An Ant Colony Optimization Approach for Estimating the Parameters of the Nonlinear Muskingum Flood Routing Model

Halil KARAHAN, Gurhan GURARSLAN, Ozgur BASKAN
Department of Civil Engineering, Pamukkale University, Denizli, Turkey, hkarahan@pau.edu.tr, gurarslan@pau.edu.tr, obaskan@pau.edu.tr

EXTENDED ABSTRACT

In this paper, an ant colony optimization approach for the parameter estimation of nonlinear Muskingum model is proposed. To prevent negativity of outflows and storages, an indirect penalty approach is imposed in the numerical solution of the model. The performance of the method is compared with reported techniques given in the literature through both synthetic and real-life example. The results obtained for both applications demonstrate that the proposed algorithm can confidently be applied to estimate optimal parameter values of the nonlinear Muskingum model.

1. INTRODUCTION

The ability to accurately predict the movement of flood waves is of vital importance to river *engineers and* managers. In the design of hydraulic structures, river improvement works, flood protection and warning schemes, knowledge of water levels and discharge is of the essence. This is reflected in the large amount of research effort which has been expended in the area of flood routing and river modelling over many years [Wormleaton and Karmegam, 1984].

Flow routing may be considered as an analysis to trace the flow thorough a hydrologic system, given the input. The difference between lumped and distributed system routing is that in a lumped system model, the flow is calculated as a function of space and time throughout the system. Routing by lumped system methods is sometimes called hydrologic routing, and routing by distributed systems methods is sometimes referred to as hydraulic routing [Chow et al., 1988].

Hydraulic methods of routing involve the numerical solutions of either the convective-diffusion equations or the one-dimensional Saint-Venant equations of gradually varied unsteady flow in open channels. Hydrological methods use the principle of continuity and a relationship between discharge and the temporary storage of excess volumes of water during the flood period. The hydraulic methods are more accurate than hydrological methods, but hydraulic methods are more complicated than hydrological methods. Also, hydraulic methods are required high demand on computing technology on quantity and quality of input data [Singh, 1988]. In practical applications, the hydrological routing methods are relatively simple to implement and reasonably accurate [Haktanir and Ozmen, 1997]. An example of a simple hydrological flood routing technique used in natural channels is the Muskingum flood routing method [Karahan, 2012].

The Muskingum method was first developed by U.S. Corps of Engineers for the flood control studies of the Muskingum River basin in Ohio [McCarthy, 1938]. 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 [Das, 2004].

The following hydrologic continuity and nonlinear storage equations are commonly used in the Muskingum model.

$$\frac{dS_t}{dt} = I_t - O_t \tag{1}$$

$$S_t = K\left[\chi I_t + (1-\chi) O_t\right]^m \tag{2}$$

In the above equations S_t [L^3], I_t [L^3/T], and O_t [L^3/T] are simultaneous amounts of storage, inflow and outflow, respectively, at time t; K [$L^{3(1-m)}T^m$] is storage-time constant and is greater than 0, and χ is a weighting factor usually varying between 0 and 0.5 [Tung, 1985]; m is an exponent for considering the effects of nonlinearity and is greater than 1 for nonlinear models (the original linear model can be a special case of the nonlinear model where m is equal to 1). In the model, K, χ, and m are unknown parameters and S_t and O_t must be handled as non-negative variables. Several mathematical techniques, such as segmented least-squares method [Gill, 1978], hybrid of pattern search and local search [Tung, 1985], nonlinear least-squares method [Yoon and Padmanabhan, 1993], Lagrange multiplier method [Das, 2004] and Broyden-Fletcher-Goldfarb-Shanno method [Geem, 2006] have been applied for estimating three parameter values of the model. However, these techniques have drawbacks of complex derivative requirement and/or good initial vector assumption [Geem, 2011]. Thus, last decade, several researchers have also proposed various heuristic algorithms such as genetic algorithm [Mohan, 1997; Karahan and Gurarslan, 2011], harmony search [Kim et al., 2001; Geem, 2011; Karahan et al., 2013], particle swarm optimization [Chu and Chang, 2009; Gurarslan and Karahan, 2011], differential evolution [Xu et al., 2012; Karahan and Gurarslan, 2013], Nelder-Mead simplex [Barati, 2011; Karahan, 2013] and immune clonal selection [Luo and Xie, 2010] to the parameter estimation of nonlinear Muskingum model.

