Optimization of Parameters in Linear Muskingum Routing

Mehsa Mohammed , N Sajikumar
Department of Civil Engineering
Government Engineering College Thrissur, India
mehzas@gmail.com, saji@gectcr.ac.in

Abstract— **The Muskingum model is the most widely used method for flow routing. The effective utilisation of the routing procedure requires the determination of optimum parameters. The optimization of parameters in Muskingum routing method represented in the MATLAB Simulink module is attempted by means of different optimization procedures namely Genetic algorithm and Least square curve fit. The procedure is developed based on control system concept and developed using simulink tool of MATLAB. The result of optimization indicates that the methods can be used as an alternative way for parameter estimation.**

Keywords—control system; simulink; optimization toolbox; Muskingum method.

I. INTRODUCTION

Routing is used to predict the magnitudes, volumes, and temporal patterns of the flow (often a flood wave) as it moves down a channel. There are a number of conventional simplified approaches to flow routing [1]. For example, there is the Muskingum model with multiple inputs, multiple regression (MR) models, and autoregressive (AR) models [1]. A considerable amount of research has been published recently on the application of nonlinear methods in flow simulation. A nonlinear prediction approach was applied to multivariate flow routing and compared it successfully with ARMAX model forecasts [2]. Due to its simplicity and modest data requirements, the Muskingum method coming under the category of hydrologic routing becomes the most popular and widely used routing method. Though the method has the inherent problem of linearity, the method finds application in many rainfall-runoff models. The standard procedure for applying the Muskingum method involves two basic steps: calibration and prediction. In the calibration step, a parameter-estimation problem is solved in which the parameter values for the Muskingum model of a river are determined by using historical inflow-outflow hydrograph data. The prediction step is the solution of a routing problem in which the outflow hydrograph for a given inflow hydrograph is determined by using the routing equations [3].

The parameter-estimation problem has been the interest of many researchers for a long time. A lot of studies have been reported for the estimation of parameters of the Muskingum model. The early method for the linear model is based on a trial-and-error graphical approach which is considered as obsolete due to subjective interpretation. The least-squares method (LSM) was suggested in 1978 to solve the values of the three parameters in the nonlinear Muskingum method [4]. The three-parameter estimation procedures was proposed in 1985 using the Hook-Jeeve (HJ) pattern search technique in conjunction with simple linear regression (LR), the conjugate gradient (CG), and the Davidson-Fletcher Powell (DFP) techniques and used the state variable technique for routing[5]. The genetic algorithm (GA) was suggested in 1997 [6] and the Lagrange multiplier in 2004 [3] for parameter estimation. The Broyden-Fletcher-Goldfarb-Shanno (BFGS) technique was introduced in 2006 [7] etc. All these studies indicate that the solution procedures for the parameter estimation differ in many aspects and yield many different parameter sets for the same data. They also indicate that the non-linear Muskingum method for a particular reach could have many combinations of the parameters which yield almost same level of accuracy and hence multiple solutions exist. A representation of Muskingum models was proposed in MATLAB Simulink module [8]. Though the application of the model was successful, the parameter estimation of the model, the real advantage of such representation, was not successful in that study.

The routing component can be conveniently represented by a Simulink model of the Muskingum method [8]. However, application of such model requires that the parameter estimation should have to be carried out from the model itself. However such procedure was not successful in the earlier study [8]. Hence, in this study, optimization of simulink model parameters for linear Muskingum routing, is performed utilizing two different optimization procedures.

II. LITERATURE REVIEW

In general, there are two types of channel routing, hydrologic routing, and hydraulic routing. Hydrologic routing methods employ the continuity equation along with the equation of storage [9].

