA but delay and rise time are little more than BFA but setting time is improved with Hybrid PSO-BFA.

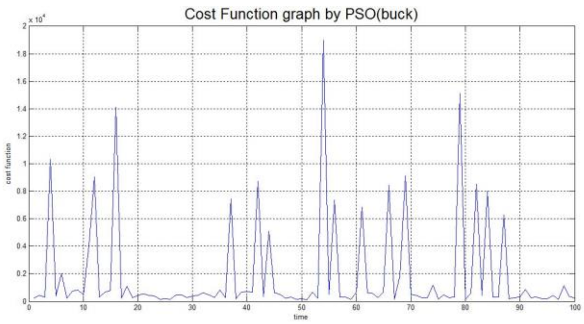

Figure 5-20 Cost function graph for PSO (Buck)

Figure 5-20 shows cost convergence graph of PSO for buck converter. From the graph obtain we can observed that good minimum cost function is obtained during run time of algorithm.

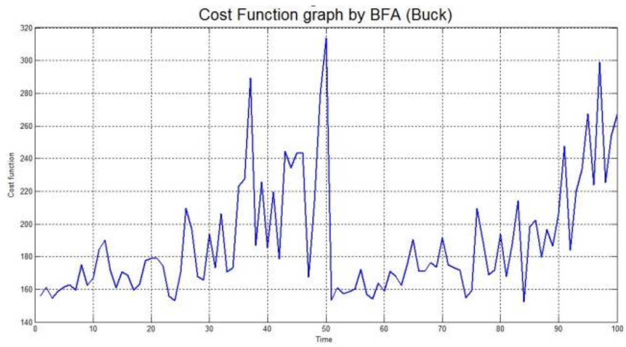

Figure 5-21 Cost function graph for BFA (Buck)

Figure 5-21 shows the cost convergence graph for BFA , the graph gives poor result then as large variations are observed.

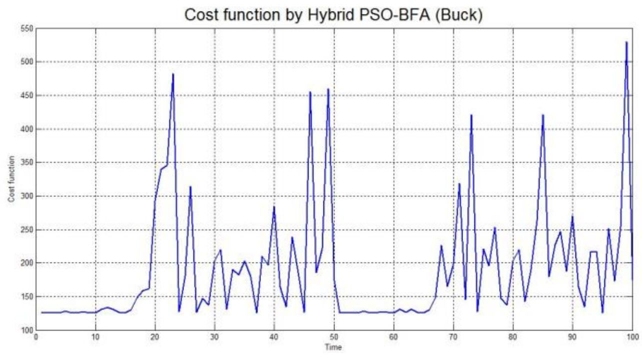

Figure 5-22 Cost function graph for Hybrid PSO-BFA (Buck)

Figure 5-22 shows the cost convergence graph for the Hybrid PSO-BFA and the graph obtain is better than the one obtain for BFA.

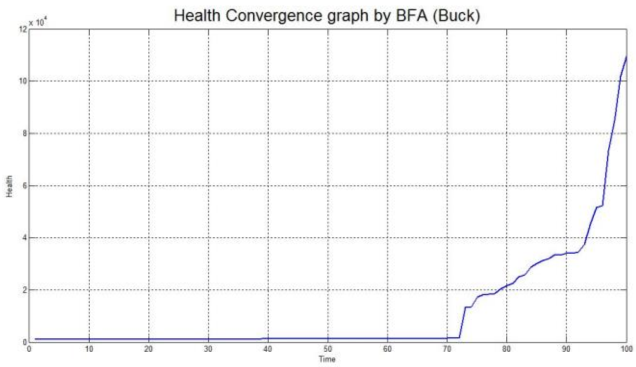

Figure 5-23 Bacteria health graph for BFA (Buck)

Figure 5-23 shows the health graph of bacteria's for BFA. The health of bacteria remain deteriorated for large span of time, i.e. more than three forth of the time span and shows good health in remaining time.

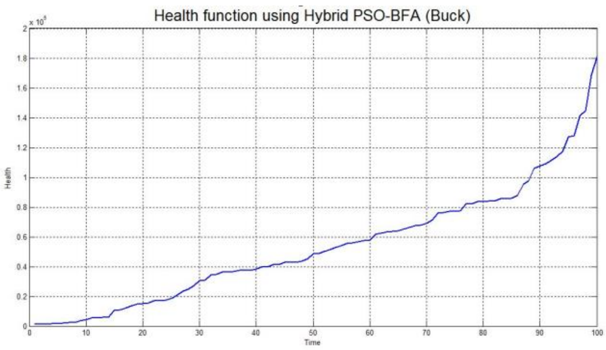

Figure 5-24 Bacteria health graph for Hybrid PSO-BFA(buck)

Figure 5-24 shows the health graph of bacteria for Hybrid PSO-BFA and the health of bacteria is continuously improving with time.The values of parameters obtain is summarized in the table 5-8 given below:

Table 5-8 Controller values obtain for buck converter

Parameter	PSO	BFA	Hybrid PSO-BFA
K_P	3.397	2.898	0.892
K_i	3.83	1.88	0.718
K_d	0.042	0.044	0.030

Table 5-9 show comparison between the parameters obtained for the all three algorithm.

Table 5-9 Comparison of result obtain from PSO, BFA and Hybrid PSO-BFA

Parameters	Delay Time (sec)	Rise Time (sec)	Peak Time (sec)	Overshoot (%)	Settling Time (sec)
PSO	0.08	0.17	-	0	Offset
BFA	0.089	0.2	-	0	1.6
Hybrid PSO-BFA	0.14	0.30	-	0	1.3

Response obtain with BFA and Hybrid PSO-BFA is almost similar in terms of smooth transition but settling time get reduce in Hybrid PSO-BFA.

