ANN BASED GOVERNING AND CONTROLLING OF SHP

A DISSERTATION

Submitted in partial fulfillment of the requirements for the award of the degree

of

MASTER OF TECHNOLOGY

in

ALTERNATE HYDRO ENERGY SYSTEMS

By

CHALLA MOHANA KRISHNA

ALTERNATE HYDRO ENERGY CENTRE
INDIAN INSTITUTE OF TECHNOLOGY ROORKEE
ROORKEE-247 667 (INDIA)

JUNE, 2009

CANDIDATE'S DECLARATION

I hereby certify that the work which is being presented in this dissertation, entitled, **"ANN Based Governing and Controlling of SHP"**, in partial fulfillment of the requirements for the award of the degree of **Master of Technology** in **"Alternate Hydro Energy Systems"**, submitted to **Alternate Hydro Energy Centre, Indian Institute of Technology Roorkee** is an authentic record of my own work carried out from July 2008 to June 2009 under the supervision of **Shri.S.N.Singh**, Senior Scientific Officer and **Shri.M.K.Singhal,** Senior Scientific Officer, **Alternate Hydro Energy Centre, Indian Institute of Technology Roorkee, India.**

I have not submitted the matter embodied in this work for award of any other degree.

Date: June 29,2009

Place: Roorkee.

(CHALLA MOHANA KRISHNA)

CERTIFICATE

This is to certify that the above statement made by the candidate is correct to the best of my knowledge.

(S.N.SINGH)
Senior Scientific Officer,
Alternate Hydro Energy Centre,
Indian Institute of Technology,
Roorkee – 247 677.

(M.K.SINGHAL)
Senior Scientific Officer,
Alternate Hydro Energy Centre,
Indian Institute of Technology,
Roorkee – 247 677.

ACKNOWLEDGEMENT

It is my proud privilege to express my sincere gratitude to my supervisors **Shri.S.N.Singh**, Senior Scientific Officer and **Shri.M.K.Singhal**, Senior scientific Officer, **Alternate Hydro Energy Centre, Indian Institute of Technology Roorkee** for their kind cooperation, meticulous guidance & constant inspiration throughout the dissertation work.

I wish to express my profound gratitude to **Dr. Arun Kumar, Head, Alternate Hydro Energy Centre, Indian Institute of Technology Roorkee** for providing all the facilities, without which it would have been impossible for me to complete this dissertation. The cooperation he gave is invaluable.

I am also grateful to all the faculty members and staff of **Alternate Hydro Energy Centre, Indian Institute of Technology Roorkee** for their kind support.

I would like to express my humble respect and special thanks to my parents, my sisters, my friends & others who directly or indirectly helped me during the completion of this work.

Last but not the least, I express my sincere gratitude to **God** for enabling me to achieve success in completion of this work.

Dated: June 2009

Ch. mohana krishna
(CHALLA MOHANA KRISHNA)

ABSTRACT

The basic function of a governor in hydro power plant is to control speed and /or load. Normally flow control based governing is used and many of the governors used are conventional governors. They suffer from mechanical wear due to their aging effects and the operation based on the droop characteristics.

They have to be replaced with the recent advanced governors in order to improve the performance. One of the replacement alternatives would be PID (Proportional Integral Derivative). Further, to improve performance of PID governor Artificial Intelligence controllers can be used.

In this dissertation, Hydro power plant simulation has been carried out to design a ANN governor and various components of hydro power plants like Penstock, turbine, Governor and Exciter. In the design of ANN based governor, primarily a normal hydraulic governor is designed and in order to improve the performance results PID based governor is developed.

The values of the PID are different for different sites. Generally these are carried out through the manual tuning. In this work, the PID values of governor are optimized through Genetic algorithm in GATOOL Box in MATLAB 7.0. The response of PID governor had been taken to train the Artificial Neural network and after creating ANN network it is replaced with PID governor and the neural network based governor is independent of all the settings.

CONTENTS

LIST OF FIGURES

List of Tables

CHAPTER-1

INTRODUCTION AND LITERATURE REVIEW

1.1 GENERAL

Water resource in India is one of the major energy resources and in fact is a gift of the nature. This is formed by the snowcapped mountains, glaciers and regular monsoons. Energy is one of the key factors that influence the development of a nation providing economic and social benefits to its people. This is more important in developing countries like India, for economic development at micro level necessitating the need for the availability of secure and sustainable energy. The different sources of energy are hydro, thermal, nuclear and non-conventional energy resources like wind, solar and biomass. Presently in India the total installed capacity including all the resources is 1,27,056 MW, of this thermal is 66%, hydro 26%, nuclear 3% and renewable 5%[1], whereas an ideal hydrothermal mix should be 40:60 for grid stability. It is therefore, necessary to correct the hydro-thermal mix to meet the grid requirements and peak power shortage. Among the different sources of energy, hydropower is recognized as a renewable source of energy, which is economical, non-polluting and environment friendly. It is perhaps the oldest renewable energy technique known to the mankind for mechanical energy conversion as well as electricity generation.

Hydropower represents use of water resources towards inflation free energy due to absence of fuel cost with mature technology characterized by highest prime moving efficiency and spectacular operational flexibility. Hydropower contributes around 22% of the World electricity supply generated from about 7, 50,000 MW of installed capacity and in many countries; it is the main source of power generation e.g. Norway-99%, Brazil-86%, Switzerland-76% and Sweden-50%. Despite hydroelectric projects being recognized as the most economic and preferred source of electricity, the share of hydropower in India has been declining since 1963. The hydro share declined from 50% in 1963 to about 26% in 2005. The Government of India has announced, in August, 1998 Policy on Hydro Power Development, followed by 50,000 MW hydro-electric initiatives in May, 2003. About 70% of the population in India lives in rural areas. The rural energy scenario is characterized by inadequate, poor and unreliable supply of energy services. Realizing the fact that small hydropower projects can provide a solution for the energy problem in rural, remote and hilly areas where extension of grid system is comparatively uneconomical and also along the canal systems having sufficient drops, promoting small and mini Hydro projects is one of the objectives of the Policy on Hydro Power Development in India.

India is blessed with immense amount of hydroelectric potential and only 17% of the hydro potential has been harnessed till now [1]. Hence, a lot of importance is being given to develop the hydropower potential. Ministry of Power has been entrusted to develop large hydropower resources and Ministry of Non-conventional Energy Resources has been promoting small and mini hydro projects (≤ 25MW) [4] so as to provide energy to remote and hilly areas. Energy is the basic requirement for economic development. Every sector of Indian economy - industry, agricultural, transport, commercial and domestic - needs input of energy. According to International Energy Agency (IEA), a threefold rise in India's generation capacity is expected by 2020. As the non-renewable fossil energy sources continues to deplete, and realizing the summits held at Brazil and Kyoto, to reduce the greenhouse gas emissions, hydro power has moved towards the top power development option to meet the increasing energy demand. The conventional large hydropower plants have problems like long gestation period, ecological changes, loss due to long transmission lines, submergence of valuable forest and underground mineral resources. They also require rehabilitation of large population from area to be submerged. Due to all these factors large hydropower plants are unfavorable. On the other hand, Small Hydropower (SHP) projects are free from these aspects. Such installations are environment friendly because they causes negligible or no submergence, minimal deforestation and minimal impact on flora, fauna and biodiversity. Today, the market for small and medium sized hydroelectric power plant is more attractive than ever. During the last decade China has installed 10,000 MW Small Hydropower Plants, unfortunately this is not the situation in other countries. In India, the energy scenario in rural, remote and hilly area is characterized by inadequate, poor and unreliable supply of energy services. In such areas, the load density is low and extension of grid system is totally uneconomical, hence the Small Hydropower schemes can provide a solution for the energy problems besides solar photovoltaic, which is not available throughout the day. Fortunately, India is blessed with many rivers, natural streams, canal networks and mountains offering tremendous potential of Small Hydropower. In India, the first hydro power plant of 130 kW capacity was installed in 1897 at Sidrapong (Darjeeling) which was Run-off-river type. The worldwide contribution of Small Hydropower (SHP) has grown substantially in the last ten years. Among all the renewable energy sources available, Small Hydropower is considered as the most promising source of energy.

Table 1.1 Hydro Scenario in India [1]

Basin/Rivers	**Probable Installed Capacity(MW)**
Indus basin	33,832
Gangs basin	20,711
Central Indian river system	4,152
Western flowing rivers of southern India	9,430
Eastern flowing rivers of southern India	14,511
Brahmaputra basin	60,065
Total	**1,48,701**

In addition, 56 number of pumped storage projects have also been identified with probable installed capacity of 94,000 MW, in addition to this, hydro-potential from small, mini& micro schemes has been estimated as 15,000 MW. Thus in totality India is endowed with hydro-potential of about 2,50,000 MW. Hydro is many times a cheaper option for the country compared to thermal power due to the following reasons:

1. Life of hydro plants is 60 years minimum, against 30 years that of thermal plants.
2. Against zero cost input in case of hydro, constant escalation in cost of coal makes the operational cost of Thermal ever increasing.
3. Load carrying capacity of the grid and hence its economy improves with the peaking partnership of Hydro, reducing backing down of thermal plants and therefore increasing their PLF and efficiency.

Capital investment of hydro if half that of thermal, considering the infrastructural costs and auxiliary factors in India, average cost of hydro power generation is one 3rd to one 4th of thermal.

1.2 SMALL HYDROPOWER

There is a general tendency all over the world to define Small Hydropower by the power output. Different countries follow different norms, the upper limit ranges between 5 to 50 MW, as given in the Table 1.2.

Table 1.2 Worldwide Definitions of SHP [2]

COUNTRY	CAPACITY (MW)
UK	≤ 5
UNIDO	≤ 10
Sweden	≤ 15
Colombia	≤ 20
Australia	≤ 20
India	**≤ 25**
China	≤ 25
Philippines	≤ 50
New Zealand	≤ 50

The present status of SHP is given in Table 1.3. In India, SHP schemes are classified by the Central Electricity Authority (CEA) as given in the Table 1.4. Power stations are also classified based on the head available and is given in Table 1.5.

Table 1.3 SHP Status in India [3]

Status	Potential
Overall Potential	15,000 MW
Identified Potential	12,841.81 MW (4,861 Sites)
Installed Capacity	2,013.17 MW (581 Projects)
Under Construction	561 MW (207 Projects)
Capacity Addition During 2002-2007	500 MW
Target Capacity Addition -11th plan (2007-2012)	1,400 MW

Table 1.4 Classifications of SHP Schemes in India [2]

Type	Station Capacity	Unit Rating
Micro	Up to 100kW	Up to 100kW
Mini	101 to 2,000 kW	101 to 1,000 kW
Small	2,001 to 25,000 kW	1,001 to 5,000 kW

Table 1.5 Classification based on Head [2]

Type	Range of Head
Ultra Low Head	Below 3 m
Low Head	3 to 40 m
Medium/High Head	Above 40 m

With the advancement of technology, and increasing requirement of electricity, the thrust of electricity generation was shifted to large size hydro and thermal power stations. However, during the last 10-15 years there is a renewed interest in the development of small hydro power projects due to its benefits particularly concerning environment and ability to produce power in remote areas. Small hydro projects are economically viable and have relatively short gestation period. The major constraints associated with large hydro projects are usually not encountered in small hydro projects. The World estimated potential of small hydro is of around 1,80,000 MW. India has as an estimated potential of about 15,000 MW with perennial flow rivers, streams and a large irrigation canal network with dams & barrages. Of this 4,861 potential sites with an aggregate capacity of 12,841.81 MW have been identified.

1.3 SMALL HYDRO VIS-A-VIS OTHER RENEWABLE ENERGY RESOURCES

Now let us examine superiority of small hydro even amongst renewable

(1) It is the highest-density renewable energy source against widely spread and thinly distributed solar energy, biomass, wind resource, etc.

(2) Its cost of generation is cheapest amongst renewable.

Table 1.6 Cost of Generation of Renewable [5]

SECTOR	Rs/kWh
Small Hydro	1.00-1.50
Cogeneration	1.25-1.50
Biomass	1.75-2.00
Wind	2.00-2.75
Solar PV	10.00-12.00

(3) Small hydro efficiencies are highest amongst renewable.

Table 1.7 Efficiency of Different Renewable [5]

SECTOR	EFFICIENCY (%)
Small Hydro	85-90
Cogeneration	60
Biomass	35
Wind	40
Solar PV	15

(4) Small hydro has longest project life.

Table 1.8 Project Life of Different Renewables [5]

SECTOR	LIFE (YEARS)
Small Hydro	60
Cogeneration	30
Biomass	30
Wind	20
Solar PV	20

(5) Small hydro resource is more consistent compared to other renewable like wind, solar, biomass with regard to its availability for power generation. Relatively better capacity factors can be expected. Even run-of-the-river schemes can have small pond age to meet the daily peak requirements of power.

1.4 BASIC COMPONENTS OF SHP

The various components of SHP can be categorized in two parts.

(i) Civil works components

(ii) Electro- mechanical equipments

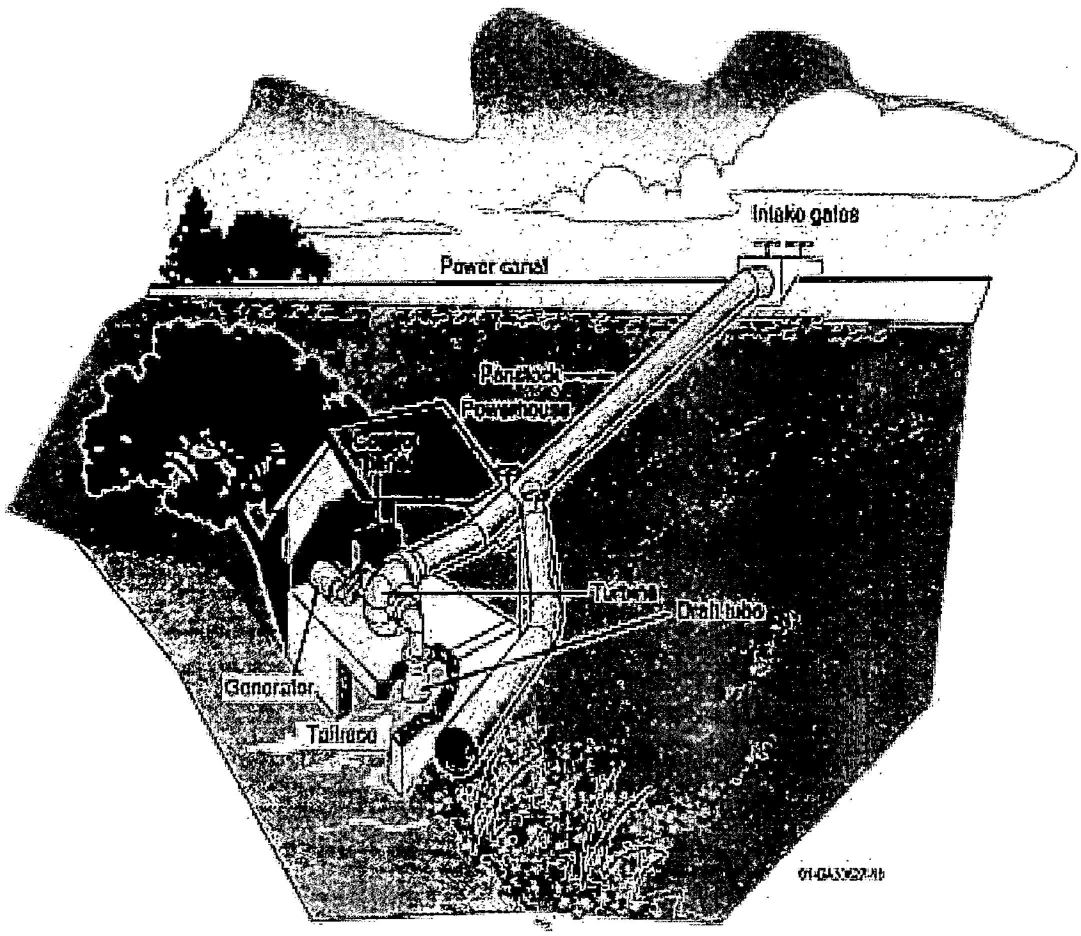

Figure 1.1 Basic Components of SHP [6]

1.4.1 Civil Works Components

The components which are contact with water and do not have any rotating parts are called as civil works components, examples: Intake weir, Desilting tank, Forebay, Power Channel, Penstock, Power House and Tailrace etc. The purpose of civil work components is to divert the water from stream and convey towards the power house. In selecting the layout and types of civil components, due consideration should be given to the requirement for reliability.

1.4.2 Electro Mechanical Equipments

Electro-Mechanical equipments mainly include hydro turbine, generator, speed increaser, governor, gates and valves and other auxiliaries. The parts which are contact with water and having rotating parts are called mechanical equipment. The parts which are not contact with water and having rotating parts are called as electrical equipment as a rule of thumb.

1.4.2.1 HYDRAULIC TURBINES

The hydraulic turbine transforms the potential energy of water into mechanical energy in the form of rotation of shaft. It can be broadly classified into two categories according to action of water on moving blades.

(i) Impulse Turbines

In case of impulse turbines the penstock is connected with the nozzle and hence the whole pressure energy of water is transformed into kinetic energy in nozzle only. The water coming out of the nozzle is in the form of a free jet, which strikes with a series of buckets mounted on the periphery of the runner. The water comes in contact with only few of the buckets at a time. Once the water comes out of the nozzle then the pressure is atmospheric throughout, hence in case of impulse turbine the casing do not have any hydraulic function to perform but it is necessary only to prevent splashing and to lead the water to the tail race, and also act as a safeguard against accidents. Examples of impulse turbines are Pelton turbine, Turgo- Impulse turbine, Cross flow turbine.