Hydrologic methods can effectively reproduce flood flows when a storage-discharge relation is calculated or routing coefficients are fitted to the storage-discharge relation. Muskingum routing is the most commonly used hydrologic routing method for handling a variable discharge-storage relationship. This method models the storage volume of flooding in a river channel by a combination of wedge and prism storages. During the advance of flood wave, inflow exceeds outflow, producing a wedge of storage. During the recession, outflow exceeds inflow resulting in a negative wedge. In addition, there is a prism of storage which is formed by a volume of constant cross section along the length of the prismatic channel [10]. There are two forms of Muskingum method; linear and nonlinear. In linear Muskingum method, the storage S in the routing reach is represented by the discharge-storage equation by means of two parameters x and k. x is a dimensionless factor that weights the influence of the inflow and outflow hydrograph to the storage within the reach. k is the travel time within the reach. In nonlinear form, an additional parameter m is utilized to represent nonlinearity.

One challenge in the application of the Muskingum model is that its parameters cannot be measured physically. There are several methods available for parameter estimation. Traditionally, the parameters are estimated by plotting accumulated storage versus weighted flow of a given reach. Three techniques to calibrate the parameters by using various curve-fitting methods was suggested in 1985 [5]. He utilized the Hooke-Jeeves (HJ) pattern search in conjunction with simple linear regression (LR), the conjugate gradient (CG), and the Davidon-Fletcher-Powell (DFP) algorithms. The performances of the methods were compared with Gill's method, and HJ+CG and HJ+DFP techniques yield better results [7].

The objective approach of genetic algorithm was proposed for the purpose of estimating the parameters of two nonlinear Muskingum routing models in 1997 [6]. The Broyden-Fletcher-Goldfarb-Shanno (BFGS) technique, which searches the solution area on the basis of mathematical gradients for estimating the parameters in the nonlinear Muskingum model as introduced in 2006 [7]. The Particle swarm optimization (PSO) was proposed in 2009 for the parameter estimation of the nonlinear Muskingum model [11]. The results demonstrate that PSO can achieve a high degree of accuracy to estimate the three parameters and these results in accurate predictions of outflow [11].

The Differential Evolution (DE) was proposed in 2012 for the parameter estimation of the nonlinear Muskingum model. The performance of DE is outstanding in the parameter estimation problem of the nonlinear Muskingum model [12]. The linear parameters of Muskingum flood routing was determined using control system concept [8]. A Simulink model was developed for linear Muskingum method. Then the parameter estimation was carried out, but the results were not promising and require further improvement. Hence, the current study was taken up for parameter estimation using two different approaches in the Simulink model of linear Muskingum method.

III. DATA USED FOR OPTIMIZATION

To evaluate the performance of optimization model for optimizing Muskingum parameters, three sets of data for routing from, [13], [14] and [9] are used, of which the first two sets of data had been utilized by Ref. [8]. The data consists of a set of observed inflow and outflow values.

IV. METHODOLOGY

A. Linear Muskingum Routing Model

Muskingum method utilizes continuity equation and a relationship connecting inflow, outflow and storage for the solution of routing problem. These equations in mathematical terms can be expressed as follows [9];

Continuity equation is

$$\frac{dS}{dt} = I - Q \qquad (1)$$

Storage equation is

$$S = k[xI + (1-x)Q] \qquad (2)$$

Substituting the value for S in the continuity equation

$$\frac{dS}{dt} = I - Q = k[\frac{dI}{dt} + (1-x)\frac{dQ}{dt} \qquad (3)$$

Then the terms are rearranged to get an equation for $\frac{dQ}{dt}$, so that an integrator could provide the unknown variable Q.

$$\frac{dQ}{dt} = \frac{\frac{I-Q}{k} - \frac{xdI}{dt}}{(1-x)} \qquad (4)$$

This system equation thus derived above was used to represent the model in Simulink using the various blocks from the Simulink Library Browser. The inflow data were provided as time series. The necessary blocks were gathered from the library browser and arranged in the order. The various blocks used were: from workspace, scope, derivative, constant, gain, divide, integrator, sum blocks etc**.** It may be noted that the equation is in an implicit form and has been incorporated with a feedback connection. Appropriate blocks have been utilised for building the right hand side of the equation and then an integrator is used for getting the variable Q, which is then fed back to the system. The parameters of the blocks were modified according to the system requirement.