5.3 LEVEL CONTROL IN A SURGE TANK SYSTEM

The Liquid level control in a surge tank system model and its equations are described in this section. The main motive here is to estimates the height of the liquid in a tank by representing the estimate in the form given as:

$$\hat{h} = \alpha + \beta u(k) \tag{5.16}$$

Where α and β are calculated by using the PSO, BFA or Hybrid PSO-BFA.The fitness function is described as

$$J(\theta^i)=\left(\hat{h}^i(k) - h(k)\right)^2 \tag{5.17}$$

This equation is calculated by using the location of particles or bacteria and the least value obtain for the fitness function is the best one. Accordingly the estimate is calculated for the next iteration and the process is repeated for certain number of iterations. Population of swarm is kept same for all the three algorithms i.e. 100 and the algorithm performs 1000 iteration with sampling time of 0.1 second. For the surge tank the reference input given is the square wave. A time scale of 100 seconds is used.

In figure 5-25 Liquid level tracking is obtain for PSO algorithm

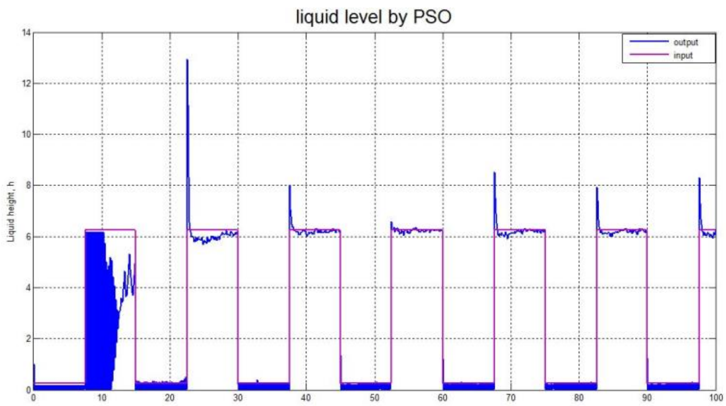

Figure 5-25 Liquid level tracking by PSO

It is shown from the figure 5-25 that the tracking of liquid level is poor as large number of oscillation are there in the tracking.

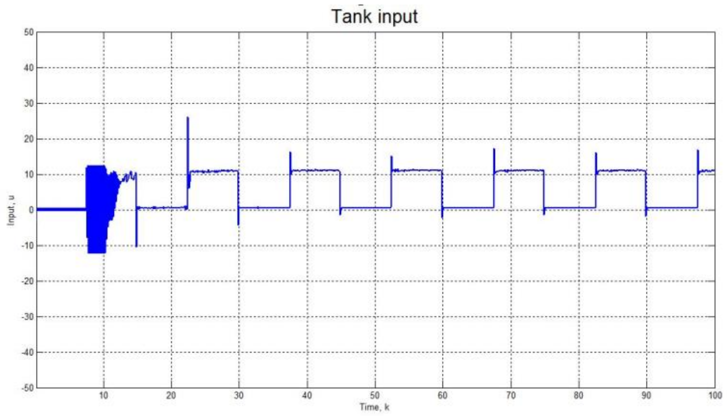

Figure 5-26 Reference input tracking by PSO

From figure 5-26 it is evident that tracking of liquid level is poor because the tracking of plant input is poor for PSO.

In the starting high disturbance is observed but after 20 sec response starts to improve and tracking improve , but persistant disturbance is there as during complete response overshoot in tracking is observed.

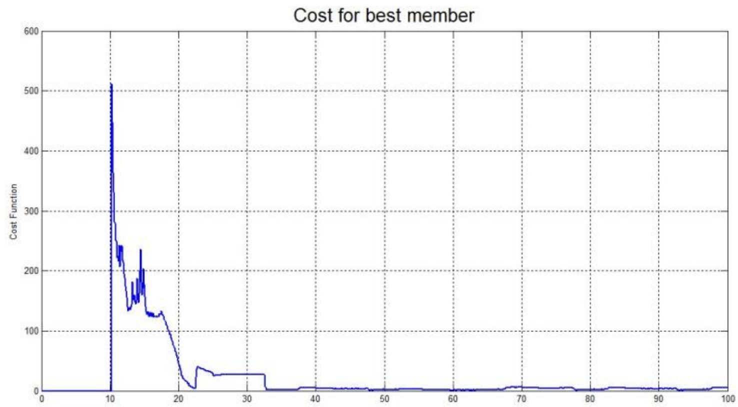

Figure 5-27 Cost function graph for PSO

From the cost function graph given in figure 5-27 it can be seen that there is very much variation in the cost function and cost function does not approaches to zero but to some minimum value.

Figure 5-28 shows Liquid level tracking by BFA. From figure 5-28 is can be concluded that tracking of liquid level by BFA algorithm is improved.. But constant type of error persist in the response till end.

