EVALUATION OF REAL-TIME FLOOD MONITORING AND EARLY WARNING SYSTEM USING ARTIFICIAL NEURAL NETWORK

by
Dimson I.C.[1], Obianyo O.R.[2], Onyeyili T.C.[3]
1,3: Dept. of Electronic and Computer Engineering, Nnamdi Azikiwe University, Awka, Nigeria.
2: Dept. of Computer Engineering, Madonna University of Nigeria, Akpugo Campus, Enugu State, Nigeria.

ABSTRACT
This research paper analyzes real-time flood detection and monitoring using Artificial Intelligence. The research was motivated by the need to develop an intelligent system that can pre-warn the inhabitants of flood prone localities on the likelihood of flooding at a specific time to initiate a process of safeguarding their lives, properties and evacuation to a safe location on time. A computer-aided software engineering methodological approach was deployed in achieving the aim and objectives of this work through characterization of a PID based conventional flood detection and monitoring system to establish the parameters that improve flood detection and monitoring system. Using the established parameters, a nonlinear model of Ugwuaji River in Enugu State (Nigeria) was developed. The sensing device that is meant to acquire real-time data from the environment was designed using pressure as the sensing element. A nonlinear model predictive control system that utilizes previous process control behaviour to foretell the future response of the system was modeled and implemented in Simulink. The developed mathematical models were transformed into a discrete form using Laplace transform to establish the transfer functions for the development of the Simulink model for real-time simulation. The model predictive control system network was trained offline using BFGS quasi-Newton Back Propagation Algorithm during simulation. Simulation results show that the proposed system achieved regression of 1 after several iterations. Results also show that the proposed system responded very fast to flood signal within 8.44s seconds as against 22 seconds achieved by the conventional PID sensor. The percentage increase in the new system performance is 21.6%. The comparative delay time between detection and prediction is 24.6s for the characterized sensor and 2.84 for the new sensor, the percentage improvement in the delay performance is therefore 88.4%.

Keywords/Phrases: Flooding, Artificial intelligence, Artificial Neural Network, Simulink.

I. INTRODUCTION

Due to high impact of flooding in the ecosystem of a locality, it is vital to analyze the flooding situation and develop a means of pre-warning the stakeholders so that they will not be taken unawares and as take a timely precaution. According to Amy Mani (2016), the causes of floods are most frequently extremely heavy rains or sudden melting snow combined with significantly reduced ability, even inability of an area to retain rainwater (due to damage to the country – e.g. dried swamps or drained of agricultural land). In African regions especially West Africa, snow is not the case but extreme or torrential rainfalls. Deforestation by those trading on timbers, urbanization as a result of development, and reduction of wetlands cause a decrease in accumulation of water in the basin and increase the runoff. Urbanization has a negative impact on the risk of flooding by increasing impervious surfaces (roofs, roads, sidewalks, parking lots, etc.).

In the course of mitigating the adverse effects of flooding in many localities, many approaches have been proposed by different researchers in recent times. Most approaches proposed are based on pre-warning the locals on the likelihood of flooding at a particular time so that appropriate measures can be put in place. Flood detection and monitoring is a non-linear situation that can only be handled using a non-linear approach (Anthone Vinicio and Satoru Oishi, 2018). Some of the existing approaches are based on linear control systems such as Proportional Integral (PI), Proportional Integral Derivative (PID) etc. Apparently, since what is being detected and monitored is non-linear in nature, it is paramount that a non-linear controller such as an artificial intelligence system is deployed in order to achieve detection and monitoring in real-time.

Artificial Intelligence (AI) is a branch of science which deals with helping machines find solutions to complex problems in a more human-like fashion (Azid I.S. and Sharma B., 2022). This generally involves borrowing characteristics from human intelligence and applying them as algorithms in a computer-friendly way. Putting it in another way, it is simply mimicking human behaviour. So many algorithms have been developed using different control techniques such as Self-tuning, Model Reference Adaptive Control (MRAC), Gain scheduling and Dual control. These techniques have their advantages and disadvantages. One of their shortcomings includes slow response time. Since this research is proposing a system that will foresee and respond in real-time, then it is vital to deploy a control technique that is adaptive such as Artificial Neural Network (ANN).

Artificial Neural Network (ANN) has found many applications in the modelling of many control processes (Yawut C., 2021). It is the choice of many designers because they can learn a nonlinear input-output mapping from process data. So, because of its learning ability, ANN, till the moment, remains the alternative and probably the most effective way for modelling. Perhaps, it has been emphasized that for successful NN modelling and control, complete controllability and observability of the process must be assumed. More so, more needs to be done in order to achieve more progress in nonlinear control theory. Invariably, model predictive control of ANN will be deployed in this proposed real-time flood detection and monitoring using artificial intelligence.

The computer-controlled algorithm which utilizes previous process control behaviour to foretell future response of plants is the whole essence of Model Predictive Control (MPC) (Basha Elizabeth et al., 2017). In other words, MPC is one approach for acquiring a feedback controller synthesis from knowledge of open-loop controllers. It measures the current process state and afterwards, computes very swiftly, the open-loop control function. The first component to this function is then used for the duration of a short interval, after which a new value of the function is computed for this measurement (Isizoh A.N et al., 2014).

It can be understood that shortcomings of the existing flood detection and monitoring systems include: Time response, Complexity of the system, High cost of implementation and maintenance.

Therefore, it is important that a system that is intelligent in the sense that it can adapt to the environment and has the capabilities of handling nonlinearity characteristics that are associated with the flood to be deployed in modeling such systems (Hopson T. M. and Webster P. J., 2016). For this reason, this work is proposing a model predictive control system of artificial neural network in modeling and simulation of real time flood detection and monitoring system to reduce the complexity and the cost of the system and ultimately optimizing the system's time of response.

II. LITERATURE SURVEY

2.1 Artificial Neural Network

Artificial Neural Network is an interconnected massive parallel computational models, units or nodes, whose functionality mimic the artificial neural network to process information from the input to the output using the connection strength (weight) obtained by adaptation or learning from a set of training patterns (Shen C. P., 2020). The model description of ANN process is shown in Figure 1.

