

## A HYBRID STOCHASTIC-ANN APPROACH FOR FLOW PARTITIONING IN THE OKAVANGO DELTA OF BOTSWANA

MOALAFHI D.B.<sup>1</sup>  
PARIDA B.P.<sup>2</sup>  
KENABATHO P.K.<sup>1,\*</sup>

<sup>1</sup>*Department of Environmental Science  
University of Botswana, P/Bag 0022  
Gaborone, Botswana*

<sup>2</sup>*Department of Civil Engineering  
University of Botswana, P/Bag 0022  
Gaborone, Botswana*

Received: 12/09/13  
Accepted: 10/10/13

\*to whom all correspondence should be addressed:  
e-mail: kenabatho@mopipi.ub.bw

### ABSTRACT

Since a spectrum of hydrological and geomorphological conditions produce flood pulse environment in a riverine or a deltaic system, it is essential to have the knowledge on spatial and temporal distributions of river flow and dependent processes for environmental flow requirements, ecosystem maintenance, water resources management, and hydrological forecasting among others. Such systems being complex as the exchange of flows between the main channel and the flood plains are not well understood, flow partitioning dynamics between the various channels on large water bodies are often difficult to represent even with sophisticated models. In view of this, an attempt has been made to apply a short-term stochastic forecasting model-an Auto Regressive Integrated Moving Average (ARIMA) aided by Artificial Neural Networks (ANNs) to partition flows into the downstream tributaries, viz.: Lopis and Gadikwe channels from the Khiandiandavhu-Maunachira (K-M) Junction Junction (the main river channel) river system of the iconic Okavango delta in Botswana.

As such, observed monthly flow data between October 2005 and September 2008 at the K-M Junction, and the two downstream tributaries were used to test the performance of these hybrid models for the complex deltaic system. It was found that the partitioned flows at Lopis and Gadikwe agree very well with observations when using a Single Input Multiple Output (SIMO) ANN (i.e. an inverse variant of the widely used Multi Input Single Output (MISO) ANN architecture) and an ARIMA (1,1,1) model. The Mean Squared Errors (MSEs) in the forecasts were also minimal, thus giving some hope on the use of such a hybrid mode for the rest of the branched river networks of the whole Okavango delta.

**KEYWORDS:** Artificial Neural Network, Autoregressive Integrated Moving Average, Mean Squared Error, Forecasting

### 1. INTRODUCTION

It is generally agreed that effective water management strategies are crucial for sustaining economic development, meeting water demand for growing and urbanizing populations, and sustaining ecological integrity of catchment systems (Pike and Scatena, 2010). Understanding the hydrological processes at catchment level is key to dislodging the linkages between riparian ecosystems, geomorphic and ecological processes (Shaw and Cooper, 2008). Over the past few decades, a great deal of research has been devoted to modeling and forecasting of river flow dynamics (Sivakumar *et al.*, 2002; Mazvimavi and Wolski, 2006; Parida *et al.*, 2006; Pechlivanidis *et al.*, 2010). This has led to great advances in the understanding and modeling of complex interplays between catchment processes and variables characterized by non-linearity. Hydrologic models can particularly assist decision makers in

dealing with these challenges by providing systematic and consistent information on water availability, spatial activities particularly land use change and its impacts on the hydrologic systems.

In river flow forecasting, models may be grouped into physically-based and black-box models. In between these spectrums lie the so-called conceptual models (Wheater, 2002). The physically-based models are specifically designed to mathematically simulate or approximate the general internal sub-processes and physical mechanisms that govern the river flow process, whereas the black-box models are designed to identify the connection between the inputs and the outputs, without going into the analysis of the internal structure of the physical process (Sivakumar *et al.*, 2002). Conceptual models seek to represent important component processes as perceived by the modeler, and as a result the model structure is specified a priori (Wheater *et al.*, 1993). They typically conceptualise the catchment water balance using series of interconnected storages that do have a physical interpretation, but usually employ empirically parameterised equations to describe the movement of water between the storages (Young *et al.*, 2006).

Hydrological models may be used to analyze runoff data (and other related variables) at daily, monthly and annual timescales. Among these three timescales, the monthly runoff generation analysis is probably the most difficult to model as it needs to take into account both short-and long-term hydrologic processes (Anmala *et al.*, 2000; Lungu, 1991). Despite this, monthly runoff predictions are useful for long-term simulation over periods of decades as it is relevant for many processes of subsurface water movement, where residence times of several months in a watershed are not unusual (Anmala *et al.*, 2000). In a delta these aspects are well noticed where in addition to the storage characteristics due to long residence time, the flow partitioning into several stream is a dynamic process and very complex to make a good forecast for management purposes.

Therefore, in this study, a hybrid modeling approach is proposed. This consists of a (i) stochastic model to capture the persistence component inherent in the flow because of long storages, and to forecast flow volumes at the desired river junctions; (ii) then the use of artificial intelligence to partition these flows into the emerging downstream branches in the Okavango delta in Botswana, potentially solving an inverse modeling problem using the ANNs. This is because the ANNs are usually used to obtain a single output based on multiple input variables (i.e. Multiple Input-Single-Output) (Emamgholizadeh, 2012). This approach was tested using observed monthly discharge volumes to simulate flow volumes for long-term management of the ecological functioning of this world's largest dynamic wetland system.