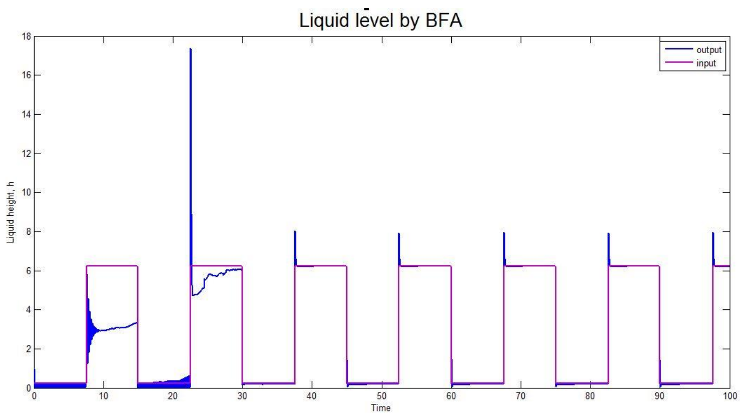

Figure 5-28 Liquid level tracking by BFA

In comparison to the PSO response, less number of vibration is observed in booth liquid level and input tracking. Maximum error obtain for liquid level tracking using BFA is obtained only once after that tracking is improved.

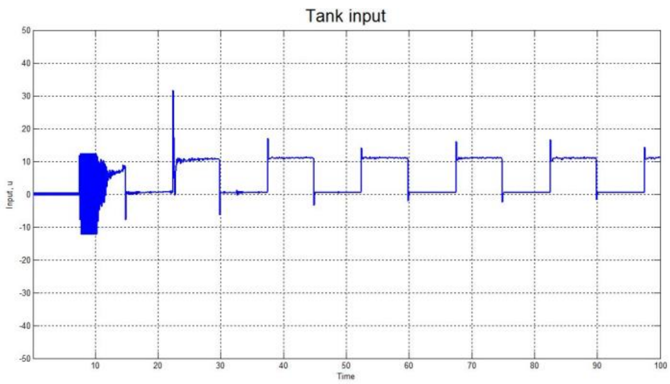

Figure 5-29 Reference input tracking by BFA

From figure 5-29 Tank input liquid level tracking obtain is similar to what we obtain using PSO. Not much improvement is observed in this response.

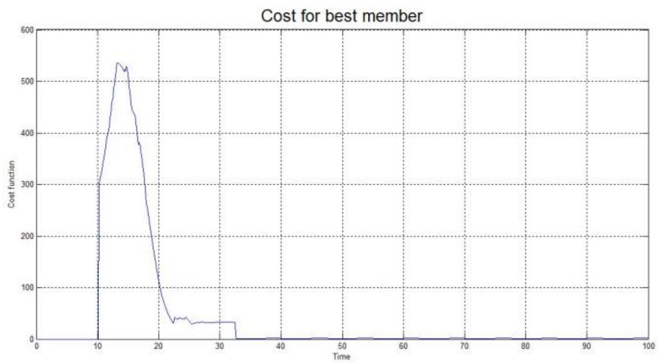

Figure 5-30 Cost function graph for BFA

From Figure 5-30 it can be stated that in the stating time interval we can observed large variation in cost function but after 30th iteration cost function graph is showed good convergence to minimum cost.

Figure 5-31 is liquid level response obtain for Hybrid PSO-BFA.

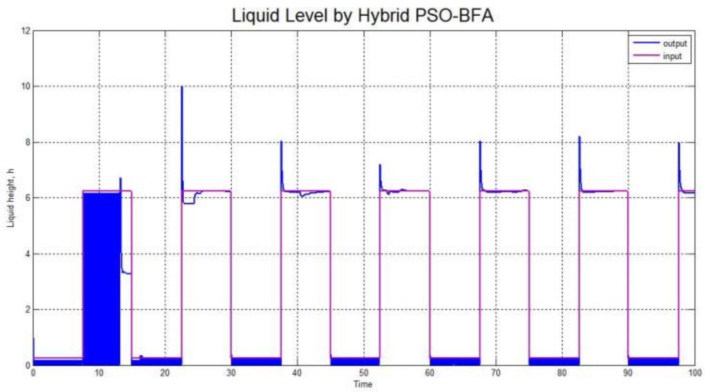

Figure 5-31 Liquid level tracking by Hybrid PSO-BFA

From figure 5.31 it is observed that Hybrid PSO-BFA obtain is successful in tracking the liquid level therefore response is improved than PSO and Less oscillation are seen. But Results by Hybrid PSO-BFA are not much better than BFA.

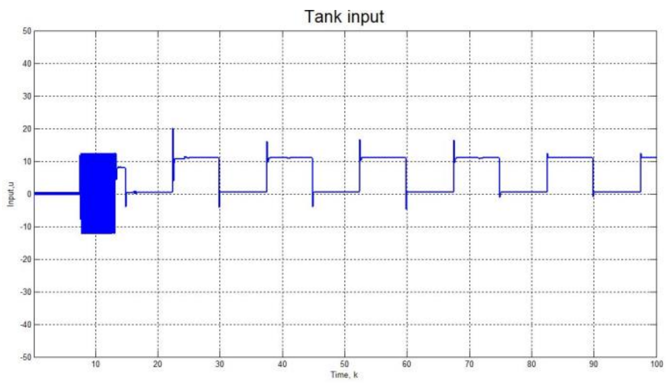

Figure 5-32 Reference tracking by Hybrid PSO-BFA

From figure 5-32 it is observed that Hybrid PSO-BFA reference input tracking is successfully obtain glitches finally goes reducing with time and in the last it completely tracks reference input.

From figure 5-33 it is evident that value of minimum cost function approaches zero and cost convergence is better than the PSO and BFA because little variations are observed during given span of time.

