Journal of Advanced Informatics in Water, Soil, and Structure 2025 1 (1) 47-61

Journal of Advanced Informatics in Water, Soil, and Structure

journal homepage: www.wss.torbath.ac.ir

Research Article

Performance Evaluation of Different Numerical Schemes for Muskingum Flood Routing (Case study: Karoon River, Iran)

Badfar, M.[a], Barati, R.[b,*], Dogan, E.[a], Tayfur,G.[c]

[a] *Department of Civil Engineering, Sakarya University, Turkey*
[b] *Khorasan Razavi Water Authority, Mashhad, Iran,*
[c] *Department of Civil Engineering, Izmir Institute of Technology, Turkey*

ARTICLE INFO

Received date: 29 Nov 2023
Accept date: 26 Feb 2024
Published date:11 Jan 2025

Keywords:
Hydrological Flood Routing, Numerical Methods, Optimization, Karoon River.

Abstract

The Muskingum model, widely used in flood routing, is the first order differential equation. The accuracy of the routing storage equation of the model depends on correct parameter estimation and a numerical method employed for its solution. In the present study, different explicit numerical methods are used for solving the equation. These methods include Euler, modified Euler, Runge-Kutta 4th order and Runge-Kutta-Fehlberg. For optimal parameter estimation, Shuffled Complex Evolution (SCE) algorithm is adopted. The performance of different numerical schemes are studied by appropriate evaluation criteria. The methods are tested against three historical and well-known flood data from literature and a field data from Karoon River, Iran as a natural river. Results indicated a good performance of the SCE algorithm and the Runge-Kutta 4th order method in the flood hydrograph numerical simulations. Regarding the sum of squared deviations and for the flood data of Karoon River, the Runge-Kutta 4th order method yielded 18% better results than traditional Euler method in the field condition of a natural river. Similar results can be observed for the first case study, where the Runge-Kutta 4th order method yielded 178% better results than traditional Euler method. However, for second and third case studies the results of all considered numerical methods nearly are in a same level of accuracy. Therefore, it can be said that an appropriate numerical scheme for a hydrological flood routing problem can be adopted by considering the relationship between storage volume and weighted flow.

Homepage: www.wss.torbath.ac.ir

*Corresponding Author:
Barati, Reza
Email: r88barati@gmail.com
ORCID: 0000-0003-2362-2227

https://doi.org/10.22048/WSS.2024.427805.1004

How to cite this article:

Badfar, M., Barati, R., Dogan, E., Tayfur,G. (2025). Performance Evaluation of Different Numerical Schemes for Muskingum Flood Routing (Case study: Karoon River, Iran). *Journal of Advanced Informatics in Water, Soil, and Structure*, *1*(1), 47-61.

1- Introduction

Flood routing can be used to calculate flood wave characteristics along a river reach. It is a computational approach that analyses changes in the velocity and flood waveform as a function of time at one or more points along waterways (Vatankhah 2014; Niazkar and Afzali 2015). The issue of the flood wave propagation along a river or channel reach and the determination of flood characteristics at specified periods and times have several applications in reducing flood damage and designing structures such as levees (Akbari and Barati, 2012; Akbari et al. 2012).

Flood routing analysis is carried out using various hydraulic and hydrological methods to predict intensity of river flows (; Barati et al. 2012; Barati et al. 2013; Vatankhah 2021; Atashi et al. 2023a, b, c). In hydrological flood routing, the equation of motion is neglected and only one-dimensional continuity equation is considered with an equation which describes relation between discharge and water storage. In the hydraulic flood routing, the continuity and momentum equations are used in the calculations simultaneously. Selecting a solution for flood routing problem is subject to on the nature of the considered river reach and available data. The complexity and cost of hydraulic methods limit the use of these approaches in comparison with hydrological models, because of the need for data such as slope, topography, river reach changes, roughness coefficient and cross section characteristics of the considered river reach. This study employs the hydrological flood routing and uses the Muskingum model that has extensive application for the flood routing calculations. In this approach, the routing parameters are estimated based on the upstream and downstream hydrographs of a flood data that previously occurred in the study area. The Muskingum model has been used in many studies (e.g. Easa et al. 2014; Barati 2018; Badfar et al. 2021; Niazkar and Zakwan, 2022). The linear state of this model may not have appropriate results when the relationship between discharge and storage in rivers is non-linear. In all previous researches, various optimization algorithms have been established to improve the fitting between computational and observational hydrographs.

Most of the previous researchers used Euler's method to calculate the output hydrograph of Wilson (1974), based on the Tung's method as discussed and analyzed by Vatankhah (2014). He compared several numerical methods using a historical flood data. In the present research, different explicit numerical methods are used for solving Muskingum's equation to reduce the difference between computational and observational hydrographs by considering several flood data events with different hydrological conditions. Moreover, the focus of the present study is on the performance evaluation of different numerical schemes in a natural river. It should be noted that Das (2004) applied implicit numerical methods such as the Lagrange coefficient method to solve the same problem.

Another important issue for the application of the Muskingum model is the parameter estimation when using a nonlinear form of the storage equation. For this purpose, during last three decades, different algorithms are proposed. These techniques consist of the least-squares method (LSM; Gill 1978); the hybrid of the Hooke–Jeeves (HJ) pattern search and the linear regression (LR), the conjugate gradient (CG), and the Davidon–Fletcher–Powell (DFP) algorithms (HJ+LR, HJ+CG, HJ+DFP; Tung 1985); the nonlinear least squares regression method (NONLR; Yoon and Padmanabhan 1993); the genetic algorithm (GA; Mohan 1997); the harmony search algorithm (HS; Kim et al. 2001); the Lagrange multiplier method (LMM; Das 2004); the Broyden–Fletcher–Goldfarb–Shanno algorithm (BFGS; Geem 2006); the chance-constrained model (CCM; Das 2007); the particle swarm optimization technique (PSO; Chu and Chang 2009; Chu 2010; Das 2010; Vatankhah 2010); the immune clonal selection algorithm (ICSA; Luo and Xie 2010; Barati 2011a); the parameter-setting-free harmony search technique (PSF-HS; Geem 2011); the Nelder-Mead simplex algorithm (NMS; Barati 2011); the differential evolution technique (DE; Xu et al. 2011); the modified honey bee mating optimization (HBMO, Niazkar and Afzali 2015); bat algorithm (BA, Farzin et al. 2018); continuous ant colony algorithm (CACA, Zeinali andPourreza-Bilondi 2018); kidney algorithm (KA, Farahani et al. 2018); hybrid of particle swarm optimization and bat algorithm (Ehteram et al. 2018); Improved BFGS Method (Yang et al. 2020); Goa algorithm (Khalifeh et al. 2020); and grey wolf optimizer algorithm (Akbari and Hessami-Kermani 2021). In general, some of these procedures such as the BFGS and NMS can obtain optimal solution and other methods such as GA, HS, PSO, ICSA, PSF-HS, HBMO, BA, CACA, KA and DE may fail to reach the same solution. BFGS and NMS that can yield a unique solution in different runs need to assume initial value for design parameters, and also some of conventional algorithms such as BFGS needs complicated calculus for searching direction (Barati 2011; Barati 2018). On the other hand, the stochastic or random algorithms that may yield different solutions in different runs need to determine the algorithm parameters such as; crossover rate and mutation rate in GA, harmony memory considering rate and pitch adjusting rate in HS, etc., for each case study. In summary, the evolutionary algorithms such as GA, HS, PSO, ICSA, PSF-HS, and DE do not require the initial value assumption for the hydrologic routing parameters, and do not need complex derivative, but these techniques yield far away-optimal or near-optimal solution instead of the optimal solution for the nonlinear Muskingum model (Barati 2013). On the other hand, the deterministic algorithms such as BFGS and NMS which yield the optimal solution do not need the determination of the algorithm parameters, and do not include any uncertainty, and also have shorter operating time for the convergence than the evolutionary algorithms, but the results of these techniques are sensitive to the starting points.