In this paper, a novel optimal parameter estimation method for the nonlinear Muskingum model is proposed. In the proposed method, to prevent negativity of outflows and storages, a penalty term approach is applied in the numerical solution of the model. We used Ant Colony Optimization Reduced Search Space (ACORSES) algorithm [Baskan et al., 2009] in the model with a numerical routing procedure given by Tung [1985]. The proposed algorithm finds the best solution regardless of the initial parameter values with fast convergence and few control parameters. The performance of the method is compared with the reported techniques given in the literature through Wilson data and River Wye 1960 December flood data. The results demonstrate that the proposed algorithm can confidently be applied to estimate best parameter values of the nonlinear Muskingum model.

2. ROUTING PROCEDURE OF THE NONLINEAR MUSKINGUM MODEL

Rearranging Eq. (2), the rate of outflow O_t can be obtained as [Tung, 1985],

$$O_t = \left(\frac{1}{1-\chi}\right)\left(\frac{S_t}{K}\right)^{1/m} - \left(\frac{\chi}{1-\chi}\right) I_t \tag{3}$$

Combining Eq. (3) and the continuity Eq. (1), the state equation can be obtained as,

$$\frac{dS_t}{dt} = -\left(\frac{1}{1-\chi}\right)\left(\frac{S_t}{K}\right)^{1/m} + \left(\frac{1}{1-\chi}\right) I_t \tag{4}$$

$$S_{t+1} = S_t + \Delta S_t \tag{5}$$

Note that the unit time step is used in the Eq. (5). If infeasible values of K, χ, m are selected, negative values of O_t and S_t can be obtained in the numerical solution of Muskingum model.

Hence, an indirect penalty function approach [Gurarslan and Karahan, 2011; Karahan et al., 2013] is imposed to the numerical solution in order to prevent negativity (outflows and storages cannot be negative quantities) as follows:

$$S^*_{t+1} = \lambda_1 \left| S_{t+1} \right| \ \ , \ if \ \ \mathrm{S}_{t+1} < 0 \tag{6.a}$$

$$O^*_{t+1} = \lambda_2 \left| O_{t+1} \right| \ , \ if \ \ O_{t+1} < 0 \tag{6.b}$$

where, λ_1 and λ_2 are penalty constants (used as 1000 for this study), S^*_{t+1} is penalized next storage, O^*_{t+1} is penalized next outflow. Note that, S^*_{t+1} and O^*_{t+1} are positive but unrealistic values. The zero-order numerical routing procedure given by Tung [1985] is preferred for making a comparison and is slightly modified for preventing negativity of outflows and storages. Main steps of this procedure are given as follows:

Step 1: Assume values for three parameters K, χ and m.
Step 2: Calculate S_t using Eq. (2), where initial outflow is the same as the initial inflow.
Step 3: Calculate the time rate of change of storage volume using Eq. (4).
Step 4: Estimate the next accumulated storage (S_{t+1}) using Eq. (5) or Eq. (6.a).
Step 5: Calculate next outflow (O_{t+1}) using Eq. (3) or Eq. (6.b).
Step 6: Repeat steps 2-5 for all times.

3. ANT COLONY OPTIMIZATION REDUCED SEARCH SPACE (ACORSES) ALGORITHM

Ant Colony Optimization (ACO) is a part of the swarm intelligence and has been developed by Dorigo et al. [1996] based on the fact that ants are able to find the shortest path between their nest and food sources. ACO belongs to the class of biologically inspired heuristics. The procedure of the ACO algorithms simulates the decision-making processes of ant colonies as they forage for food and is similar to other artificial intelligent techniques. ACO is the one of the most recent meta-heuristic technique that uses artificial ants to find solutions to optimization problems. The main idea is that it is indirect local communication among the individuals of a population of artificial ants. In nature, an individual ant is unable to communicate, but as a group, ants posses the ability to collect food for their colony [Bell and McMullen, 2004]. The core of ant's behaviour is the communication between the ants by means of chemical pheromone trails, which enables them to find shortest paths between their nest and food sources. The role of pheromone is to guide the other ants towards the target points [Baskan et al., 2009].

Although there are many studies in literature with different heuristic methods to estimate the parameters the Muskingum flood routing model, there is no application of ACO to this area. Thus, in this study, a heuristic algorithm named as ACORSES proposed by Baskan et al. [Baskan et al., 2009] was used to estimate the parameters of nonlinear Muskingum model. The ACORSES is based on each ant searches only around the best solution of the previous iteration with β. The ACORSES differs from other ACO's in that its feasible search space (FSS) is reduced with β and its best solution obtained from the previous iteration. In ACORSES model, ants search randomly the solution within the FSS to reach global or near global optimum values. At the end of the each iteration, only one of the ants is near to global optimum. After the first iteration, when global optimum is searched around the best solution of the previous iteration using β, the algorithm will quickly reach to the global optimum [Baskan et al., 2012]. The ACORSES is consisted of three main phases; Initialization, pheromone update and solution phase as can be seen in Figure 1.