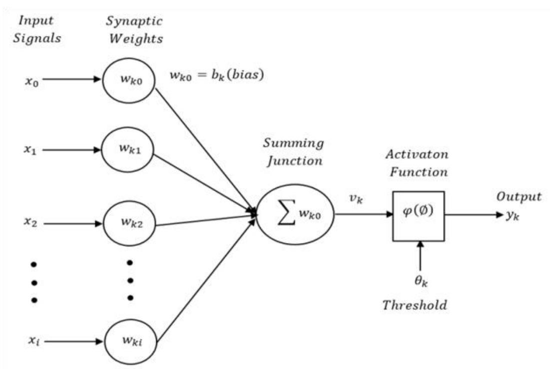

Figure 1: Model Description of an Artificial Neuron

The neuron is a unit of computation that reads the inputs given, processes the input and gives the output in processed form (Victor Seal et al., 2022). To get the output of the Artificial Neuron from the activation function, we compute the weighted sum of the inputs as: $v_k = \sum_{i=1}^{N} w_{ki}x_i$ (1)

Where:

x_i is the neuron's input from the training dataset.

w_{ki} is the corresponding weight to the input x_i.

The neuron's output is obtained by sending the weighted sum v_k as the activation function φinput that resolves the output of the specific neuron. $y_k = \varphi(v_k)$. A step function with threshold t can be used to

express a simple activation as: $\varphi(x) = \begin{cases} 1\, if x \ \geq t \\ 0\ if x < t \end{cases}$ (2)

However, bias is most time used instead of a threshold in the network to learn optimal threshold by itself by adding $x_o = 1$ to every neuron in the network. The step activation function for the bias becomes:

$$\varphi(x) = \begin{cases} 1\, if\ x \ \geq 0 \\ 0\ if x < 0 \end{cases} \quad (3)$$

For the learning process to speed up and also adaptive learning capacity, multiple neurons are were used as a multi layered network of neurons formed by feeding the output of one neuron to the input of another neuron as shown in Figure 2.

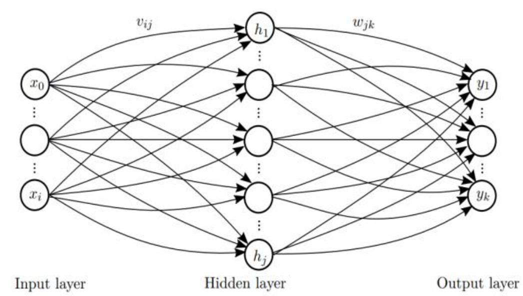

Fig. 2: Basic Structure of a Multilayer Artificial Neuron.

The layers between the input and output layers are termed hidden layers. Each layer of the multilayer network is made up of a bunch of neuron nodes with each input feed with the class of the data set. The neurons are connected by a link that has a weight which represents the connection strength between each interconnected neurons. In Figure 2, the w_{ij}^{l} denotes the weight for a link between unit j in layer l and unit i in layer $l+1$. Also b_i^lrepresents the bias of the unit i in layer $l+1$. For any neural network, the associated parameters inside it are expressed as a function of the weight and the bias of the neurons as:

$$(w, b) = \left(w^1 b^1, w^2 b^2, w^3 b^3, \ldots\right) \quad (4)$$

The components of equation (4) can be written in the form of $w^1 \in \mathbb{R}^{3\times3}$ and $w^1 \in \mathbb{R}^{1\times3}$.

Let the activation of unit i in layer $l$be represented by a_i^l, then the input for the layer labelled as L_1 we have $a_i^1 = x_i$ for the ith input of the whole network. Other layers are given by $a_i^l = f\left(z_i^l\right)$, where z_i^l is the total weighted sum of the inputs to unit i in layer l in addition to the bias term. The activation function of a nine-input ANN with bias can be computed as:

$$a_{\mathrm{n}}^2 = f\left(w_{\mathrm{n}1}^1 x_1 + w_{\mathrm{n}2}^1 x_2 + w_{\mathrm{n}3}^1 x_3 \ldots\ldots\ldots\ldots + b_{\mathrm{n}}^1\right) \quad (5)$$

Where n is the number of input classes from the dataset.
The equation can be re-written as:

$$h_{w,b}(x) = a_1^3 = f\left(w_{1\mathrm{n}}^2 a_{\mathrm{n}}^2 + w_{1\mathrm{n}}^2 a_{\mathrm{n}}^2 + w_{1\mathrm{n}}^2 a_{\mathrm{n}}^2 + b_{\mathrm{n}}^1\right) \quad (6)$$

Where $h_{w,b}(x)$ is a real number representing the output of the ANN, n is the number of inputs from the dataset. The activation function $f(\mathrm{Zn})$ can be applied to vectors in element-wise as $f([z_1, z_2, z_3, , , , , , , \mathrm{Zn}]) = [f(z_1), f(z_2), f(z_3)$ (Zn),]. Therefore equation (6) can be written as:

$$h_{w,b}(x) = a^3 = f\left(z^3\right) \quad (7)$$

So, for any given layer l with activation a^l, the activation a^{l+1} of the next layer $l+1$ is obtained as:

$$z^{l+} = w^l a^l + b^l, a^{l+1} = f\left(z^{l+1}\right) \quad (8)$$

When the computation of the signal moves from the input to the output of the ANN (feed forward network), it is called Forward Propagation. To make the network recurrent, the ANN could have a closed-loop back to itself from a neuron. Also, when an ANN has every neuron in each layer connected to the neurons in the next layer, it is called a fully-connected network.
A nonlinear activation function is used in multilayer networks, which is why it can solve nonlinear issues. A common activation function in ANN is the sigmoid functions which are like the logistic function as shown in equation (9). $\sigma(z) = \frac{1}{1+e^{-z}}$ (9)

Where z is the activation function, and when z is large, then e^{-z} tends to zero (0), so $\sigma(z) = 1$. Conversely, if z is a small or very large or very large negative number, then e^{-z} tends to one (1), so $\sigma(z) = 0$.

2.2 Review of Past Related Literatures

Anthone Vinicio and Satoru Oishi, 2018 presented a comprehensive study of the Flood analysis and prediction using Geographical Information system (GIS) i.e. they are using an Arc GIS simulation tool to identify pre and post disaster Flood risk analysis and an Ad hoc wireless Sensor Network Architecture. A model is proposed for Flood risk analysis and prediction for calculating the impact of Flood damage in disaster hit regions, however the response time of the system is poor with over 20 sec.