Shuffled complex evolution algorithm (SCE) is one of the very powerful methods for optimization. The ability of this method has been confirmed by many researchers in hydrological problems modeling (Duan et al. 1993). The SCE algorithm combined best characteristics of both deterministic and stochastic algorithms. In the present research, the SCE algorithm is used to optimize the parameters of the Muskingum model in the MATLAB software environment. Four different numerical methods include Euler, modified Euler, Runge-Kutta 4th order and Runge-Kutta-Fehlberg are used for solving the equations of the Muskingum model. At first, three well-known flood data of Wilson (1974), Ramirez (2010) and Brutsaert (2023) were used to evaluate the model. The main novelty of the present study is the application of aforementioned approaches in a natural river (i.e. Karoon river, Iran), which will be considered as last case study. Moreover, the evaluation of the different numerical schemes for different conditions of relationship between storage volume and weighted flow is a necessary issue, which will be considered in the present study.

2- Materials and Methods

2-1- Nonlinear Muskingum model

The relationship between weight discharge and storage in the Muskingum model is usually nonlinear for the most of the natural rivers around the world. Therefore, the use of the linear mode of the Muskingum model may cause a significant error in the flood routing calculation (Yoon and Padmanabehan, 1993; Barati 2013; Barati 2018). The nonlinear storage equations are more complex in comparison with the linear storage model because of the additional exponent parameter *m*, which results more complex calibration process. In the nonlinear Muskingum model, the following equations can be used as continuity and storage equations:

$$\frac{ds}{dt} = I_t - O_t \qquad (1)$$

$$S_t = K\left[XI_t + (1-X)O_t\right]^m \qquad (2)$$

$$S_t = K\left[XI_t^m + (1-X)O_t^m\right] \qquad (3)$$

where, S_t, O_t, and I_t show storage values, output and input flow at time t, respectively. *K* is a time storage constant and *X* is a weighting factor. In this research, *K*, *X* and *m* parameters are optimized using the SCE algorithm by minimizing the objective function of the sum of the squared residuals between computed and observed outputs (SSQ). By reordering Eq. (3) in terms of O:

$$O_i = \left(\frac{1}{1-X}\right)\left(\frac{S_i}{K}\right)^{\frac{1}{m}} - \left(\frac{X}{1-X}\right)I_i \qquad (4)$$

$$\frac{ds}{dt} = -\left(\frac{1}{1-X}\right)\left(\frac{S_i}{K}\right)^{\frac{1}{m}} + \left(\frac{1}{1-X}\right)I_i \qquad (5)$$

The nonlinear Muskingum model can be performed for flood routing according to the following procedure:

1. Assuming *K*, *X*, and *m* values.
2. Calculating the initial storage, S_0, for given values of I_0 and O_0
3. Calculating the rate of time storage changes.
4. Calculating the next storage by using Eq. (6).
5. Calculating the next outflow by using Eq. (7).
6. Repeating steps 3 to 5 for the following time intervals.

There is no analytical solution for the differential Eq. (5), so explicit numerical solutions such as Euler, modified Euler, Runge-Kutta 4th Order and Runge-Kutta-Fehlberg methods (Gerald and Wheatley, 2004; Vatankhah 2014; Badfar et al. 2021), or implicit numerical methods such as the Lagrange coefficient method are used to solve this equation (Das, 2004). In this research, explicit numerical methods are considered. In the following sections, explicit numerical methods are presented for flood routing by the Muskingum model and the performance of the numerical methods is evaluated using appropriate evaluation criteria and application for four case studies including a natural river.

2-1-1- Euler method

The Euler method is the simplest solution method for the first-order ordinary differential equation. Eq. (5) can be solved by this method for $i = 0,1,2,\ldots,n-1$ (n: number of steps) as (Gerald and Wheatley, 2004; Vatankhah 2014; Badfar et al. 2021):

$$S_{i+1} = S_i + \Delta t \left(\frac{ds}{dt}\right)_i \tag{6}$$

$$S_{i+1} = S_i + \Delta t \left(\frac{I_i - \left(\frac{S_i}{K}\right)^{\frac{1}{m}}}{1-X}\right)_i$$

Δt: time step, by solving Eq. 6, S_{i+1} is calculated for given values of I_i and S_i for each routing step and also *K*, *X* and *m* parameters are obtained. The value of O_{i+1} is determined by the following equation:

$$O_{i+1} = \left(\frac{1}{1-X}\right)\left(\frac{S_{i+1}}{K}\right)^{\frac{1}{m}} - \left(\frac{X}{1-X}\right) I_{i+1} \tag{7}$$

The routing process based on Euler's method includes the following steps:

1. Assuming *K*, *X* and *m* values.
2. Calculating the initial storage, S_0, for given values of I_0 and O_0 by using Eq. (2).
3. Calculating the next storage by using Eq. (6).
4. Calculating the next outflow by using Eq. (7).
5. Repeating steps 3 to 4 for the following time intervals.

During calibration, these steps should be repeated for different values of *K*, *X*, and m, until a suitable fit between computed and observed hydrographs is obtained.