## 2. THE STUDY AREA

The Okavango Delta is situated in the northern part of Botswana in the Ngamiland District with its drainage basin covering three countries namely, Botswana, Namibia and Angola. The delta is a significant water source for each of these countries (Wolski *et al.*, 2006). The basin in Botswana is enclosed between coordinates 21° 41" and 23° 53' East and 18° 15' and 20° 45' South (DWA, 2004). Rainfall over Angola, southeast of the Luanda Divide, is the primary source of water for this river. The Okavango River enters Botswana at Mohembo (in the month of April), having travelled some 1000 km from its source, and it is then channeled through a 15 km wide corridor between two secondary faults oriented north–east to south–west known as 'panhandle'. Downstream of the panhandle, the Okavango comprises a near-perfect conical surface with a gradient of approximately 1:3600 radiating from the apex of the 'delta' which is more correctly classified as an alluvial fan. From Mohembo, the Okavango River flows southeast to Seronga before dispersing into a series of complex and poorly understood vegetation-choked channels and shallow basins that form the largest inland delta in the world. It then takes 4–5 months for this flood wave to travel approximately 300 km, through the wetland, to reach the town of Maun, at the toe of the delta, in August (McCarthy *et al.*, 1998). This pattern of flooding means that the Okavango Delta experiences its maximum extent of inundation during the dry winter season. The area flooded at least every decade is approximately 14 000 km<sup>2</sup>, of which 9000 km<sup>2</sup> is classified as actual wetland (Kurugundla *et al.*, 2009).

The region of the Okavango delta is semi-arid with evaporation four times higher than rainfall (Ringrose *et al.*, 2005), and is of the order of 2300 mm in a year. There is a large difference between the volumes of water entering and leaving the Okavango Delta each year.

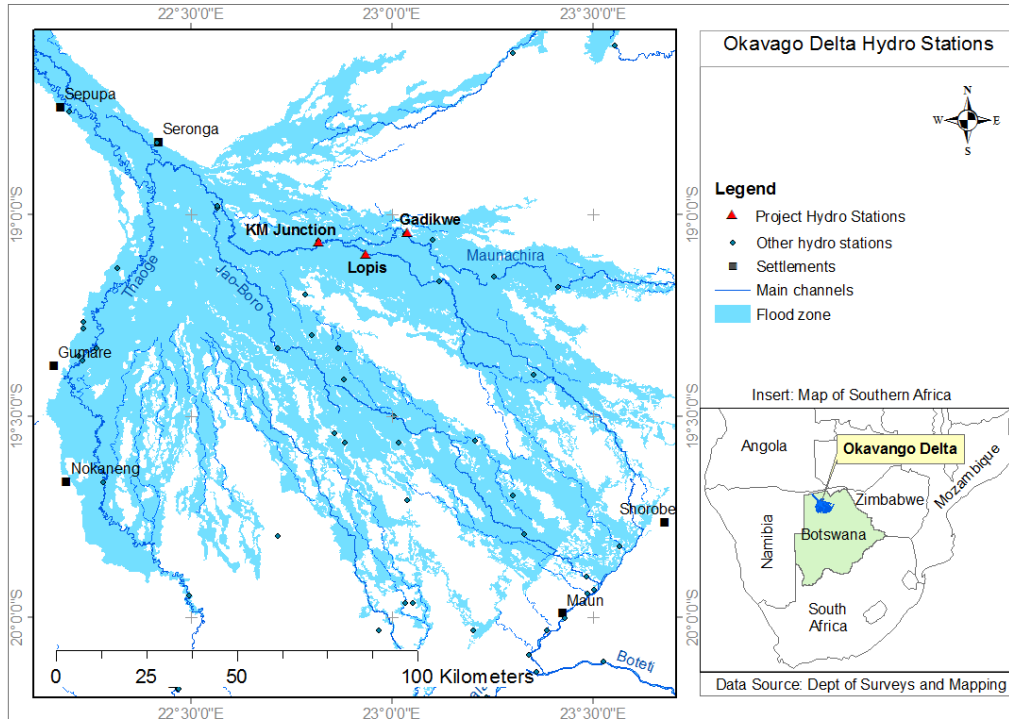


Figure 1. The spatial extend of the Okavango delta in Botswana. [Station 8=KM-Junction, 9=Lopis, 11=Gadikwe]. (Source: Kurugundla *et al.*, 2009)

The outflow from the delta is related to the extent of the inundation of the wetland, and several attempts have been made to predict the outflow using the variables that affect the spread of this inundation, such as rainfall, inflow and evapo–transpiration (Bauer, 2002; Dincer *et al.*, 1987; Gieske, 1997; Ellery *et al.*, 1993 and McCarthy *et al.*, 1998). The conceptually simple hydrological models of McCarthy *et al.*, (1998) and Wolski *et al.*, (2006) illustrate the importance of inflow, rainfall, previous outflow and evaporation as important variables that contribute to the observed variation in outflow.

With the outflow of the Delta standing at about 2% of the inflow, the southernmost Lake Ngami or the Boteti River only receives substantial water during higher flow years. The Boteti River, which is now mostly dry, ultimately connects to the Makgadikgadi salt pan to the southeast of the Delta.

### 3. METHODOLOGY

As said earlier, due to long residence time of flow into the delta as storage, it has been proposed to model the flow volumes in main channel before bifurcation into smaller streams using stochastic approaches and artificial neural networks (ANNs).