Azid I.S. and Sharma B. (2022) proposed a model which is based on Open Geospatial Consortium's (OGC) Sensor Web Enablement (SWE) standards, that collects data to be shared in an interoperable and flexible manner. A Spatial Data Infrastructure (SDI), geospatial software platforms which were used to manage the environmental risks, however this work employs geographical image information system for its data source, hence the implementation will be expensive.
In Basha Elizabeth et al. (2017), they employed Sacramento Soil Moisture Accounting (SAC-SMA) for easy detection of flood, and it is very efficient but expensive to implement and maintain. Although several commercial models of Flood warning systems are currently available, many of them are either expensive to implement and maintain or unable to identify water levels early enough and alert the residence. In fact, some water detection devices are triggered by a single event and their alerts are monitored by agencies responsible for Flood prevention. However, it is often too late for people to protect their belongings and evacuate to safe ground if their Flood warning appliance is solely activated by a certain water level without a pre-flood warning. Therefore, this thesis creates a low-cost artificial

intelligence-based flood detection system that senses rising water and pressure in real-time and determines any potential flash Floods.

Isizoh A.N et al. (2014) researched on a real-time model which is based on an alarm system that operates in such a way that when the plain area is flooded, then the alarm will start blowing. The technique cannot be used for a real live flood detection system because of its unreliability.

Hopson T. M. and Webster P. J. (2016) researched on a real-time model which is based on an Alarm System that means when the plain area is Flooded then the alarm will start blowing. The technique cannot be used for a real live flood detection system because of its unreliability.

Hughes D. et al. (2018) implemented the sensor network in Honduras for early detection of Flood & alert the community is done. They have analyzed the significance of sensor networks in developing countries, sensor networks for flood detection and the available current operational systems for Flood detection. This work cannot be justified as the response time was not presented.

Several models have been proposed and also implemented but the necessity of another model arises when we consider flood depending upon the flushing style at the barrage. At the time of flooding, large volume of discharged water is accumulated at the barrage. When the water holding capacity of the barrage exceeds more than the normal level, the excess volume is flushed out. When the excess volume of water is flushed at a sudden then the water flows to the basin of the river and floods the plain area. So, if a model could predict the upcoming amount of volume and the time to reach the barrage, it can be ready to receive the excess volume without being over flooded. Barrage can flush out the water in a well and stipulated manner so that flooding due to flush out of the excess volume of water suddenly will not occur. Deploying the wireless sensor network, a cost-effective model is conceived which could be used by developing or poor country, this adds to the advantages of the proposed model. However, there is a better need for an intelligent system that can monitor and respond to future non-linear river performance.

III PROPOSED METHODODOLOGY AND DESIGN

It will be good to study and characterize an existing flood monitoring and detection system, considering necessary parameters and then design a new system using an artificial intelligence technique to help optimize the response time of the flood. This will be accomplished using the necessary mathematical representations and implemented using a high-level programming language.

3.1 System Design

This section discusses the mathematical models of the respective parameters which are employed for the development of the proposed system. The non-linear model of the river will be developed, then the sensor will be designed considering the dynamic nature of flood in real-time, the calibration of the river to determine the height of the water level as shown in Figure 3; then the sensor designed will be improved using artificial intelligence technique, however this time a better technique different from the characterized will be employed. This sensor alongside other engineering devices will be employed to build the proposed system with improved performance, and real-time indication of a flood.

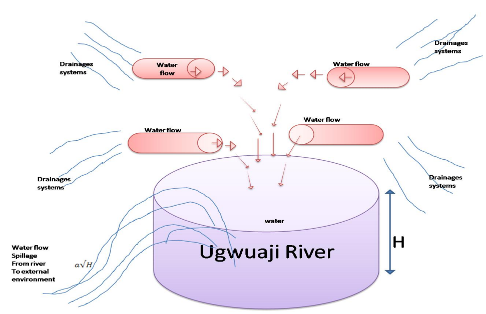

Figure 3: Proposed River and Water Flow from other drainage

3.2 Nonlinear River Model

The architecture in figure 3 models the river dynamics, showing how water from various drainage systems flows into the river proportional to pressure P. The river when overflowed spilled through the environment at a rate that is proportional to the square root of the water height, H, of the river. The presence of the square root in the water flow rate makes the river nonlinear. This is represented using the differential equation in equation (10).

$$\frac{d}{dt}\text{vol} = A\ \frac{dH}{dt} = bP - a\sqrt{H} \qquad (10)$$

Where H is the river height
V is the volume of the water
F is the applied Pressure
A is the cross-sectional area of the river
b is the flow rate constant into the river
a is the flow rate constant out of the river
The transfer function of the river base is presented using the Simulink model in figure 4.

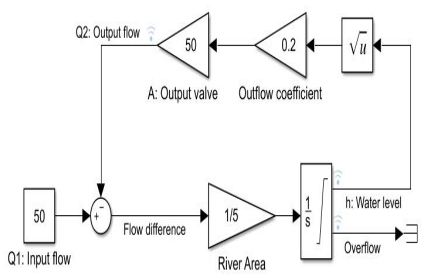

Figure 4: Simulink Model of the River

3.3 Sensor Model

In designing a flood sensor, pressure is very important to consider as a key factor in the system design, this is to help differentiate flood from erosion. The two natural phenomena involve water spillage; however, that of flood involves high pressure (Hopson T. M. and Webster P. J., 2016).

The sensing device is designed using a logic gate which considers water and pressure as the input devices. A model of the sensing device is developed using the block diagram in Figure 5.

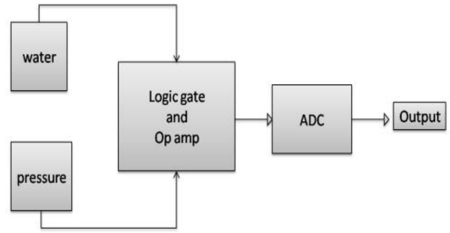

Figure 5: The Sensing Block Diagram

The Flowchart of the Water Sensing System is shown in Figure 6.