2-1-2- Modified-Euler method

In many flood routing cases, to achieve meaningful results, a small computational step is required, especially in field conditions. In Euler method, the starting point of each sub-range is used to find the curve slope, which yields the correct results only when it has a linear function. Euler's method can be improved by averaging the slopes at i and i + 1 time steps (Gerald and Wheatley, 2004; Vatankhah 2014; Badfar et al. 2021). By applying the modified Euler method for solving Eq. (5):

$$S_{i+1} = S_i + \frac{\Delta t}{2}\left(\left(\frac{ds}{dt}\right)_i + \left(\frac{ds}{dt}\right)_{i+1}\right) \tag{8}$$

Using Euler method to guess, the amount of solution at i + 1 is:

$$S_{i+1} = S_i + \frac{\Delta t}{2}\left[\left(\frac{dS}{dt}\right)_i + \frac{I_{i+1} - \left(\frac{S_i}{K} + \frac{\Delta t}{K} \times \left(\frac{dS}{dt}\right)_i\right)^{1/m}}{1-X}\right] \tag{9}$$

And we already have:

$$\frac{ds}{dt} = -\left(\frac{1}{1-X}\right)\left(\frac{S_i}{K}\right)^{\frac{1}{m}} + \left(\frac{1}{1-X}\right) I_i \tag{10}$$

The routing procedure based on the improved Euler's method includes the following steps:

1. Assuming *K*, *X* and *m* values.
2. Calculating the initial storage, S_0, for given values of I_0 and O_0 by using Eq. (2).
3. Calculating the next storage by using Eq. (9).
4. Calculating the next outflow by using Eq. (7).
5. Repeating steps 3 to 4 for the following time intervals.

2-1-3- Fourth-Order Runge-Kutta Method

The routing procedure of the fourth-order Runge-Kutta method is based on the following steps (Gerald and Wheatley, 2004; Vatankhah 2014; Badfar et al. 2021):

1. Assuming values of *K*, *X* and *m*.
2. Calculating the initial storage, S_0, for given values of I_0 and O_0 by using Eq. (2).
3. Calculating the storage at the next time step S_{i+1}.which is estimated from S_i with a slope. The obtained slope is the weighted average of the following steps:

$$K_1 = -\left(\frac{1}{1-X}\right)\left(\frac{S_i}{K}\right)^{1/m} + \left(\frac{1}{1-X}\right)I_i \quad (11)$$

$$K_2 = -\left(\frac{1}{1-X}\right)\left(\frac{S_i + 0.5\times\Delta t\times K_1}{K}\right)^{1/m} + \left(\frac{1}{1-X}\right)\left(\frac{I_i + I_{i+1}}{2}\right)I_i \quad (12)$$

$$K_3 = -\left(\frac{1}{1-X}\right)\left(\frac{S_i + 0.5\times\Delta t\times K_2}{K}\right)^{1/m} + \left(\frac{1}{1-X}\right)\left(\frac{I_i + I_{i+1}}{2}\right)I_i \quad (13)$$

$$K_4 = -\left(\frac{1}{1-X}\right)\left(\frac{S_i + \Delta t\times K_3}{K}\right)^{1/m} + \left(\frac{1}{1-X}\right)I_{i+1} \quad (14)$$

With the weighted average of the four equations above, the amount of storage in the next step can be obtained by Eq. (15):

$$S_{i+1} = S_i + \Delta t\left(\frac{K_1 + 2K_2 + 2K_3 + K_4}{6}\right) \quad (15)$$

4. Calculating the next outflow by using Eq. (7).
5. Repeating steps 3 and 4 for the following time intervals.

2-1-4- Runge-Kutta-Fehlberg Method

The Runge-Kutta-Fehlberg method is used for solving problems with a specified initial value. The routing procedure using the Runge-Kutta Fehlberg method includes the following steps (Gerald and Wheatley, 2004; Vatankhah 2014; Badfar et al. 2021):

1. Assuming K, X and m values.
2. Calculating the initial storage, S_0, for given values of I_0 and O_0 by using Eq. (2).
3. Calculating the next storage. The slope will be a function of the following slopes:

$$K_1 = -\left(\frac{1}{1-X}\right)\left(\frac{S_i}{K}\right)^{1/m} + \left(\frac{1}{1-X}\right)I_i \quad (16)$$

$$K_2 = -\left(\frac{1}{1-X}\right)\left(\frac{S_i + 0.25\times\Delta t\times K_1}{K}\right)^{1/m} + \left(\frac{1}{1-X}\right)\left[I_i + 0.25\times\left(I_{i+1} - I_i\right)\right] \quad (17)$$

$$K_3 = -\left(\frac{1}{1-X}\right)\left(\frac{S_i + \frac{3}{32}\Delta t K_1 + \frac{9}{32}\Delta t K_2}{K}\right)^{1/m} + \left(\frac{1}{1-X}\right)\left[I_i + \frac{3}{8}\left(I_{i+1} - I_i\right)\right] \quad (18)$$

$$K_4 = -\left(\frac{1}{1-X}\right)\left(\frac{S_i + \frac{1932}{2197}\Delta t K_1 - \frac{7200}{2197}\Delta t K_2 + \frac{7296}{2197}\Delta t K_3}{K}\right)^{1/m} + \left(\frac{1}{1-X}\right)\left[I_i + \frac{12}{13}\left(I_{i+1} - I_i\right)\right] \quad (19)$$

$$K_5 = -\left(\frac{1}{1-X}\right)\left(\frac{S_i + \frac{439}{216}\Delta t K_1 - 8\Delta t K_2 + \frac{3680}{513}\Delta t K_3 - \frac{845}{4104}\Delta t K_4}{K}\right)^{1/m} + \left(\frac{1}{1-X}\right)\left[I_{i+1}\right] \quad (20)$$

$$K_6 = -\left(\frac{1}{1-X}\right)\left(\frac{S_i - \frac{8}{27}\Delta t K_1 + 2\Delta t K_2 - \frac{3544}{2565}\Delta t K_3 + \frac{1859}{4104}\Delta t K_4 - \frac{11}{40}\Delta t K_5}{K}\right)^{1/m} + \left(\frac{1}{1-X}\right)\left[I_i + \frac{1}{2}(I_{i+1} - I_i)\right] \quad (21)$$

Finally, the next storage can be obtained by Eq. (22) as following (22):

$$S_{i+1} = S_i + \Delta t\left(\frac{16}{135}K_1 + \frac{6656}{12825}K_3 + \frac{28561}{56430}K_4 - \frac{9}{50}K_5 + \frac{2}{55}K_6\right) \quad (22)$$

1. Calculating the next outflow using Eq. (7).
2. Repeating steps 3 and 4 for the following time steps.

2-2- Shuffled Complex Evolution Algorithm

The SCE algorithm starts with creating an initial population of design parameters which are sampled from a feasible space randomly (Duan, et al 1992; Duan, et al 1993; Gupta et al. 1999; Duan 2003). The population is divided into a number of complexes. The partition of the population facilitates an extensive exploration of the solution space in different directions, thereby reducing the search getting trapped in local optima. Each complex develops based on a competitive evolution technique that uses the downhill simplex method (Nelder and Mead, 1965) for a set number of evolutions. Then, the complexes are shuffled and reassigned into new complexes to enable information sharing. A new set of evolutions for each complex is performed if the convergence is not reached. These multiple complex shuffling and complex evolution provide a reasonably good balance between the exploration and exploitation. The procedure of the SCE algorithm could be found in Duan et al. (1992; 1993; see Fig. 1):

Four stopping criteria including (1) difference between best and worst function evaluation in population is smaller than the tolerance (e.g. 0.001); (2) maximum difference between the coordinates of the vertices in simplex is less than the tolerance (e.g. 0.001); (3) maximum number of function assessments or iterations are touched (e.g. 2500); and (4) maximum duration of optimization is touched (e.g. 30 sec); were used in the SCE algorithm at each generation.