At the beginning of the first iteration, all ants search randomly to the best solution of a given problem within the FSS, and old ant colony is created at initialization phase. After that, quantity of pheromone is updated. In the solution phase, new ant colony is created based on the best solution from the old ant colony using Equation (7) and (8). Then, the best solutions of two colonies are compared. At the end of the first iteration, FSS is reduced by β, where β is a vector, and best solution obtained from the previous iteration. The range of the β may be chosen between minimum and maximum bounds of any given problem as proposed by Baskan et al. [2009]. The solution vector of the each ant is updated using following expression:

Initialization

FOR i=1 TO I (I=iteration number)
IF i=1 THEN generate m random ants within FSS
ELSE reduce FSS with range [$x_{t-1}^{best} + \beta; x_{t-1}^{best} - \beta$]
END IF
FOR i=1 TO m
Determine $f(x_t^{best})$ (old ant colony)
Save x_t^{best}
END

Pheromone update

Pheromone evaporation using (9)
Update pheromone trail using (10)

Solution phase

Determine search direction using (8)
Generate the values of α vector

FOR i=1 TO m
Determine the values of new colony using (7)
Determine $f(x_t^{best})$ (new ant colony)
Save x_t^{best}
END

IF $f(x_t^{best})^{new} \leq f(x_t^{best})^{old}$ THEN $x^{global\min} = (x_t^{best})^{new}$
ELSE $x^{global\min} = (x_t^{best})^{old}$
END IF
$\alpha_t = \alpha_{t-1} * 0.99$
$\beta_t = \beta_{t-1} * 0.99$
END

Figure 1: *Steps of ACORSES [Baskan et al., 2009].*

$$x_t^{k(new)} = x_t^{k(old)} \pm \alpha \qquad (t = 1,2,.....,I) \tag{7}$$

where $x_t^{k(new)}$ is the solution vector of the kth ant at iteration t, $x_t^{k(old)}$ is the solution obtained from the previous step at iteration t, and α is a vector generated randomly to determine the length of jump.

In Eq. (7), (+) sign is used when point x_t^k is on the left of the best solution on the x coordinate axis. (-) sign is used when point x_t^k is on the right of the best solution on the same axis. The direction of search is defined by Eq. (8).

$$\bar{x}_t^{best} = x_t^{best} + (x_t^{best} * 0.01) \tag{8}$$

If $f(\overline{x}_t^{best}) \le f(x_t^{best})$, (+) sign is used in Eq. (7). Otherwise, (-) sign is used. (+) sign defines the search direction to reach to the global optimum. α value is used to define the length of jump, and it will be gradually decreased in order not to pass over global optimum. At the end of the each iteration, a new ant colony is developed as the number of colony size that is generated at the beginning of the each iteration. Any of the newly created solution vectors may be outside the reduced search space that is created at the beginning of the each iteration. Therefore, created new ant colony prevents being trapped in bad local optimum [Baskan et al., 2009]. Quantity of pheromone is reduced to simulate the evaporation process of real ant colonies using Eq. (9) in the pheromone update phase. After reducing of the number of pheromone, it is updated using Eq. (10).

$$\tau_t = 0.1 * \tau_{t-1} \tag{9}$$

$$\tau_t = \tau_{t-1} + 0.01 * f(x_{t-1}^{best}) \tag{10}$$

4. NUMERICAL APPLICATIONS

In order to fairly compare the results, two examples are solved through the developed model with Wilson data [Wilson, 1974] and River Wye December 1960 flood data [NERC, 1975]. The objective function to be minimized is the sum of the squared residuals (SSQ) between observed and calculated outflows as follows:

$$\text{minimize } SSQ = \sum_{t=1}^{N} \left[O_t - \hat{O}_t \left(K, \chi, m \right) \right]^2 \tag{11}$$

where, O_t denotes the observed outflow, $\hat{O}_t$ denotes the calculated outflow and N denotes the number of time step. The ranges of three parameters used in the applications are selected as $K = 0.0 - 1.0$, $\chi = 0.0 - 0.5$ and $m = 1.0 - 3.0$. The stopping condition was selected as a fixed iteration number. ACORSES algorithm was run 100 times both applications to achieve average results. For each run, the initial population was randomly created by means of using different seed numbers. In the ACORSES algorithm, required iteration number and colony size were selected according to the results of the sensitivity analysis with Wilson data.