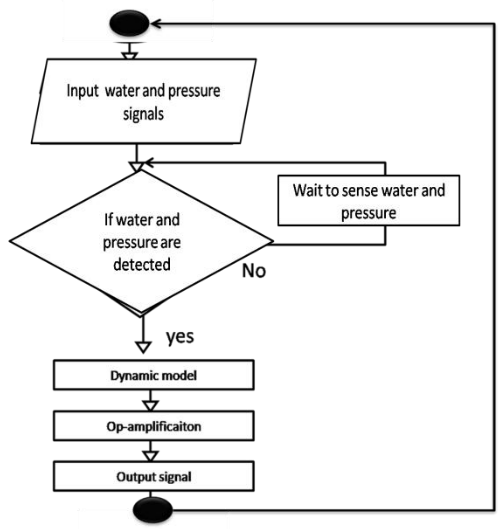

Figure 6: Flowchart of the Water Sensing System

3.4 Sensor Calibration

This is the linear relationship between the sensor output and the height of the river. This is determined by considering the volume at varying height (meters) and then measuring the sensor differential output Y, and V is volume as shown in equation (11):

$$Y = V_{water} + V_{pressure} \qquad (11)$$

Then sensor is then calibrated using the linear relationship which considers the change in the two parameters as shown in the equation (12):

$$\Delta V_{sensor} = \frac{\Delta Y}{x} \text{ x Height} \qquad (12)$$

The linear relationship can be employed to find the sensor predicted output ΔVsensor at the new height. The logic circuitry in Figure 7 presents the internal logic architecture of the sensor with the Truth Table in Table 1.

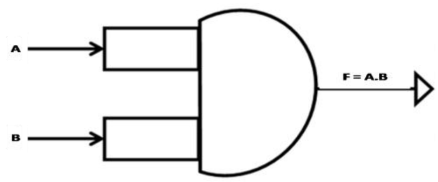

Figure 7: Logic Circuit for the Sensor Design

Table 1: Digital Logic AND Gate truth table

A	B	OUTPUT
0	0	0
0	1	0
1	0	0
1	1	1

3.5 Nonlinear Neuro Model Predictive Control (NNMPC)

This is an artificial intelligence technique that employs a control strategy adopting the model of the system (plant) in order to make next prediction on which an optimal input sequence is determined, so as to minimize an objective function not neglecting Constraints. The basic components for the process are:

a) The process model which combines the non-linear state space rainfall model and time series rainfall input in order to predict the future output within a predetermined (objective function) volume and height.

b) The objective function is minimized taking into account constraints on the input and output as a quadratic function, trying to minimize the deviation of the water level with the reference level and the rate of increment in volume (in this case with artificial neural network).

c) Model of the neuro predictive Controller (NPC)

d) Training of the plant with the NPC to obtained an improved neuro sensor

3.6 System Identification

This process identified the nonlinear flood model of the logic output (plant) as a nonlinear auto regressive model using the structure in Figure 8.

$$y(k+d) = N(y(k),y(k-1),\ldots,y(k-n+1),u(k),u(k-1),\ldots\ldots u(k-n+1)) \quad (13)$$

Where u(k) is the feature vectors inputs, N is the non-linear slip force, and y(k) is the system output as shown using the neural network architecture in Figure 8.

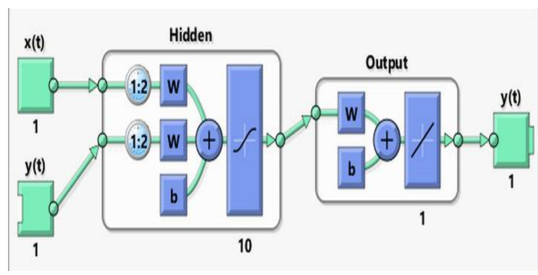

Figure 8: Neural Network model

3.7 The Training Dataset

To train the neural network, data of flood was collected from the Nigerian meteorological agency (NIMAX). The data collected was uploaded to the neural network tool which uses its activation function tool to extract the features and then learn the patterns for predict future fold problems using a process called training.

3.8 Neural Network Training

Considering the nature of the dynamic model identified, the BFGS quasi-Newton back propagation algorithm was used to train the model. The algorithm was used to calculate derivative of performance with respect to the weight and bias variables x. each variable is adjusted according to the following

$$X = X + a*dx \quad (14)$$

Where dx is the search direction, the parameters a is selected to minimize the performance along the search direction. The line search function is used to determine the minimum point.

The first search direction is the negative of the gradient of performance. In succeeding iterations the search direction is computed according to the following formula:

$$dX = -Hi\backslash gX \quad (15)$$

Where gX is the gradient and Hi is an approximate Hessian matrix. The training stops when any of these conditions occurs:

a) The maximum number of epochs (repetitions) is reached.
b) The maximum amount of time is exceeded.
c) Performance is minimized to the goal.
d) Precision problems have occurred in the matrix inversion.

The Flowchart for ANN Training is shown in Figure 9.

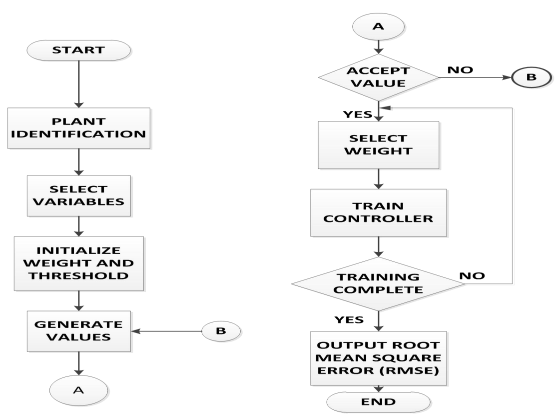

Figure 9: Flowchart for ANN Training

In training the network, the aim is to achieve optimal number of hidden layer neurons and also the learning parameter. So, through training of different combination of hidden layer neurons and the learning parameter, the optimal number of hidden layer neurons and the learning parameter were obtained. The following parameters as shown in Table 2 were used.