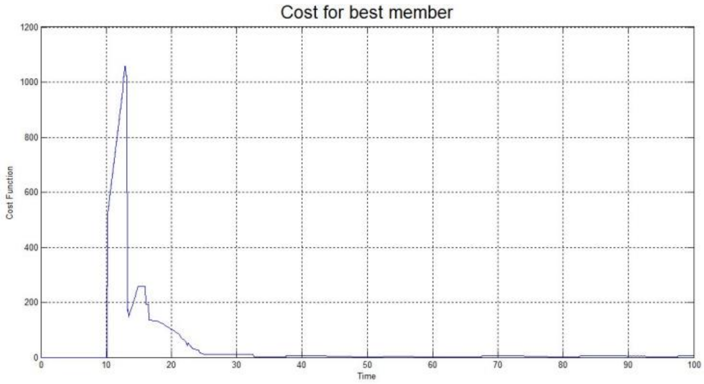

Figure 5-33 Cost function graph for Hybrid PSO-BFA

From the above figures obtain for liquid level tracking of surge tank by PSO, BFA and Hybrid PSO-BFA. Over all it can be concluded that the best response was obtain by Hybrid algorithm for tracking of liquid level and reference input and this happens because the tracking of plant nonlinearities improve tremendously by this algorithm. Even the minimum value of cost function obtain for Hybrid PSO-BFA reached zero much faster than obtain by other two algorithm . Table 5-10 Shows minimum cost value obtains from mention three algorithms.

Table 5-10 Minimum cost for surge tank level system

Algorithm	**Minimum Cost**
PSO	8.5008
BFA	0
Hybrid PSO-BFA	0

CHAPTER 6

CONCLUSION AND FURTHER SCOPE OF WORK

6.1 Conclusion

This dissertation includes the study and designing of Evolutionary algorithm based controller. Three different evolutionary algorithms are used and they are: Particle Swarm Optimization (PSO) Algorithm, Bacterial Foraging Optimization Algorithm (BFOA) and Hybrid PSO-BFA Algorithm. In this thesis design of PID controller using above mentioned algorithms is implemented. On the basis of tuning of PID controller by these mentioned algorithms, converter controller is also designed using concept of PID tuning.

The main thrust of the work presented in this thesis is to suggest a suitable strategy for adaptive control of two nonlinear systems using three methods i.e. PSO, BFOA, hybrid PSO-BFA. The nonlinear systems which are used for analysis were buck converter , boost converter and Surge tank system.

The result was obtained for Adaptive PID tuning of simple second order plant. From the result it was found that little overshoot was seen with PSO step response while other two algorithms gave no overshoot and less oscillation. Hybrid algorithm converges more smoothly towards the step response then the PSO and BFOA. The best value for PID parameters were obtain by Hybrid algorithm then the BFOA and PSO. The response time was also best for hybrid one. BFOA also shows better result than PSO this is because due to the introduction of randomness during optimization which help in reduction of time needed for computation. The result obtain for boost converter shows that three of the mention algorithm were successful in controlling the output of in accordance with the desired trajectory.

The result obtain for boost converter shows that three of the mention algorithm were successful in controlling the output of in accordance with the desired trajectory. If analyzed on a comparative basis it can be seen that in almost all the response hybrid algorithm overall gave best. From the output voltage graph it was observed that for PSO offset was seen and settling time was also higher than the other two algorithms. BFOA also shows good response and settling time also improved with BFOA but the best result was obtained by the hybrid algorithm. In cost convergence graph also best result was provided by Hybrid algorithm whereas PSO shows very poor convergence and BFA shows very large variation in cost convergence graph. BFOA show inferior performance, because during chemotaxis it search in random direction and this may lead to delay in reaching the global solution, so this characteristic is improved hybrid PSO-BFA. Similarly in case of buck converter best result was obtain by the hybrid algorithm. Here also the values obtain by PSO were less better than the other two as offset was observed during complete response.

In the liquid level control of surge tank the result obtain by both BFA and Hybrid PSO-BFA were comparable but overall Hybrid PSO-BFA shows good results than the other two algorithm as the tank input was completely tracked in it and cost convergence reached minimum value faster than BFA and PSO. PSO shows the worst result as we observed large deviation in the graph obtain by this algorithm.

From the simulation results, we can conclude the Hybrid PSO-BFA algorithm in which tumbling of bacteria is decided by PSO strategy is used greatly improve the optimization performance. It can be effectively used when the real problems evaluation of a function become difficult and a long parameter tuning is not feasible and the robustness of the algorithm with respect to the variation of control parameters becomes a necessity.

6.2 FUTURE SCOPE

In this dissertation implementation of adaptive control techniques employing three different bio inspired algorithms i.e. BFOA, PSO and hybrid PSO-BFOA presented and effective and improved control of complex nonlinear system was obtained.

BFOA can be applied to problem having large search space because it is a random search algorithm and shows good result. BFOA also suffer from a problem, as all bacteria involve take time in reaching a particular solution as they search vast space. We can also make BFOA effective by having its certain parameter fixed or vary depending upon the application for which it is performed. Parameters like dispersal probability, population of bacteria, number of chemotactic steps & swim length can be used to provide diversification in result.

PSO algorithms can outperform various other conventional algorithms for solving many optimization problems which can be evidently seen in various studies PSO is preferred where there is small searching space involved. But PSO shows premature and slow convergence speed. This happens because of the PSO parameters chosen for particular application remains constant throughout the run. But these disadvantages of PSO can be overcome by making parameters like learning or acceleration constant rate adaptive in nature; so that they can vary according to the environment change.