5. APPLICATION TO DATA SET GIVEN BY WILSON

The parameter estimation technique for the nonlinear Muskingum model using ACORSES algorithm is applied to data set given by Wilson [1974], which has been studied by Mohan [1997], Kim et al. [2001], Das [2004], Geem [2006], Luo and Xie [2010] , Geem [2011], Barati [2011], Xu et al. [2012] and Karahan et al. [2013].

In order to determine the suitable maximum iteration number and colony size values, a sensitivity analysis was carried out thorough data set given by Wilson [1974]. Statistical evaluation of the parameters, SSQ's, iteration number and CPU times for different colony sizes (*CS*) is presented in Table 1. The suitable *CS* value is selected as 10 for this problem in terms of requiring less CPU time. As can clearly be seen in Table 1, there is extremely small difference between the best solution and worst solution of 100 model runs. Standard deviation of SSQ's of 100 runs is 1.17E-13 for *CS=10.* Global optimum is always found for all *CS* values in the 100 model runs. The proposed algorithm is free from infeasible starting vectors and computational divergence. Required CPU time of the proposed algorithm for 2,000 iterations is 6.15 second.

Table 2 presents the statistical evaluation of the parameters, SSQ's and CPU times according to the different fixed iteration number. As can be seen in Table 2, there was not a significant change in the results after 2,000 iterations. Thus, this fixed iteration value is appropriate for a stopping criteria in both applications.

Table 3 shows the comparison of the best parameters and SSQ values for Wilson data obtained from various techniques such as genetic algorithm (GA), harmony search (HS), Broyden-Fletcher-Goldfarb-Shanno (BFGS) technique, parameter-setting-free harmony search (PSFHS), differential evolution (DE), hybrid harmony search BFGS (HS-BFGS) algorithm and ACORSES algorithm. As can be given in Table 3, the best parameters (0.086249, 0.286917, 1.868087) and the corresponding SSQ value (36.767888) obtained by ACORSES algorithm is better than all techniques given in the literature. CPU time of the proposed algorithm for 2000 iteration is only 6.15 second.

Computed outflows of Wilson data which were obtained by different methods are given in Table 4. As can be seen in Table 4, computed outflows from ACORSES are better than the other methods except that HS-BFGS. ACORSES and HS-BFGS algorithms give same results.

The comparison of the observed and computed hydrograph of Wilson data for the best solution vector is presented in Figure 2. As can be seen in Figure 2, computed hydrograph is well suited to the observed hydrograph.

Table 1: Statistical evaluation of the parameters, SSQ's, iteration number and CPU times for different colony sizes CS values (2,000 iteration).

CS	**Statistical Values**	***K***	χ	***m***	**SSQ**	**CPU**
10	Best	0.086249	0.286917	1.868087	36.767888	6.15
	Worst	0.086249	0.286917	1.868087	36.767888	6.72
	Mean	0.086249	0.286917	1.868087	36.767888	6.34
	Std. Dev.	3.59E-09	1.65E-09	9.20E-09	1.17E-13	0.13
20	Best	0.086249	0.286917	1.868087	36.767888	10.84
	Worst	0.086249	0.286917	1.868087	36.767888	12.39
	Mean	0.086249	0.286917	1.868087	36.767888	11.77
	Std. Dev.	2.99E-09	1.31E-09	7.67E-09	9.42E-14	0.36
30	Best	0.086249	0.286917	1.868087	36.767888	17.00
	Worst	0.086249	0.286917	1.868087	36.767888	18.78
	Mean	0.086249	0.286917	1.868087	36.767888	17.85
	Std. Dev.	3.19E-09	1.48E-09	8.20E-09	1.09E-13	0.38
40	Best	0.086249	0.286917	1.868087	36.767888	23.03
	Worst	0.086249	0.286917	1.868087	36.767888	26.47
	Mean	0.086249	0.286917	1.868087	36.767888	24.25
	Std. Dev.	2.96E-09	1.27E-09	7.66E-09	9.63E-14	0.45
50	Best	0.086249	0.286917	1.868087	36.767888	29.81
	Worst	0.086249	0.286917	1.868087	36.767888	32.79
	Mean	0.086249	0.286917	1.868087	36.767888	30.53
	Std. Dev.	2.95E-09	1.44E-09	7.54E-09	9.94E-14	0.46

Table 2: Statistical evaluation of the parameters, SSQ's and CPU times for different iteration number (CS=10).