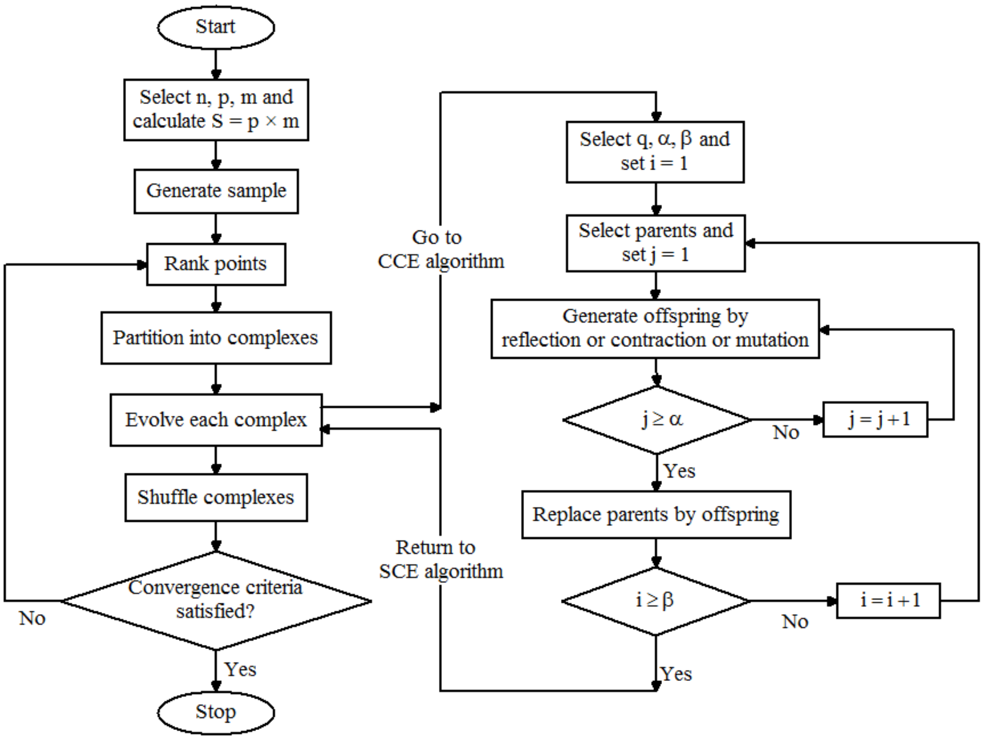

Figure. 1. Flowchart of Shuffled Complex Evolution (SCE) algorithm (adopted from Duan et al. 1992; 1993)

The SCE algorithm considers several probabilistic and deterministic components that are measured by some algorithmic parameters, and they are: m=the number of points in a complex ($m \geq 2$); q=the number of points in a sub-complex ($2 \leq q \leq m$); p=the number of complexes ($p \geq 1$); α =the number of consecutive offspring generated by each sub-complex $(\alpha \geq 1)$; and β=the number of evolution steps taken by each complex $(\beta \geq 1)$. Duan et al. (1994) mentioned values of these parameters based on several experimental analysis using different algorithmic parameter setups. The suitable values for these parameters as a function of the number of parameters to be optimized are: m= (2n+1); q=(n+1); α =1; and β =(2n+1). The p is the only parameter should be determined based on the characteristics of the considered problem (Barati et al. 2014).

3- Results and Discussion

Four case studies including Wilson's data (1974), Ramirez (2010), Brutsaert (2023) and a natural river flood event of the Karoon River; are used to compare the efficiency and accuracy of explicit numerical methods and also the performance of the SCE algorithm for the flood routing using the nonlinear Muskingum model. The objective function for the optimal estimation of the parameters *K*, *X* and *m* is to minimize the sum of the squared residuals between calculated and observed outputs (SSQ), which is expressed as follows:

$$\min SSQ = \sum_{i=1}^{N} \left\{ O_i - \left[\left(\frac{1}{1-X} \right) \left(\frac{S_i}{K} \right)^{\frac{1}{m}} - \left(\frac{X}{1-X} \right) I_i \right] \right\} \tag{23}$$

**

The following four criteria are used to evaluate the numerical methods: 1- The sum of the squared residuals between computed and observed outputs (SSQ); 2- The sum of the absolute difference between the observed and routed values of the discharge flow (SAD); 3- The difference between observed and routed peak flow (DPO); and 4- The difference between time to compute and routed peak flow (DPOT). The optimal value for four criteria is zero.

$$SSQ = \sum_{j=1}^{M} \left[O_j - \hat{O}_j \right]^2 \quad (24)$$

$$SAD = \sum_{j=1}^{M} \left| O_j - \hat{O}_j \right| \quad (25)$$

$$DPO = \left| Q_{po} - Q_{pc} \right| \quad (26)$$

$$DPOT = \left| T_{po} - T_{pc} \right| / \Delta t \quad (27)$$

where O_j and $\hat{O}_j$ are respectively the observed and routed output flow rates at the time *j*, *M* the number of data, Q_{po} observed peak outflow and Q_{pc} routed peak outflow, T_{po} the time of the observed peak outflow, T_{pc} the time of the computed peak outflow.

3-1-1- The first case study

Wilson's data (1974) have been widely used by researchers to estimate optimal values of the routing parameters in the optimization phase (i.e. calibration phase). For exploring more details of this historical data please refer to Atashi et al. (2023a,b,c). The simulated and observed hydrographs using four numerical methods are presented by Figure 2. The values of the parameters (*K*, *X* and *m*) and criteria obtained are listed in Table 1.