(ii) Reaction Turbines

The water pressure can apply a force on the face of the runner blades, which decreases as it proceeds through the turbine. Turbines that operate in this way are called reaction turbines. It operates with its runner submerged in water. The water before entering the turbine has pressure as well as kinetic energy. All pressure energy is not transformed into kinetic energy as in case of impulse turbine. The moment on the runner is produced by both kinetic and pressure energies. The water leaving the turbine has still some of the pressure as well as the kinetic energy. The pressure at the inlet to the turbine is much higher than the pressure at the outlet. Thus, there is a possibility of water flowing through some passage other than the runner and escape without doing any work. Hence a casing is absolutely essential due to the difference of pressure in reaction turbine. The reaction turbines can be further classified into mixed and axial flow turbines. Mixed flow turbine water enters from outer periphery of the runner, moves inwards in radial direction and comes out from center in axial direction. Example of mixed flow turbine is Francis turbine. Axial flow turbines water enters from the wicket gates to the runner in the axial direction, moves along the axial direction and comes out in axial direction. Examples of axial flow turbines are: Propeller turbine, Kaplan turbine, Bulb turbine, Star flow turbine.

1.4.2.2 Generator

Generator transforms mechanical energy into electrical energy. Nowadays, only 3-phase A.C. (alternating current) generators are used in normal practice. There are basically two types of generators: namely synchronous generator and Induction generator. The Induction generators are also called as Asynchronous generator based on the super synchronous speed at which they operate.

1.5 ADVANTAGES OF SMALL HYDROPOWER

Amongst renewable energy sources, hydroelectric power seems to be the most desirable for utilities and its economic feasibility has been successfully proven. The advantages of small hydro are;

(a) A fast way to increase rural electrification, improved living standards and simulation of rural industries.

(b) Flexibility of installation and operation in an isolated mode and also in a localized or regional grid system.

(c) Relatively small investments required as compared to large hydro.

(d) Low operational cost with cheap and simple maintenance.

(e) Standard indigenous technologies and maintenance base available which require only minor adaptation to specific site condition.

(f) Compatible with use of water for other purposes such as irrigation, drinking etc.

(g) Long life of 60 and above years.

(h) Perennial source of income generation.

(i) Environment friendly due to negligible or no submergence, least deforestation.

(j) Hydro power involves a clean process of power generation.

(k) It is a renewable source of energy and contributes to the upliftment of the rural masses, especially projects located in remote and inaccessible areas.

(l) It is the most cost effective option for power supply because it does not suffer from the limitation on account of fuel consumption.

(m) Most small hydro projects in Himalayan region are being developed in remote and backward areas where substantial support for economic development is actually needed.

(n) Small hydro power contributes in solving the low voltage problem in the remote hilly areas and helping reducing the losses in transmission and distribution.

(o) In certain cases projects are helpful in providing drinking water and irrigation facilities.

(p) It helps in promoting the local industries in remote areas.

(q) The development of small hydro projects requires minimum rehabilitation and resettlement as well as environmental problems.

(r) Small hydro projects help in generating self employment in remote areas of the state.

(s) Small hydro power projects help in providing stable electricity supply at remote areas where such facility by other source shall be much costlier and unreliable.

1.6. TYPES OF SHP SCHEMES

Small Hydropower can also be broadly categorized in three types as follows:

1.6.1 Run-Off River Scheme

Run-of-River hydroelectric schemes are those, in which water is diverted towards power house, as it comes in the stream. Practically, water is not stored during flood periods as well as during low electricity demand periods, hence water is wasted. Seasonal changes in river flow and weather conditions affect the plant's output. After power generation water is again discharged back to the stream. Generally, these are high head and low discharge schemes. The typical run-off river scheme is shown in Fig. 1.2.

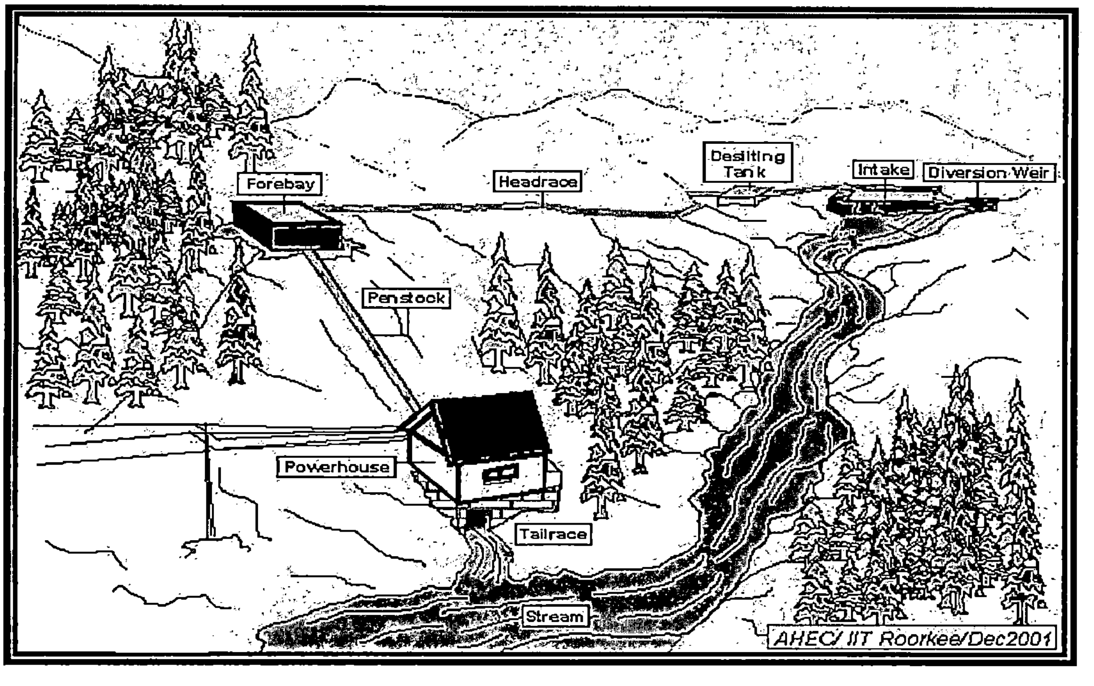

Fig. 1.2 Typical arrangement of run-off river scheme [3]

1.6.2 Canal Based Scheme

Canal based small hydropower scheme is planned to generate power by utilizing the fall in the canal. These schemes may be planned in the canal itself or in the bye pass channel. These are low head and high discharge schemes. These schemes are associated with advantages such as low gestation period, simple layout, no submergence and rehabilitation problems and practically no environmental problems. The typical canal based scheme is shown in Fig. 1.3.

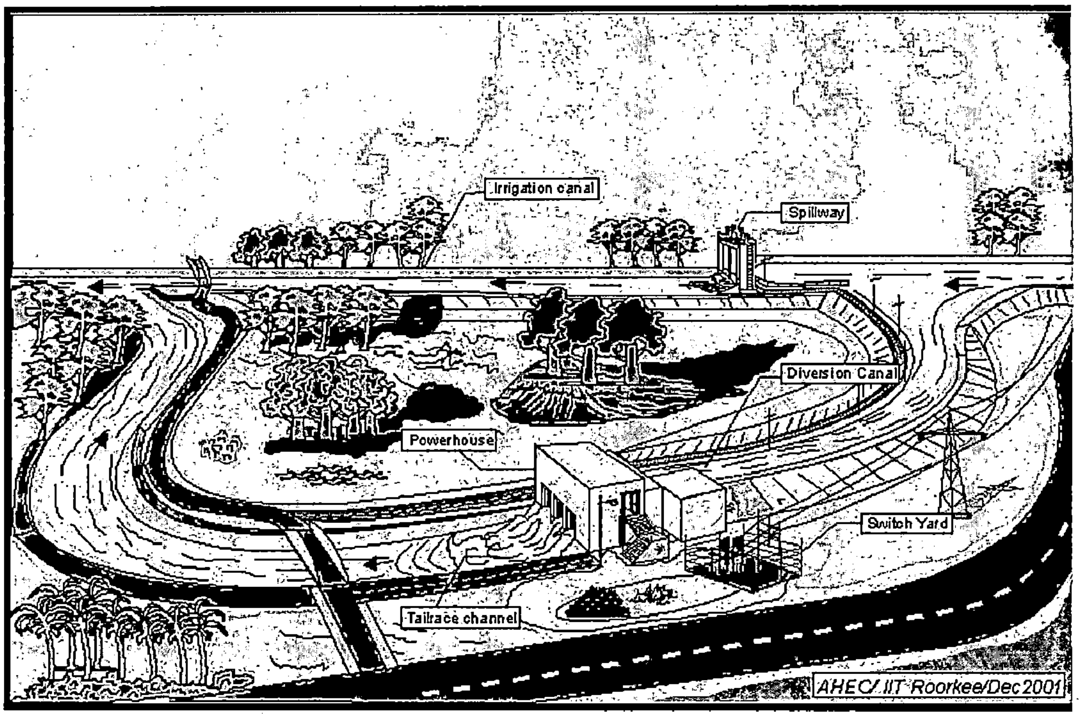

Fig. 1.3 Typical arrangement of canal based scheme [3]

1.6.3 Dam Toe Based Scheme

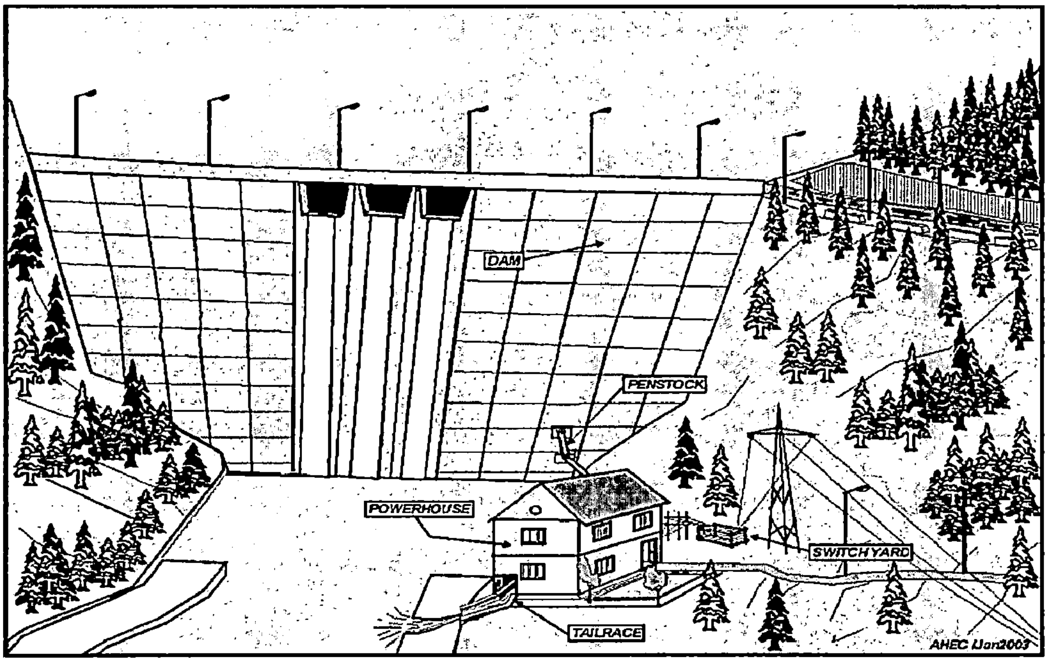

Fig. 1.4 Typical arrangement of dam toe based scheme [3]

In this case, head is created by raising the water level behind the dam by storing natural flow and the power house is placed at the toe of the dam or along the axis of the dam on either sides. The water is carried to the powerhouse through penstock. Such schemes utilize the head created by the dam and the natural drop in the valley. Typical dam toe based scheme is shown in the Fig. 1.4

1.7. CONTROLLING OF HYDRO POWER PLANT

The generated output power of the hydro generator can be controlled by two ways:

(a) Flow control

(b) Load control

Load control is used normally in isolated power plants.

The control is used to set the frequency of the system. Mostly flow control can be used. This flow control can be done by the governor. The governors control the speed of prime movers. In hydroelectric power plant the prime mover is hydraulic turbine.

Many of the governors used are conventional governors. They suffer from mechanical wear due to their aging effects. They have to be replaced with the recent advanced governors. One of the replacement choices would be PID (Proportional Integral Derivative).The PID controllers are best suited in the SISO (Single Input and Single Output) systems. They are unstable for complex systems and lack ability to change over time.

Intelligent controllers will be replacement for PID. Intelligent controllers include ANN (Artificial Neural Networks) and fuzzy control systems.

The value of the PID is site specific and is different for different hydro power plants. In the present work, these values can be tuned using the Genetic algorithm and to make it independent of PID values, ANN governing can be done.

1.8 LITERATURE REVIEW

There are a lot of literatures available on neural as well as hydropower plant modelling. The literatures required for in modeling and simulation of hydro power plants and about Genetic Algorithm to set the values of PID. Application of neural networks to tune the developed model and in the preparation making report along with the features incorporated is discussed as below:

The literature used for modeling purpose is as follows: -

For modeling of different components of hydroelectric power plant the book by Kundur P., [8] has proved been covered the details about creating the mathematical model of Hydropower plant. It also describes the modelling types of synchronous generator, and also explains different types of excitation systems as recommended by IEEE.

In paper [9], C. K. Sanathanan has demonstrated an accurate low order model for hydraulic turbine and penstock.

In the Book [10] by K. R. Padiyar analysis electric system has been taken on excitation systems and generator.

Paper [11] by Jin Jian, gives the detail about different type of hydraulic governors and their historical development. While the model of the PID governor has been taken from paper [17], i.e., IEEE Transaction on Power Systems, this paper discuss the details about different types of governors employed in hydro turbine governing and also the tuning of PID governor.

Paper [18], i.e.; IEEE STD 421.5-1992, deals with the need and the requirement of exact modeling of excitation system for synchronous generator.

The details of books and paper used for ANN and GA are:

The details of artificial neurons have been taken from the book [14] by J. M. Zurada, the back propagation training algorithm different types of activation functions, types of learning and learning factors and their effects are also taken from the same.

The concept of usage of back propagation for PID networks has been taken from Introduction to Neural Networks Using MATLAB 6.0 [16].It has been used as quick reference and to abridge theoretical and in practical implementation.

The paper [15] Martin T. Hagan and Mohammad B. Menhaj gives the details about the Back propagation algorithm for training of ANN; it also covers the historical development of the Back propagation algorithm, its benefits and limitations. The paper, Dionisio S. Pereira gives the detailed concept of Genetic Algorithm to tune the PID values.

1.9 Organization Of Report

Chapter 1 describes;

(i) Small hydro power, various schemes of small hydro and its components.
(ii) Introduction on the various Governing systems.
(iii) Literature survey on Modelling of small hydro power plant, ANN and GA.

Chapter 2 describes the modelling of various components of hydro power plant in the equations and in block diagrams.

Chapter 3 gives detailed description on ANN and various learning algorithms and about the GA tool box application to PID systems.

Chapter 4 explains on the simulation carried out for various blocks developed in SIMULINK

Chapter 5 is about the data used in simulation and the results and discussion on the results.

Chapter 6 includes the conclusion and scope of the future work.

CHAPTER 2

MODELLING OF VARIOUS COMPONENTS OF SMALL HYDRO-ELECTRIC POWER PLANT

2.1 MODELLING OF SHP STATION

The modelling of various components of hydro electric plant is required and equations of various components as below.

(a)Hydraulic Turbine Modelling

(b) Modelling of Governor

(c)Modelling of Excitation System

(d) Modelling of Synchronous Generator

2.1.1 Hydraulic Turbine Modelling

The hydraulic turbine and water column in transient performance study is usually based on the certain assumptions:

1. The hydraulic resistance is neglected.
2. The penstock pipe is considered as inelastic and water as incompressible.
3. The velocity of water in penstock varies directly with gate opening.
4. The turbine output power is proportional to the product of head and velocity of flow.

A schematic diagram of hydroelectric plant is shown in Figure 2.1

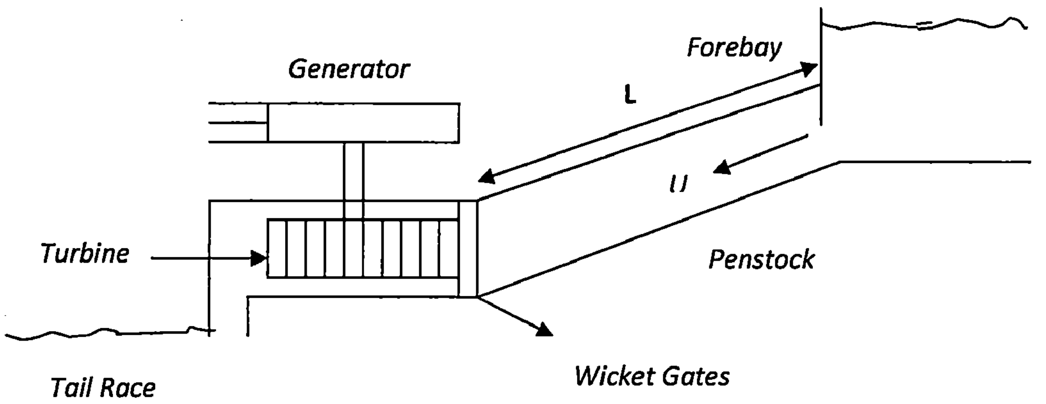

Fig. 2.1 Schematic Diagram of Small Hydro Power Station

The characteristics of the turbine and penstock are determined from Velocity of water in penstock, Turbine mechanical power, Acceleration of water.