The inflow data could be either typed in or imported to the MATLAB workspace [8]. A small modification is applied to the linear model developed by [8] by removing the Mux Block in the model which is used for combining several inputs into a single vector output. Only the outflow (Q) obtained from the loop was directly given to the scope. Also the values in the blocks corresponding to x and k parameters of Muskingum were given as x and k itself. After developing the model, optimization of Muskingum parameters was carried out by invoking optimization routine in the optimization toolbox of MATLAB with necessary M-Files for objective function. Fig.1 shows the representation of linear model in Simulink for parameter optimization. Separate M- File is required to be used for invoking optimization using the Simulink model for linear method. Such a methodology was not attempted earlier.

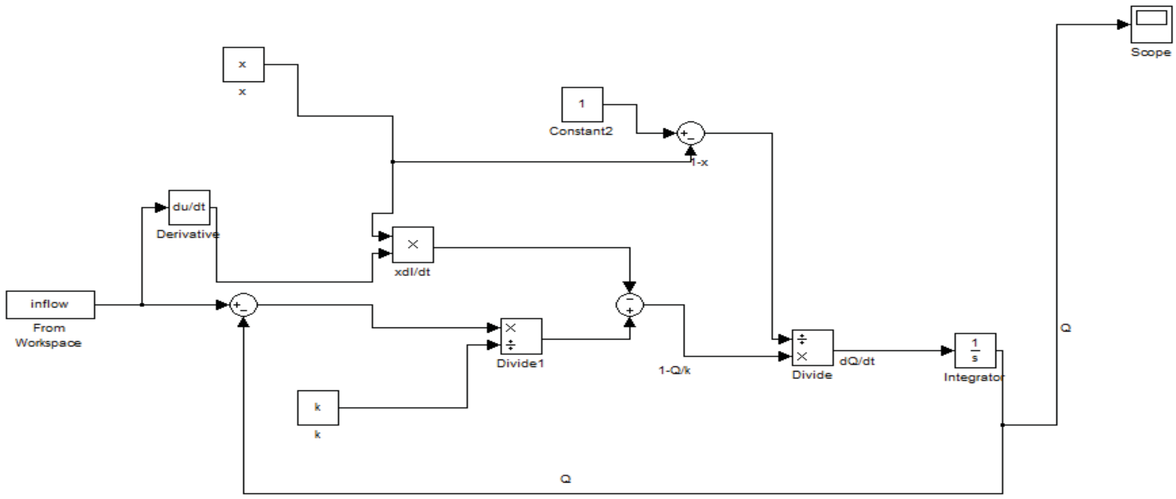

Fig.1 Representation of linear model in Simulink for parameter optimization

B. Optimization toolbox

Optimization Toolbox is a collection of functions thatextend the capability of MATLAB. The Lsqcurvefit solver and GA solver in the optimization toolbox are utilized in this study.

1) Least Square Curve fit (Lsqcurvefit) Solver

Lsqcurvefit solver solves the nonlinear curve-fitting (data-fitting) problems in least-squares sense. That is, with the given input data 'xdata', and the observed output 'ydata', it find the coefficients 'x' that "best-fit" the equation F(x, xdata), where xdata and ydata are vectors and F(x, xdata) is a vector valued function.. The solver demands certain data to be given by the user for performing optimization. These include algorithm, objective function, start point, xdata, ydata and the boundary constraints (lower bound and upper bound).

2) Genetic Algorithm Solver

The genetic algorithm is a method for solving both constrained and unconstrained optimization problems that is based on natural selection, the process that drives biological evolution. At each step, the genetic algorithm selects individuals at random from the current population to be parents and uses them to produce the children for the next generation. Over successive generations, the population "evolves" toward an optimal solution by means of selection process. Like the Lsqcurvefit solver, GA solver also demands certain inputs to be supplied by the user. These includes the fitness function, number of variables, boundary constraints (lower bound and upper bound) etc. Here also, the results as described in the case of Lsqcurvefit solver can be visualized.