#### 3.1 Autoregressive Integrated Moving Average (ARIMA)

Autoregressive integrated moving average (ARIMA) models are useful in representing many time series ( $z_t$ ,  $t = 1, 2, \dots, N$ ) exhibiting non-stationary behaviour and in particular, which do not vary about a fixed mean. The models have the general form

$$\Phi_p(B)z_t = \phi_p(B)\nabla^d z_t = \theta_q(B)a_t \quad (1)$$

where  $\Phi_p(B) = \phi(B)(1-B)^d$  is the generalized Auto Regressive (AR) operator, in which one or more of the zeros of the polynomial  $\Phi_p(B)$  lie on the unit circle. The order of an ARIMA process is  $(p, d, q)$ , also written as ARIMA  $(p, d, q)$ , represents the  $p^{\text{th}}$  order AR,  $q^{\text{th}}$  order Moving Average (MA) components with 'd' time steps differenced series. In practice, d is usually 0, 1, or at most 2.

If  $\nabla^d z_t$  is replaced by  $z_t - \mu$  in Eq.(1), when  $d = 0$ , it represents the stationary ARMA (p, q) process as a special case, or even just a simple AR(p) or MA(q) processes. It is therefore legitimate to denote the AR (p) model by ARMA (p, 0) or ARIMA (p, 0, 0), the model MA(q) by ARMA (0, q) or ARIMA (0, 0, q), and the model ARMA (p, q) by ARIMA (p, 0, q) (Box, Jenkins, and Reinsel, 1994).

As time series data that are observed on some regular calendar basis may have some seasonal effect, there will generally be autocorrelation at lag s or at its multiples, depending on the persistence and nature of that autocorrelation (Wadsworth, 1998). To account for the seasonal dependence, Eq. (1) may be written as

$$\Phi_P(B^s)\nabla_s^D z_t = \Theta_Q(B^s)e_t \quad (2)$$

where  $e_t$  is a normal random deviate, D is the degree of seasonal differencing, s is the periodicity and is equal to 12 for the data considered in this study, and

$$\Phi_P(B^s) = 1 - \Phi_1 B^s - \Phi_2 B^{2s} - \dots - \Phi_P B^{Ps} \quad (3)$$

$$\Theta_Q(B^s) = 1 - \Theta_1 B^s - \Theta_2 B^{2s} - \dots - \Theta_Q B^{Qs} \quad (4)$$

The operators  $\Phi_P(B^s)$  and  $\Theta_Q(B^s)$  are seasonal autoregressive and moving average operators, which are polynomials in  $B^s$  of degrees P and Q, respectively, and satisfy stationarity and invertibility conditions.  $\nabla_s = 1 - B^s$  is the simplifying operator. Useful operations in the analysis are

$$B^s z_t = z_{t-s}, \nabla_s z_t = (1 - B^s)z_t = z_t - z_{t-s} \quad (5)$$

Similar to Eq. (2), a model

$$\Phi_P(B^s)\nabla_s^D z_{t-1} = \Theta_Q(B^s)e_{t-1} \quad (6)$$

can be used to link the current behaviour with previous observations, and so on. In Eq.s (2), (6) and other similar equations, the error terms  $e_t, e_{t-1}, \dots$  would in general be correlated (Lungu, 1991). Such relationships can be taken care of by the model

$$\phi_p(B)\nabla^d e_t = \theta_q(B)a_t \quad (7)$$

To account for both the seasonal and non-seasonal dependencies, Eq.s (6) and (7) are combined, to give the general multiplicative seasonal ARIMA model of order  $(p, d, q) \times (P, D, Q)_s$ :

$$\Phi_P(B^s)\phi_p(B)\nabla_s^D \nabla^d z_t = \Theta_Q(B^s)\theta_q(B)a_t \quad (8)$$

where subscripts p, q, Q, P have been added as a reminder of the orders of the various operators.

The specific objectives of stochastic modeling are (i) to identify the order (p, d, q) for a non-seasonal model, or the order  $(p, d, q) \times (P, D, Q)_s$  for a seasonal model, that provides a parsimonious representation for both the total series under consideration and the stochastic component (ii) to determine the proportions of the variance of the total series of the serially correlated data, explained by the periodic and stochastic components (iii) to determine the persistence pattern (if any) of the stochastic component and (iv) to calculate least square estimates for the parameters of the equations. Against all these, ARIMA models are based on three parts: (1) an autoregressive part, (2) a contribution from a moving average and (3) a part involving the first derivative of the time series. The formula used for describing AR(p) models is;

$$Z_t = \sum_{i=1}^p \alpha_i Z_{(t-i)} \quad (9)$$

where the order of the model is determined by  $p$ . This is described by linear models based on preceding observations.

Past estimation or forecasting errors are taken into account when estimating the next time series value. The difference between the estimation  $\hat{Z}_t$  and the actually observed value  $Z_t$  is denoted  $\varepsilon_t$ .

The description of MA( $q$ ) model is:

$$Z_t = \sum_{i=1}^q \beta_i \varepsilon_{(t-i)} \quad (10)$$

ARMA models are obtained through combining AR and MA models. An ARMA ( $p, q$ ) is described by:

$$X_{(t)} = \sum_{i=1}^p \alpha_i X_{(t-i)} - \sum_{i=1}^q \beta_i \varepsilon_{(t-i)} \quad (11)$$

After additional differentiation and integration of the time series, ARIMA models are realized. The parameter  $d$  of ARIMA( $p, d, q$ ) determines the number of differentiation steps until the time series is stationery. It is from this that a suitable ARMA ( $p, q$ ) is fitted to the resulting series. Finally, the estimated forecasts are integrated  $d$  times to obtain results in the normal domain.