While in this dissertation PSO hybrid with BFOA is also provided. According to the results obtain Hybrid PSO-BFO algorithm it can be concluded that it offers advantage over both PSO and BFA as it tries to find out of local optimal solution and make effort to converge at global optimal solution. From it convergence characteristics it is found to be better and provide solution of higher quality than PSO and BFOA. But values of parameters should be chosen carefully because if not then they can cause slow or no convergence of algorithm. These problems can also be solved or result can be improved by using other intelligent algorithm like monkey, firefly, cockroach etc. For Hybrid PSO-BFA, one of the future research which can be implemented is trying to make the algorithm perform faster by as it tends to be slower than PSO and BFA. Another research area is by hybridizing PSO or BFA with other algorithms and applying them on problems and comparing result with original ones.

REFRENCES

1. G.Theraulaz And J.L. Deneubourg, “On Formal Constraints In Swarm Dynamics”. Proceedings Of The 1992 IEEE International Symposium On Intelligent Control, 1992, pp 225-233
2. R.Eberhart And J. K Ennedy “A New Optimizer Using Particle Swarm Theory.” Proc Of 6th Int. Symp Micro Machine And Human Sciences(MHS 95), 1995, pp 39-43.
3. J.Kennedy And R.Eberhart, “PSO”. Proc. Of IEEE Int. Conf. Neural Networks, Vol.4 1995, pp 1942-1948.
4. K.Ogata “Modern Control Engineering” 1997 3nd Edition Prentice Hall.
5. Mitchell Melanie “An Introduction To Genetic Algorithms” 5th printing 1999 The MIT Press Cambridge.
6. E.Bonabeau, M.Dorigo and G. Theraulaz, “Swarm Intelligence: From Natural To Artifical Intelligence” 1999 Oxford University Press.
7. Gang Feng and Rigelio Lozano “Adaptive control system” 1999 Reed Educational and Professional Publishing Ltd.
8. B.K Bose “Modern power electronics and AC drives”. 2001 Prentice Hall Publication.
9. I.J. Nagrat and M.Gopal “Control System Engineering” 2002 New Age International.
10. Wan Nur Eliana, Afif Wan Titik Khawa, Abdul Rahman & Zuhaina Zakaria, “Optimal Load Shedding Using Bacterial Foraging Optimization Algorithm”. IEEE 4th Control And System Graduate Research Colloquium, Aug 2003, pp 93-97.
11. Ray. T. and Liew. K “Society and civilization: An optimization algorithm based on the simulation of social behavior”. IEEE Transactions on Evolutionary Computation 2003, pp 386-396.

12. Riza, A.Y "A novel particle swarm optimization approach for product design and manufacturing". International Journal for Advance Manufacture Technology, Springer-Verlag 2003, pp.617-628.
13. Ashish Tewari "Modern Control Design with Matlab and Simulink" 2003 Wiley.J and Sons, pp 1-12.
14. B.C Kuo and Farid Golnaraghi "Automatic Control System" 8th 2003 Edition Wiley.J and Sons, pp 138-195.
15. S.Mishra "Hybrid Least-Square Adaptive Bacterial Foraging Strategy For Harmonic Estimation" .IEEE Proc.-Gener. Transm. Distrib., Vol. 152, No. 3, May 2005, pp 379-389.
16. Seok Lee, Zong Woo and Geem Kang "A new meta-heuristic algorithm for continuous engineering optimization: harmony search theory and Practice" Elsevier Journal Computational Methods Application Mechanical Engineering 2005 pp. 3902–3933.
17. K.M Passino,"Adaptive Control", In Biomimicry For Optimization, Control, And Automation, 1st Edition., London Springer-Verlag, 2005, pp.547-600
18. D.N. Jeyakumar, T.Jayabharthi And T. Raghunathan, "Particle Swarm Optimization For Various Types Of Economic Dispatch Problems". Electrical Power And Energy System 28, Elsevier, 2006, pp. 36-42.
19. M.Gopal "Control System Principle and Design" 2006 2nd edition Tata Mc Graw Hill.
20. S. Mishra, C. N. Bhende & L. L. Lai "Optimization Of A Distribution Static Compensator By Bacterial Foraging Technique". IEEE Fifth International Conference On Machine Learning And Cybernetics, 2006 , pp 4075-4082.
21. M. Tripathy & S. Mishra "Bacteria Foraging-Based Solution To Optimize Both Real Power Loss And Voltage Stability Limit Ieee Transactions On Power Systems" Vol. 22, No. 1, Feb 2007, pp 240-249.
22. B.Yang, Y.Chen And Z.Zhao "A Hybrid Evolutionary Algorithm By Combination Of PSO And GA For Unconstrained And Constrained Optimization Problems", Proc Of IEEE Int. Conf. On Control And Automation, 2007, pp 166-170.