C. Optimization Procedure

The first step is the representation of linear Muskingum method in Simulink. The optimization procedure adopted for optimizing the parameters in Muskingum routing method represented in the MATLAB Simulink module are discussed in detail in the subsequent sections.

1) Using Lsqcurvefit Solver

The most important step in the optimization is the formulation of objective function. A simple M-File was created for creating the objective function. Here the optimization was performed by finding the parameters of Muskingum routing that best fit the equation F (v, xdata) by least square curve fitting method. The result obtained from simulink modelling was saved in the structure format in the scope of the output in simulink. Also, it should be noted that the ydata and simulink output should be of same size. The parameters were defined in terms of a single variable 'v'. Then the 'option' for setting up the simulation in simulink was defined. The simulation was performed by using 'sim' command. The value obtained after each simulation was compared with the observed value and the parameter value obtained with the best fit would yield the optimized result. Thus the objective function formulation was completed. In the 'Start point', the initial guesses have to be provided.

TABLE I. BOUND CONSTRAINTS USED FOR DIFFERENT DATASETS

Model	Dataset	Parameters	Lower bound (LB)	Upper bound (UB)	Start point (SP)
Linear	Data 1	x, k	[0.1 , 10]	[0.4 ,40]	[0.25,25]
Linear	Data 2	x ,k	[0.1 , 10]	[0.4 ,40]	[0.25,25]
Linear	Data 3	x, k	[0.1 , 10]	[0.4 ,40]	[0.25,25]
Non-linear	Wilson (1974)	k, x, m	[0.01,0.2,1.5]	[0.2,0.3,2.5]	[0.105 ,0.25,2]

'Xdata' corresponds to the inflow values of data sets given in the structure format in the workspace of MATLAB and the same variable name was specified here. 'Ydata' corresponds to the observed outflow values of data sets. Constraints were the bounds (Lower and Upper) which should be provided according to the problem concerned. Table 1 show the bound constraints used. The start point was taken as the average of lower and upper bound values. The other options were left with the default values. Then the optimization was performed by clicking the 'Start'.

2) Using GA Solver

The fitness function for GA was minimization of the sum of squares of error between the observed and simulated value. The number of variables was to be given as 2 for linear. Constraints were only the lower and upper bound constraints. The same upper and lower bound values as in the case of Lsqcurvefit were provided here since the datasets were same. The other options were left with the default values. Then the optimization was performed by clicking the 'Start'.

V. RESULTS AND DISCUSSIONS

For optimization of linear model, first the model was represented in Simulink by utilizing the continuity and storage equations [8]. The optimization process was applied to the datasets. In Lsqcurvefit solver, the objective function was formulated in such a way that the value obtained after each simulation is compared with the observed value and the parameter values corresponding to the best fit will yield the optimized result. The problem set up corresponding to each dataset was made and optimization was run for each dataset. Thus after optimization process, the optimized values for x and k parameters were obtained for all the datasets. But during optimization, it was observed that for dataset 1, the value of 'x' parameter shows some tendency to align with the lower bound value. i.e., the x value always obtained as lower bound value. i.e., if the value is given as 0.0, the x value obtained as 0.0. It indicates that the parameter optimization has not been done in proper fashion. The reason behind such behaviour has to be ascertained. Such behaviour was not acceptable and hence the lower bound was fixed as 0.1.

Because the genetic algorithm uses random number generators, the algorithm returns slightly different results in each trial. Hence, a number of trials have been performed and the parameter values obtained corresponding to the least objective function value was taken as the result. The fitness function in GA was formulated in such a way that the sum of squares of error between the observed and simulated value should be minimum. The problem set up corresponding to each dataset was made and optimization was run for each dataset. Thus after optimization process, the optimized values for x and k parameters were obtained for the datasets considered. For dataset 1, the optimized parameter values obtained from both the optimization procedures were the same. Like in the previous case, it was observed that for dataset 1, the value of 'x' parameter shows some tendency to align with the lower bound value. The reason has to be ascertained. The computed and observed hydrographs were given in Fig.2. Table 2 shows the details of result of the optimization in dataset 1.