With the Linear models, it is not possible to represent a practical system. since they are unable to study large fluctuations of power output and frequency. The non-linear model is more appropriate for large signal time domain simulations.

$$U = k_u G\sqrt{H} \tag{2.1}$$

$$P = K_p HU \tag{2.2}$$

$$\frac{dU}{dt} = -\frac{a_g}{L}(H - H_o) \tag{2.3}$$

$$Q = AU \tag{2.4}$$

Where U= water Velocity ; G=Ideal gate Opening; H=Hydraulic Head at the gate

H_0= Initial Steady state value of head; P=Turbine Power Output; Q=Water Flow Rate

A=Pipe Area; a_g= Acceleration due to gravity; t=time in seconds

To convert into per unit values normalize the above equations with Rated values.

$$\frac{U}{U_r}=\frac{G}{G_r}\sqrt{\frac{H}{H_r}} \tag{2.5}$$

$$\frac{P}{P_r}=\frac{U}{U_r}.\frac{H}{H_r} \tag{2.6}$$

Where subscript 'r' denotes rated values.

$$\overline{U}=\overline{G}\sqrt{\overline{H}} \text{ or}$$

$$\overline{H}=\left(\frac{\overline{U}}{\overline{G}}\right)^2 \tag{2.7}$$

The per unit form of the equation (2.7) can be rewritten by dividing it throughout by H_r U_r, as below.

$$\frac{d}{dt}\left(\frac{U}{Ur}\right)=\frac{a_g}{L}.\frac{Hr}{Ur}\left(\frac{H}{Hr}-\frac{Ho}{Hr}\right) \tag{2.8}$$

i.e. $$\frac{d\overline{U}}{dt}=\frac{1}{T_w}(\overline{H}-\overline{H}_o) \tag{2.9}$$

Where $T_w=\frac{LU_r}{a_gH_r}=\frac{LQ_r}{a_gH_r}$ (Water starting time at rated load.)

The equation in the laplace form can be,

$$\frac{\overline{U}}{\overline{H}-\overline{Ho}}=\frac{-1}{T_ws} \tag{2.10}$$

The mechanical power output is given by

$$P_m = P - P_L$$

The loss in the turbine system can be calculated by,

$$P_L = U_{NL} H$$

Where, U_{NL} represent the no-load speed.

The above equation gives per unit power output on a base equal to turbine MW rating and has to be converted into MVA in P.U for system studies.

$$\overline{T}_m = \frac{\omega_o}{\omega}.\overline{P}_m(\frac{P_r}{MVA_{base}}) \qquad (2.11)$$

Where ω per unit speed and MVA_{base}, is the MVA on which turbine torque is to be made per unit.

$$\overline{Tm} = \frac{1}{\overline{\omega}}(\overline{U} - \overline{U}_{NL})\overline{H}\,\overline{P}_r \qquad (2.12)$$

$$\text{Pr} = \frac{TurbineMWrating}{MVA_{base}} \qquad (2.13)$$

In the equation, $\overline{G}$ is the ideal gate opening based on the change from no load to full load.

Ideal gate opening is related to real gate opening as follows

$$\overline{G} = A_t\,\overline{g}$$ Where A_t is turbine gain given by

$$A_t = \frac{1}{\overline{g}_{FL} - \overline{g}_{NL}} \qquad (2.14)$$

Equations (2.14), (2.8), (2.10) and (2.12) and (2.13) completely describes the water column and turbine characteristics. This may also be shown in block diagram as shown in Figure 2.2.

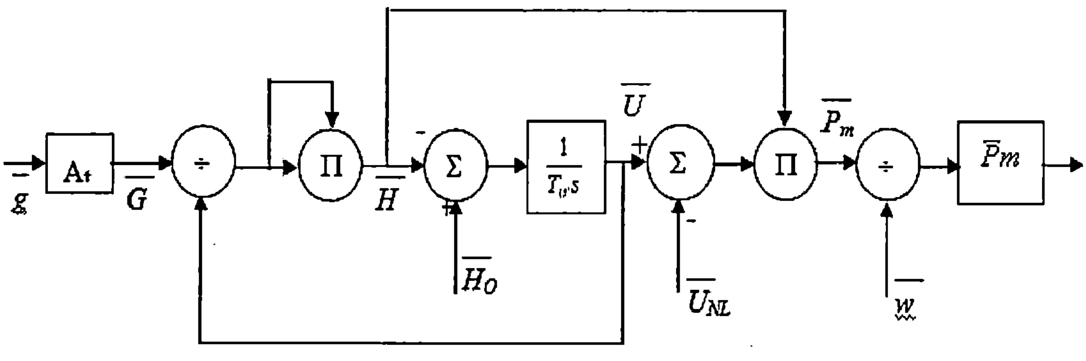

Figure 2.2: Block diagram of turbine penstock model

The difference between the mechanical torque and electrical torque will give acceleration torque of a electrical machine.

$$T_a = T_m - T_e \tag{2.15}$$

Where T_a= accelerating torque N-m; T_m =mechanical torque N-m and

Te =electromagnetic torque in N-m.

In above T_m and T_e are positive for generator and negative for motor. The combined inertia of rotor and generator is accelerated by unbalance in the mechanical and electrical torque. Thus equation (2.15) can be written as

$$J \frac{d\omega_m}{dt} = T_a = T_m - T_e \tag{2.16}$$

Where J = Combined inertia of generator and turbine kg-m^2

ω_m =Angular velocity of rotor mech rad/sec; t = time in seconds

If the synchronous generator which is connected to turbine rotating at angular velocity ω_r rad/sec and δ is the angular position of rotor in radians calculated with respect to reference position. The per unit form of the equation can be written as

$$\frac{d\Delta\overline{\omega}_r}{dt} = \frac{1}{2H}(\overline{T}_M - \overline{T}_e - K_D\Delta\overline{\omega}_r) \qquad (2.17)$$

Where K_D is damping coefficient, H is the normalized inertia constant represents K.E stored in the rotor. The mechanical starting time $T_M = 2H$ sec, is defined as time required for rated torque to accelerate the rotor from stand still to rated speed. The change in angular position of rotor is given by

$$\frac{d\delta}{dt} = \omega_o\Delta\overline{\omega}_r \qquad (2.18)$$

Where $\Delta\overline{\omega}_r$ is the per unit angular velocity of rotor in electrical rad/sec.

2.1.2 Modelling of Governor

The basic function of a governor is to control speed and /or load. The primary speed/load control function involves feeding back speed error to control the gate position. Droop characteristics can be helpful for parallel operation

Modern speed governors for hydraulic turbines use electro-hydraulic systems. Functionally their operation is very similar to that of mechanical-hydraulic governors. Speed sensing, permanent droop, temporary droop, and other measuring and computing functions are performed electrically. The electrical components gives improved performance compared to mechanical regard to Dead bands and time lags. The dynamic characteristics adjusted to me chanical hydraulic governor [8].

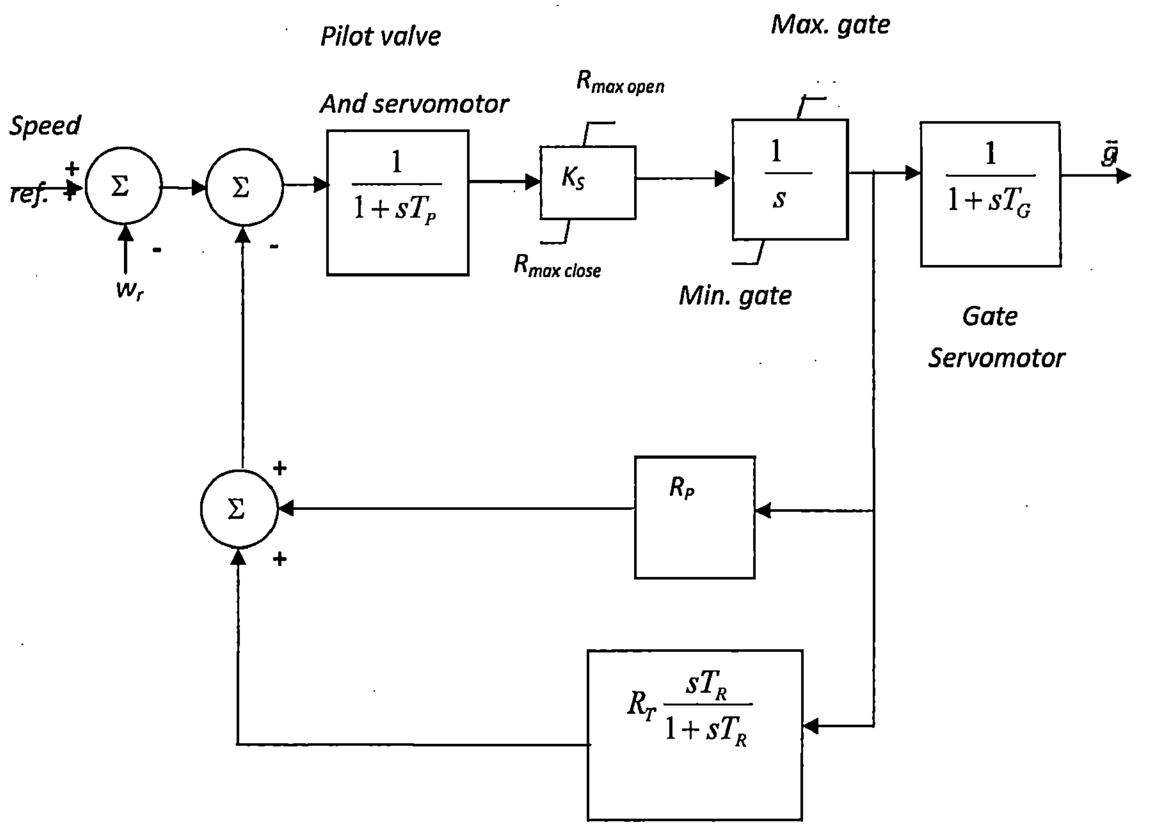

Figure 2.3 Model of Hydraulic Governor for Hydraulic Turbines

ω_r = rotor speed; T_P = pilot valve and servomotor time constant; K_S = servo gain;

T_G = main servo motor time constant of hydraulic governor; R_P = permanent droop;

R_T = temporary droop of hydraulic governor; T_R = reset time;

$R_{max\ open}$ = maximum gate opening rate limit; $R_{max\ close}$ = minimum gate opening rate limit,

2.1.2.1 Tuning Of Speed –Governing Systems

There are two important considerations in the selection of governor settings:

(1) Stable operation during when operating in isolated mode and

(2) Acceptable speed response when operating in variable load under synchrous operation.

For stable operation under islanding conditions, the optimum choice of the temporary droop R_T and reset time T_R is related to water starting time constant T_W and mechanical starting time T_M as follows:

$$R_T = [2.3 - (T_W - 1.0)0.15]\frac{T_W}{T_M} \qquad (2.19)$$

$$T_R = [5.0 - (T_W - 1.0)0.5]T_W \qquad (2.20)$$

The value of Ks servo system gain should be kept as maximum as possible such that stable operation can be maintained in isolated load.

For loading and unloading during normal interconnected system operation, the above setting results in too slow response. For satisfactory loading rates, the reset time T_R should be less than 1.0s, preferably close to 0.5s. [8]

2.1.2.2 TERMS RELATED WITH GOVERNING

Droop

Droop in a governor refers to the amount speed drops as load is applied. If we set a governor for 50 Hz no load, and it runs at 48 Hz full load, we would have 4% droop. Droop is used to stabilize outputs in a system of several units operating in parallel, and also to limit load upon some unit.

Water starting time

It is the time in seconds required to bring water in the penstock from zero velocity to full load velocity through the turbine, if the gates or valves were to open simultaneously.

Mechanical starting time

It is the time in seconds required for full turbine torque to accelerate the turbine-generator-flywheel unit from zero speed to full operating speed.

Gate timing

It is the minimum time in seconds turbine gates or valve may be completely opened or closed, and must be set by the hydraulic designer to keep water hammer pressure or vacuum within the penstock design limits.

Effective governor time

It is the length of time in seconds required by the governor to completely open or close the gates or valves on a turbine, and is generally several seconds longer than the gate timing.

2.1.2.3 PID Governor

Because of the important role that governor play in control of prime-movers of electrical power system, a great deal of research work has been made to investigate the effects of governor parameter settings on overall performance, mainly stability and transient behaviors for major load disturbance.

The evolution of governors for hydraulic turbine generating units certainly reflects the technology advance. The most commonly used governors before the 60's were mechanical hydraulic type with fly ball for speed sensing and dashpots to provide temporary droop characteristics. In the early 70's, electrical hydraulic type of governor employing PID (Proportional, Integral and Derivative) controllers became popular. [5]

A model of PID Governor is shown in Figure 2.4 [10], without derivative action, it is equivalent to hydraulic governor. The proportional and integral gains can be adjusted to obtain desired temporary droop and reset time. The derivative action is beneficial for isolated operation. There are cases when specific governors require more complex representation than shown in Figure 2.4. The difference may be due to added time constants in hardware and also where derivative action is included. In case of PID Governor we do not have the liberty to neglect the impact of traveling wave whereas in case of PI Governor we have got this liberty.

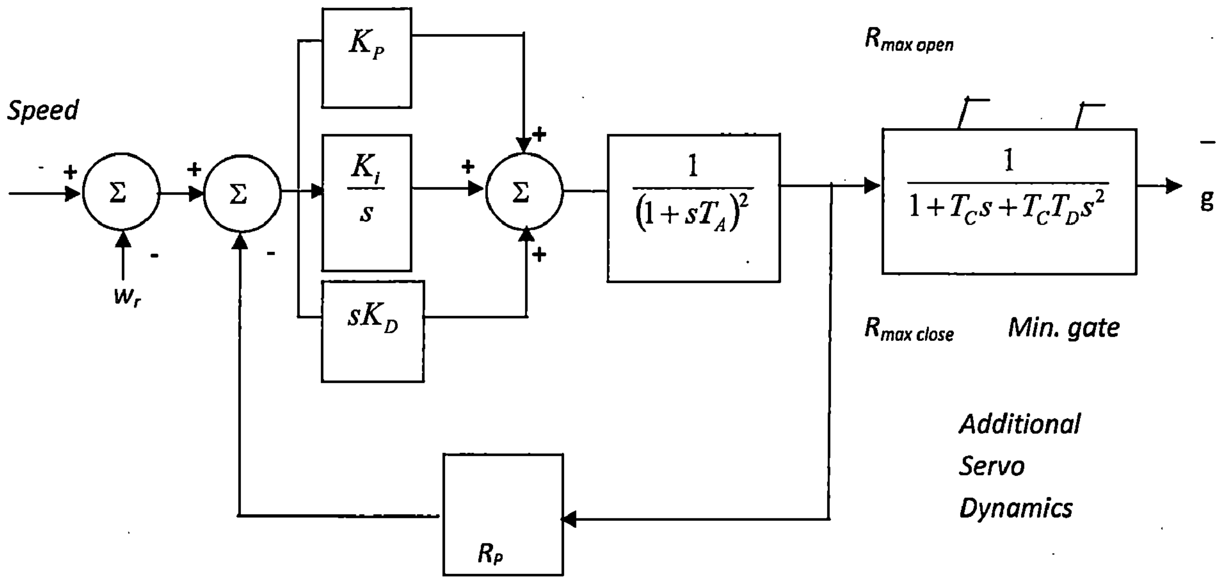

Figure 2.4 Model of PID Governor for Hydraulic Turbines

ω_r = rotor speed; K_P = proportional gain; K_I = integral gain; K_D = derivative gain;

T_A = pilot valve time constant; T_C & T_D = gate servo motor time constant of PID governor;

R_P = permanent droop; $R_{max\ open}$ = maximum gate opening rate limit;

$R_{max\ close}$ = minimum gate opening rate limit

2.1.3 Modelling of Excitation System

The basic requirement is that the excitation system supply and automatically adjusts the field current of the synchronous generator to maintain the terminal voltage as the output varies within the continuous capability of generator. In addition, the excitation system must be able to respond to transient disturbances with field forcing consistent with the generator instantaneous and short-term capabilities. From power system viewpoint, the excitation system should contribute to effective control of voltage and enhancement of system stability [1].

When the behavior of synchronous machines is to be simulated accurately in power system stability studies, it is essential that the excitation systems of the synchronous machines be modeled in sufficient detail; the desired models must be suitable for representing the actual excitation equipment performance for large, severe disturbances as well as for small perturbations [11].

2.1.3.1. Type DC1A Excitation System Model

The type DC1A exciter model represents field-controlled dc commutator exciters, with continuously acting voltage regulators. The exciter may be separately excited or self excited, the later type being more common. When self-excited, K_E is selected so that initially V_R =0, representing operator action of tracking the voltage regulator by periodically trimming the shunt field rheostat set point.[8]

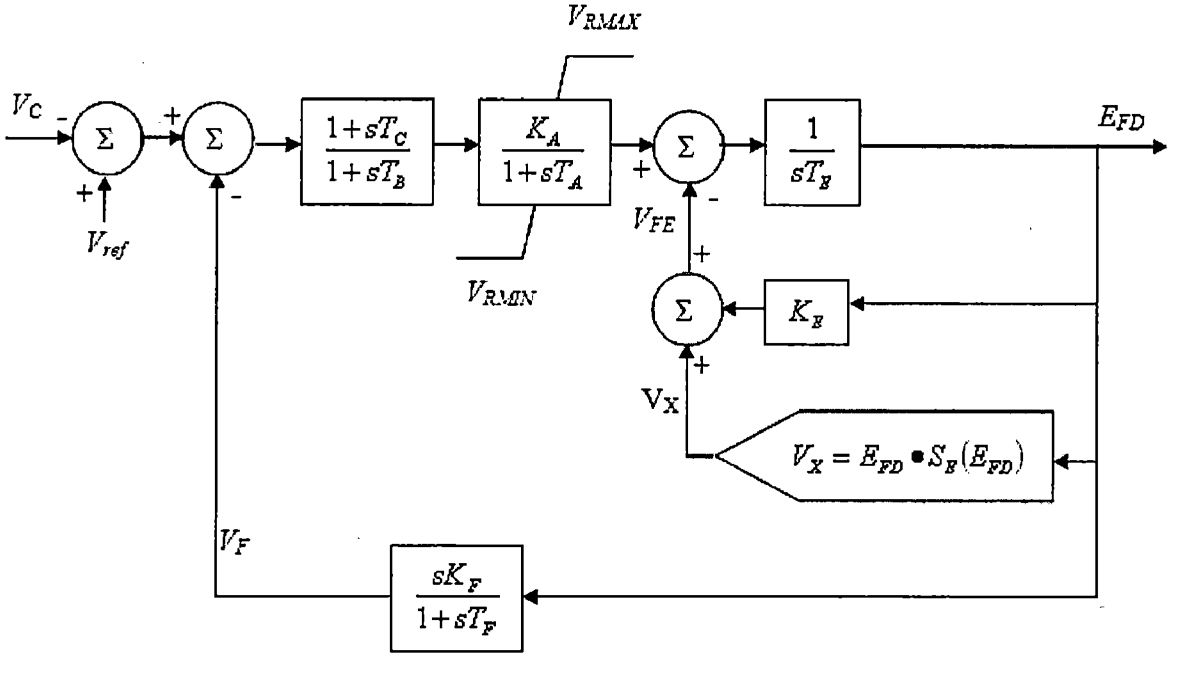

Figure 2.5: IEEE type D1A Excitation System Model

2.1.4 Modelling of Synchronous Generator

In developing equations of a synchronous machine, the following assumptions are made:

(a) The stator windings are sinusoidally distributed along the air-gap as far as the, mutual effects with the rotor are concerned.