Table 1. Criteria values and estimated parameters for Wilson (1974) by the nonlinear Muskingum model

Criteria values and estimated parameters	Numerical method			
	Euler method	Modified Euler	Runge-Kutta 4th Order	Runge Kutta Fehlberg
SSQ	178.98	90.97	62.59	62.15
SAD	48.98	37.41	29.43	29.45
DPO ($m^3s/1$)	1.08	0.36	0.32	0.35
DPOT (h)	0	0	0	0
K	0.0798	0.0717	0.0541	0.0541
X	0.1864	0.2934	0.2830	0.2827
m	2.2831	2.3081	2.3724	2.3724

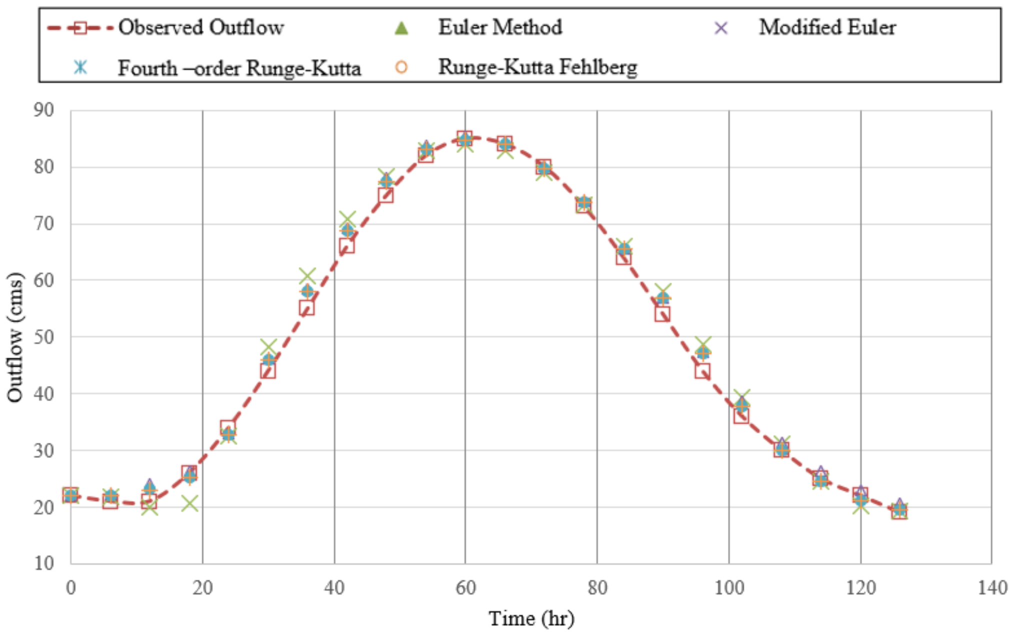

Figure. 2. Observed outflow (qout) and the routed outflows by different numerical methods for Wilson (1974)

The Fourth-Order Runge-Kutta and Runge-Kutta-Fehlberg methods have similar accuracy in terms of four statistical indexes. However, the results obtained from Fourth-Order Runge-Kutta and Runge-Kutta-Fehlberg methods have greatly improved the results of Euler and modified Euler methods. Regarding the SSQ, the Runge-Kutta 4th order method yielded 178% and 45% better results than traditional Euler and modified Euler methods, respectively. Regarding the SAD, the Runge-Kutta 4th order method yielded 66% and 27% better results than traditional Euler and modified Euler methods, respectively. Due to the fact that the results of Runge-Kutta 4th order and Runge-Kutta-Fehlberg method are very close to each other, the use of both methods is recommended for flood hydrological routing by the Muskingum nonlinear model for flood events such as Wilson (1974).

3-1-2- The second case study

The results of flood routing of Ramirez's data (2010) are presented in Table 2. Observational and computational output hydrographs are also displayed in Figure 3. As can be seen in Table 2, the results of the Fourth –order Runge-Kutta method and the Runge Kutta Fehlberg method are similar in terms of four performance evaluation criteria, and both methods are recommended for flood hydrological routing by the Muskingum nonlinear model. Moreover, Euler and modified Euler methods has good results for this case study in terms of statistical indexes and these explicit numerical methods could also be applied for flood events such as Ramirez (2010).

3-1-3- The third case study

By examining the objective functions using the SCE algorithm, the results of Brutsaert's data (2005) are displayed in Table 3. The observational and computational output hydrographs are also depicted in Figure 4. Since the results of Fourth–order Runge-Kutta method and the Runge Kutta Fehlberg method are almost the same in terms of four statistical indexes, the use of both methods for flood hydrological routing by the Muskingum nonlinear model is recommended. It should be considered that Euler method has better accuracy for this case study, whereas modified Euler has the last accuracy among considered methods.

Table 2. Criteria values and estimated parameters for Ramirez (2010) by the nonlinear Muskingum model

Criteria values and estimated parameters	Numerical method			
	Euler method	Modified Euler	Fourth –order Runge-Kutta	Runge Kutta Fehlberg
SSQ	2.001	2.12	2.07	2.07
SAD	5.51	5.72	5.66	5.66
DPO (m^3/s)	0.074	0.038	0.051	0.051
DPOT (hr)	0	0	0	0
K	2.261	2.2838	2.2723	2.2704
X	0.0647	0.2391	0.1721	0.1713
m	1.002	1.0010	1.0018	1.0019

Table 3. Criteria values and estimated parameters for Brutsaert (2023) by the nonlinear Muskingum model

Criteria values and estimated parameters	Numerical method			
	Euler method	Modified Euler	Fourth –order Runge-Kutta	Runge Kutta Fehlberg
SSQ	12144.81	15369.13	14435.70	14441.01
SAD	478.49	481.41	494.22	494.77
DPO (m^3/s)	46.90	54.21	46.88	47.05
DPOT (hr)	1	0	1	1
K	0.8198	1.3003	1.0856	1.0807
X	0.0387	0.4999	0.3090	0.3061
m	1.1106	1.0533	1.0748	1.0753