TABLE II RESULT OF OPTIMIZATION IN LINEAR

DATASET 1

Data set	Method	x	K	Number of Iterations Required	Objective Function Value
Data 1	Lsqcurvefit	0.1	45.465	7	4.581
Data 1	GA	0.1	45.465	51	4.581

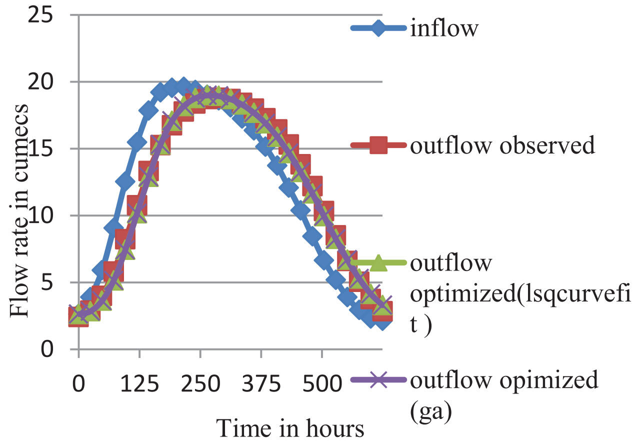

Fig.2 Graphical representation of the result for dataset 1

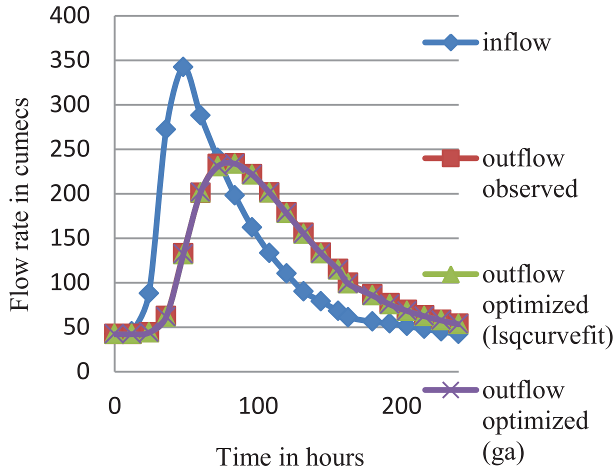

Fig.3 Graphical representation of the result for dataset 2

The objective function (for Lsqcurvefit) and fitness function (for GA) was formulated in the same manner as well for the second set. For second dataset, 'x' value obtained from both the methods were same. But there was a slight difference in 'k' value. The GA gave better value of objective function, i.e., lower value of objective function. It was observed that the computed and measured output matches very well (Fig.3). Table 3 shows the details of result of the optimization in dataset 2.

In the same manner, the optimization was run for dataset 3. The values of parameters obtained from both the methods were the same. The perfect match of computed and measured discharge (Fig.4) clearly showed that the performance of optimization was very good. Table 4 shows the details of the result of optimization in dataset 3. It may be noted that the optimization procedure worked well for second and third set while it misbehaved for dataset 1. The reason behind this behavior of the model is to be ascertained

TABLE III. RESULT OF OPTIMIZATION IN LINEAR DATASET 2

Data set	Method	x	K	Number of iterations required	Objective function value
Data 2	Lsqcurvefit	0.177	35.186	5	15.996
Data 2	GA	0.177	35.158	51	13.803

TABLE IV. RESULT OF OPTIMIZATION IN LINEAR DATASET 3

Data set	Method	x	K	Number of iterations required	Objective function value
Data 3	Lsqcurvefit	0.239	13.243	7	1.157
Data 3	GA	0.239	13.243	51	1.157