### 3.2. Artificial Neural Network

An Artificial Neural Networks (ANN) could be conceptualized as an information processing paradigm that is inspired by the structural and operational functioning of the biological nervous systems, such as the brain. Central to this paradigm is the novel structure of the information processing system that is composed of a large number of highly interconnected processing elements (neurons) working in unison to solve specific problems.

Neural Networks, with their remarkable ability to derive meaning from complicated or imprecise data, can be used to extract patterns and detect trends that are too complex to be noticed by either humans or other computer techniques. Their use has to some degree alleviated the problem of modeling the complexity between rainfall and runoff in flow regime studies (Michaelides *et al.*, 2001; Shamseldin, 1997). This is achieved through determination of neural network optimal architecture, which is included as a part of the learning strategy. Artificial Neural Networks approach has successfully been used in hydrologic studies by a considerable number of researchers (Anmala *et al.*, 2000; Furundzi 1997; Parida and Moalafhi 2008; Sivakumar *et al.*, 2002; Uvo *et al.*, 2000). Over the years, neural networks have proved to be uniquely powerful tools in applications where formal analysis would be difficult or impossible, such as pattern recognition and non-linear system identification and control (The Math Works, Inc. 1994-2002). ANNs have widely been used in modeling a large variety of dynamic systems that are characterized by non-linearity (Coulibaly *et al.*, 2001; Hsu *et al.*, 2002; Johnson and Rogers, 2000), and such non-linearity is a prime characteristic of issues related to the atmospheric and hydrologic sciences.

#### 3.2.1. Feed forward back propagation neural network

The use of ANNs manifested through the feed-forward back propagation entails signals traveling one way only; from input to output. Mapping of the input to output vectors does not have feedback connections but errors are back-propagated during training to optimize for the performance function which is mostly Mean Squared Errors (MSEs). The simplest implementation of back propagation learning is realized through updating the network weights and biases in the direction in which the performance function decreases most rapidly-the negative of the gradient.

One of the advantages of use of ANNs is the inherent provision of adaptive learning in which the network can learn how to do tasks based on the data given for training or initial experience. This is mostly realized through model predictive control platform which offers a collection of functions (commands) developed mostly for use in multivariable control algorithm in many dynamic systems.

Depending on the number of system inputs and outputs, network architectures are classified as multi-input-single-output (MISO) and single-input-multi-output (SIMO) systems. With the SIMO configuration, the practical approach is to control the most important process variable and leave out the other variables uncontrolled or use the output variables to control the input of a  $1 \times M$  system after which the observed and controlled (simulated) variables of the input variables could be used as controller variables of each individual of original output variable of the  $1 \times M$  system separately (Cheng and Mulkey, 2004).

The control system comprises an  $M$ -input and 1-output controller in which  $r_1(t), r_2(t), \dots, r_M(t)$  are set-points  $1, 2, \dots, M$ .  $x_1(t), x_2(t), \dots, x_M(t)$  are system outputs of the  $1 \times M$  system.  $d_1(t), d_2(t), \dots, d_M(t)$  are disturbances  $1, 2, \dots, M$  caused by noise or load.  $y_1(t), y_2(t), \dots, y_M(t)$  are measured process variables of the  $1 \times M$  system. The control objective is to produce output  $u(t)$  to manipulate the manipulated variable so that the measured process variable  $y_1(t), y_2(t), \dots, y_M(t)$  track the given trajectory of their set-points  $r_1(t), r_2(t), \dots, r_M(t)$ , respectively, under variations in set-point, disturbance, and process dynamics. In this regard, the controller is focused on optimizing for errors  $e_1(t), e_2(t), \dots, e_M(t)$ . Here the control objective is to optimize the error for the more critical loop, or for all  $M$  loops with no weighing on the importance so that these may be static errors in all the loops.

During training, neural networks are adjusted so that a particular input leads to a specific target output. The weighed activations are the connections that inhibit or excite activity. The network is usually configured on the batch training mode in which training proceeds by making weight and bias changes based on a comparison of the output and the target until the network output matches the target. This is called supervised training.

A neuron with a single  $R$ -element input vector can be represented as;

$$P_1, P_2, \dots, P_R \quad (12)$$

where  $p_i$  is an  $i^{\text{th}}$  element input and  $R$  is the number of elements in the input vector.

The input vector elements enter the network through the weight matrix  $W$

$$W = \begin{bmatrix} W_{1,1} & W_{1,2} & \dots & W_{1,R} \\ W_{2,1} & W_{2,2} & \dots & W_{2,R} \\ \vdots & \vdots & \dots & \vdots \\ W_{s,1} & W_{s,2} & \dots & W_{s,R} \end{bmatrix} \quad (13)$$

The row indices on the elements of the matrix indicate the destination neuron of the weight, and the column indices indicate which source is the input for that weight. Here  $s$  is the number of neurons in a layer. The weighed activations are summed and fed to each neuron in the following layer,

$$n = w_{1,1}p_1 + w_{1,2}p_2 + \dots + w_{1,R}p_R + b \quad (14)$$

where,  $w_p$  is the dot product of the (single) row matrix  $w$  and the vector  $p$  is the argument of the transfer function  $f(\cdot)$  with the output;

$$a = f(W_p + b) \quad (15)$$

The multi-input—single-output (MISO) implementation of prediction usually adopted is based on historical record of the output  $y_1(k)$  and the inputs  $v_1(k); v_2(k), \dots, v_{n_v}(k)$ . The step response coefficients are estimated for  $n_v$  inputs and  $n_y$  outputs and this enables computation of the system output for any input sequence;

$$y(k) = \sum_{i=1}^n s_i \Delta v(k-i) + s_n v(k-n-1) \quad (16)$$

where  $s_i$  is the output change for a unit input change  $\Delta v$  for  $n$  steps

A three layer 2x1 system controller neural network architecture is shown as in Figure 2.