(b) The stator slots cause no appreciable variation of the rotor inductances with rotor position.

(c) Magnetic hysteresis is negligible.

(d) Magnetic saturation effects are negligible.

Assumptions (a), (b), and (c) are reasonable. The principal justification comes from the comparison of calculated performances based on these assumptions and actual measured performances. Assumption (d) is made for convenience in analysis. With magnetic saturation neglected, we are required to deal with only linear coupled circuits, making superposition applicable. However, saturation effects are important.

In the circuits involved in the analysis of a synchronous machine, the stator circuits consist of 3-ϕ armature windings carrying alternating currents. The rotor circuits comprise field and amortisseur windings. But while developing the equations the amortisseur windings have been neglected .The field winding is connected to a source of direct current [1].

In figure 2.6, θ is defined as the angle by which the d-axis leads the axis of phase a winding in the direction of rotation. Since the rotor is rotating with respect to the stator, angle θ is continuously increasing and is related to the rotor angular velocity, ω_r and time, t as follows,

$$\theta = \omega_r t \qquad (2.21)$$

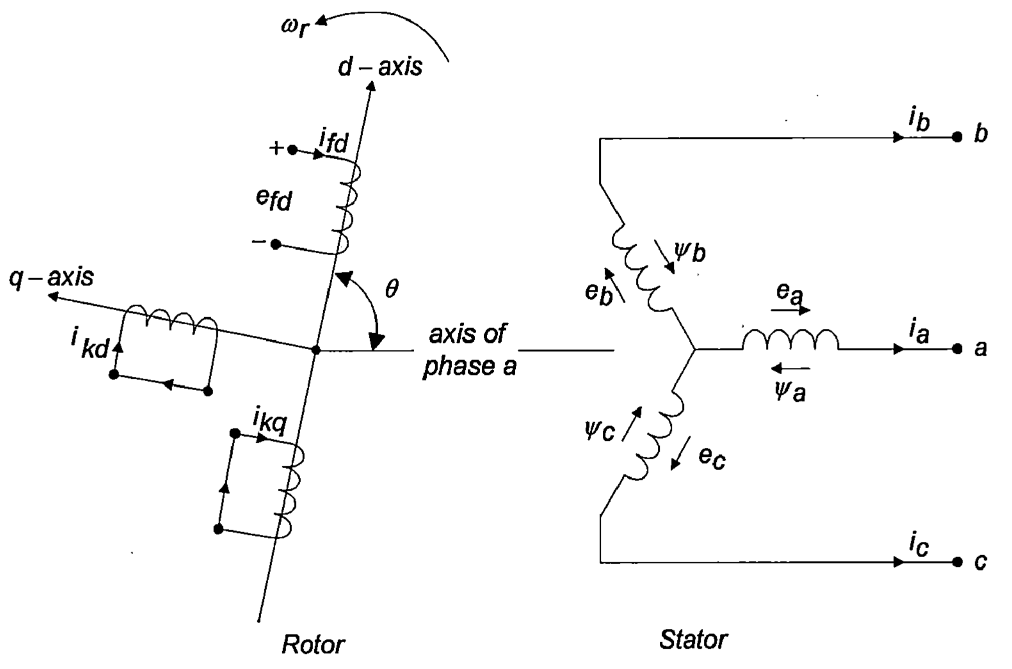

Figure 2.6: Stator and Rotor Circuits of Synchronous Generator

2.1.4.1 Basic Equations of a Synchronous Machine

In addition to the large number of circuits involved, the fact that mutual and self-inductances of the stator circuits vary with rotor position complicates the synchronous machine equations. The variation of inductances is caused by the variation of permeances of the magnetic flux path due to non-uniform air-gap. This is pronounced in a salient pole machine in which the permeances in two axes are significantly different. Even in the round rotor machine there are differences in the two axes due to large no. of slot associated with the field winding.

The voltage equations of stator and rotor circuits are: -

$$\left.\begin{aligned} e_a &= p\psi_a - i_a R_a \\ e_b &= p\psi_b - i_b R_a \\ e_c &= p\psi_c - i_c R_a \\ e_{fd} &= p\psi_{fd} + R_{fd} i_{fd} \end{aligned}\right\} \qquad (2.22)$$

The flux-linkages with stator and rotor windings at any instant t are given by: -

$$\psi_a = -i_a\left[L_{aao} + L_{aa2}\cos 2\theta\right] + i_b\left[L_{abo} + L_{aa2}\cos\left(2\theta + \frac{\pi}{3}\right)\right] + i_c\left[L_{abo} + L_{aa2}\cos\left(2\theta - \frac{\pi}{3}\right)\right]$$
$$+ i_{fd}L_{afd}\cos\theta$$

$$\psi_b = i_a\left[L_{abo} + L_{aa2}\cos\left(2\theta + \frac{\pi}{3}\right)\right] - i_b\left[L_{aao} + L_{aa2}\cos 2\left(\theta - \frac{2\pi}{3}\right)\right] + i_c\left[L_{abo} + L_{aa2}\cos(2\theta - \pi)\right]$$
$$+ i_{fd}L_{afd}\cos\left(\theta + \frac{2\pi}{3}\right)$$

$$\psi_c = i_a\left[L_{abo} + L_{aa2}\cos\left(2\theta - \frac{\pi}{3}\right)\right] + i_b\left[L_{abo} + L_{aa0}\cos(2\pi - \theta)\right] - i_c\left[L_{aao} + L_{aa2}\cos 2\left(\theta + \frac{2\pi}{3}\right)\right]$$
$$+ i_{fd}L_{afd}\cos\left(\theta + \frac{2\pi}{3}\right)$$

$$\psi_{fd} = L_{ffd}i_{fd} - L_{afd}\left[i_a\cos\theta + i_b\cos\left(\theta - \frac{2\pi}{3}\right) + i_c\cos\left(\theta + \frac{2\pi}{3}\right)\right] \qquad (2.23)$$

All inductances are functions of the rotor position, θ and thus time varying.

The dqo Transformation:

The dqo transformation is related to the abc phase variables as: -

$$\begin{bmatrix} X_d \\ X_q \\ X_o \end{bmatrix} = \frac{2}{3}\begin{bmatrix} \cos\theta & \cos\left(\theta - \frac{2\pi}{3}\right) & \cos\left(\theta + \frac{2\pi}{3}\right) \\ -\sin\theta & -\sin\left(\theta - \frac{2\pi}{3}\right) & -\sin\left(\theta + \frac{2\pi}{3}\right) \\ \frac{1}{2} & \frac{1}{2} & \frac{1}{2} \end{bmatrix}\begin{bmatrix} X_a \\ X_b \\ X_c \end{bmatrix}$$

Where *abc* represents phase variable and *dqo* represents dqo components. Using this stator flux in terms of d-q frame can be written as:

$$\left.\begin{aligned}\psi_d &= -L_d i_d + L_{afd} i_{fd} \\ \psi_q &= -L_q i_q \\ \psi_o &= -L_o i_o\end{aligned}\right\} \quad (2.24)$$

Where, $L_d = L_{aao} + \frac{3}{2} L_{aa2} + L_{abo}$

$$L_q = L_{aao} - \frac{3}{2} L_{aa2} + L_{abo}$$

$$L_o = L_{aao} - 2L_{abo}$$

Per unit stator voltage equations in d-q frame can be expressed as:-

$$\left.\begin{aligned}e_d &= \frac{2}{3}\left(e_a \cos\theta + e_b \cos\left(\theta - \frac{2\pi}{3}\right) + e_c \cos\left(\theta + \frac{2\pi}{3}\right)\right) \\ e_q &= -\frac{2}{3}\left(e_a \sin\theta + e_b \sin\left(\theta - \frac{2\pi}{3}\right) + e_c \sin\left(\theta + \frac{2\pi}{3}\right)\right) \\ e_o &= \frac{1}{3}(e_a + e_b + e_c)\end{aligned}\right\} \quad (2.25)$$

An alternative form of the voltages is:-

$$\left.\begin{aligned}e_d &= \tfrac{1}{\omega_{base}} p\psi_d - \psi_q \omega_r - R_a i_d \\ e_q &= \tfrac{1}{\omega_{base}} p\psi_q + \psi_d \omega_r - R_a i_q \\ e_o &= \tfrac{1}{\omega_{base}} p\psi_o - R_a i_o\end{aligned}\right\}$$

Neglecting speed variations and stator transients, above equations can be written as: -

$$\left.\begin{aligned} e_d &= -\psi_q \omega_s - R_a i_d \\ e_q &= \psi_d \omega_s - R_a i_q \end{aligned}\right\} \tag{2.26}$$

Stator current can be expressed in d-q frame as: -

$$\left.\begin{aligned} i_d &= \frac{2}{3}\left(i_a \cos\theta + i_b \cos\left(\theta - \frac{2\pi}{3}\right) + i_c \cos\left(\theta + \frac{2\pi}{3}\right)\right) \\ i_q &= -\frac{2}{3}\left(i_a \sin\theta + i_b \sin\left(\theta - \frac{2\pi}{3}\right) + i_c \sin\left(\theta + \frac{2\pi}{3}\right)\right) \\ i_o &= \frac{1}{3}\left(i_a + i_b + i_c\right) \end{aligned}\right\} \tag{2.27}$$

The per unit electrical power output can be expressed as: -

$$P_{elect} = e_d i_d + e_q i_q + 2e_o i_o \tag{2.28}$$

The per unit air-gap torque can be expressed as: -

$$T_e = \left(\psi_d i_q - \psi_q i_d\right)\frac{P_f}{2} \tag{2.29}$$

CHAPTER 3

ARTIFICIAL NEURAL NETWORKS AND GENETIC ALGORITHM

3.1 INTRODUCTION TO ANN

An artificial neural network is a system based on the operation of biological neural networks, in other words, it is an emulation of biological neural system. It is composed of a large number of highly interconnected processing elements (neurons) working in unity to solve specific problems.

Dr. Robert Hecht-Nielsen, inventor of one of the first neuro computers defines a neural network as described below:

"...A computing system made up of a number of simple, highly interconnected processing elements, which process information by their dynamic state response to external inputs.

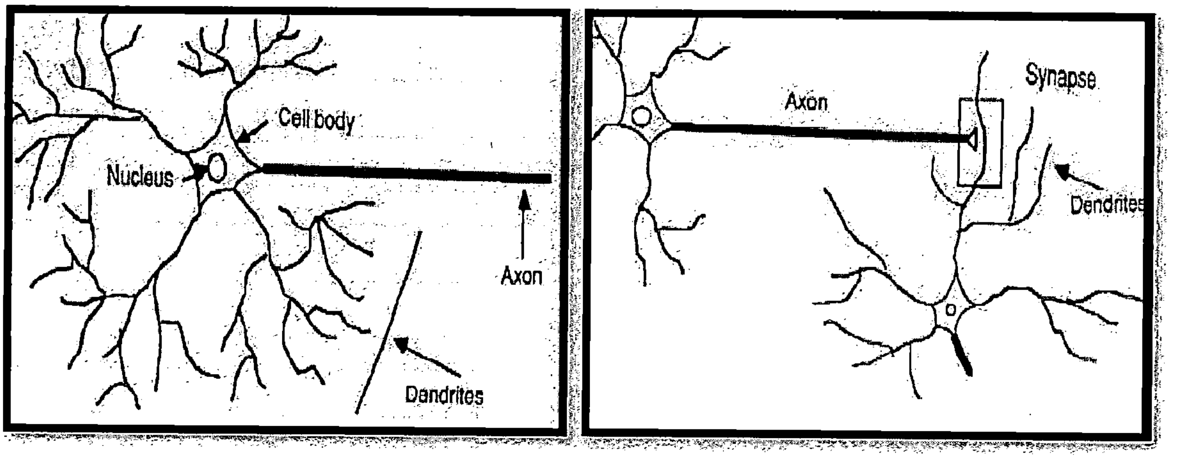

Figure 3.1.Physical structure of biological neuron

ANNs, like people, learn by example. An ANN is configured for a specific application, such as pattern recognition or data classification, through a learning process. Learning in biological systems involves adjustments to the synaptic connections that exist between the neurons. This is true of ANNs as well.

The motivation for the development of neural network technology stemmed from the desire to develop an artificial system that could perform "intelligent" tasks similar to those performed by the human brain. Neural networks resemble the human brain in the following two ways:

1. A neural network acquires knowledge through learning.
2. A neural network's knowledge is stored within inter-neuron connection strengths known as synaptic weights (as shown in Fig. 3.1).

The first artificial neural network was invented in 1958 by psychologist Frank Rosenblatt called Perceptron, it was intended to model how the human brain processed visual data and learned to recognize objects.

3.2 STRUCTURE OF ARTIFICIAL NEURON

ANN model has a fundamental processing unit called a neuron. Fig.3.2 shows a conceptual model of the neuron, which receives several inputs through connections called synapses. The incoming activations are multiplied by the synaptic weights and summed up. The outgoing activation is determined by applying a threshold function (like sigmoid, hyperbolic tangent, Gaussian, bipolar linear etc.,) to the summation. The neuron has only one outgoing activation value although it might have several connections to the other neuron. The threshold function is a non-linear function that decides the output of a particular neuron.

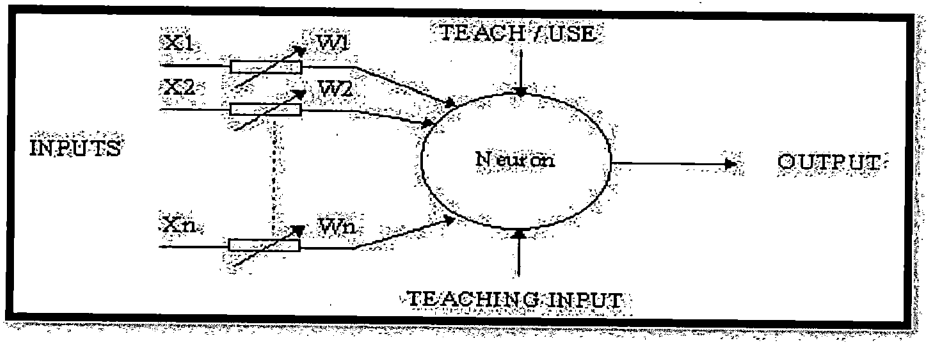

Fig.3.2.Structure of an artificial neuron

$$\text{Net} = \sum x_i w_i \quad (3.1)$$

In mathematical terms, the neuron fires if and only if;

$X1W1 + X2W2 + X3W3 + \ldots > T$, the threshold value.

In general, the higher a weight of an artificial neuron, the stronger the input which is multiplied by it will be. Weights can also be negative, so we can say that the signal is inhibited by the negative weight. Depending on the weights, the computation of the neuron will be different. By adjusting the weights of an artificial neuron we can obtain the output we want for specific inputs.

3.3 BASICS OF ANN ARCHITECTURE

A neural network consists of many neurons and the method by which neurons are organized is called as "network architecture". Each neuron in any architecture is a simple processing element that responds to the weighted inputs from the other neurons. ANN consists of mostly three types of layers.

Input layer to the neural network is the passage through which data is presented to neural network. The input data might consist of the wavebands of a data set, the texture of the image or other complex parameters. In this study, MODIS ten selective bands were used as a part of training and also for validation of the network.

Hidden layers-refer to one or more layers of neurons that are arranged between the input and output layer. These layers do not directly interact with the external environment but these layers have tremendous influence on the final output and hence on the network performance. An increase in the number of hidden layers enables the network to learn more complex problems, but the capacity to generalize is reduced and there is an associate increase in training time [15]. Lipman suggests that *if* a second hidden layer is used in the network, the maximum number of nodes in the second hidden layer should be three times the number in the first hidden layer.

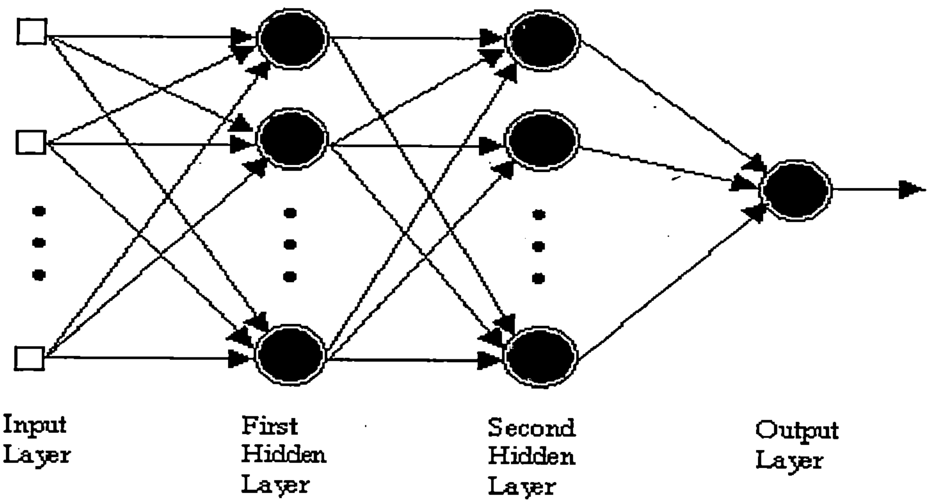

FIG.3.3. Block diagram of two hidden layer MLP

Output layer of the neural network is what actually presents the output data/results to the user. For example, for remote sensing image classification, the number of nodes in the output layer is equal to the *c* classes in the classification

In each hidden layer and output layer, the processing unit sums its input from the previous layer and then applies the threshold function to compute its output to the next layer. The data is repeatedly presented to the neural network. With each presentation, the error between the network output and desired output is computed and is then fed back to the neural network. The neural network uses this error to adjust its synaptic weights such that the error will be decreased. This sequence of events is usually repeated until an acceptable error has been reached.