Iteration Number	Statistical Values	K	χ	m	SSQ	CPU(s)
1000	Best	0.086210	0.286902	1.868027	36.767889	2.95
	Worst	0.086273	0.286928	1.868187	36.767894	4.01
	Mean	0.086249	0.286915	1.868087	36.767890	3.36
	Std. Dev.	1.22E-05	5.81E-06	3.15E-05	1.13E-06	0.23
1500	Best	0.086249	0.286916	1.868087	36.767888	4.60
	Worst	0.086249	0.286917	1.868088	36.767888	5.12
	Mean	0.086249	0.286917	1.868087	36.767888	4.75
	Std. Dev.	7.92E-08	3.85E-08	2.03E-07	4.92E-11	0.09
2000	Best	0.086249	0.286917	1.868087	36.767888	6.10
	Worst	0.086249	0.286917	1.868087	36.767888	7.33
	Mean	0.086249	0.286917	1.868087	36.767888	6.40
	Std. Dev.	3.45E-09	1.37E-09	8.90E-09	1.50E-13	0.18
2500	Best	0.086249	0.286917	1.868087	36.767888	7.68
	Worst	0.086249	0.286917	1.868087	36.767888	8.19
	Mean	0.086249	0.286917	1.868087	36.767888	7.85
	Std. Dev.	3.51E-09	1.58E-09	8.99E-09	1.02E-13	0.09
3000	Best	0.086249	0.286917	1.868087	36.767888	9.14
	Worst	0.086249	0.286917	1.868087	36.767888	10.48
	Mean	0.086249	0.286917	1.868087	36.767888	9.37
	Std. Dev.	3.50E-09	1.64E-09	8.96E-09	1.09E-13	0.18

Table 3: Comparison of the best parameter and SSQ values obtained by diffferent methods for Wilson data.

Method	K	χ	m	SSQ
GA	0.1033	0.2813	1.8282	38.2363
HS	0.0883	0.2873	1.8630	36.7829
BFGS	0.0863	0.2869	1.8679	36.7679
PSF-HS	0.0864	0.2869	1.8677	36.7680
DE	0.5175	0.2869	1.868	36.77
HS-BFGS	0.086249	0.286917	1.868088	36.767888
ACORSES	0.086249	0.286917	1.868088	36.767888

Table 4: Comparison of the observed and computed outflows for Wilson data.

Time (h)	I_t (cms)	O_t (cms)	Computed outflows (cms) GA	HS	BFGS	PSF -HS	DE	HS-BFGS	ACORSES
0	22	22	22.0	22.0	22.0	22.0	22.0	22.0	22.0
6	23	21	22.0	22.0	22.0	22.0	22.0	22.0	22.0
12	35	21	22.4	22.4	22.4	22.4	22.4	22.4	22.4
18	71	26	26.4	26.6	26.6	26.6	26.6	26.6	26.6
24	103	34	34.2	34.4	34.5	34.5	34.5	34.5	34.5
30	111	44	44.2	44.1	44.2	44.2	44.2	44.2	44.2
36	109	55	57.0	56.8	56.9	56.9	56.9	56.9	56.9
42	100	66	68.2	68.1	68.1	68.1	68.1	68.1	68.1
48	86	75	77.2	77.1	77.1	77.1	77.1	77.1	77.1
54	71	82	83.3	83.3	83.3	83.3	83.3	83.3	83.3
60	59	85	85.7	85.9	85.9	85.9	85.9	85.9	85.9
66	47	84	84.2	84.5	84.5	84.5	84.5	84.5	84.5
72	39	80	80.2	80.6	80.6	80.6	80.6	80.6	80.6
78	32	73	73.3	73.7	73.7	73.7	73.7	73.7	73.7
84	28	64	65.1	65.4	65.4	65.4	65.4	65.4	65.4
90	24	54	55.8	56.0	56.0	56.0	56.0	56.0	56.0
96	22	44	46.7	46.7	46.7	46.7	46.7	46.6	46.6
102	21	36	38.0	37.8	37.7	37.8	37.8	37.7	37.7
108	20	30	30.9	30.5	30.5	30.5	30.5	30.4	30.4
114	19	25	25.7	25.3	25.2	25.2	25.2	25.2	25.2
120	19	22	22.2	21.8	21.7	21.7	21.7	21.7	21.7
126	18	19	20.3	20.0	20.0	20.0	20.0	20.0	20.0