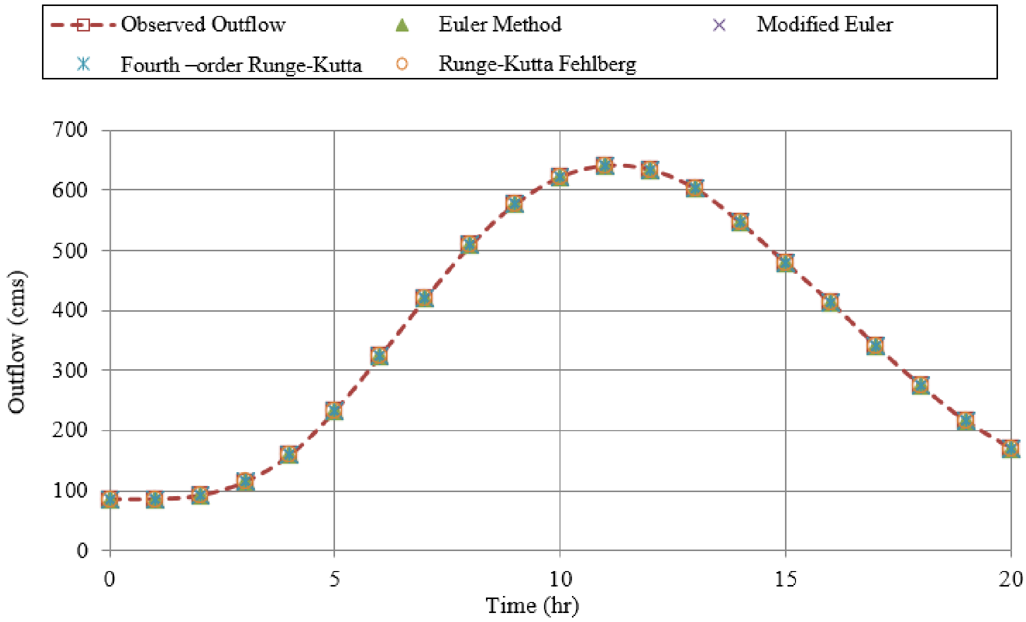

Figure. 3. Observed Outflow and the Routed Outflows by different numerical methods for Ramirez (2010)

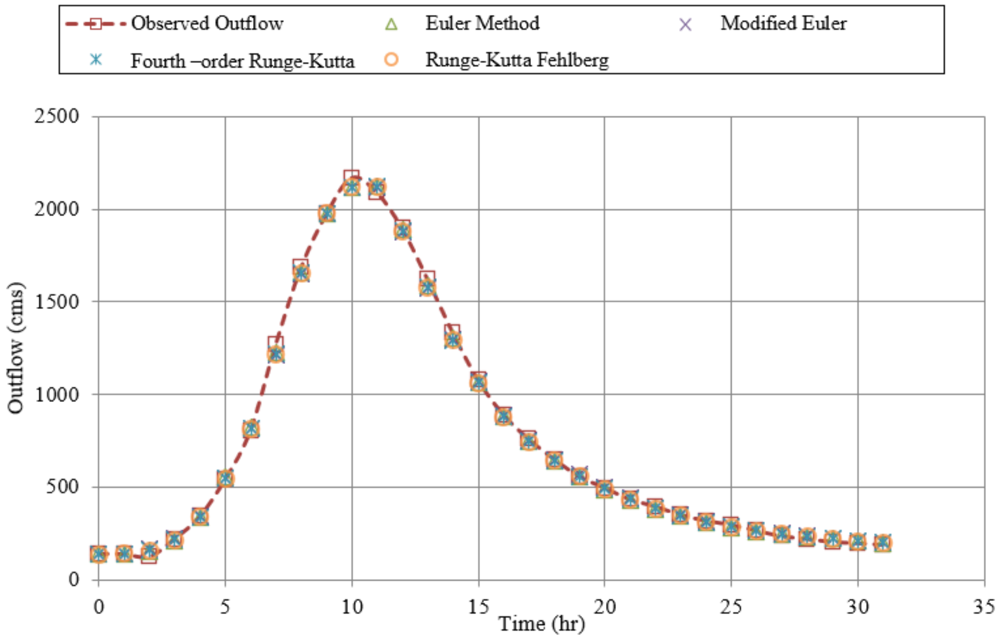

Figure. 4. Observed Outflow and the Routed Outflows by different numerical methods for Brutsaert (2023)

3-2- A natural river reach

As last case study, a flood event data of the Karoon River, Iran is used. The length of river reach is 60.5 km, which is located between Mollasani and Ahvaz stations. Ahvaz (downstream) and Mollasani (upstream) stations are respectively built in 1895 and 1966 to determine the discharge of the Karoon River. The equipment of these stations includes scaffolding, limnograph and data logger. The average width of the river is almost 268 meters and the average slope between the two stations Mollasani and Ahvaz is 0.00011. For exploring more details of study area please refer to Akbari and Barati (2012).

Table 4. Criteria values and estimated parameters for Karoon River by the nonlinear Muskingum model

Criteria	Numerical method

values and estimated parameters	Euler Method	Modified Euler	Fourth –order Runge-Kutta	Runge Kutta Fehlberg
SSQ	45600	40114	38640	38640
SAD	1072	1001	974	974
DPO (m^3/s)	49.3	26.2	24.1	24.1
DPOT (hr)	1	0	1	1
K	4.489	6.801	6.787	6.787
X	0.00000173	0.03490	0.00175	0.00166
m	1.052	1.0000	1.0000	1.0000

By examining the objective functions using the SCE algorithm, the results are displayed in Figure 5. Values of the performance evaluation criteria and estimated parameters are listed in Table 4. Fourth-Order Runge-Kutta and Runge-Kutta-Fehlberg methods have similar accuracy by considering SSQ, SAD, DPO and DPOT. The results obtained from Fourth-Order Runge-Kutta and Runge-Kutta-Fehlberg methods are better than the results of Euler and modified Euler methods, but not similar to the first case study. Regarding the SSQ, the Runge-Kutta 4th order method yielded 18% and 4% better results than traditional Euler and modified Euler methods, respectively. Regarding the SAD, the Runge-Kutta 4th order method yielded 10% and 3% better results than traditional Euler and modified Euler methods, respectively. Therefore, due to the fact that the results of Runge-Kutta 4th order and Runge-Kutta-Fehlberg method are very close to each other, the use of both methods is recommended for flood hydrological routing by the Muskingum nonlinear model for flood events such as the Karoon River.

3-3- Discussion

Vatankhah (2014) stated that fourth-order Runge-Kutta method is an accurate and suitable explicit solution method for calculating the storage time variation of the Muskingum model. The results of the present research are consistent with those of Vatankhah (2014). However, it was found that by considering the results of the flood routing for four flood data events (i.e., Wilson 1974; Ramirez 2010; Brutsaert 2023 and Karoon River), for the first and last case studies the results of the Fourth-Order Runge-Kutta and Runge-Kutta-Fehlberg methods are better than the those of the Euler and modified Euler methods. However, for second and third case studies the results of all considered numerical methods nearly are in a same level of accuracy by considering four statistical indexes of SSQ, SAD, DPO and DPOT. It is interesting that the flood data of Wilson (1974) has the most nonlinear relationship between storage volume and weighted flow as it is well-known (Badfar et al. 2021), while both flood data of Ramirez (2010) and Brutsaert (2023) have almost linear relationship between storage volume and weighted flow. The relationship between storage volume and weighted flow for flood data of the Karoon River is linear, but some data point were not recorded in the falling limb of the hydrograph, which imposed semi-nonlinear relationship in the calculations. Therefore, it can be said that the selection of a numerical method for a flood routing problem depends on the relationship between storage volume and weighted flow. For nonlinear and semi-nonlinear relationships the Fourth-Order Runge-Kutta and Runge-Kutta-Fehlberg methods could be adopted, while for a linear and partly linear relationships the Euler and modified Euler methods could be utilized. However, more flood data should be analyzed to reach a more concrete discussion and conclusion about this issue.