There are various types of network architectures and one of the most popular and successful is that of the multi-layer perceptron (MLP) and is shown in fig. 3.3. This consists of identical neurons that are all interconnected and organized in layers. Neurons in one layer are connected to those in the next layer so that the outputs in one layer become the inputs in the subsequent layer.

3.4 TYPES OF NEURAL NETWORKS AND TYPES OF CONNECTIONS

There are two broad classes of neural networks based on the connection pattern:

(i) One type is multilayer perceptron network or MLP network, which is a generalization of single layer neural network. This network does not have any feedback, and thus it is also referred to as feed forward neural network (as shown in Fig 3.4).

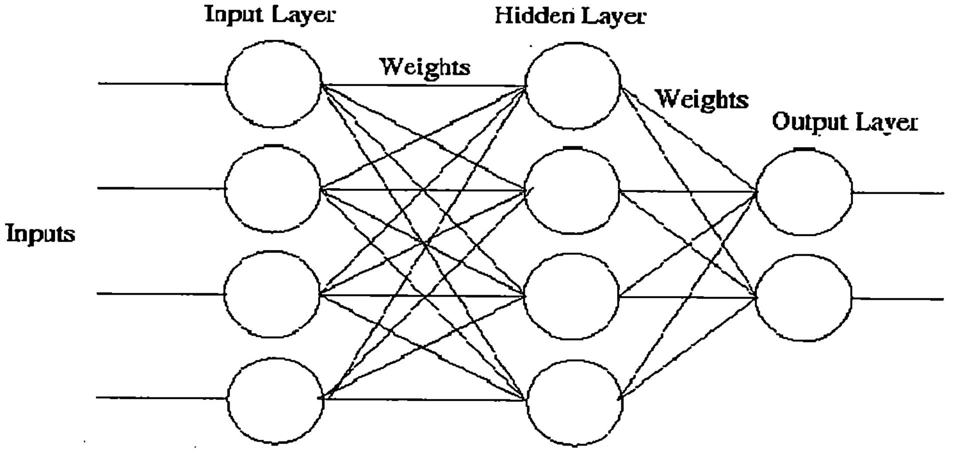

Fig.3.4.A. Block diagram of three layer feed forward neural network (FFNN)

(ii) The other class of network has feedback, where the output of each neuron, in general, is fed back to itself and all the other neurons. Such networks are referred to as recurrent networks.

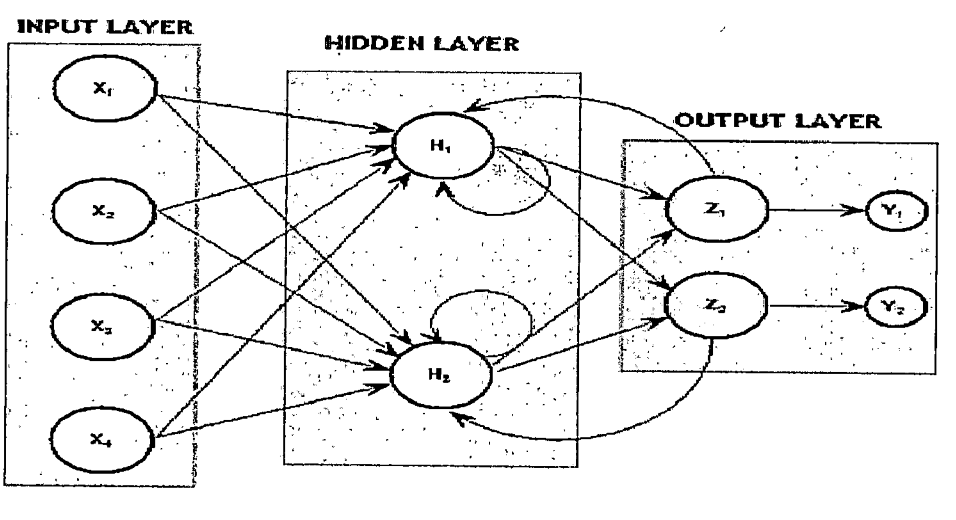

Fig.3.4.B.Typical diagram of Recurrent Neural Network

Generally speaking, feed forward networks are *static* in the sense that they produce only one set of output values rather than a sequence of values from a given input. Feed forward networks are memory less in the sense that their response to an input is independent of the previous network state. Recurrent or feedback networks, on the other hand, are dynamic systems. When a new input pattern is presented, the neuron outputs are computed. Because of the feedback paths, the inputs to each neuron are then modified, which leads the network to enter a new state.

The connections between layers are called inter-layer connections and they are of following types:

Fully connected – each neuron on the first layer is connected to every neuron the second layer.

Partially connected- a neuron of the first layer does not have to be connected to all neurons on the second layer.

Bi – directional- there is another set of connections carrying the output of the neurons of the second layer into the neurons of the first layer.

3.5 HOW TO CHOOSE NUMBER OF NODES AND NUMBER OF HIDDEN LAYERS?

The number of neurons or nodes in the input and output layers is automatically determined by the number of input and output variables in the task to be handled. On the other hand, we need to make two important key decisions with regard to the hidden layers, namely, the number of hidden layers and the number of neurons in each such layer. Using too few neurons in the hidden layer will result in under fitting, which occurs due to failure in adequately detecting the signals in a complicated data set. Using too many neurons in the hidden layers will result in several problems such as over-fitting and an increase in the training time to such an extent that it may be impossible to adequately train the network for the practical purposes[16].

Therefore, a compromise has to be reached between too many and too few neurons in the hidden layers. There exists no rigid rule for determining the hidden neuron count. One approach is to start with as few neurons as possible, usually two, to train the neural network and test it. The number of hidden neurons is then increased and the process is repeated so long as the overall results of the training and testing improves significantly.

Second approach for deciding the number of nodes in the hidden layer is to follow Kolmogorov's theorem. According to Kolmogorov's theorem, it is understood that twice the number of input nodes plus one is sufficient to compute any arbitrary continuous function (i.e. if we have two input node, then number of nodes in hidden layer to start with, would be calculated as 2 * 2+1=5). The general trend is to start with Kolmogorov's number of hidden nodes and then go on increasing the number of nodes until a network with the least mean squared error is attained.

To decide the number of hidden layer is a key decision to be made. Although in theory, any number of hidden layers may be added, in general only one hidden layer is required for the majority of pharmaceutical applications. A rule of thumb indicates that 85% of all problems can be learned by neural network with single hidden layer, and this network architecture is most commonly used in formulation development. Multiple hidden layers are only necessary for those applications with extensive non-linear behavior.

3.6 TRAINING OR LEARNING OF ANN

The aim of network training is to build a model of the data generating process so that the network can generalize and predict outputs from inputs that it has not seen before. The process of adjusting the weights to make the network learn the relationship between the inputs and targets is called learning or training.

3.6.1 Types of Learning

Artificial neural networks typically start out with randomized weights for all their neurons. This means that they don't "know" anything and must be trained to solve the particular problem for which they are intended. Broadly speaking, there are three main categories for learning, namely supervised, unsupervised, and hybrid learning [16].

Supervised learning

The vast majority of artificial neural networks are trained by supervised learning. In supervised learning data are presented to neural networks as records, each of which corresponds to a formulation experiment. Each record contains both inputs and outputs. The network is initialized by putting small random weights and an output is calculated. The output is compared with the observed data. Weights that are usually randomly set to begin with, are then adjusted by the network and are compared to the desired and actual output. The learning method tries to minimize the current errors of all processing elements. This global error reduction is created over time by continuously modifying the input weights until acceptable network accuracy is reached. The method used most frequently for this is called back propagation. Back propagation can be standard incremental back propagation (here the weights are updated after each pattern) or standard batch propagation (here the weights are updated only after all the patterns have been presented to network).

When no further learning is necessary, the weights are typically frozen for the application. After a supervised network performs well on the training data, it is important to see what it can do with data it has not seen before. If a system does not give reasonable outputs for this test set, the training period is not over. Therefore, this testing is critical to insure that the network has not simply memorized a given set of data but has learned the general patterns involved within application.

Unsupervised learning

Unsupervised learning is sometimes called self-supervised learning. With unsupervised learning, there is no feedback from the environment to indicate if the outputs of the network are correct. A sequence of input vectors is provided, but no target vectors are specified. The network must discover the features, regulations, correlations, or categories in the input data automatically. Thus, unsupervised learning involves trial and error type strategies and these are the type of situations which one often encountered in practice. These networks use no external influences to adjust their weights. Instead they internally monitor their performance. These network look for regularities or trends in the input signals and makes adaptations according to the function of the network.

Hybrid Learning

Hybrid learning combines supervised and unsupervised learning. Parts of the weights are usually determined through supervised learning, while the others are obtained through unsupervised learning.

3.6.2 Learning Rules

These rules are mathematical algorithms used to update the connection weights. Given below are few of the major rules

(a) **Hebbian learning rule-** according to this rule, if a neuron receives input from another neuron and if both are highly active (mathematically have the same sign), the weight between the neurons should be strengthened. For the neurons operating in the opposite phase, the weights between them should be weakened. If there is no correlation, the weights should remain unchanged.

(b) **Correlation learning rule-** it is based on similar principles as the Hebbian learning rule. It assumes that weights between simultaneously responding neurons should be largely positive and between the neurons with opposite reaction should be largely negative. Instead of actual response, the desired response is used for weight change calculation.

(c) **Competitive learning rules-** here the output neurons of an ANN compete among themselves to become activated or fired, with the result that only one output neuron, or one neuron per group, is on at one time. An output neuron that wins the competition is called winner neuron. One way of inducing competition among the output neurons is to use lateral inhibitory connections between them.

(d) **The delta rule-** this is basically an error correction learning rule. It is one of the most commonly used. This rule is based on the simple idea of continuously modifying the strengths of input connections to reduce the difference (the delta) between the desired output value and the actual output of processing element. This rule changes the synaptic weights in the way that minimizes the mean squared error of the network. This rule is also referred to as Widrow- Hoff Learning rule or Least Mean square (LMS) learning rule. Out of the above mentioned rules the most popular one is the delta rule. The *generalized delta rule* [14] can be written as:

$$\Delta\omega_{ji}(n+1) = \lambda\,(\delta_j\, o_i) + \acute{\alpha}\,\Delta\omega_{ji}(n)$$

Where ω_{ji} represents the weights between node *i* and node *j*, and o_i is the output from node *i*, λ is the learning rate parameter, δj is an index of the rate of change of the error, and $\acute{\alpha}$ is the momentum parameter. This process of feeding forward signals and back-propagating the error is repeated iteratively until the error of the network as a whole is minimized or reaches an acceptable magnitude.

The neural networks are entirely data driven and the issue of "how much data" needs to train the networks is an important factor. In practice, a good model can be developed from 3 to 4 times as many experiments as there are inputs, provided that the data are of good quality.

3.6.3 Data set for ANNs

ANNs must be provided with relevant and large data set as possible for its training purpose. In case of ANN any non-numerical data must be processed in such a way to obtain a numeric representation. A "data set" is a matrix containing several samples. In ANN methodology, the data set is often sub divided into "Learning" and "Test" sets. A Learning set is a set of examples used for learning, which is to fit the parameters [i.e., weights] of the classifier. A validation set is a set of examples used to tune the parameters [i.e., architecture, not weights] of a classifier, for example, number of hidden units.

A test set is a set of examples used only to assess the performance [generalization] of a fully specified classifier. Learning sets needs to be fairly large to contain all the needed information if the network is to learn the features and relationship that are important. Not only do the sets have to be large but also the training session must include a wide variety of data.

3.7 ANN VERSUS CONVENTIONAL COMPUTING

To better understand artificial neural computing it is important to know first how a conventional 'serial' computer and its software process information. A serial computer has a central processor that can address an array of memory locations where data and instructions are stored. Computations are made by the processor reading an instruction as well as any data the instruction requires from memory addresses, the instruction is then executed and the results are saved in a specified memory location as required. In a serial system (and a standard parallel one as well) the computational steps are deterministic, sequential and logical, and the state of a given variable can be tracked from one operation to another.

In comparison, ANNs are not sequential or necessarily deterministic. There are no complex central processors, rather there are many simple ones which generally do nothing more than take the weighted sum of their inputs from other processors. ANNs do not execute programmed instructions; they respond in parallel (either simulated or actual) to the pattern of inputs presented to it. There are also no separate memory addresses for storing data. Instead, information is contained in the overall activation 'state' of the network. 'Knowledge' is thus represented by the network itself, which is quite literally more than the sum of its individual components.

3.8 ADVANTAGES AND DISADVANTAGES OF ANN

The main advantages and disadvantages/limitations of ANNs are briefly described below [21].

3.8.1 Advantages

(a) The true power and advantage of neural networks lies in their ability to represent both linear and non-linear relationships and in their ability to learn these relationships directly from the data being modeled. Thus a neural network can perform tasks that a linear program can't.

(b) ANNs provide an analytical alternative to conventional techniques which are often limited by strict assumptions of normality, linearity, variable independence etc

(c) When an element of the neural network fails, it can continue without any problem by their parallel nature.

(d) ANN can perform more accurately than other techniques such as statistical classifiers, particularly when the feature space is complex and the source data has different statistical distributions.

(e) Perform more rapidly than other techniques such as statistical classifiers [12]

(f) Incorporate *a priori* knowledge and realistic physical constraints into the analysis .

(g) ANNs can incorporate different types of remote sensing data (including those from different sensors) into the analysis.

(h) ANN can capture many kinds of relationships and it allows the user to quickly and relatively easily model phenomena which otherwise may have been very difficult or impossible to explain otherwise.

(i) A neural network learns by itself and does not need to be reprogrammed.

(j) It can be implemented in any application.

(k) It can be implemented without any problem.

(l) An ANN can create its own organization or representation of the information it receives during learning time.

(m)An ANN model, unlike statistical models operates upon the experimental data without data transformations.

(n) Further, the neural networks (once trained) are faster than other approaches, and they may allow the incorporation of data from different sources into the estimation.

3.8.2 Disadvantages or Limitations

(a) The major disadvantage of ANN is that, they are by nature black boxes; the relationship that the network finds cannot be expressed easily in mathematical form.

(b) The primary risk in developing a model is that of overtraining, a situation in which the neural network starts to reproduce the noise specific to a particular sample in the training data, which may cause it to lose its ability to predict accurately. This disadvantage can be removed as mentioned earlier by performing network validation.

(c) Back propagational neural networks (and many other types of networks) are in a sense the ultimate 'black boxes'. Apart from defining the general architecture of a network and perhaps initially seeding it with a random numbers, the user has no other role than to feed it input and watch it train and await the output. In fact, it has been said that with back propagation, "you almost don't know what you're doing". Some software packages do allow the user to sample the networks 'progress' at regular time intervals, but the learning itself progresses on its own. The final product of this activity is a trained network that provides no equations or coefficients defining a relationship (as in regression) beyond its own internal mathematics. The network is the final equation of the relationship.

(d) Back propagational networks also tend to be slower to train than other types of networks and sometimes require thousands of epochs. If run on a truly parallel computer system this issue is not really a problem, but if the BPNN is being simulated on standard serial machine (i.e. a single SPARC, Mac or PC) training can take some time. This is because the machines CPU must compute the function of each node and connection separately, which can be problematic in very large networks with a large amount of data. However, the speed of most current machines is such that this is typically not much of an issue

(e) In general the neural network needs training to operate.

(f) The architecture of a neural network is different from the architecture of micro processors therefore needs to be emulated.

(g) Requires high processing time for large neural networks.

3.9 NETWORK VALIDATION

The primary risk in developing a ANN model is that of over training, a situation in which the neural network starts to reproduce the noise specific to a particular sample in the training data, which may cause it to lose its ability to predict accurately. There are certain techniques that are used to avoid this; the most popular of this is network validation. The testing of the network prediction is done by reserving some of the data, which is excluded from the training data sets. The network is used to predict the outputs for these reserve data records, and the calculated outputs are compared with the observed values.If they are found to be sufficiently close, the network is considered to be sufficiently predictive and the network is said to be validated .

3.10 NETWORK GENERALIZATION

Generalization is the ability of a neural network to produce reasonable responses to input patterns that are similar, but not identical, to training patterns. Once it is claimed that neural network has been trained properly or it is said that the neural network has learned the concept. Then it is expected that neural network produces the correct output for previously unknown inputs, but belonging to the same concept class. Then, the trained neural network has the capability to generalize.

Several factors affect the generalizing capability of neural network. These include:

(i) Number of nodes and architecture

If a large number of simple processing elements are used the mathematical structure can be made very flexible and the neural network can be used for a wide range of applications. This may not be necessary for all applications. In general terms, the larger the number of nodes in the hidden layer(s), the better the neural network is able to represent the training data, but at the expense of the ability to generalize.

(ii) Size of training set

If the extent of the distribution of the data in feature space is not covered adequately the network may fail to classify new data accurately. A consequence of this for the MLP algorithm is that large quantities of data are often required for training, and researchers are often concerned with finding the minimum size of data set necessary. The requirement for large training data sets also means that training times may be long. To speed up the training process, several modifications to the MLP algorithm have been introduced including the momentum term, the delta-bar-delta rule, and optimization procedures.