23. B .Y.Y. Chen And Z.Zhao, " Survey On Application Of Particle Swarm Optimization In Electric Power System ", IEEE Int. Conf On Control And Automation, 2007, pp 481-486.
24. S.F. Zheng " A Modified PSO Algorithm And Application". Proc Of The Sixth Intnl. Conf. On Machine Learning And Cybernetics, IEEE,2007, pp 945-951.
25. Yamile Del Valle and G.K. Venayagamoorthy . "PSO: Basic Concepts. Variants And Applications In Power System". IEEE Transaction On Evolutionary Computation. Vol.12 No-2, April 2008, pp. 171-195.
26. Mezura-Montes E. and Coello-Coello C.A"Constrained Optimization via Multiobjective Evolutionary Algorithms". Springer 2008, pp 53-75.
27. M.O'Neil And A.Brabazon. "Self Organizing Swarm(Soswarm) For Financial Credit-Risk Assessment". IEEE Congress On Evolutionary Comp. 2008, pp.3087-3093
28. Sambarta Dasgupta, Arijit Biswas, Ajith Abraham And Swagatam Das "Adaptive Computational Chemotaxis In Bacterial Foraging Algorithm" IEEE International Conference On Complex, Intelligent And Software Intensive Systems 2008.
29. P.S Bhimbra "Power Electronics", 2008 Khanna Publishers, pp 347-413
30. Z.Zhan, J.Zhang And H.S.Chung," Adaptive Particle Swarm Optimization", IEEE Transaction On Systems, Man, And Cybernetics-Part B,Decemner 2009, Vol.39, pp 1362-1381.
31. Li Zhi-Jie, Liu Xiang-Dong, Uuan Xiao-Dong And Wang Cun-Rui." An Inproved Particle Swarm Algorithm For Search Optimization". Global Congress On Intelligent Systems, IEEE, 2009, pp 154-158.
32. Janardan Nanda, S. Mishra & Lalit Chandra Saikia "Maiden Application Of Bacterial Foraging-Based Optimization Technique In Multiarea Automatic Generation Control". IEEE Transactions On Power Systems, Vol. 24, No. 2, 2009, pp 602-609
33. Vahid Rashtchi, Akbar Bayat And Resan Vahedi "Adaptive Step Length Bacterial Foraging Algorithm" .IEEE International Conference of Intelligent Computing And Intelligent System, 2009, pp 322 - 326

34. Abolfazl Jalilvand, Hesan Vahedi & Akbar Bayat "Optimal Tuning of the PID Controller for a Buck Converter Using Bacterial Foraging Algorithm".IEEE International Conference on Intelligent and Advanced Systems (ICIAS), June 2010, pp 1-5.

35. H. Zang, S.Zhang and K.Hapeshi, "A Review Of Nature Inspired Algorithms", Journal Of Bionic Engineering 7 Suppliment.' Sept. ,2010 pp 232-237

36. Liu Xiao Long, Li Rong Jun & Yang Ping" A Bacterial Foraging Global Optimization Algorithm Based On The Particle Swarm Optimization". IEEE International Conference on Intelligent Computing and Intelligent Systems, Oct. 2010 , pp 22 – 27.

37. Wael M. Korani, Hassen Taher Dorrah & Hassan M. Emara "Bacterial Foraging Oriented By Particle Swarm Optimization Strategy For PID Tuning". IEEE International Symposium on Computational Intelligence in Robotics and Automation (CIRA), 2009 IEEE ,Dec. 2009, pp 445-450.

38. Frank Neumann And Carsten Witt "Bioinspired Computation In Combinatorial Optimization – Algorithms And Their Computational Complexity". Published By Springer 2010.

39. Xin-She Yang, "Engineering Optimization", 2010, Wiley.J and Sons.

40. T.Y.Chen And T.M.Chi, " On The Improvement Of The Particle Swarm Optimization Algorithms". Advances In Engineering Software 41, 2010, pp 229-239

41. M.Senthil Kumar and Dr.P.Renuga "Bacterial Foraging Algorithm based Enhancement of Voltage Profile and Minimization of Losses Using Thyristor Controlled Series Capacitor (TCSC)". International Journal of Computer Applications Volume 7 No.2, 2010.

42. John Oyekan & Huosheng Hu "A Novel Bacterial Foraging Algorithm for Automated tuning of PID controllers of UAVs. IEEE International Conference on Information and Automation 2010, pp 693-698

43. H. Vahedi, S. H. Hosseini & R. Noroozian "Bacterial Foraging Algorithm for Security Constrained Optimal Power Flow". IEEE 2010, pp 1-7.

44. S. M. Tabatabaei, B. Vahidi, S. H. Hosseinian & S. M. Madani " Bacterial Foraging-Based Solution for Optimal Capacitor Allocation in Distribution Systems".IEEE International conference on power and energy 2010 pp 253-258.
45. K.M Pasino "Bacterial Foraging Optimization". International journal of swarm intelligence research, 2010.
46. Rong-Jong Wai, Jeng-Dao Lee And Kun-Lun Chuang, "Real-Time PID Control Stratergy For Maglev Transportation". Industrial Electronics, IEEE Transactions on Vol 58 Issue: 2), Feb 2011, pp 629 – 646.
47. Serkan Kiranyaz , Jenni Pulkkinen ,&Moncef Gabbouj "Multi-dimensional particle swarm optimization in dynamic environments". Elsevier International Journal Expert Systems with Applications Volume 38, Issue 3, March 2011, pp 2212–2223
48. Adel Boughelala Xiaoming Luan and Said Leghmizi "An Adaptive Channel Equalizer Using Bacterial Foraging Oriented By Particle Swarm Optimization Strategy" .IEEE 3rd International Conference on Computer Research and Development , March 2011 , pp 24 – 29.
49. Fateme Nournjad & Rasool Kazemzadeh "Modified Bacterial Foraging Algorithm Applied to Economic Dispatch Problem and Comparison with other Solutions". IEEE International Conference Electrical Engineering, May 2011 , pp 1-6.
50. Bharat Bhushan and Madhusudan Singh "Adaptive control of non linear system using Bacterial Foraging Algorithm" . International Journal of Computer and Electrical Engineering, Vol. 3, No. 3, June 2011, pp. 335-342.
51. Y.Zhao, Z.Zhang, Li Guo And B.Fang "Revisting The Swarm Evolution: A Long Term Perspective" IEEE Symposium On Computers And Communications, 2011, pp 1062-1067.
52. A.K Yadav , P.Kaur, A.P. Mittal and M.Anzar."Comparative Analysis Of Various Control Techniques For Inverted Pendulum .Indian International Conference On Power Elcetronics ,2011, pp.1-6
53. Parminder Singh, Navdeep Kaur and Loveleen Kaur "Satellite Image Classification by Hybridization of FPAB Algorithm and Bacterial Chemotaxis"

International Journal of Computer Technology and Electronics Engineering (IJCTEE) Vol 1, Issue 3 , 2011, pp.23-27.