Table 2: Neural Network Parameters

Maximum number of epoch to train	100
Epoch between display	25
Maximum time to train in sec	Infinity
Maximum validation failure	5
Scale factor for tolerance	0.001
Scale factor for step size	0.1
Initial step size	0.01
Minimum performance gradient	1e-6
Cost horizon	7
Control horizon	2
Control weighting factor	0.05
Search parameter	0.001

From the training process of the neural network, the reference model of the input plant was achieved as shown in equation (16), and also the plant output was observed as shown in Figure 10.

$$\hat{y}(k+d)=f(y(k),y(k-1),\dots,y(k-n+1),u(k-1),\dots,u(k-m+1))+g(y(k),y(k-1),\dots,y(k-n+1),u(k-1),\dots,u(k-m+1))\cdot u(k) \quad \dots\dots\dots\dots\dots\dots \quad (16)$$

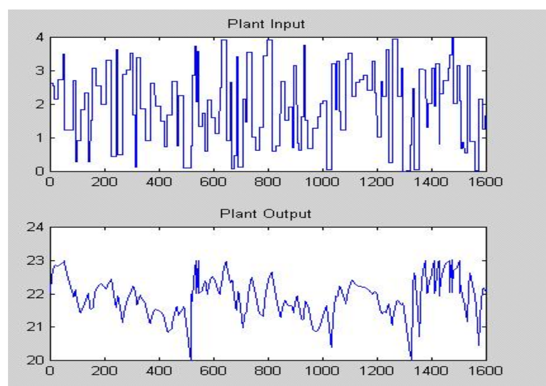

Figure 10: System Identification with ANN

Now that the approximate flood model has been identified using the ANN via the training algorithm. This will be used to develop the neuro controller structure as shown:

$$u(k+1) = \frac{yr(k+d) - f[y(k),\dots,y(k-n+1), u(k), \dots, u(k-n+1)]}{g[y(k),\dots, y(k-n+1), u(k),\dots, u(k-n+1)]} \quad (17)$$

The model of the neuro controller equation (3.8), is developed as the flood controller using the

Flowchart in Figure 9, this model is classified with the approximate flood model to detect flood and indicated alarm (see training parameters in Table 2).

3.9 Neuro Predictive Controller (NPS)

The predictive control technique is based on the plant response over specified time horizon. The predictions are numerical optimization programs to determine the control signal that minimizes the following criterion over the specified horizon as shown in equation (18):

$$J = \sum_{j=N1}^{N2} (y_r(t+j) - y_m(t+j))^2 + p\sum_{j=1}^{Nu} (u'(t+j-1) - u'(t+j-2))^2 \qquad (18)$$

Where N_1, N_2, and N_u define the horizons over which the tracking error and the control increments are evaluated. The u' variable is the tentative control signal, y_r is the desired response, and y_m is the network model response. The p value determines the contribution that the sum of the squares of the control increments has on the performance index. The controller consists of the neural network plant model and the optimization block as shown in Figure 11.

The optimization block determines the values of u' that minimize J, and then the optimal u is input to the plant. The controller block is described as in Figure 11.

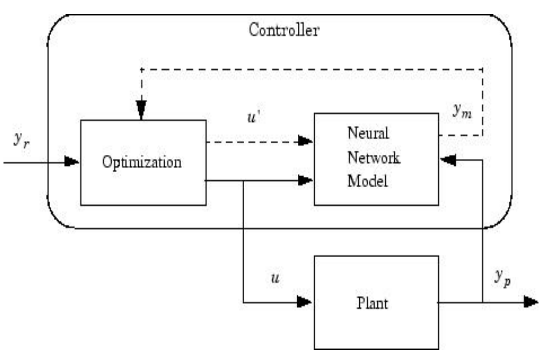

Figure 11: Block Diagram of the Neuro Controller

The result of the developed controller is presented in Figure 12; showing the random plant inputs steps and the output result of the simulink plant model. It was observed that the neural network model is one step ahead prediction while the error margin (difference between the plant output and the neural network output) is zero.

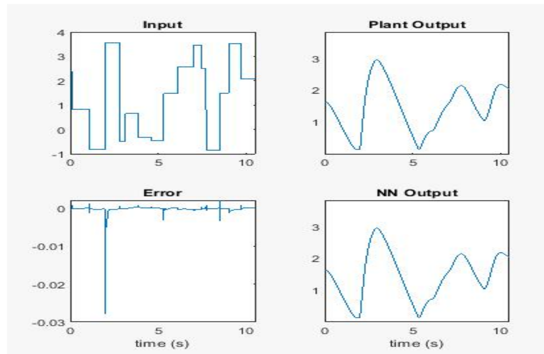

Figure 12: Result of the Neuro Controller

The controller was used to optimize the performance of the sensor model, thus producing a neuro sensor as shown in the implementation model of Simulink in the Figure 13.

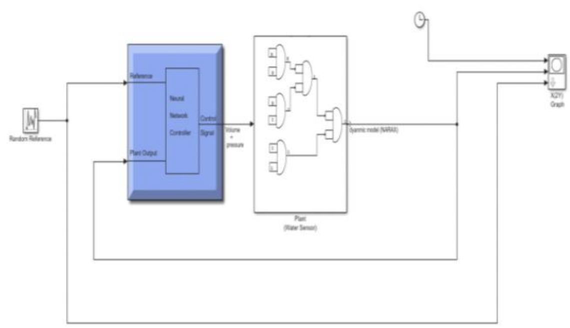

Figure 13: Simulink Model of the Neuro Sensor

The implementation model was simulated to determine the performance of the newly developed sensor using the simulation parameters in Table 3. The controlled response is shown in Figure 14. It was observed that the response time of the neuro sensor is 5.6secs.

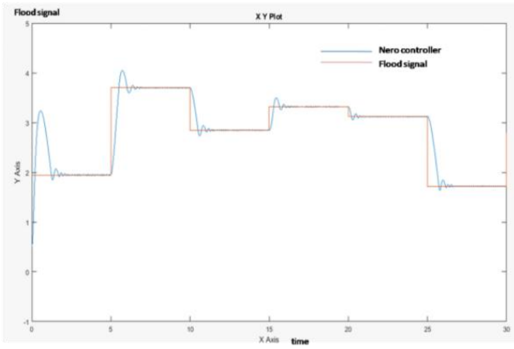

Figure 14: Step Response of the Neuro Controller

3.10 Block Diagram of the System

This section explains the complete architecture and operation of the system using the improved developed neuro sensor. The system is made up of four main sections which are; the power supply, sensing unit, monitoring and alert section.