(iii) Training time

The time taken for training also affects the generalizing capabilities of the network.The longer that a network is trained on a specific data set, the more accurately it will be able to classify those data, but at the expense of the ability to classify previously unseen data. In particular, it is possible to over train a network so that it is able to memorize the training data but is not able to generalize when it is applied to different data.

3.12 ACTIVATION FUNCTION

The *net* signal is usually further processed by an activation function F to produce the neuron's output signal, *out*. This may be simple linear function.

Out = K (net) (3.2)

Where, K is a constant threshold function such that

out = I if net > T, where 'T' is a constant threshold value.

out = 0, otherwise

In fig.1 the block labeled f accepts the *net* output and produces the signal labeled *out*. If the f block compresses the range of *net*, so that the *out* never exceeds some low limit regardless of the value of *net*, F is called squashing function. The Squashing function is chosen to be a logistic function or sigmoid as shown in Fig. 3.5. Sigmoidal activation function is mathematically expressed as [16]:

$F(x) = 1/(1 + e^{-X})$ thus

out = 1/(1 + exp(net)) (3.3)

It is symmetrical about the origin, resulting in *out* having a value' 0' when net is '0'.

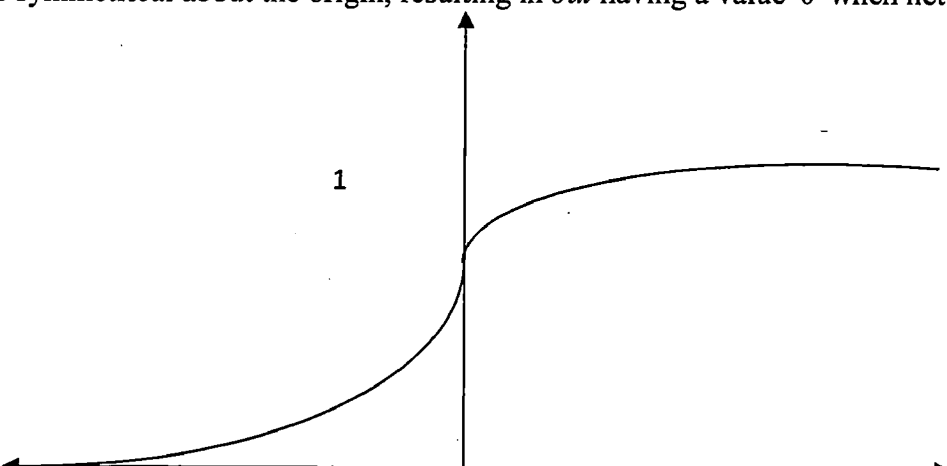

Figure 3.5 Sigmoid

3.13 THE BACKPROPAGATION ALGORITHM

3.13.1 History of the Back propagation algorithm

Rumelhart et al. [1986] popularized the back propagation algorithm in *Parallel Distributed Processing* **(PDP)** edited volumes in the late 1980s. The PDP volumes summarized the neural research of several psychologist and computer scientists at the University of Calfornia at San Diego. The PDP volumes helped secure for neural network a broad interdisciplinary audience.

Back propagation modifies synaptic connection strengths with nonlocal error information. Non-locality, synchrony, supervision, and lengthy training cycles preclude biological plausibility.

The back propagation did not always converge in discrete simulations. Some choice of initial conditions led to oscillations, even chaotic wandering. [14]

3.13.2 Back propagation

This learning algorithm is applied to multilayer feed forward networks consisting of processing elements with continuous differentiable activation functions. Such network associated with the back propagation learning algorithm is also called back propagation networks. Given a training set of input-output pairs, the algorithm provides a procedure for changing the weight in a back-propagation network to classify the given input patterns correctly. [14]

3.13.2.1. The Back propagation Algorithm

For a given set of training pairs say P

Where $P = \{\mathbf{z}_1, \mathbf{d}_1, \mathbf{z}_2, \mathbf{d}_2 \ldots \mathbf{z}_p, \mathbf{d}_p,\}$,

$\mathbf{z}_i$ is (I x 1), $\mathbf{d}_i$ id (K x 1), and i =1,2…P. Note that the Ith component of each $\mathbf{z}_i$ is of value −1 since input vector has been augmented. Size J-1 of the hidden layer having outputs $\mathbf{y}$ is selected. Note that the Jth component of $\mathbf{y}$ is of value −1, since hidden layer outputs have also been augmented; $\mathbf{y}$ is (J x 1) and $\mathbf{o}$ is (K x 1).

Step 1: $\eta>0$, E_{max} is chosen.

Weights **W** and **V** are initialized at small random values; **W** is (K x J), **V** is (J x I).

$$q \leftarrow 1,\ p \leftarrow 1,\ E \leftarrow 0 \tag{3.4}$$

Step 2: Training step starts here

Input is presented and the layer's outputs computed

$$\mathbf{z} \leftarrow \mathbf{z_p},\ \mathbf{d} \leftarrow \mathbf{d_p}$$

$$y_j \leftarrow f(\mathbf{v}^t_j \mathbf{z}), \quad \text{For } j=1,2,\ldots,J \tag{3.5}$$

Where $\mathbf{v}_j$ is a column vector, is the j[th] row of **V**, and

$$o_k \leftarrow f(\mathbf{w}^t_k \mathbf{y}), \text{ For } k=1,2,\ldots,K \tag{3.6}$$

Where $\mathbf{w}_k$ a column vector, is the k[th] row of **W**

Step 3: Error value is computed:

$$E \leftarrow \frac{1}{2}(d_k - o_k)^2 + E, \text{ For } k=1,2,\ldots,K \tag{3.7}$$

Step 4: Error signal vector δ_o and δ_y of both layers are computed. Vector δ_o is (K x 1), δ_y is (J x 1).

The error signal terms of output layer in this step are

$$\delta_{ok} = \frac{1}{2}(d_k - o_k)(1 - o_k^2), \text{ For } k=1,2,\ldots,K \tag{3.8}$$

the error signal terms of hidden layer in this step are

$$\delta_{yj} = \frac{1}{2}(1 - y_j^2)\sum_{k=1}^{K}\delta_{ok} w_{kj}, \text{ For } j=1,2,\ldots,J \tag{3.9}$$

Step 5: Output layer weights are adjusted:

$$w_{kj} \leftarrow w_{kj} + \eta \delta_{ok} y_j, \text{ for } k=1,2,\ldots,K \text{ and } j=1,2,\ldots,J \tag{3.10}$$

Step 6: Hidden layer weights are adjusted:

$$v_{kj} \leftarrow v_j + \eta \delta_{yj} z_i, \text{ for } k=1,2,\ldots,K \text{ and } i=1,2,\ldots,I \tag{3.11}$$

Step 7: If $p<P$ then $p \leftarrow p+1, q \leftarrow q+1$, and go to step 2; otherwise go to step 8.

Step 8: The training cycle is completed.

For $E< E_{max}$ terminate the training session. Output weights **W**, **V**, q, and E.

If $E> E_{max}$, then $E \leftarrow 0, p \leftarrow 1$, and initiate the new training cycle by going to step 2.

3.13.2.2 Implementation consideration of Back propagation Algorithm:

For back propagation algorithm to be feasible there are some criterion, which are:

1. The activation function must be differentiable.

2. The error function should not reach to zero otherwise it may stick at local minima. Instead the algorithm should stop when $E_{max} < \in$ where $\in$ is a small positive real number.

3. Weights should be randomly initialized as for the same weight, the change would be same. After updating, weights will remain same.

4. The convergence speed is limited by the slow adaptation of the connections in layer close to the input layer. To overcome this problem, shortcut connections from lower layer directly to the output units may be made.

3.14 LEARNING FACTORS AND THEIR EFFECT

The essence of the error back propagation algorithm is the evaluation of the contribution of each particle weight to the output error. In practice, implementation of the algorithm may encounter different difficulties. The solution of the difficulties can be obtained by adjusting the various parameters or factors to the proper value. One of the problems is that the error minimization procedure may produce only local minima of the error function. In practice the learning would be considered successful for error below an acceptable minimum error value. To get stable global convergence we have to adjust various factors so that incremental weight adjustment size can be chosen of suitable value. These factors are as follows [14]:

(1) Initial weights: The weights of the network to be trained are typically initialized at small random values. If all weights start out with equal values and if the solution requires that unequal weights be developed, the network may not trained properly. So, random initialization of weights is essential.

(2) Non-linearity factor: The neuron's continuous activation function is characterized by its non-linearity factor. Also the derivative of the activation function serves as a multiplying factor in building components of the error signal vectors. Thus both the choice and the shape of the activation function would strongly affect the speed of network learning. The standard value of non-linearity factor is unity.

(3) Learning constant: The effectiveness and convergence of the error back propagation learning algorithm depend significantly on the value of the learning constant. Broad minima yield small gradient values, then a larger value of learning constant will result in a more rapid convergence. However, for the problems with steep and narrow minima, a small value of learning constant must be chosen to avoid overshooting the solution. This leads to the conclusion that learning constant should indeed be experimentally for each problem. The value of learning constant ranging from 10^{-3} to 10 has been taken throughout the technical literature as successful for many computational back propagation experiments.

(4) Momentum factor: The purpose of momentum factor is to accelerate the convergence of the error back propagation learning algorithm. The method involves supplementing the current weight adjustments with a fraction of the most recent (previous) weight adjustment. Typically momentum factor is chosen between 0.1 and 0.8.

3.15 SELF-TUNING USING GENETIC ALGORITHM:

In the above section the disadvantages of manual tuning has been discussed. Now in order to get consistent tuning, software tools are used for tuning a PID controller. One such method is the Genetic Algorithm (GA tool) available in the MATLAB software. Most modern industrial facilities no longer tune loops using the manual calculation methods shown above. Instead, PID tuning and loop optimization software are used to ensure consistent results. These software packages will gather the data, develop process models, and suggest optimal tuning. Some software packages can even develop tuning by gathering data from reference changes.

A genetic algorithm (GA) is a search technique used in computing to find exact or approximate solutions to optimization and search problems[12]. Genetic algorithms are implemented in a computer simulation in which a population of abstract representations (called chromosomes or the genotype of the genome) of candidate solutions (called individual, creatures, or phenotypes) to an optimization problem evolves toward better solutions. Traditionally, solutions are represented in binary as strings of 0s and 1s, but other encodings are also possible. The evolution usually starts from a population of randomly generated individuals and happens in generations. In each generation, the fitness of every individual in the population is evaluated, multiple individuals are stochastically selected from the current population (based on their fitness), and modified (recombined and possibly randomly mutated) to form a new population. The new population is then used in the next iteration of the algorithm. Commonly, the algorithm terminated when either a maximum number of generations has been produce, or a satisfactory fitness level has been reached for the population. If the algorithm has terminated due to a maximum number of generations, a satisfactory solution may or may not have been reached. A typical genetic algorithm requires:

(1) A genetic representation of the solution domain

(2) A fitness function to evaluate the solution domain

The main property that makes these genetic representations convenient is that their parts are easily aligned due to their fixed size, which facilitates simple crossover operations. Variable length representations may also be used, but crossover implementation is more complex in this case.

The fitness function is defined over the genetic representation and measures the quality of the represented solution. The fitness function is always problem dependent. Once we have the genetic representations and the fitness function defined, GA proceeds to initialize a population of solutions randomly, and then improve it through repetitive application of mutation, crossover, and inversion and selection operators.

Initialization:

Initially many individual solutions are randomly generated to form an initial population. The population size depends on the nature of the problem, but typically contains several hundreds or thousands of possible solutions. Traditionally, the population is generated randomly, covering the entire range of possible solutions. Occasionally, the solutions may be "seeded" in areas where optimal solutions are likely to be found.

Selection:

During each successive generation, a proportion of the existing population is selected to breed a new generation. Individual solutions are selected are selected through a fitness-based process, where fitter solutions are typically more likely to be selected. Certain selection methods rate the fitness of each solution and preferentially select the best solutions. Other methods rate only a random sample of the population, as this process may be very time-consuming. Most functions are stochastic and designed so that a small proportion of less fit solutions are selected. This helps keep the diversity of the population large, preventing premature convergence on poor solutions. Popular and well-studied selection methods include Rolette wheel selection and tournament selection.

Reproduction:

The next step is to generate a second generation population of solutions from those selected through genetic operators: crossover (also called recombination), and/or mutation. For each new solution to be produced, a pair of "parent" solutions for breeding from the pool selected previously. By producing a "child" solution using the above methods of crossover and mutation, a new solution is created which typically shares many of the characteristics of its "parents". New parents are selected for each child, and the process continues until a new population of solutions of appropriate size is generated. Although reproduction methods that are based on the use of two parents are more "biology inspired", recent researches suggested more than two "parents" are better to be used to reproduce a good quality chromosome. These processes ultimately result in the next generation population of chromosomes that is different from the initial generation. Generally the average fitness will have increased by this procedure for the population, since only the best organisms form the first generation are selected for breeding, along with a small proportion of less fit solutions, for reason already mentioned above.

Termination:

This generational process is repeated until a termination condition has been reached. Common terminating conditions are:

(a) A solution is found that satisfies minimum criteria
(b) Fixed number of generations reached
(c) Allocated computation time reached
(d) The highest ranking solution's fitness is reaching or has reached a plateau such that successive iteration no longer produces better results.
(e) Manual inspection
(f) Combinations of the above

Simple generational genetic algorithm pseudo code:

1. Choose initial population
2. Evaluate the fitness of each individual in the population
3. Repeat until termination: (time limit or sufficient fitness achieved)
    (1) select best-ranking individual to reproduce
    (2) Breed new generation through crossover and/or mutation and give birth to offspring
    (3) Evaluate the individual fitness of the offspring
    (4) Replace worst ranked part of population with offspring

Observations:

(a) Repeated fitness function evaluation for complex problems is often the most prohibitive and limiting segment of artificial evolutionary algorithms. Finding optimal solution to complex high dimensional, multimodal problems often requires very expensive fitness function evaluations. In real world problems such as structural optimization problems, one single function evaluation may require several hours to several days of complete simulation. Typical optimization method cannot deal with such a type of problem. In this case, it may be necessary to forgo an exact evaluation and use an approximated fitness that is computationally efficient. It is apparent that amalgamation of approximate models may be one of the most promising approaches to convincingly use EA to solve complex real life problems [12].

(b) The "better" is only in comparison to other solution. As a result, the stop criterion is not clear.

(c) GAs cannot effectively solve problems in which the only fitness measure is a single right/wrong measure (like decision problems), as there is no way to converge on the solution (no hill to climb). In these cases, a random search may find a solution as quickly as a GA. However, if the situation allows the success/failure trial to be repeated giving (possibly) different results, then the ratio of successes to failures provides a suitable fitness measure.

(d) Often, GAs can rapidly locate *good* solutions, even for difficult search spaces. The same is of course also true for evolution strategies and evolutionary programming.

(e) The implementation and evaluation of the fitness function is an important factor in the speed and efficiency of the algorithm.

(f) The advantage of software tools (like MATLAB) is consistent tuning, online or offline method, and it allows simulation before downloading. But the only disadvantage is that some cost and training are involved.

CHAPTER-4

SIMULATION OF SHP

The simulation of hydro plant has been implemented in MATLAB Simulink .The various models which were given in this chapter are modeled through MATLAB Simulink. MATLAB provides built in mathematical functions .These functions provide broad range of values. SIMULINK provides a graphical user interface for constructing block diagram models by drag and use .The modelling of various components has been done through SIMULINK. The various blocks which have been developed in SIMULINK have been given.

4.1 SIMULATION OF PENSTOCK AND TURBINE

Fig.4.1 is the simulation work for penstock and turbine. The position of the gate opening to know the power generated by the turbine .The modelling is discussed in chapter 2. The equations from equation 2.1 to 2.11 are used in modelling penstock and turbine torque output.

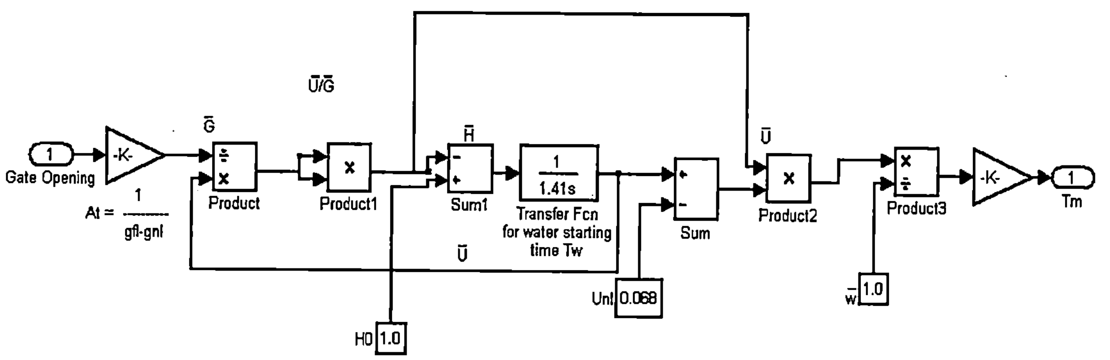

Figure 4.1 Penstock and Turbine Modelling in SIMULINK

In this simulation is to get the turbine output torque or power can be modeled in terms gate opening. For this per unit modeling has been done to get a relation between gate opening and output power. The various parameters that modeled to per unit are velocity of water, head, Mechanical power. In this simulated with the water constant used to get the power and losses are subtracted to get power generated by the turbine and with the speed division ,it will be torque generation which it will be useful to load model to get the loss(in the governing system .shown in fig .4.1

4.2 GOVERNING SYSTEMS

4.2.1 Simulation of Hydraulic Governing system

The basic function of a governor is to control speed and /or load. The primary speed/load control function involves feeding back speed error to control the gate position. In order to ensure satisfactory and stable parallel operation of multiple units, the speed governor is provided with a droop characteristic. Governor models shown in fig 4.2, has two types of drop characteristics.