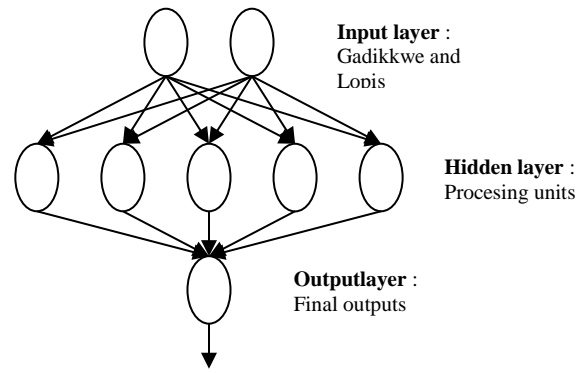


Figure 2. A three –layer 2X1 system controller neural network that was adopted for both Gadikwe and Loopis discharge volume predictions

For this study, the input variable was monthly river flow data at KM-junction station with output variables being river flows at the downstream branches, Lopis and Gadikwe. Then a 2-input-1-output Feed Forward Neural Network (FFNN) was developed as the controller to reproduce the discharge volumes at KM-junction. The historical and simulated volumes at KM-junction were subsequently used as inputs with volumes at Gadikwe and Lopis as targets separately for both timeseries of first entire observations (set A) and secondly the entire observations with the last two months volumes replaced by ARIMA forecasts (set B) for KM-junction. The input vectors, discharge volumes, were scaled to fall within [-1 and 1] as some form of normalization to improve on generalization. The output of the network was then post-processed and transformed back to the original form of inputs and targets. The outputs were compared with the target discharge volumes as the inputs were applied to the network. A learning rule (training algorithm) was then used to adjust the weights and biases to nodes of the network in order to find the optimum network architecture that could map the input vectors to target output vectors.

Back-propagation learning algorithm was used alongside the sigmoidal transfer function. The back-propagation algorithm is differentiable and it gives a network the needed ability to learn and/or model non-linearity between input(s) and output(s) while sigmoid transfer functions are usually preferable to threshold activation functions since with sigmoid units, even a small change in the weights will usually produce an appreciable change in the outputs, which makes it possible to tell whether that change in the weights is good or bad. The input layer used the identity transfer function because no processing takes place in the input layer. The hidden layer neurons used the log-sigmoidal transfer function which is commonly used with the back-propagation networks. The output neuron used *purelin* transfer function.

The network was developed to use the inputs that contributed at least 95 % of the total variation in the data set. One third of evenly representative data were equally divided between training, validation and testing. The training stopped when the validation error increased for a default number of iterations at which the weights and biases at the minimum of the validation error were returned finally

#### 4. RESULTS AND DISCUSSIONS

Observed monthly discharge volumes at KM-Junction from October 2005 up to July 2008 (say Set A) were used as to develop an ARIMA(1,1,1) model and subsequently to use it to make short time

forecasts. In this regard, discharge volumes for KM-Junction were forecasted for the months of August and September 2008 (Table 1).

Table 1. Observed and ARIMA(1,1,1) forecasted discharge volumes at KM Junction

Month	Observed-Discharge Volumes, MCM	ARIMA-Forecasted Discharge Volumes, MCM
August 2008	75.71	78.62
September 2008	75.08	77.40

The last two (2) river flows (i.e., August and September 2008) of set A were then replaced by the two (2) forecasts, which constituted say Set B. Discharge volumes at KM-Junction for both timeseries (Sets A and B) were used separately as the input in development of the M-input-1-output controller at which input variables were discharge volumes at both Gadikwe and Lopis. A network was developed and trained to simulate discharge volumes at KM-Junction. These simulated and observed (sets A and B separately) discharge volumes at KM-Junction were then used as inputs with single outputs being Gadikwe and Lopis discharge volumes separately.

The observed and simulated (using both sets A and B) discharge volumes are plotted together as shown by Figures 2 and 3 for both Lopis and Gadikwe respectively.