The power supply is designed using a 220/12V step down transformer, the 12V/AC signal is converted to direct current (dc) by the rectifier (diode, IN4004) and filtered using the capacitor (24v/1200uf) before two voltage regulators are connected (+5V and +12V) to supply the required power to the monitoring and control system.

The sensors are mounted at various water level to detect the volume (height) and pressure of water, this sensor is designed to trigger when both high pressure and water are sensed (see Figure 15), this is to ensure accuracy and not send wrong signal to the control center or environment. This is important because wrong signal can cause human panic and also system unreliability.

The signal input is connected to the operational amplifier using a logic gate device which triggers when all the sensors trigger as shown in Table 3; the output data are transmitted to a microcontroller (Atmel 80S52) using an analogue to digital converter (ADC). The ADC is a device which converts signal types either analogue or digital signal to the desired output. All these aforementioned components in the monitoring section operates with +5V.

The alert section is designed using a switch and a buzzer, all powered by +12V dc. The buzzer is employed for alert, while the switch is used to trigger the indicators with respect to the water level. This is further buttressed using the system block diagram in Figure 15.

Table 3: Neuro Sensor Operation

Sensors	A	B	C	Output	ALERT
LEVEL 1	1	0	0	1	GREEN
LEVEL 2	1	1	0	1	YELLOW
LEVEL 3	1	1	1	1	RED

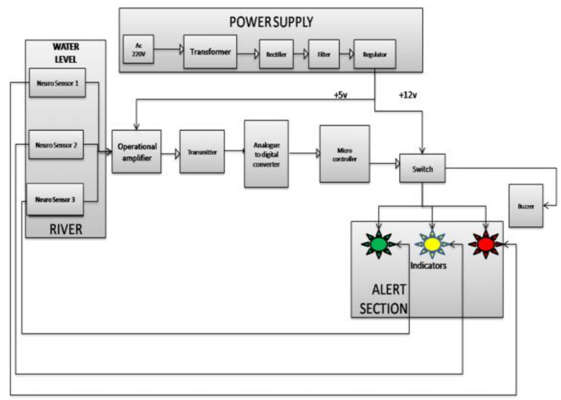

Figure 15: System Block Diagram

IV. RESULTS AND DISCUSSIONS

4.1 Results of the Neural Network Training

When a neural network was loaded with the flood dataset collected from NIMAX, it automatically spit the activated feature vectors into multi set of training, testing and validation sets in the ratio of 70:15:15. The training process was done, then test set was used to test the performance of the training to check it truly the data is been learned, the validation set is used to validate the result.

During the training of the neural network, the mean square error performance toll was used to evaluate the performance as shown in Figure 16.

Figure 16: Training Error Analysis

From the result it was observed that the best training performance has a root mean error of 0.02377 at epoch 2. Further training evaluation was performed to justify this result in Figure 17 using a regression model. The regression result is employed to monitor the fittings of the neural network performance in line with the reference plant model. This is achieved by creating a linear relationship between the output and the target. If the fitness is 100%, the linear relationship is R=1 which is the precised result for the plant. Although it is rare to achieve in practical, however if the relationship is R= < 0.5, then the performance of the neural network controller is very poor and need to be re-trained.

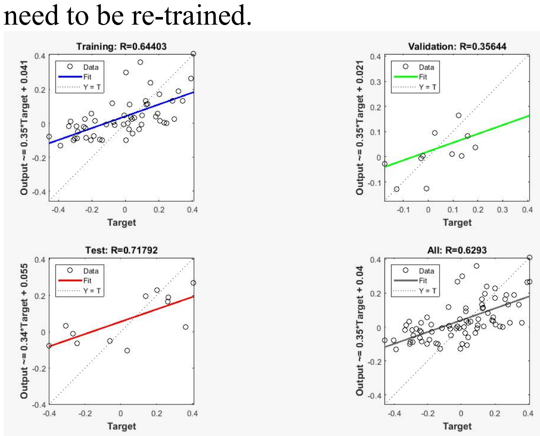

Figure 17: Training Error Analysis

From the result in Figure 17 it was revealed that the relationship between the training, testing and validation result are not accurate (0.6293). This is not acceptable in this case and will greatly affect the performance of the neuro sensor response. To improve this result, the system was retrained and this time produced a desired validation result as shown in Figure 18.

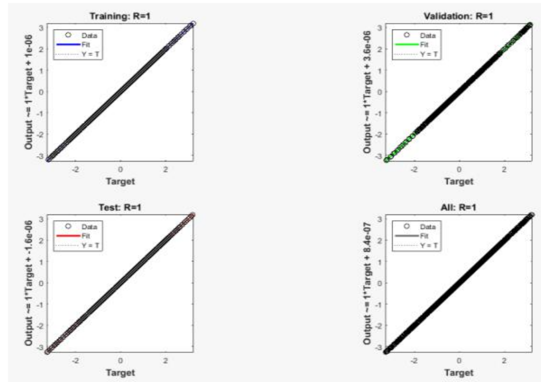

Figure 18: Regression Model for the Second Training

From the result in Figure 18; it was observed that the relationship between the target and the output at R= 1 was achieved. It is true that a regression value of 1 cannot be achieved in practical but the implication of this performance as achieved in simulation also shows that even though other physical factor will not allow the realization of this result (100%) in practical, however when implemented it indicates that a high level of precision will be guaranteed in the performance. The step response performance of the sensor is presented in Figure 19. While the Performance, when tested at the Ugwuaji River was shown in Table 4.

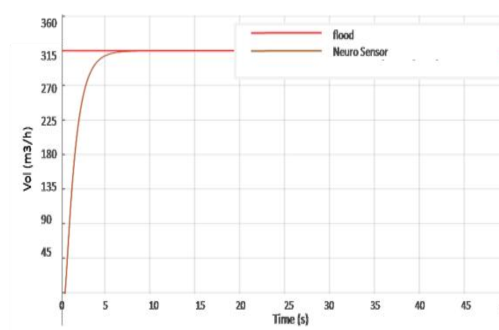

Figure 19: The Step Response Result of the New Sensor

In the Figure 19 the step response performance of the new sensor was presented when tested against flood. The result showed that the sensor was able to detect flood at 5.6s and then predict it at 8.44s, the delay time between detection and prediction of flood is 2.84s.