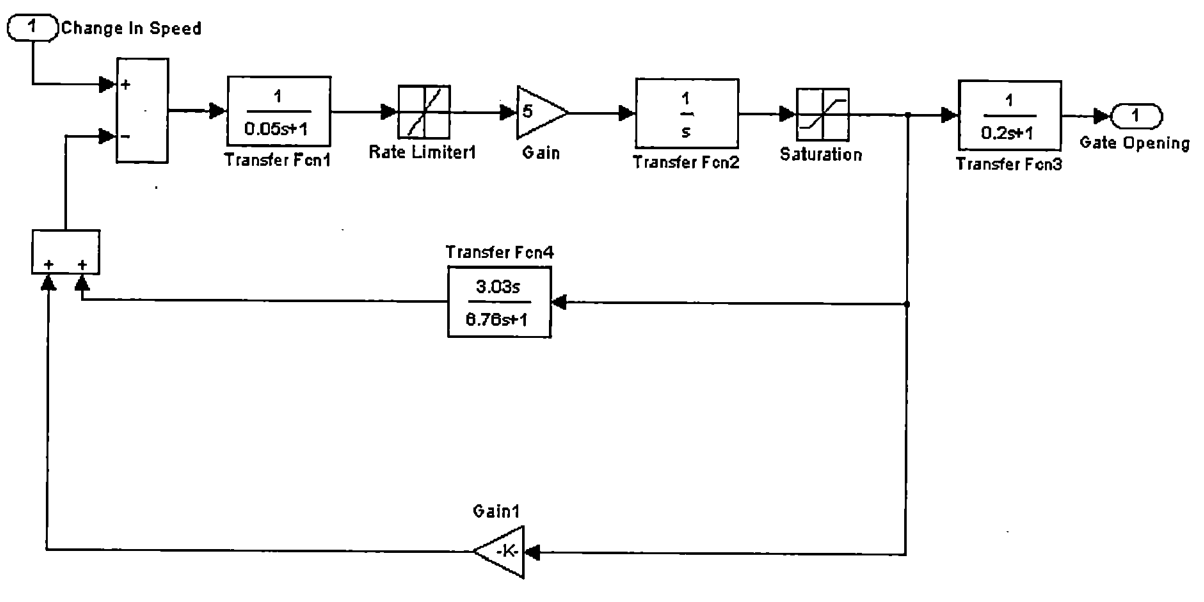

Figure 4.2 Hydraulic Governor in SIMULINK

Temporary droop determines the regulation under steady state conditions as frequency drop in percentage and transient response gives the fast frequency changes the control system provides higher regulation. Modelling of hydraulic governor is given in chapter2. The difference of the speed is given to the pilot valve of Servomotor and the gate opening of servomotor depend on the drop characteristics finally it opens for the servo motor. The variables are taken from reference [8].

4.2.2 Simulation of PID Governing System

In the Figure 4.3, the values of the PID get tuned with the GA. About this tuning is explained in Implementation of GA.

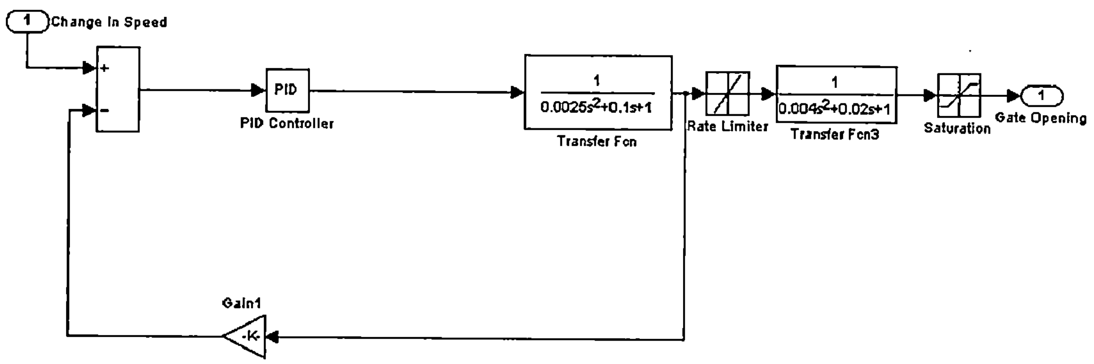

Figure 4.3 Block Diagram of PID Governor in SIMULINK

The error which is taken to PID and with the time constant T_A servo motor transfer function has been made and a rate limiter is used before pilot valve transfer function and servomotor transfer functions are created and it finally with the saturation gives the gate opening modeling.

4.2.2.1 Implementation of GA

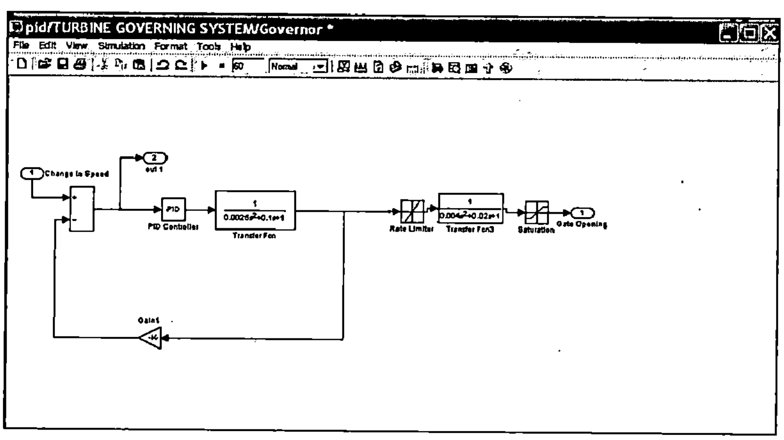

Figure 4.4 PID Governing System

From the Simulink, PID Simulink model can be brought to GA tool by using fitness function. In the fitness function number of variables to optimize or the values to be found out in the block is initialized in the functional block not to initialize any values.

The variables kp, ki, kd which are to be optimized are given in command window.

In the GA Tool box has to get the data or constrains have to give in the fields to optimize the PID values.

Required Data

(a) Fitness function

(b) Number of variables

(c) Population size

(d) Number of Iterations

(e) Initial range

Then click on start to run to find out PID values. It will run and finally The values will be displayed in the field called final points.

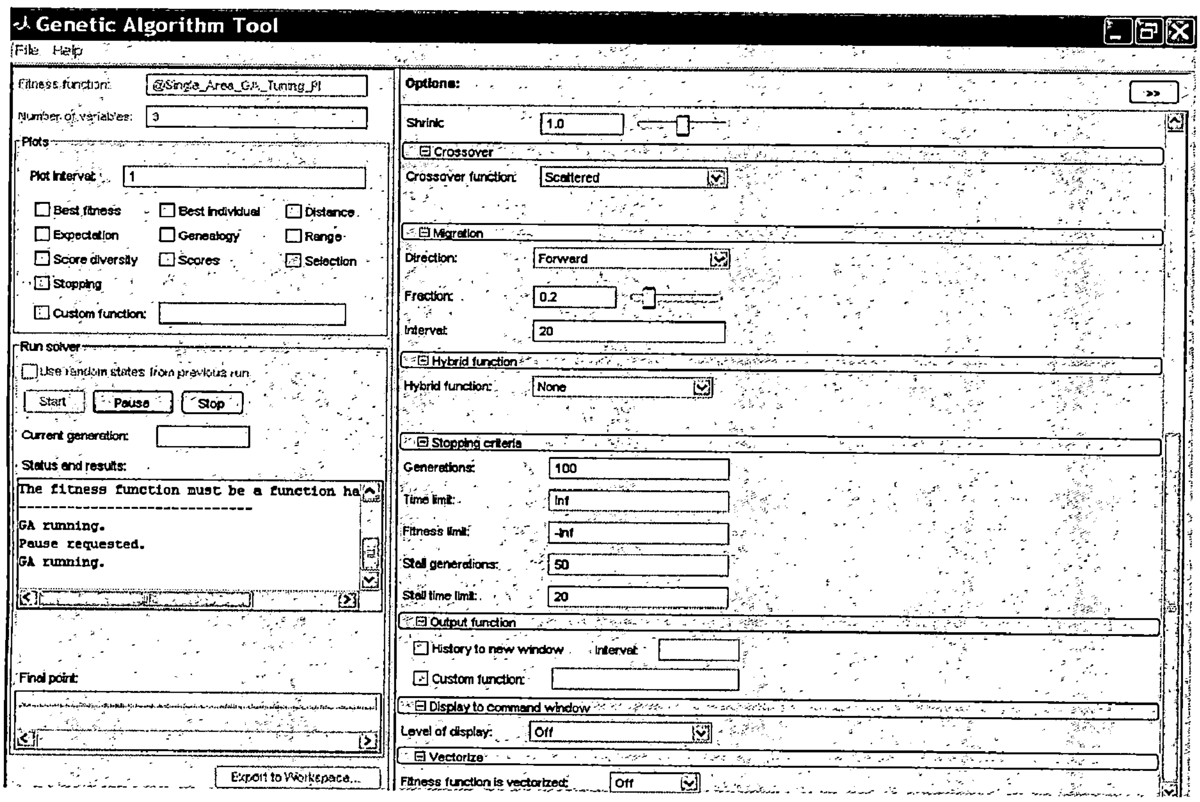

Figure.4.5 GA TOOL Box

4.2.3 Simulation of ANN Based Governing System

In the fig 4.6, the basic PID governing system included was replaced by ANN governor which is formed by taking error as the input vector and Gate as the target. The governor below shown was trained for full load.

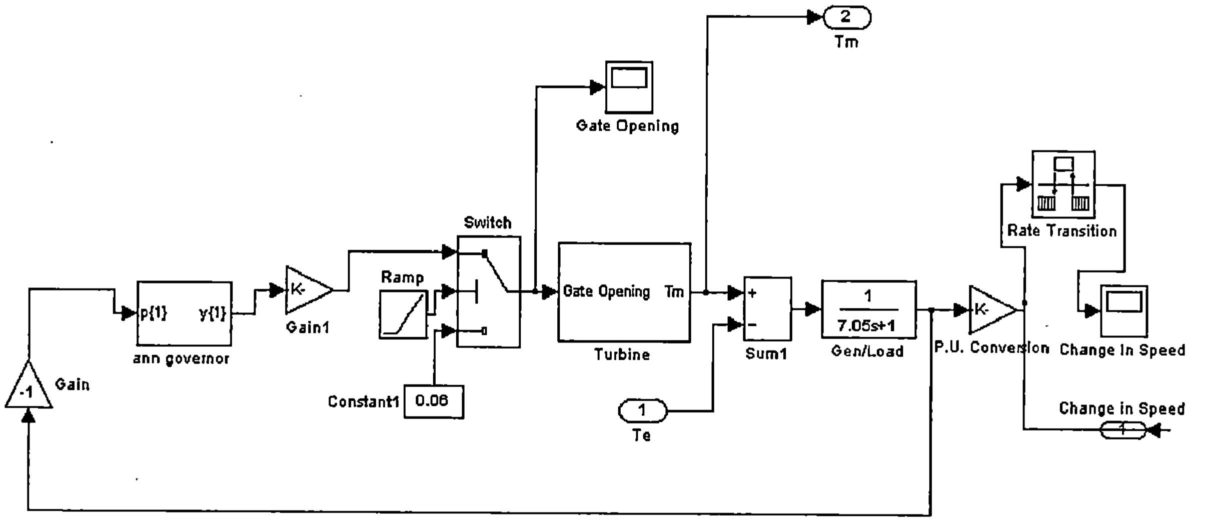

Figure 4.6 ANN based governing system in SIMULINK

In the ANN block, 2 layers are there, which it has 15 neurons in the hidden layer and 3 neurons in the output layer. The input data has been extracted as error and the Gate opening as the Target data .the data has been extracted from the workspace a 6000 row values have been taken because simulation had run through 60sec linspace command cuts each interval by 100.The weights are initialized with random values and the transfer function in the hidden layer as logsig and in the output layer it as the linear function is used. The functional net has been calculated for each neuron in hidden layer and the weights in the hidden layer will be updated .The weights are updated through back propagation algorithm. The weights get increased in each iteration. The weights of the hidden layer the error value are compared in each iteration and if the error is less than the prescribed value, the training gets stopped.

4.4 SIMULATION OF DC41A EXCITER

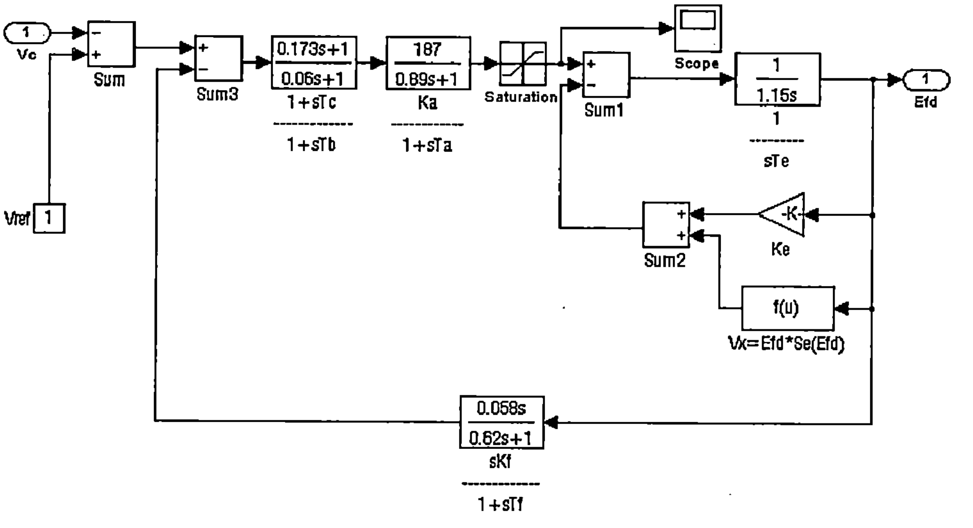

Figure 4.7 DC41A exciter in Simulink

The function of the exciter is to stabilize the voltage of the synchronous generator automatically when a disturbance occurs in the terminal voltage. the voltage of the generator is compare with reference voltage and the error passed and two transfer functions are created ,one is for the amplification and second to control the transient voltage (low pass filter) and the regulator limits are provided . An exponential function is created in the feedback of the circuit.

4.5 SIMULATION OF GOVERNING SYSTEM

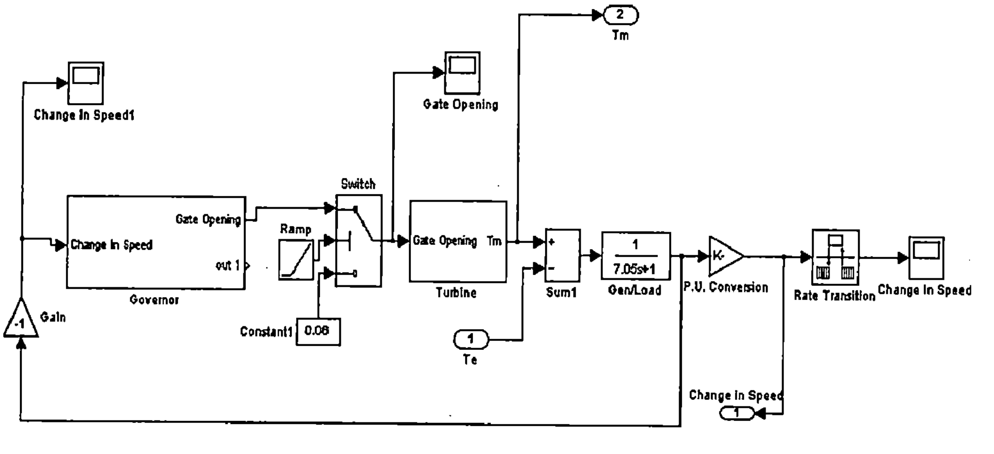

Figure 4.8 Simulation Diagram of Governing System

The function of the governor is to sense the speed changes have to accept the change with respect to load changes and change the position of gate opening and produces required torque with respect to acceleration or deceleration of torque and the explanation is given, why does speed change when a load change occurs and load model has been done and after again set it back to i/p to governor model that explained through the equations 2.15 to 2.17. This is modeled

Because to get the speed change has been modeled through load modeling and that one was converted to per unit form. In this present work, three types of governors were modeled they are Hydraulic, PID based and ANN. Output of the governor block is given to the penstock and turbine modeling block and this block is given load model to get the change in speed. Here a rate transition block is used for setting purposes in configuring parameters.

The exciter output and turbine output are connected to the generator to make the complete system and set the parameters of generator.

CHAPTER-5

DATA & RESULT

5.1 DATA

The model developed in Matlab is of a hydro power plant supplying power in isolated mode. The data for various components hydro power plant, which have been used for simulation purpose in this dissertation work, are given below: -

5.1.1 Penstock and Turbine [8]

U	=	7.26 m/s	U_{NL}	=	0.068 p.u.
L	=	450 m	A	=	1.54 m^2
Q	=	3.58	H_r	=	75 m
H_o	=	1 p.u.	g_{FL}	=	0.94
g_{NL}	=	0.06	A_t	=	1.136
T_w	=	1.41 s	P_t	=	2252.4KW
K_D	=	1			

5.1.2 Synchronous Generator [8]

Rating = 2000KVA   Poles = 3

Voltage rating = 11kV   Frequency = 50Hz

R_a = 0.0006 p.u.

L_l = 0.150 p.u. L_d = 1.81 p.u.

L_q = 1.81 p.u. L'_d = 0.30 p.u.

L'_q = 0.65 p.u. L''_d = 0.23 p.u.