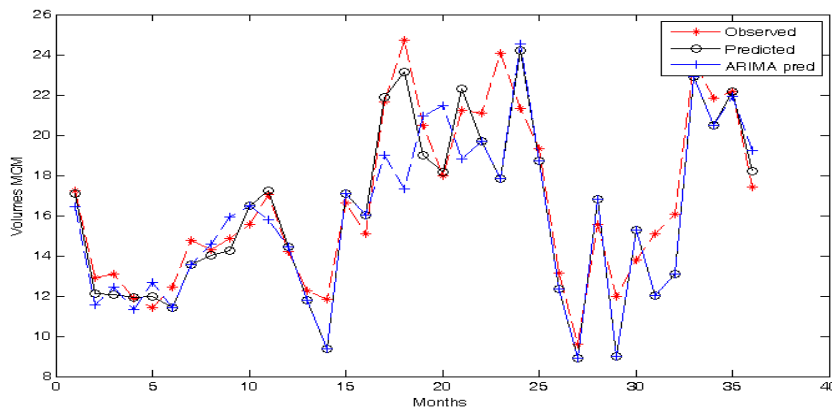


Figure 2. Observed and simulated discharge volumes at Lopis gauging station

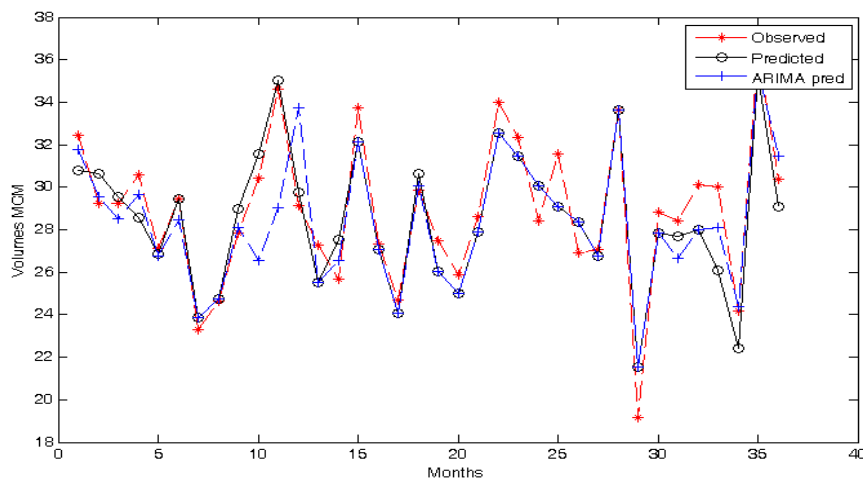


Figure 3. Observed and simulated discharge volumes at Gadikwe gauging station



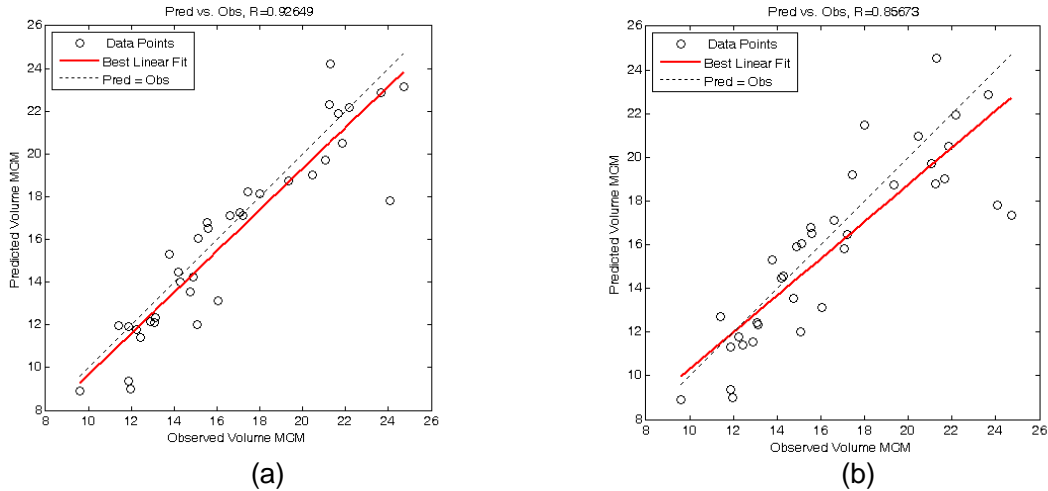


Figure 4. Simple Linear regression between the observed and simulated discharge volumes at Lopis gauging station. (a) set A and (b) set B at KM Junction

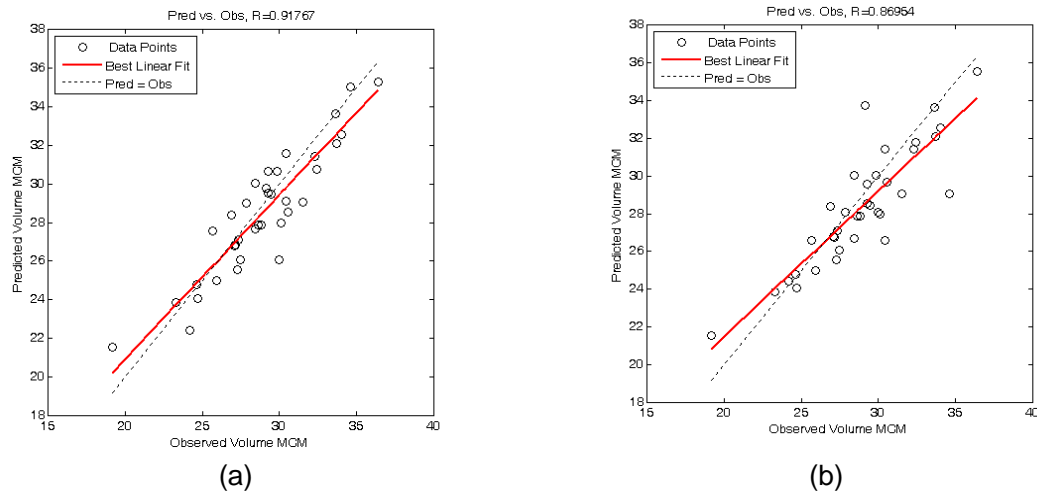


Figure 5. Simple Linear regression between the observed and simulated discharge volumes at Gadikwe gauging station. (a) using set A and (b) set B at KM Junction

The simulations using both sets A and B generally were in good agreement with correlation coefficients between observed and simulated volumes being greater than 0.8 for both the sites. This offers promise for advancing the use of this hybrid modeling approach for flow partitioning of the Okavango river system. Mean Squared Errors (MSEs) appear to be minimal for Gadikwe than for Lopis and thus the discharge volumes were better reproduced at Gadikwe gauging station with the testing data set having the highest MSE in both cases (Figures 6 & 7).