Table 4: Performance When Tested at the Ugwuaji River

Response time(sec)	Sensor Data (m^3)	Prediction (m^3)	Status	**Results**
8.400	96.9	95.8	Green Alert	Water level normal
8.450	94.8	93.8	Green Alert	
8.500	90.9	91.9	Green Alert	
8.550	92.9	93.5	Green Alert	
8.600	94.7	95.6	Green Alert	
8.650	95.7	96.9	Green Alert	
8.700	98.7	99.8	Green Alert	
8.750	99.9	102.6	Yellow Alert	Warning signal
8.800	100.2	100.7	Yellow Alert	
8.850	103	102.8	Yellow Alert	
8.900	101.4	101.3	Yellow Alert	
8.950	100.2	100.7	Yellow Alert	
8.000	100.7	101.8	Yellow Alert	
8.050	104.8	103.5	Yellow Alert	
8.1OO	106.3	106.8	Yellow Alert	
8.15O	80.7	81.8	Yellow Alert	
8.2OO	86.8	87.5	Yellow Alert	
8.25O	187.9	189.6	Yellow Alert	
8.30O	198.9	199.5	Yellow Alert	
8.35O	198.6	199.9	Yellow Alert	
8.4OO	199.7	199.4	Yellow Alert	
8.45O	240.1	248.6	Yellow Alert	
8.5OO	270.3	280	Yellow Alert	
8.55O	290.7	292.1	Yellow Alert	
8.00O	305.1	305.3	Yellow Alert	
8.05O	315.4	317.4	Red Alert	Flood signal
8.2OO	392.9	393.9	Red Alert	
8.15O	493.3	493.3	Red Alert	
8.2OO	482.9	488.9	Red Alert	
8.25O	493.1	497.1	Red Alert	
8.3OO	493.8	494.8	Red Alert	
8.35O	483.8	489.8	Red Alert	
8.4OO	472.8	479.8	Red Alert	
8.45O	472.9	474.3	Red Alert	
8.5OO	482.9	483.1	Red Alert	
8.55O	493.2	498.7	Red Alert	

The data collected and presented in the Table 4 shows the performance of the new flood detection system developed. The data was analyzed and the result is presented in Figure 20.

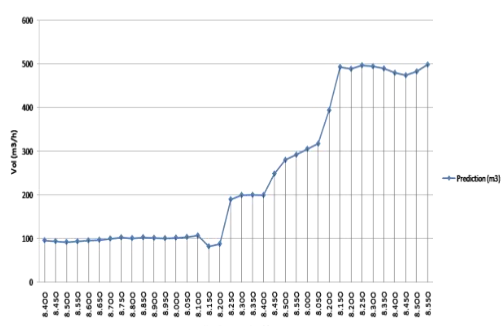

Figure 20: Performance of the New Sensor

The result showed that response time of the new sensor when tested at the case study Ugwuaji River. The result showed that the sensor was able to detect flood at 8.44s as the average time of detection. This is good and acceptable as the early detection and warning will facilitate fast evacuation.

4.2 Comparative Analysis

This section presents the comparative performance of the new and characterized sensor used for flood detection and monitoring. The result was collected from the characterized data and the result of the new sensor and presented in the Table 5.

Table 5: Comparative Sensor Performance

New Response time(sec)	Sensor Data (m^3)	New Prediction(m^3)	Characterized Response time(sec)	Characterized Sensor Data (m^3)	Characterized Prediction(m^3)
Response time(sec)	96.9	95.8	21.400	96.9	94.8
8.400	94.8	93.8	21.400	94.8	91.8
8.450	90.9	91.9	21.450	90.9	91.9
8.500	92.9	93.5	21.500	92.9	94.5
8.550	94.7	95.6	21.550	94.7	95.6
8.600	95.7	96.9	21.600	95.7	97.9
8.650	98.7	99.8	21.650	98.7	99.8
8.700	99.9	102.6	21.700	99.9	102.6
8.750	100.2	100.7	21.750	100.2	97.7
8.800	103	102.8	21.800	103	107.8
8.850	101.4	101.3	21.850	101.4	108.3
8.900	100.2	100.7	21.900	100.2	109.7
8.950	100.7	101.8	21.950	100.7	121.8
8.000	104.8	103.5	22.000	104.8	131.5
8.050	106.3	106.8	22.050	106.3	131.8
8.100	120.7	121.8	22.100	120.7	131.8
8.150	126.8	127.5	22.150	126.8	131.5
8.200	187.9	189.6	22.200	187.9	157.6
8.250	198.9	199.5	22.250	198.9	171.5
8.300	198.6	199.9	22.300	198.6	165.9
8.350	199.7	199.4	22.350	199.7	187.4
8.400	240.1	248.6	22.400	240.1	198.6
8.450	270.3	280	22.450	270.3	210
8.500	290.7	292.1	22.500	290.7	250.1
8.550	305.1	305.3	22.550	305.1	325.3
8.000	315.4	317.4	23.000	315.4	337.4

The Table 5 presented the comparative performance of flood detection systems. The comparative response time is presented as shown in Figure 21.

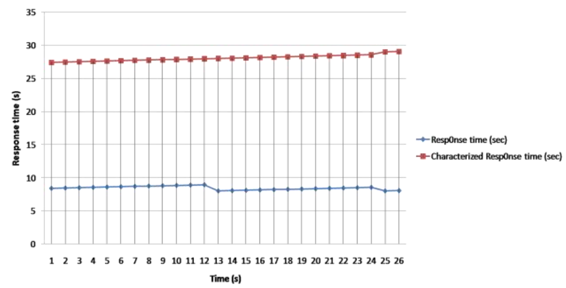

Figure 21: Comparative Response Time

The result in Figure 21 presented the comparative performance of the sensor response time when used to predict flood in the river. The result showed that the new sensor was able to predict flood faster than the characterized sensor. The new sensor predicted flood at 8.44s while the characterized sensor predicted flood at 22s. The percentage increase in the new system performance is 21.6%. The comparative delay time between detection and prediction is 24.6s for the characterized sensor and 2.84 for the new sensor, the percentage improvement in the delay performance is therefore 88.4%. The next result compared the sensors prediction performance as shown in Figure 22.