54. Ismail Ahmed A. Al-Hadi, Siti Zaiton Mohd Hashim, Siti Mariyam & Hj Shamsuddin "Bacterial Foraging Optimization Algorithm For Neural Network Learning Enhancemen" 11th International Conference On Hybrid Intelligent Systems,2011.
55. Grant H. Kruger , Albert J. Shih , Danie G. Hattingh & Theo I. van Niekerk "Intelligent machine agent architecture for adaptive control optimization of manufacturing processes" Elsevier Advanced Engineering Informatics Journal, 2011, pp 783–796.
56. Navneet Kaur, Ramandeep Singh, Dr. R.K. Tulli "Bacteria Foraging Based Image Segmentation". International Journal of Engineering Sciences Vol. 6, July 2012, pp 141.
57. Yu-Sheng Lin, Yih-Guang Leu & Wei-Cheng Lu "Fuzzy Bacterial Foraging System And Its Applications In Control Of Servo Motors". International Conference On Fuzzy Theory And Its Applications ,2012.
58. Susmita Das, Ayan Chakraborty, Jayanta Kumar Ray, Soumyendu Bhattacharjee, & Dr. Biswarup Neogi "Study on Different Tuning Approach with Incorporation of Simulation Aspect for Z-N (Ziegler-Nichols) Rules" International Journal of Scientific and Research Publications, Vol 2, Issue 8, 2012.
59. Bharat Bhushan and Sarath S. Pillai "Particle Swarm Optimization and Firefly Algorithm: Performance Analysis". 3rd IEEE International Advance Computing Conference , Feb 2013, pp. 746-751.
60. K. Sundareswaran , Kuruvinashetti Kiran , Varsha Padhee , P Sankar , P. Srinivasa Rao Nayak & Abhilash " Buck-Boost Converter Controller Design Using Bacterial Foraging ". IEEE International Conference on Control Applications (CCA) Part of IEEE Multi-Conference on Systems and Control, March, 2013.
61. Xutao Li, Minjie Chen and Yoshihara Tsutomu "A Method of Searching PID Controller's Optimized Coefficients for Buck Converter Using Particle Swarm

Optimization".,IEEE 10th International Conference Power Electronics and Drive Systems, April 2013 , pp 238-243

62. Vladimir Sousa , Enrique C. Quispe, Julia R. Gomez Sarduy, "Bacterial Foraging Algorithm For Induction Motor Field Efficiency Estimation And Under Harmonics And Unbalanced Voltages" . IEEE International conference on Electric Machines & Drives Conference, May 2013 ,pp 1108 – 1111
63. Iztok Fister Jr and Xin-She Yang "A Brief Review of Nature-Inspired Algorithms for Optimization". IEEE June, 2013
64. Serdar Carbas1, Ibrahim Aydogdu2, Mehmet Polat Saka "A Comparative Study of Three Metaheuristics for Optimum Design of Engineering Structures ". 10th World Congress on Structural and Multidisciplinary Optimization, 2013,.
65. Guozhong Wu Application Of Adaptive PID Controller Based On Bacterial Foraging Optimization Algorithm IEEE 25th Chinese Control And Decision Conference ,2013, pp 2353-2356.
66. Megha Jaiswal & Mohna Phadnis "Speed Control of DC Motor Using Genetic Algorithm Based PID Controller". International Journal of Advanced Research in Computer Science and Software Engineering Vol 3, Issue 7, 2013.
67. G. Seshagiri Rao, S. Raghu, N. Rajasekaran "Design of Feedback Controller for Boost Converter Using Optimization Technique" . International Journal of Power Electronics and Drive System ,Vol. 3, Issue No. 1, 2013, pp 117-128.
68. Sarath Pillia "A Novel Approach To Adaptive Control" DTU M.tech (C&I) 2013.

APPENDIX I

ADAPTIVE CONTROL DESIGN

Class of plant used is represented by

$$y(k+d) = \alpha(x\,k) + ((x(k)\,u(k) \quad \text{(A.1)}$$

$\alpha(x(k))$ *and* $\beta(x(k))$ are unknown smooth functions of the state $x(k)$ while $y(k+d)$ is a nonlinear function of past values of $u(x(k))$ is requiring to be bounded away from zero. $d \geq 1$ is the delay between the input and output. For d=1

$$y(k+1) = \alpha(x(k)) + \beta(x(k))\,u(k) \quad \text{(A.2)}$$

$$y(k+d) = \alpha(x(k)) + \beta(x(k))\,u(k) \quad \text{(A.3)}$$

$$y(k+d) = \alpha_u(x(k)) + \alpha_k(k) + (\beta_u(x(k)) + \beta_k(x(k)))\mathrm{u}(k) \quad \text{(A.4)}$$