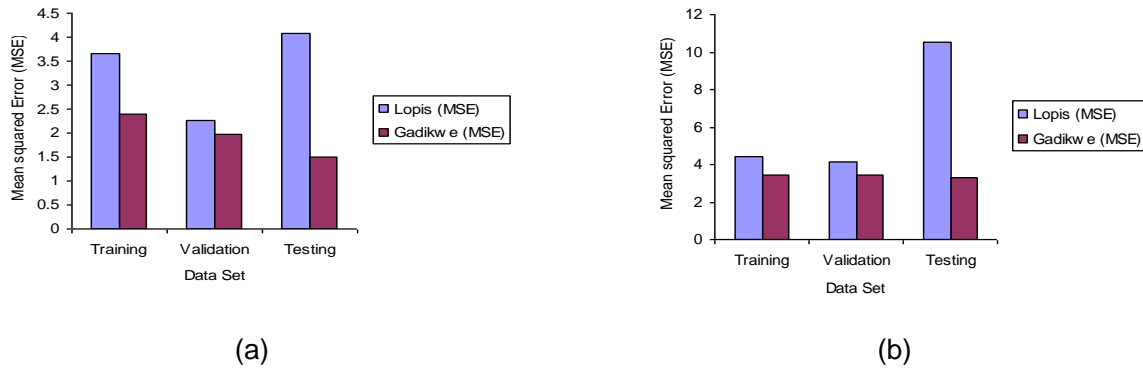


Figure 6. Mean Squared Errors (MSEs) for the data sets (a) using set A and (b) set B at KM Junction

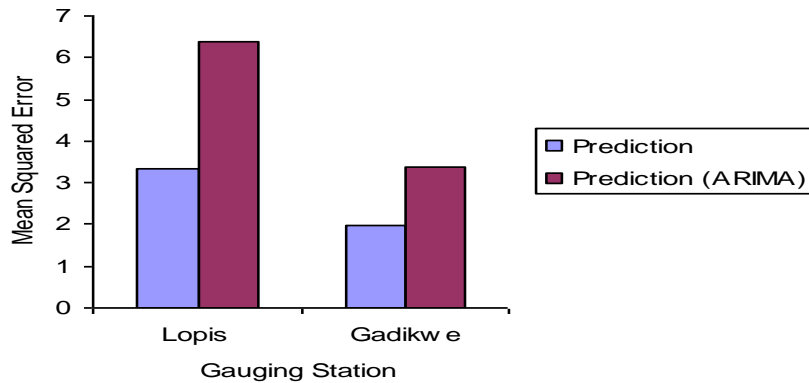


Figure 7. Mean Squared Errors (MSEs) for the simulations to predict partitioned flow into Lopis and Gadikwe gaugingstations

**5. CONCLUSIONS**

A stochastic forecasting for monthly river flow partitioning downstream of KM-Junction of the Okavango delta in Botswana has been attempted through an ARIMA forecasting model and subsequently evaluated using ANNs at both Lopis and Gadikwe gauging sites. For both sites, simulations were attempted using one set of observed flows at KM-Junction spanning between the period October 2005 through September 2008, all inclusive (set A). With the other set, the last two monthly discharge volumes (August and September 2008) were replaced with the ARIMA (1,1,1) forecasted flows (set B). The simulations at Lopis were achieved at correlation with observations of 0.93 (MSE=3.4) and 0.86 (MSE=6.2) respectively using both sets A and B respectively. The simulations were correlated with observations at Gadikwe at 0.92 (MSE=1.9) and 0.87 (MSE=3.4) respectively.

Although the MSEs using the ARIMA (1,1,1) extended time series were higher than the ones using entirely observed time series for KM-Junction, the use of a stochastic forecasting (ARIMA) model aided by SIMO ANN model for monthly river flow partitioning could be a viable option in the Okavango delta with correlation coefficients being higher than 0.8 in all the cases. Incorporating a greater range of river systems and developing nested models that can capture majority of riverbank flows will improve the resolution that can be used to include ecological relationships and extend the study to the whole of Okavango deltaic system. Having demonstrated use of stochastic forecasting for river flow partitioning at monthly time scales, finer resolutions of daily time scales, for example, could be very helpful and with synchronization of the time scales short term forecasts of branched river systems over the whole delta aided by inclusion of other hydro-climatic, ecological, and geomorphological data could be a

phenomenal achievement for hydrological forecasting, water resources management, ecosystem studies, flood warning, and tourism-related activities among others.

#### **Acknowledgements:**

The authors would like to thank the Department of Water Affairs and Department of Surveys and Mapping in Botswana for supplying the data used in this study. Special thanks also goes to Mr Othusitse Lekoko, Department of Environmental Science at University of Botswana for sourcing and providing maps used in this study. We also thank anonymous reviewers for improving the quality of this paper.