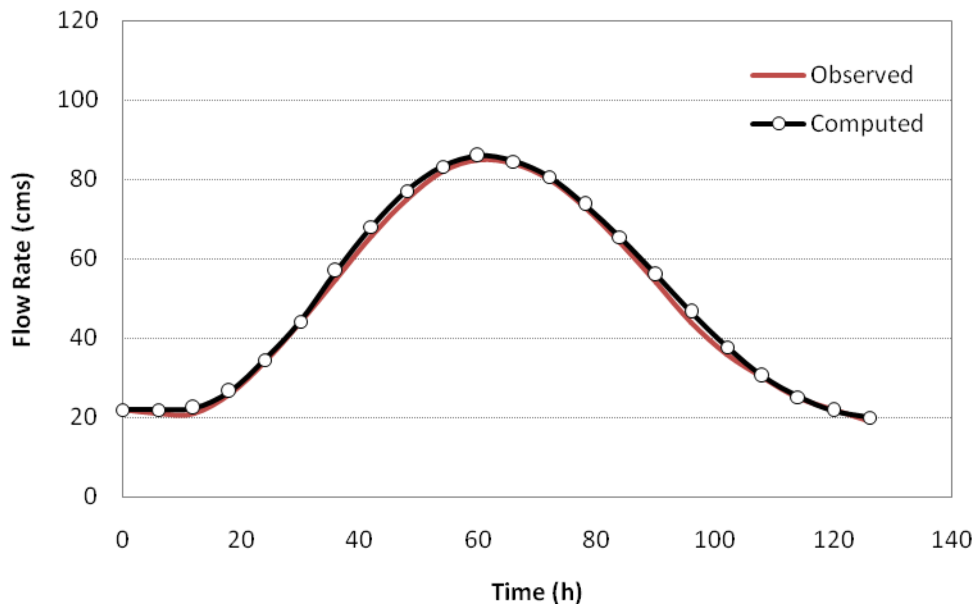

Figure 2: Comparison of the observed and computed hydrograph of Wilson data for the best solution vector.

6. APPLICATION TO RIVER WYE DECEMBER 1960 FLOOD

An example of the 1960 flood in the River Wye in the UK is presented here [NERC, 1975]. The 69.75 km stretch of the River Wye from Erwood to Belmont has no tributaries and very small lateral inflow. It is, thus, an excellent example to demonstrate the use of flood routing techniques [Bajracharya and Barry, 1997].

Computed outflows of River Wye December 1960 Flood which were obtained by a three-parameter linear Muskingum model considering lateral flow (LMM-L) [O'Donnell, 1985] and ACORSES methods are given in Table 5. As can be seen in Table 5, computed outflows from ACORSES are better than the LMM-L method. The comparison of the observed and computed hydrograph of these two methods is presented in Figure 3. As can be seen in Figure 3, computed hydrograph obtained by ACORSES is well suited to the observed hydrograph.

The best parameters (0.079235, 0.409238, 1.581483) and the corresponding SSQ value (37944.14) obtained by ACORSES algorithm is better than LMM-L method. CPU time of the proposed algorithm for 2000 iteration is only 8.16 second. The SSQ value of LMM-L method is computed as 251802 for the given parameters. ACORSES and HS-BFGS algorithms give same results for both examples. It can be said that HS-BFGS is faster than ACORSES for obtaining global optimum. But, HS-BFGS needs a good knowledge of gradient-based computations.

Table 5: Comparison of the observed and computed outflows for River Wye December 1960 flood.

Time	I_t	O_t	Computed outflows (cms)		
(h)	(cms)	(cms)	LMM-L	HS-BFGS	ACORSES
0	154	102	102	154	154
6	150	140	116	154	154
12	219	169	120	152	152
18	182	190	147	181	181
24	182	209	158	191	191
30	192	218	165	185	185
36	165	210	176	187	187
42	150	194	178	179	179
48	128	172	176	162	162
54	168	149	164	141	141
60	260	136	160	154	154
66	471	228	167	198	198
72	717	303	218	264	264
78	1092	366	303	344	344
84	1145	456	484	416	416
90	600	615	690	599	599
96	365	830	700	871	871
102	277	969	642	834	834
108	227	665	572	689	689
114	187	519	505	535	535
120	161	444	442	397	397
126	143	321	386	283	283
132	126	208	338	202	202
138	115	176	296	152	152
144	102	148	260	124	124
150	93	125	228	106	106

Table 5: Comparison of the observed and computed outflows for River Wye December 1960 flood (continued).