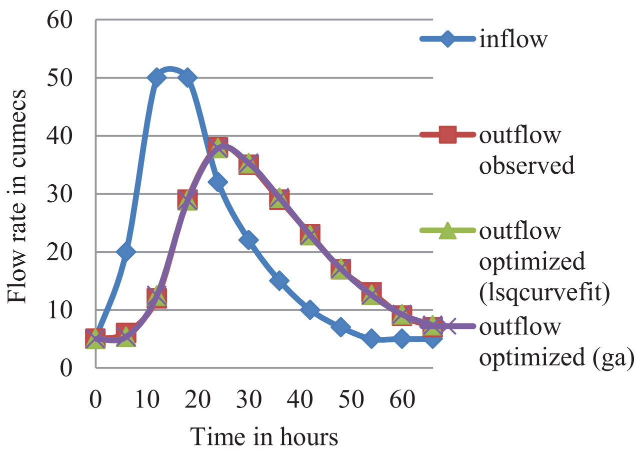

Fig.4 Graphical representation of the result for dataset 3

In order to ascertain the improper behaviour of the optimization model while using the dataset1, discharge storage curve was drawn for the three datasets by using the parameter values obtained from Lsqcurvefit optimization method. The discharge storage curve obtained for dataset 1, 2 and 3 are shown in Fig.5, Fig.6 and Fig.7 respectively. From these graphs, it was found that the plot for the first dataset showed some peculiar behaviour which was different from the other two datasets. i.e., the relationship was not linear as expected, rather it gave non-linear relationship. The dataset 1 could not be represented by a linear model. Under this circumstance, it becomes essential to build the non-linear model of Muskingum method. Hence, while applying the linear model its applicability needs to be verified.