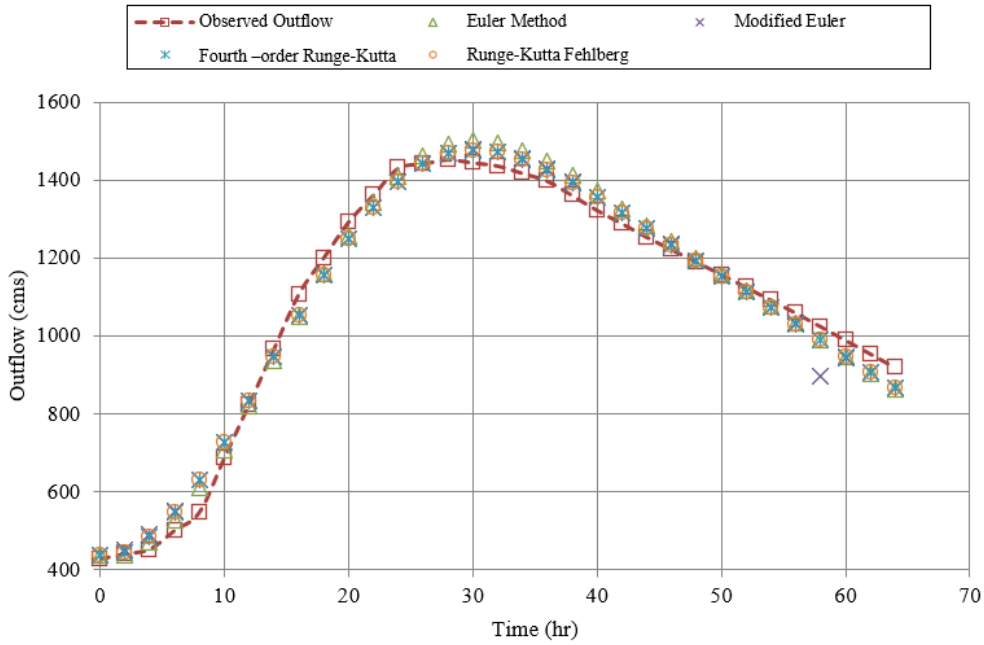

Figure. 5. Observed Outflow and the Routed Outflows by different numerical methods for Karoon River

4- Conclusion

In the present research, four explicit numerical methods were used to solve the Muskingum nonlinear model for three well-known flood data. Moreover, the available data of the Karoon River was used as a natural river to compare the explicit numerical methods in the field condition. By assessing the used statistical criteria (i.e., SSQ, SAD, DPO and DPOT), the results of this study indicated that the Euler and modified Euler methods were not sufficiently accurate than other methods, especially for flood data with nonlinear relationship between storage volume and weighted flow. Also, the results of the Fourth-Order Runge-Kutta and Runge-Kutta-Fehlberg methods were very close, but because of the simplicity of using the Fourth-Order Runge-Kutta method, this method is considered as a convenient and accurate method among different explicit numerical methods for solving nonlinear models of the Muskingum routing approach. Concluding, it can be said that an appropriate numerical scheme for a hydrological flood routing problem could be selected by considering the relationship between storage volume and weighted flow.

Funding sources

There is no funding source.

Competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Authors contribution

MB conceived the study and drafted the manuscript; RB supervised the work and provided resources and data curation, validated the data, and reviewed and edited the manuscript; ED and GT provided data curation and reviewed and edited the manuscript.

References

Akbari, R., and Hessami-Kermani, M. R. (2021). Parameter estimation of Muskingum model using grey wolf optimizer algorithm. *MethodsX*, 8, 101589.

Akbari, G. H., Nezhad, A. H., & Barati, R. (2012). Developing a model for analysis of uncertainties in prediction of floods. *Journal of Advanced Research*, 3(1), 73-79.

Akbari, G. H., & Barati, R. (2012, April). Comprehensive analysis of flooding in unmanaged catchments. In *Proceedings of the Institution of Civil Engineers-Water Management* (Vol. 165, No. 4, pp. 229-238). Thomas Telford Ltd.

Atashi, V., Barati, R., & Lim, Y. H. (2023). Distributed Muskingum model with a Whale Optimization Algorithm for river flood routing. *Journal of Hydroinformatics*, *25*(6), 2210-2222.

Atashi, V., Barati, R., & Lim, Y. H. (2023). Improved river flood routing with spatially variable exponent Muskingum model and sine cosine optimization algorithm. *Environmental Processes*, *10*(3), 42.

Atashi, V., Barati, R., & Lim, Y. H. (2023). Development of a distributed nonlinear Muskingum model by considering snowmelt effects for flood routing in the Red River. *Scientific Reports*, *13*(1), 21356.

Badfar, M., Barati, R., Dogan, E., & Tayfur, G. (2021). Reverse Flood Routing in Rivers Using Linear and Nonlinear Muskingum Models. *Journal of Hydrologic Engineering*, 26(6), 04021018.

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

Barati, R. (2013). Application of excel solver for parameter estimation of the nonlinear Muskingum models. *KSCE Journal of Civil Engineering*, *17*, 1139-1148.

Barati, R. 2013. Application of excel solver for parameter estimation of the nonlinear Muskingum models. KSCE *Journal of Civil Engineering*, 17(5), 1139-1148.

Barati, R., Rahimi, S., & Akbari, G. H. (2012). Analysis of dynamic wave model for flood routing in natural rivers. *Water Science and Engineering*, 5(3), 243-258.

Barati, R., Akbari, G. H., & Rahimi, S. (2013). Flood Routing of an Unmanaged River Basin Using Muskingum--Cunge Model; Field Application and Numerical Experiments. Caspian *Journal of Applied Sciences Research*, 2(6). 8-20.

Barati, R., Neyshabouri, S. S., & Ahmadi, G. (2014). Sphere drag revisited using shuffled complex evolution algorithm. *River flow*, 978, 345-353.

Barati, R. (2018). Discussion of "application of genetic programming to flow routing in simple and compound channels" by Elahe Fallah-Mehdipour, Omid Bozorg-Haddad, Hossein Orouji, and Miguel A. Mariño. *Journal of Irrigation and Drainage Engineering*, 144(5), 07018015.

Brutsaert, W. (2023). *Hydrology*. Cambridge university press.

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(9), 1024-1027.

Chu, H. J. (2010) "Closure to 'Applying particle swarm optimization to parameter estimation of the nonlinear Muskingum model" by H.-J. Chu and L.-C. Chang' " *Journal of Hydrologic Engineering*, 15(11), 952–954.