Time (h)	I_t (cms)	O_t (cms)	Computed outflows (cms)		
			LMM-L	HS-BFGS	ACORSES
156	88	114	201	94	94
162	82	106	179	88	88
168	76	97	160	82	82
174	73	89	144	75	75
180	70	81	130	73	73
186	67	76	118	69	69
192	63	71	109	66	66
198	59	66	100	62	62

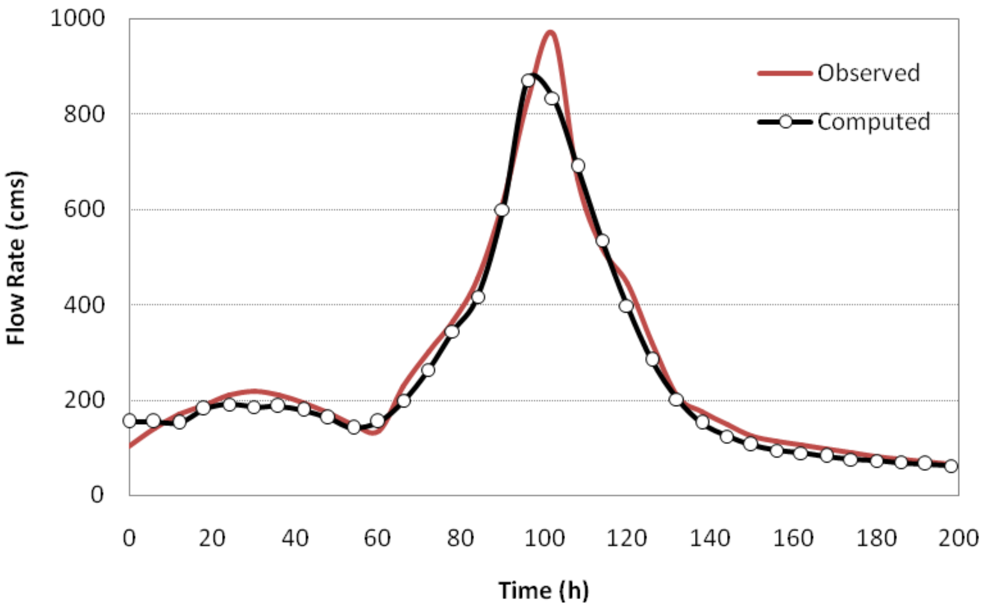

Figure 3: Comparison to observed and computed hydrograph of River Wye December 1960 Flood data.

7. CONCLUSIONS

This study proposes an ant colony algorithm for a nonlinear Muskingum flood routing model. The proposed algorithm overcame the disadvantages of mathematical techniques (initial vector setting, local optima and diverging). In each different model run, true global optimum is always obtained. In the proposed method, non-negativity restrictions are imposed on the model with an effective indirect penalty approach. ACORSES algorithm found the best solution among different methods given in the literature. The proposed algorithm is very robust. The results obtained demonstrate that the ACORSES algorithm can confidently be applied to estimate optimal parameter values of the nonlinear Muskingum model.

REFERENCES

Bajracharya, K., and Barry, D.A. (1997). Accuracy criteria for linearised diffusion wave flood routing, *Journal Hydrology*, 195:200-217.

Barati, R. (2011). Parameter estimation of nonlinear Muskingum models using Nelder-Mead simplex algorithm, *Journal of Hydrologic Engineering*, 16(11):946-954.

Baskan, O., Haldenbilen, S., Ceylan, H., and Ceylan, H. (2009). A new solution algorithm for improving performance of ant colony optimization, *Applied Mathematics and Compututation*, 211:75-84.

Baskan, O., Haldenbilen, S., Ceylan, H., and Ceylan, H. (2012). Estimating transport energy demand using ant colony optimization, *Energy Sources Part B: Economics, Planning, and Policy*, 7(2):188-199.

Bell, J.E., and McMullen, P.R. (2004). Ant colony optimization techniques for the vehicle routing problem", *Advanced Engineering Informatics*, 18:41-48.

Chow, V.T., Maidment, D.R., and Mays, L.W. (1988). *Applied Hydrology*, McGraw-Hill, New York.

Chu, H.J., and Chang, L.C. (2009). Applying particle swarm optimization to parameter estimation of the nonlinear Muskingum model, *Journal of Hydrologic Engineering*, 14:1024-1027.