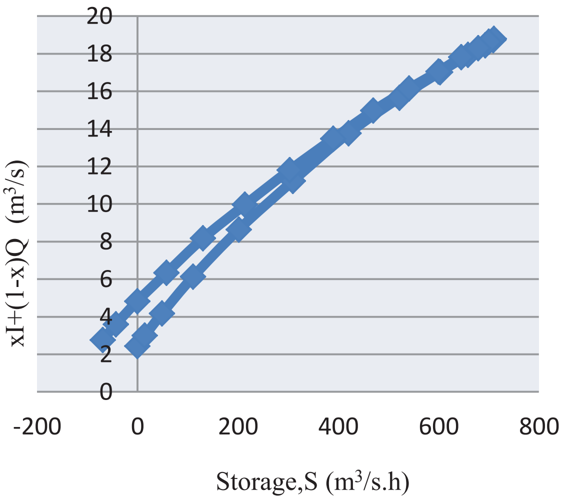

Fig. 5 Discharge storage curve for dataset 1

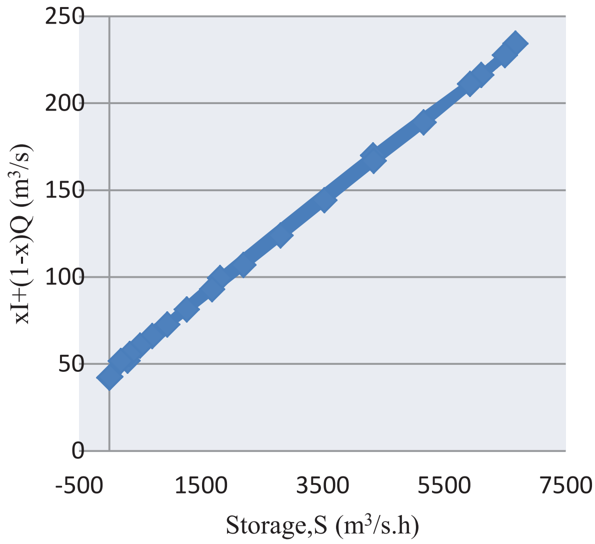

Fig. 6 Discharge storage curve for dataset 2

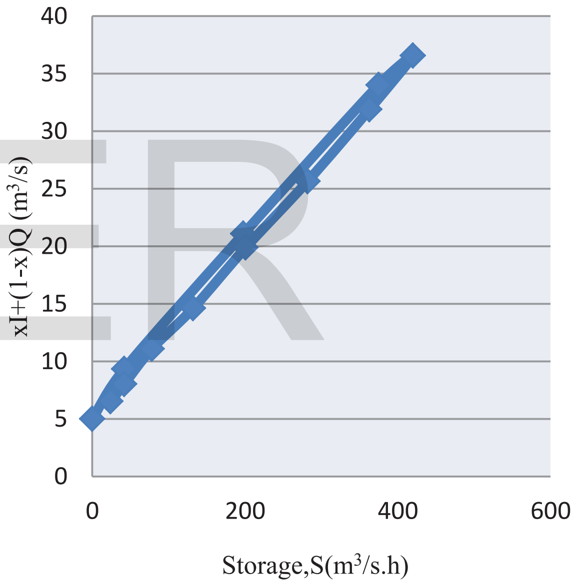

Fig. 7 Discharge storage curve for dataset 3

VI. CONCLUSION

Optimization of parameters in Muskingum routing methods represented in the MATLAB Simulink module was done by means of using two different optimization procedures. The optimization was performed on three sets of linear data. The results obtained from both the methods were analyzed and compared. The methodology can be extended to a situation where there many such routing component in network of channels and hence will be useful in practical sense.

The following are conclusions derived from the study.

- Muskingum linear model expressed in simulink could also be used for parameter estimation. This procedure, just like any other optimization for such a purpose, alleviates the subjective judgement in the determination of parameters of Muskingum method (especially in the case of graphical method).
- The Simulink representation of the model makes the optimisation procedure handy and easy.
- The blind application of linear Muskingum model to a non-linear dataset can lead to unrealistic solution and hence lead to unrealistic parameters while carrying out parameter optimization.

References

[1] Romanowicz R.J., and Ousch M., "An integrated data based mechanistic lowland catchment model for the upper narew," *Publications of the Institute of Geophysics Polish Academy of Sciences*, vol.9, pp.57-74, 2008.

[2] Porporato A., and Ridolfi L., "Multivariate nonlinear prediction of river flows," *Journal of Hydrology*, vol.248, pp.109– 122, 2001.

[3] Das A. ,"Parameter estimation for muskingum models," *Journal of Irrigation and Drainage Engineering*, vol.130, pp 140–147, April 2004.

[4] Gill M. A. ,"Flood routing by the muskingum method," *Journal of Hydrology,* vol.36, pp. 353–363,1978.

[5] Tung Y. K., "River flood routing by nonlinear muskingum method," *Journal of Hydraulic Eng*., vol.111, 1985.

[6] Mohan S., "Parameter estimation of nonlinear muskingum models using genetic algorithm," *Journal of Hydraulic Engineering*, ,vol.123, pp.137–142 , Feb. 1997.

[7] Geem Z. W. ,"Parameter estimation for the nonlinear muskingum model using the bfgs technique, "*Journal Irrigation Drain Eng*., vol.132, pp.474–478,2006.

[8] Gyncy I., "Hydrologic routing of river using control system concept", July 2013, A Thesis submitted to Calicut University.

[9] Subramanya K. , Engineering hydrology, Edition 3, *Tata Mc. Graw Hill*., New Delhi, 2008.

[10] Chow V.T., Maidment D. R. and Mays L.W., Applied hydrology, *McGraw-Hill Book Company*, 1988.

[11] Chu H. J. and Chang L. C. ,"Applying particle swarm optimization to parameter estimation of the nonlinear muskingum model," *Journal of Hydrology Engineering*, vol. 14, pp.1024–1027, September 2009.

[12] Dong-Mei Xu., Lin Qiu and Shou-Yu Chen ,"Estimation of nonlinear muskingum model parameter using differential evolution," *Journal of Hydrologic Engineering*, vol.17, pp 348-353, February 2012.

[13] Singh V.P., Engineering hydrology, *Edition 1, MacMillan Education*, Hampshire, UK, 1974.

[14] Raghunath H.M., Hydrology- principles, analysis, and design, *New Age International*, 2005.