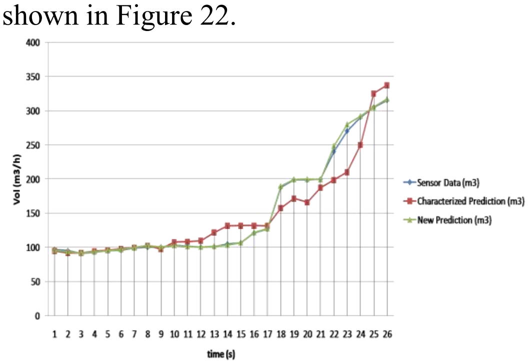

Figure 22: Comparative Flood Level Prediction Performance

From the result of Figure 22, the prediction performance of the new and characterized sensor was presented. The result showed that the new sensor predicted performance closely flow the sensor data which is good indicating the reliability of the new system, when compared to the characterized prediction performance which differs from the sensing data.

4.3 Discussions

This work has successfully developed an artificial intelligent-based system for real time flood detection and monitoring. The work employed the BFGS quasi-Newton back propagation algorithm for the training process. This helps achieve the desired response of the neuro sensor developed. From the results, the characteristic performance of the sensor has been presented and analyzed, starting with the response time of the sensor. This was achieved by comparing the result of the actual experiment time and the response time of the sensor. Furthermore, the sensing and prediction capacity of the sensor was also looked into and observe that the sensor was able to predict the future behavior of the river for some time. This is because the sensor is designed and calibrated to trigger for flood at 315v/h. the result showed that the new system was able to detect flood accurate at 8.44s.

The result of the neural network performance was also examined to justify the improvement made, this was done using the regression analysis and mean square error result. The regression result shows that the later algorithm (existing system) will not produce the desired response behavior of the sensor with a regression value of R = 0.6293, this result is good but far from precision.

The former algorithm (new system) after training at various iteration eventually produced a regression result of R=1. This is good because the time of response is vital for fast information and evacuation ahead of flood. From the result of the neuro sensor designed, it was observed that the sensor responded very fast to flood signal within 8.44 seconds. This is perfect when compared to the existing system characterized in table 5 with a delay response time of 22 seconds.

V. CONCLUSION

The proposed Real Time Flood Detection and Monitoring system has been successfully designed and simulated using Simulink toolbox. The requirement specification was to develop a nonlinear system that is intelligent in such a way that it can readjust its parameters and produce an output based on reference input. It is required that the system should not be complex structurally, response time should be at real time and lastly the system implementation and maintenance cost should be relatively low when compared to existing systems. It was observed from the work carried out that the requirement specifications were met.

A PID based conventional flood detection and monitoring system was successfully characterized to establish those parameters that improve flood detection and monitoring system. Using those established parameters, a nonlinear model of Ugwuaji River was developed. The sensor that is meant to acquire real time data from the environment was designed using pressure as sensing the sensing element. A nonlinear model predictive control system which utilizes previous process control behavior to foretell future response of the system was modeled and implemented in Simulink. The developed mathematical models were transformed into discrete form using Laplace transform to establish the transfer functions for development of the Simulink model for real time simulation. The model predictive control system network was successfully trained offline using BFGS quasi-Newton back propagation algorithm. Simulation was carried out to evaluate the performance of the developed system in the Simulink environment.
Simulation results show that the proposed system achieved a regression result of R=1 after various iteration during training. The significance of this result is that the monitoring of changes in the parameters of the sensor which is time dependent can be done in real time. From the result also, it was observed that the proposed system responded very fast to flood signal within 8.44 seconds as against 22 seconds achieved from the work of Amy Mani (2016). The proposed system was validated to determine improvement by comparing the response times achieved by the two systems. It was found that the proposed system achieved a 21.6% improvement when compared with the conventional PID system.

REFERENCES

1. Amy Mani (2016), "Design and Implementation of a Fully-Automated Water Level Indicator", CA: Proposal Life.
2. Anthone Vinicio and Satoru Oishi (2018), "A Wireless Mesh Sensor Network Framework for River Flood Detection Which Can be Used as An Emergency Communications Network In Case Of Disaster", 11th International Conference on Hydroinformatics HIC.
3. Azid I.S. and Sharma B. (2022), "An SMS Based Flood Level Monitoring System, Advances in Computer Science and Engineering", Springer. Vol.8, (2).
4. Basha, Elizabeth, and Daniela Rus (2017), "Design of early warning flood detection systems for developing countries", Information and Communication Technologies and Development (ICTD), International Conference on, IEEE.
5. Isizoh A.N, Anazia A.E, Okide S.O, Okwaraoka C.A (2014), "Improved Automatic Process Control System Using Artificial Intelligence", Natural and Applied Science Journal, Vol 13, Issue 1.
6. Hopson T. M. and Webster P. J. (2016), "Operational Short-term Flood Forecasting for Bangaldesh: Application of ECMWF Ensemble Precipitation Forecasts," Vol. 8.
7. Hughes D., Greenwood P., Blair G., Coulson G., Pappenberger F. (2018), "An Intelligent and Adaptable Grid-Based Flood Monitoring and Warning System", in Proceedings of the 5th UK eScience All Hands Meeting.
8. Shen C. P. (2020), "A Real-Time Flood Monitoring System Based on GIS and Hydrological Model," 2nd Conference on Environmental Science and Information Application Technology, pp. 476- 485.
10. Victor Seal, Arnab Raha, Shovan Maity, Souvik Kr Mitra, Amitava Mukherjee (2022), "A Simple Flood Forecasting Scheme Using Wireless Sensor Networks", International Journal of Ad hoc Sensor & Ubiquitous Computing (IJASUC) Vol.3, No.1.
11. Yawut C. (2021), "A Wireless Sensor Network for Weather and Disaster Alarm Systems", Int. Conf. on Information and Electronics Engineering, Singapore, Vol.6.