Das, A. (2004). Parameter estimation of Muskingum models, *Journal of Irrigation and Drainage Engineering*, 130:140-147.

Dorigo, M., Maniezzo, V., and Colorni, A. (1996). Optimization by a colony of cooperating agents, *IEEE Transactions on Systems, Man, and Cybernetics-Part B*, 26(1):29-41.

Geem, Z.W. (2006). Parameter estimation for the nonlinear Muskingum model using the BFGS technique, *Journal of Irrigation and Drainage Engineering*, 132: 474-478.

Geem, Z.W. (2011). Parameter estimation of the nonlinear Muskingum model using parameter-setting-free harmony search algorithm, *Journal of Hydrologic Engineering*, 16(8):684-688.

Gill, M.A. (1978). Flood routing by Muskingum method, *Journal of Hydrology*, 36:353-363.

Gurarslan, G., and Karahan, H. (2011). A parameter estimation technique for the nonlinear Muskingum flood routing model, *6th EWRA International Symposium-Water Engineering and Management in a Changing Environment*, Catania, Italy.

Haktanir T., and Ozmen, H. (1997). Comparison of hydraulic and hydrologic routing on three long reservoirs, *Journal of Hydraulic Engineering*, 123: 153-156.

Karahan, H. (2012). Predicting Muskingum flood routing parameters using spreadsheets, *Computer Applications in Engineering Education*, 20(2):280-286.

Karahan, H. (2013). Discussion of "Parameter estimation of nonlinear Muskingum models using Nelder-Mead simplex algorithm" by Reza Barati, *Journal of Hydrologic Engineering*, 18(3): 365-367.

Karahan, H., and Gurarslan, G. (2011). Identification of the parameters of the nonlinear Muskingum flood routing model by using genetic algorithm (in Turkish), *5th National Symposium on Water Engineering*, Istanbul, Turkey.

Karahan, H., and Gurarslan, G. (2013). Discussion of "Estimation of nonlinear Muskingum model parameter using differential evolution" by Dong-Mei Xu, Lin Qiu and Shou-Yu Chen, *Journal of Hydrologic Engineering*, 10.1061/(ASCE)HE.1943-5584.0000641 (in press).

Karahan, H., Gurarslan, G., and Geem, Z.W. (2013). Parameter Estimation of the nonlinear Muskingum flood routing model using a hybrid harmony search algorithm, *Journal of Hydrologic Engineering*, 18(3):352-360.

Kim, J.H., Geem, Z.W., and Kim, E.S. (2001). Parameter estimation of the nonlinear Muskingum model using harmony search, *Journal of American Water Resources Association*, 37:1131-1138.

Luo, J., and Xie, J. (2010). Parameter estimation for the nonlinear Muskingum model based on immune clonal selection algorithm, *Journal of Hydrologic Engineering*, 15(10):844-851.

McCarthy, G.T. (1938). The unit hydrograph and flood routing, *Conference of North Atlantic Division*, U.S. Army Corps of Engineers.

Mohan, S. (1997). Parameter estimation of nonlinear Muskingum models using genetic algorithm, *Journal of Hydraulic Engineering*, 123:137-142.

Natural Environment Research Council (NERC) (1975). *Flood Studies Report, Vol. 3*", Institute of Hydrology, Wallingford, UK.

O'Donnell, T. (1985). A direct three-parameter Muskingum procedure incorporating lateral inflow, *Hydrological Sciences Journal*, 30(4):479-496.

Singh, V.P. (1988). Hydrologic systems, vol. 1: rainfall-runoff modelling, Prentice Hall, NJ..

Tung, Y.K. (1985). River flood routing by nonlinear Muskingum method, *Journal of Hydraulic Engineering*, 111:1447-1460.

Wilson, E.M. (1974). *Engineering Hydrology*, MacMillan Education Ltd., Hampshire, U.K.

Wormleaton P.R., and Karmegam M. (1984). Parameter optimization in flood routing, *Journal of hydraulic engineering*, 110:1799-1814.

Xu, D.M., Qiu, L., and Chen, S.Y. (2012). Estimation of nonlinear Muskingum model parameter using differential evolution, *Journal of Hydrologic Engineering*, 17(2):348-353.

Yoon, J.W., and Padmanabhan, G. (1993). Parameter-estimation of linear and nonlinear Muskingum models, *Journal of Water Resources Planning and Management*, 119:600-610.