L''_q = 0.23 p.u. T'_{do} = 8.00 sec

T'_{qo} = 1.00 sec T''_{do} = 0.03 sec

T''_{qo} = 0.07 sec

5.1.3 Electro Hydraulic Governor [8]

T_P	=	0.05 s	K_S	=	5
T_G	=	0.2s	R_P	=	0.001
R_T	=	0.448	T_R	=	6.76 s
$R_{max\ open}$	=	0.16 pu/s	$R_{max\ close}$	=	0.16 pu/s

5.1.4 PID Governor [17]

T_A	=	0.05 s	T_C	=	0.02 s
T_D	=	0.02 s	R_P	=	0.04 s
$R_{max\ open}$	=	0.16 pu/s	$R_{max\ close}$	=	0.16 pu/s

5.1.5 DC1A Exciter [8]

T_c	=	0.173	T_b	=	0.06
V_{RMAX}	=	1.7	V_{RMIN}	=	-1.7
K_A	=	187	T_A	=	0.89
T_E	=	1.15	K_F	=	0.058
T_F	=	0.62			

5 .1.6 For ANN

Pattern	=	Change in speed
Target	=	Gate opening
Type	=	Fed forward Network
Algorithm	=	Levenberg Marquardt (Backpropagation)
No. of hidden layer	=	2
No. of output layer	=	1
No. of input layer	=	1

5.2 RESULTS

5.2.1 Gate opening for Different Governors

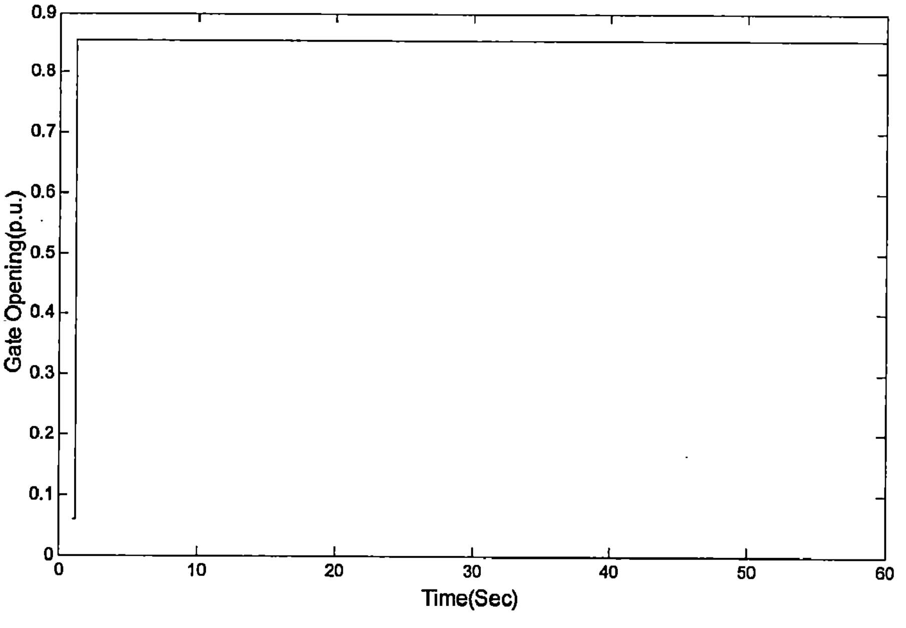

Figure.5.1.Gate Opening Vs Time (ANN)

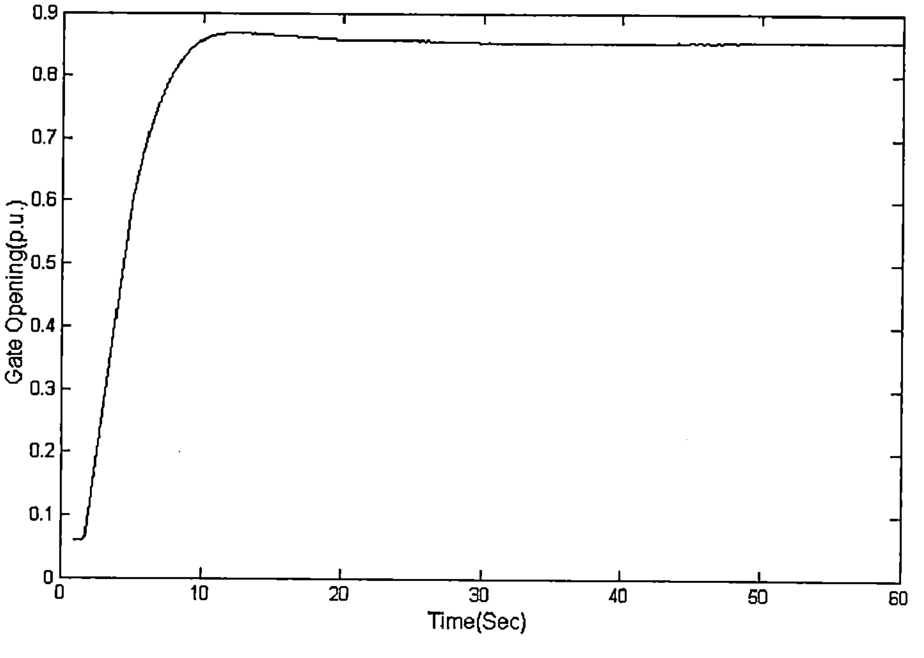

Figure 5.2.Gate Opening Vs Time (PID)

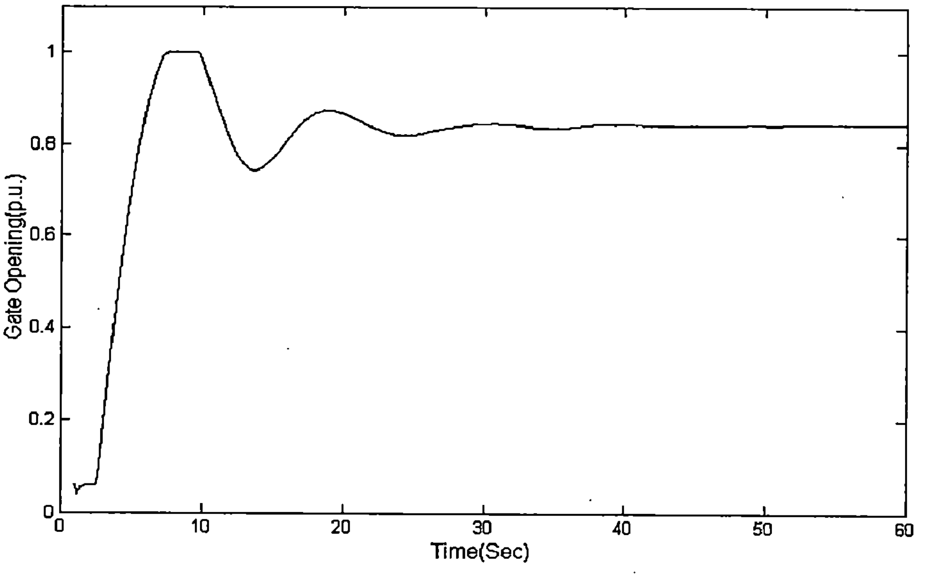

Figure 5.3.Gate Opening Vs Time (Hydraulic governor)

5.2.2 Speed vs. Time for Different Governors

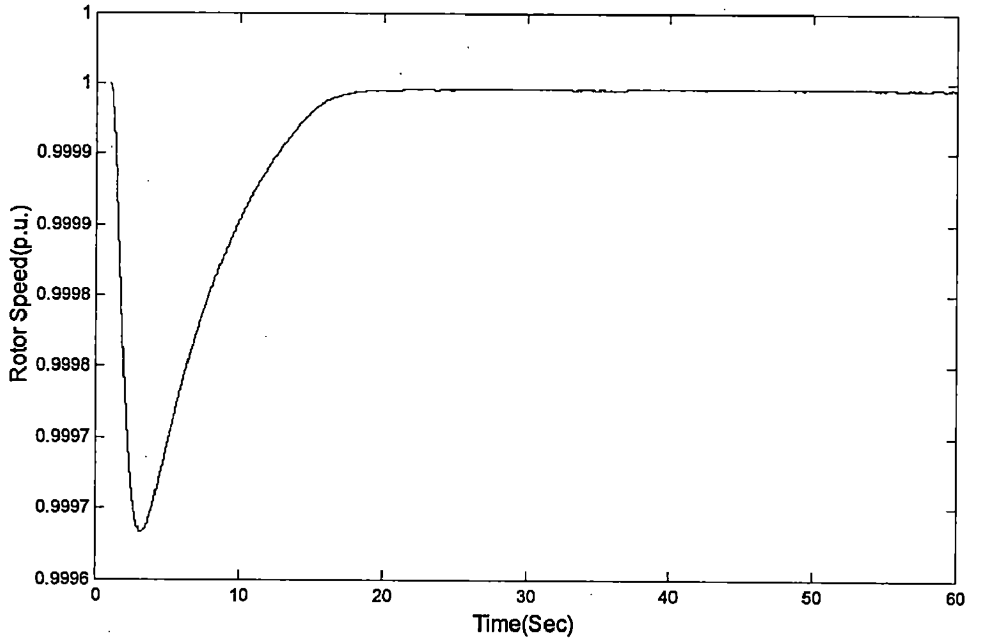

Figure 5.4 Rotor Speed vs Time (ANN)

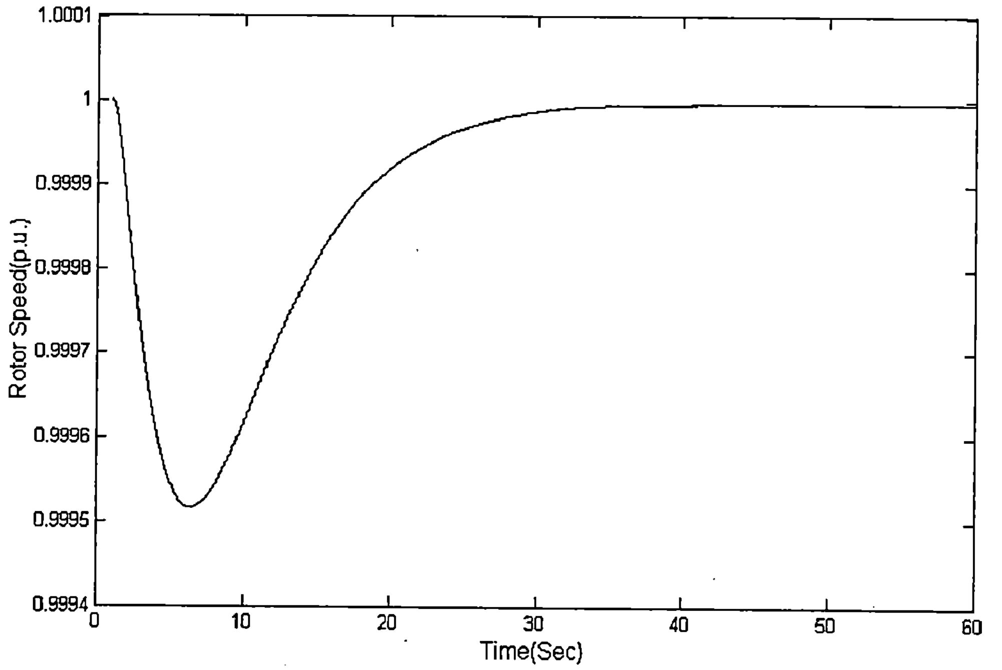

Figure 5.5. Rotor Speed vs Time (PID)

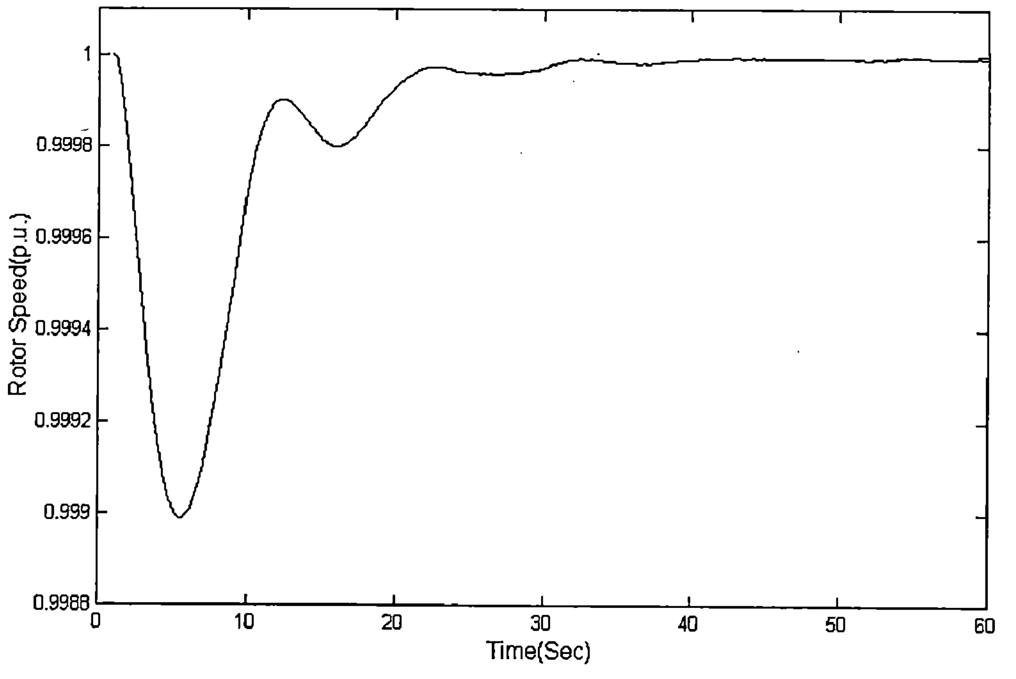

Figure 5.6. Rotor Speed vs Time (Hydraulic governor)

5.3 DISCUSSION

From the above results, in the hydraulic governor when the load changes, the flyball moves and will increase the water input to turbine and hence, it will be beyond the final settled value. Therefore, overshoot occurs and flyball moves to and fro. This gets oscillations and gets rapidity which gets damaged quickly due to aging.

The gate opening in the ANN is tuned, giving faster and smooth response. The hydraulic governor which is totally dependent upon the water-time constant and error takes time to settle down. In the case of PID governor, the values of Integral and Derivative additional components present come to steady state error and fastening response will improve. Hence, the governor life improves comparatively than the normal governor. The ANN network is trained through the PID as it is dependent upon the training provided. By the above responses, ANN based governor gives much improvement in both steady state and also in the faster response. Due to Inertia the speed instantly doesn't rise. Due to oscillations present in the normal hydraulic governor, the frequency may be fluctuated disturbing the network. As SHP is one of the decentralized generations, governor action plays an important rule.

To construct a new type of governing system, these intelligent controllers depend upon the fundamental controllers. By using GA tool one can set the parameter values easily.

CHAPTER 6

CONCLUSIONS

6.1 CONCLUSIONS

Following are the conclusions that can be drawn after simulation:

1. Using GA, PID values can be tuned if any changes in the component of hydro power plant and normal off line tuning can be replaced with which optimal PID values can be found out.
2. ANN response is better than PID response in view of Settling time and overshoot.
3. Smoothness of governor operation can be improved which will decrease mechanical wear.
4. Conventional governor can be replaced with PID and Artificial Intelligence controlled Governors to improve the performance.

6.2 FUTURE SCOPE:

The present work can be carried out as extension to the following.

1. For different types of hydro power plants like run-off river and with other excitation models (type-AC1A and type-ST1A.) can be applied.
2. Online tuning of Genetic algorithm can be applied for tuning to set the PID values.
3. The work has done with static load can be done also for Dynamic loads and also for faults.
4. Any values to be optimized in the total system can be done by GA tuning.

REFERENCES

[1] Ministry of Power, Government of India. http://www.powermin.nic.in/

[2] International Course on "Small Hydropower Development" by Alternate Hydro Energy Centre, Indian Institute of Technology Roorkee, February 2004

[3] "International Association for Small - News letter", Delhi, India, Vol.9-10, No.1, January 2008,

[4] Ministry of New and Renewable Energy, Government of India. http://www.mnes.nic.in/

[5] Naidu B.S.K. "Small hydro in India: environment friendly alternative energy source" TERI Information Monitor on Environmental Science ,Vol-1, 1997.

[6] http///www.en.wikipedia.org/wiki/small-hydro

[7] Arun Kumar, "Overview of Small Hydropower Development in India", Proceedings of International Himalayan Small Hydropower Summit (HSHS), International bulletin by AHEC and Government of Uttarkhand, Dehradun (India), pp.6., October 2006.

[8] Kundur P., "Power System Stability and Control", McGraw Hill Inc., New York.

[9] Sanathanan C. K., "Accurate low order model for hydraulic turbine –penstock", IEEE transaction on energy conversion, Vol. EC-2, No. 2, June 1987.

[10] Padiyar K.R., "Power System Dynamics- Control and Stability", Interline Publishing Pvt. Ltd., Banglore.,2nd Edition

[11] Jin Jiang,"Design of an Optimal Robust Governor for Hydraulic Turbine Generating Units", IEEE transaction on energy conversion, Vol. 10. No. 1 March 1995, pp188-194.

[12] Dionisio S. Pereira,"Genetic Algorithm Based System Identification and PID Tuning for Optimum Adaptive Control "International Conference on Advanced Intelligent Mechatronics, Proceedings of the 2005 IEEE/ASME

[13] Hanmandlu M., Himani Goyal, D.P.Kothari. ,"An Advanced Control Scheme for Micro Hydro Power Plants" Power Electronics, Drives and Energy Systems,. PEDES '06. International Conference, pp.1-7,2006

[14] Zurada J. M., "Introduction to artificial neural systems", Mumbai, Jaico Publishing House, 1997

[15] Martin Hagan.T and Mohammad Menhaj B. "Training Feed Forward Networks with the Marquardt Algorithm". Ieee Transactions On Neural Networks, Vol. 5, No. 6, November 1994

[16] Sivanandam.S.N, Sumathi.S, Deepa S N., "Introduction to Neural Networks Using MATLAB 0"Tata McGraw Hill company Limited, New Delhi.

[17] Working Group on Prime Mover and Energy Supply Models for System Dynamic Performance Studies, "Hydraulic Turbine and Turbine Control Models for System Dynamic Studies", IEEE Transaction on Power Systems, Vol 7, No.1, February 1992.

[18] "IEEE Recommended Practice for Excitation System Models for Power System Stability Studies", IEEE STD 421.5-1992.

[19] "IEEE Guide for Synchronous Generator Modeling Practices in Stability Analyses" IEEE STD 1110-1991.

[20] http://www.neurosolutions.com