Das, A. (2004). "Parameter estimation for Muskingum models." *Journal of Irrigation and Drainage Engineering*, 130(2):140–147.

Das, A. (2007). "Chance-constrained optimization-based parameter estimation for Muskingum models." *Journal of Irrigation and Drainage Engineering*, 133(5):487–494.

Das, A. (2010). "Discussion of 'Applying particle swarm optimization to parameter estimation of the nonlinear Muskingum model by H.-J. Chu and L.-C. Chang.' " *Journal of Hydrologic Engineering*, 15(11), 946–948.

Duan, Q. (2003). Global optimization for watershed model calibration. *Calibration of watershed models*, *6*, 89-104.

Duan, Q. Y., Gupta, V. K., and Sorooshian, S. (1992). "Effective and efficient global optimization for conceptual rainfall-runoff models." *Water Resources Research*, 28(4), 1015–1031.

Duan, Q. Y., Gupta, V. K., and Sorooshian, S. (1993). "Shuffled complex evolution approach for effective and efficient global minimization." *Journal of Optimization Theory and*

Applications, 76(3), 501–521.

Duan, Q. Y., Sorooshian, S., and Gupta, V. K. (1994). "Optimal use of the SCE-UA global optimization method for calibrating watershed models." *Journal of Hydrology*, 158, 265–284.

Easa, S. M., Barati, R., Shahheydari, H., Nodoshan, E. J., & Barati, T. (2014). Discussion: New and improved four-parameter non-linear Muskingum model. In Proceedings of the *Institution of Civil Engineers-Water Management* (Vol. 167, No. 10, pp. 612-615). Thomas Telford Ltd.

Ehteram, M., Binti Othman, F., Mundher Yaseen, Z., Abdulmohsin Afan, H., Falah Allawi, M., Najah Ahmed, A., ... and El-Shafie, A. (2018). Improving the Muskingum flood routing method using a hybrid of particle swarm optimization and bat algorithm. *Water*, 10(6), 807.

Farahani, N., Farzin, S., and Karami, H. (2018). Flood routing by Kidney algorithm and Muskingum model. *Natural Hazards*, 1-19.

Farzin, S., Singh, V. P., Karami, H., Farahani, N., Ehteram, M., Kisi, O., ... and El-Shafie, A. (2018). Flood routing in river reaches using a three-parameter Muskingum model coupled with an improved bat algorithm. *Water*, 10(9), 1130.

Geem, Z. W. (2006). "Parameter estimation for the nonlinear Muskingum model using the BFGS technique." *Journal of Irrigation and Drainage Engineering*, 132(5), 474–478.

Geem, Z. W. (2011). "Parameter Estimation of the Nonlinear Muskingum Model Using Parameter-Setting-Free Harmony Search." *Journal of Hydrologic Engineering*, 16(8), 684–688.

Gerald, C. F., and Wheatley, P. O. 2004. Applied numerical analysis, 7th Ed., Addison-Wesley, Boston.

Gill, M. A. (1978). "Flood routing by Muskingum method." *Journal of hydrology*, 36(3–4), 353–363.

Gupta, H. V., Sorooshian, S., and Yapo, P. O. (1999). "Status of automatic calibration for hydrologic models: comparison with multilevel expert calibration." *Journal of Hydrologic Engineering*, 4(2), 135-143.

Hosseini, S. M. (2009). "Application of Spreadsheets in Developing Flexible Multiple-Reach and Multiple-Branch Methods of Muskingum Flood Routing." *Computer Applications in Engineering Education*, 17(4), 448–454.

Khalifeh, S., Esmaili, K., Khodashenas, S., and Akbarifard, S. (2020). Data on optimization of the non-linear Muskingum flood routing in Kardeh River using Goa algorithm. *Data in Brief*, 30, 105398.

Kim, J. H., Geem, Z. W., and Kim, E. S. (2001). "Parameter estimation of the nonlinear Muskingum model using harmony search." *Journal of the American Water Resources Association*, 37(5), 1131–1138.

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

Mohan, S. (1997). "Parameter estimation of nonlinear Muskingum models using genetic algorithm." *Journal of Hydraulic Engineering*, 123(2), 137–142.

Nelder, J. A., and Mead, R. (1965). "A simplex method for function minimization." *Computer Journal*., 7, 308-313.

Niazkar, M., and Afzali, S. H. (2015). Assessment of modified honey bee mating optimization for parameter estimation of nonlinear Muskingum models. *Journal of Hydrologic Engineering*, 20(4), 04014055.

Niazkar, M., and Zakwan, M. (2022). Parameter estimation of a new four-parameter Muskingum flood routing model. In *Computers in Earth and Environmental Sciences* (pp. 337-349). Elsevier.

Ramirez, J. A. (2010). Accessed October 4, 2012. http://www.engr.colostate.edu/~ramirez/ce_old / Adopted from Badfar, M., Barati, R., Dogan, E., & Tayfur, G. (2021). Reverse Flood Routing in Rivers Using Linear and Nonlinear Muskingum Models. *Journal of Hydrologic Engineering*, 26(6), 04021018.

Samani, H. M. V., and Shamsipour, G. A. (2004). "Hydrologic flood routing in branched river systems via nonlinear optimization." *Journal of Hydraulic Research*, 42(1), 55-59.

Tung, Y. K. (1985). "River flood routing by nonlinear Muskingum method." *Journal of Hydraulic Engineering*, 111(12), 1447–1460.

Vatankhah, A. (2010). "Discussion of 'Applying particle swarm optimization to parameter estimation of the nonlinear Muskingum model by H.-J. Chu and L.-C. Chang.' " *Journal of Hydrologic Engineering*, 15(11), 949–952.

Vatankhah, A. R. 2014. Evaluation of explicit numerical solution methods of the Muskingum model. *Journal of Hydrologic Engineering*, 19(8), 949-952.

Vatankhah, A. R. (2021). The lumped Muskingum flood routing model revisited: the storage relationship. *Hydrological Sciences Journal*, 66(11), 1625-1637.

Wilson, E. M. (1974). "Engineering hydrology." MacMillan, Hampshire, U.K.

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

Yang, T., Li, P., & Wang, X. (2020). Convergence analysis of an improved BFGS method and its application in the Muskingum model. *Mathematical Problems in Engineering*, *2020*(1), 4519274.

Yoon, J., and Padmanabhan, G. (1993). Parameter estimation of linear and nonlinear Muskingum models. *Journal of Water Resources Planning and Management*, *119*(5), 600-610.

Zeinali, M., and Pourreza-Bilondi, M. (2018). Estimation of optimal parameters of the nonlinear Muskingum model using continuous ant colony algorithm. Irrigation and *Water Engineering*, 8(3), 94-106.