#### **REFERENCES**

- Anmala J., Zhang B., and Govindaraju R.S. (2000), Comparison of ANNs and Empirical Approaches for Predicting Watershed Runoff, *Journal of Water Resources Planning and Management*, **126**(3), 156-166.
- Bauer P., Brunner P. and Kinzelbach W. (2002), Quantifying the net exchange of water between land and atmosphere in the Okavango Delta, Botswana, *Proceedings of MODEL-CARE 2002*, Prague.
- Box G.E.P. and Jenkins G.M. (1976), *Time Series Analysis Forecasting and Control*, Holder-day, Oakland, CA.
- Cheng S.G. and Mulkey S.L. (2004), USPTO Filing Receipt mailed Dec. 2, 2004, Specification, Claims and Drawings of U.S. Appl. No. 10/969, 85, filed on Oct. 19, 2004, applications.
- Coulibaly P., Anctil F., Aravena R. and Bobée B. (2001), Artificial neural network modeling of water table depth fluctuations, *Water Resources Research*, **37**(4), 885-896.
- Department of Water Affairs, (2004), ODMP, draft inception report.
- Dincer T., Child S., Khupe B.B.J. (1987), A simple mathematical model of a complex hydrological system, Okavango Swamp, Botswana, *Journal of Hydrology*, **93**, 41-65
- Ellery W.N., Ellery K., Rogers K.H., McCarthy T.S. and Walker B.H. (1993), Vegetation, hydrology and sedimentation processes as determinants of channel form and dynamics in northeastern Okavango Delta, Botswana, *African Journal of Ecology*, **31**, 10-25
- Emamgholizadeh S. (2012), Neural network modeling of scour cone geometry around outlet in the pressure flushing, *Global NEST Journal*, **14**(4), 540-549.
- Furundzic D. (1997), Application example of neural networks for time series analysis: Rainfall- runoff modeling, *Signal Processing*, **64**, 383-396.
- Gieske A. (1997), Modelling outflow from the Jao/Boro River system in the Okavango Delta, Botswana, *Journal of Hydrology*, **193**, 214-239
- Hsu K., Gupta H.V., Gao X., Sorooshian S. and Imam B. (2002), Self-organizing linear output map (SOLO): An artificial neural network suitable for hydrologic modeling and analysis, *Water Resources Research*, **38**(12)(10.1029/2001 WR000795).
- Johnson V.M. and Rogers L.L. (2000), Accuracy of Neural Network Approximators in Simulation-Optimization, *Journal of Water Resources Planning and management*, **126**(2), 48-56.
- Kurugundla C.N., Moleele N.M. and Dikgola K. (2009), Flow Partitioning Within Okavango Delta-A Pre-requisite for Environmental Flow Assessment for Human Livelihoods and Sustainable Biodiversity Management. Unpublished.
- Lungu E.M. (1991), Stochastic characteristics of rainfall-runoff processes in southeast Botswana, *Hydrological Sciences Journal*, **36**(5), 423-434.
- Mazvimavi D., Wolski P. (2006), Long-term variations of annual flows of the Okavango and Zambezi Rivers, *Physics and Chemistry of the Earth*, **31**, 944-951.
- McCarthy T. S. and Ellery W. N. (1998), The Okavango Delta, *Trans. Roy. Soc. S. Afr.*, **53**, 157-182.
- Michaelides S.C., Pattichis C.S. and Kleovoulou G. (2001), Classification of rainfall variability by using Artificial Neural Networks, *International Journal of Climatology*, **21**, 1401-1414.

- Neural Network Toolbox (2002.), For Use with MATLAB, User's Guide Version 4, The Mathworks Inc.
- Parida B.P. and Moalafhi D.B. (2008), Regional rainfall frequency analysis for Botswana using L-Moments and radial basis function network, *Physics and Chemistry of the Earth*, **33**, 614-620.
- Parida B.P., Moalafhi D.B. and Kenabatho P.K. (2006), Forecasting runoff coefficients using ANN for water resources management: The case of Notwane catchment in Eastern Botswana, *Physics and Chemistry of the Earth*, **31**, 928-934.
- Pechlivanidis I.G., McIntyre N.R. and Wheater H.S. (2010), Calibration of the semi-distributed PDM rainfall-runoff model in the Upper Lee catchment, UK, *Journal of Hydrology*, **386**, 198-209.
- Pike A.S. and Scatena. F.N., (2010), Riparian indicators of flow frequency in a tropical montane stream network, *Journal of Hydrology*, **382**, 72-87.
- Shamseldin A.Y. (1997), Application of a neural network technique to rainfall-runoff modeling, *Journal of Hydrology*, **199**, 272-294.
- Shaw J.R. and Cooper D.J. (2008), Linkages among watersheds, stream reaches, and riparian vegetation in dryland ephemeral stream networks, *Journal of Hydrology*, **350**, 68-82.
- Sivakumar B., Jayawardena A.W. and Fernando T.M.K.G. (2002), River flow forecasting: use of phase-space reconstruction and artificial neural networks approaches, *Journal of Hydrology*, **256**, 225-245.
- Uvo C.B., Tölle U. and Berndtsson R. (2000), Forecasting discharge in Amazon using artificial neural networks, *International Journal of Climatology*, **20**, 1495-1507.
- Wheater H.S., Jakeman A.J., Beven K.J. (1993), Progress and directions in rainfallrunoff modelling. In: Modelling Change in Environmental Systems, Jakeman A.J., Beck M.B. and McAleer M.J. (eds), Wiley., 101-132.
- Wheater H.S. (2002). Hydrological Processes in arid and semi arid areas. In Wheater H.S. and Al-Weshah (eds). Hydrology of Wadi Systems (IHP Regional Network on Wadi hydrology in the Arab region: . IHP-V, Technical Documents in Hydrology, UNESCO, 55, 5-22.
- Wolski P., Savenije H., Murray-Hudson M. and Gumbrecht T. (2006), Modelling of the flooding in the Okavango Delta, Botswana, using a hybrid reservoir-GIS model, *Journal of Hydrology*, **331**, 58-72.
- Young W., Brandis K., Kingsford R. (2006), Modelling monthly streamflows in two Australian dryland rivers: Matching model complexity to spatial scale and data availability, *Journal of Hydrology*, **331**, 242-256.