Functions $\alpha_u(x(k))$ and $\beta_u((k))$ represent the unknown nonlinear dynamics of the plant. It is these functions which require to be estimated so that a controller can be specified. $\alpha_k(x(k))$ and $\beta_k(x(k))$ are defined to be as known parts of the plant dynamics, these can be set to zero. $\beta(x(k))$ is assumed to satisfy $0 < \beta_0 < \beta(x(k))$ for some known $\beta_0 > 0$ for all $x(k)$ Estimation of an unknown ideal controller: An ideal Controller is given by

$$u^*(k) = \frac{-\alpha(x(k)) + r(k+d)}{\beta(x(k))} \quad \text{(A.5)}$$

This linearizes the dynamics of equation (A.3) such that $y(k) \rightarrow r(k)$. Substituting $u(k) = u*(k)$ in equation (A.3) we obtain $y(k+d) = r(k+d)$ so that tracking of reference input have been achieved within d steps. Since $\alpha\ (x\ (k))\ or\ \beta\ (x(\ k))$ are unknown, an estimator is developed for these plant nonlinearities and used them to form an approximation to $u*\ (k)$. Using a "Certainty equivalence controller", the control input can be defined as:

$$u(k) = \frac{-\tilde{\alpha}(x(k))+r(k+d)}{\tilde{\beta}(x(k))} \tag{A.6}$$

$\tilde{\alpha}\ (x(k\ ))$ and $\tilde{\beta}(x(k))$ are estimates of $\alpha\ (x\ (k))$ and $\beta(x(\ k)\ )$ respectively. Following Estimates

$$\alpha\ (x\ (k)) = F\ (\alpha\ (x\ (k)),\ \theta_{\alpha}\ (k)) \tag{A.7}$$

$$\beta\ (x\ (k)) = F\ (\beta\ (x\ (k)),\ \theta_{\beta}\ (k)) \tag{A.8}$$

Error $e\ (k) = r\ (k) - y\ (k)$ is not a linear function of the parameters. Also $e\ (k) = \tilde{y}\ (k) - y(k)$, where $\tilde{y}\ (k)$ is estimate of $y(k)$. A set of approximators for α and β where the i^{th} ones are denoted by: $F_{\ \alpha}\ (x,\theta^{i}_{\alpha})$ and $F_{\ \beta}\ (x,\theta^{i}_{\beta})$, for i=1,2,....,S. From a foraging perspective, θ^{i} is viewed as the location of the i^{th} foragers in the environment. In foraging method position of the forager θ^{i} is used to minimize the fitness function (θ^{i}) . Let the i^{th} estimate of the output and identification error be $\tilde{y}^{i}(k + 1\ )= F_{\alpha}\ ((\ x\ (k))\ ,\ \theta^{i}_{\alpha}(k)) + F_{\beta}\ ((x\ (k))\ ,\ \theta^{i}_{\beta}(k\ ))\ u(k)$ and $e^{i}\ (k\ )= \tilde{y}^{i}\ (k) - y(k)$ for i=1,2..., S. i^{th} Individual (bacteria) at time k can be given by

$$\theta^{i}_{k} = \left[\theta^{i^{T}}_{\alpha}(k), \theta^{i^{T}}_{\beta}(k)\right] \tag{A.9}$$

Fitness function can be defined as:

$$J\ (\theta^{\ i}(\mathrm{k}-\ 1)) = \mathrm{e}^{\ i}(\mathrm{k} = \left(\tilde{\mathrm{y}}^{\mathrm{i}}\ (\mathrm{k})\ -\ \mathrm{y}(\mathrm{k})\right)^{\ 2}$$

$$= \left(\mathrm{F}_{\alpha}\ \left((\ \mathrm{x}\ (\mathrm{k}-1)), \theta^{\mathrm{i}}_{\alpha}(\mathrm{k}-1)\right) +\ \mathrm{F}_{\beta}\left((\mathrm{x}\ (\mathrm{k}-1)), \theta^{\mathrm{i}}_{\beta}(\mathrm{k}-1\ )\right)\mathrm{u}(\mathrm{k}-1) -\ \mathrm{y}(\mathrm{k})\right)^{2} \tag{A.10}$$

which measures the size of the estimation error for the i^{th} estimate. It is required to minimize $\mathrm{J}\left(\theta^{\ i}(\mathrm{k}-\ 1)\right)$.

For BFOA, Forager's position in one dimension is given by θ_α the other dimension by θ_β so that forager's position is $\theta^i = [\theta^i_\alpha, \theta^i_\beta]^T$, i = 1,2. . , S. S is the population size of the bacteria. Foraging strategy is based on E.coli chemotaxis, but without swarming, elimination-dispersal, and reproduction. Here chemotactic hill-climbing strategy is used to adjust the parameters. At each time step, one foraging step is used, that means either one tumble-tumble step. Foraging occurs while the control system operates with searching for parameters occurring at each time step. For instance, if over one time step the cost did not decrease for an individual, then there is a tumble, and by this, a random direction is generated which update the parameters (location of the forager) in that direction. If, cost is improved from the last step, then another step in the same direction taken last time is made. In such case, forager is on a run in a good direction, down the cost function. Similarly on the basis of above analysis we can apply the process of adaptive control for other swarm or evolutionary based algorithm.

APPENDIX II

LIST OF PUBLICATION

“Bacterial Foraging Algorithm Based Adaptive Control Of Water Bath System**”.** 4th International Journal Of Information & Computation Technology, 2014. India.