

## OPTIMIZATION TO IDENTIFY MUSKINGUM MODEL PARAMETERS USING IMPERIALIST COMPETITIVE ALGORITHM

A. Tahershamsi and R. Sheikholeslami<sup>\*†</sup>

*Department of Civil and Environmental Engineering, Amirkabir University of Technology,  
Tehran, Iran*

### ABSTRACT

In engineering, flood routing is an important technique necessary for the solution of a flood-control problem and for the satisfactory operation of a flood-prediction service. A simple conceptual model like the Muskingum model is very effective for the flood routing process. One challenge in application of the Muskingum model is that its parameters cannot be measured physically. In this article we proposed imperialist competitive algorithm (ICA) for optimal parameter estimation of the linear Muskingum model. This algorithm uses imperialism and imperialistic competition process as a source of inspiration. Optimization to identify Muskingum model parameters can be considered as a suitable field to investigate the efficiency of this algorithm.

Received: 2 October 2011; Accepted: 29 December 2011

KEY WORDS: Flood routing; muskingum model; optimization; imperialist competitive algorithm

### 1. INTRODUCTION

Floods are one of the most costly types of natural disasters in the world. It was reported that the average annual cost of flood damage in some countries was about \$300 million. Therefore one of the most common problems facing the civil engineer is the estimation of the hydrograph of the rise and fall of a river at any given point on the river during the course of a flood event. This problem is solved by the techniques of flood routing, which is a process for calculating

---

<sup>\*</sup>Corresponding author: R. Sheikholeslami, Department of Civil and Environmental Engineering, Amirkabir University of Technology, Tehran, Iran

<sup>†</sup>E-mail address: [razi.tabrizu.civil@gmail.com](mailto:razi.tabrizu.civil@gmail.com)

the shape of a flood wave along a river channel. The Muskingum model is a popular model for flood routing, and its storage depends on the inflow and outflow. However, the application of the model still suffers from an absence of an efficient method for parameter estimation. There are various mathematical techniques which have been used for estimating the parameters of the Muskingum model. These mathematical methods use gradient information to search the solution space near an initial starting point.

On the other hand starting from the 1980s, a number of successful meta-heuristic optimization algorithms have been created and developed to solve optimization problems. Among them the most well known are genetic algorithms (GAs), simulated annealing (SA), particle swarm optimization (PSO), ant colony optimization (ACO), harmony search algorithm (HS), imperialist competitive algorithm (ICA) and charged system search (CSS). All of these algorithms try to find the optimal solution in a stochastic manner and avoid local optimum solutions. These algorithms impose fewer mathematical requirements and they do not require very well defined mathematical models. Meta-heuristic algorithms also provide efficacious solutions to the high-scale combinatorial and non-linear problems. Due to these advantages, these algorithms are used in different fields such as management science, engineering, and computer [1].

One of these meta-heuristic approaches is imperialist competitive algorithm. ICA is a multi-agent algorithm with each agent being a country, which is either a colony or an imperialist. These countries form some empires in the search space. Movement of the colonies toward their related imperialist and imperialistic competition among the empires forms the basis of the ICA. During these movements, the powerful imperialists are reinforced and the weak ones are weakened and gradually collapsed, directing the algorithm towards optimum points [2]. Imperialistic competition is the main part of the ICA and hopefully causes the colonies to converge to the global minimum of the cost function. This algorithm is proposed by Atashpaz-Gargari et al. [3]. Kaveh and Talatahari [4] improved the ICA by defining two new movement steps and investigated the performance of this algorithm to engineering optimization problems.

As an alternative to the conventional mathematical approaches this article utilizes ICA for identifying the parameters of the linear Muskingum model. In order to evaluate ICA, a numerical example is utilized and the results are compared to those of other algorithms. The results reveal the efficiency of this algorithm to optimum parameter estimation of Muskingum model.

## 2. PROBLEM FORMULATION

Routing of flood hydrographs by means of channel routing procedures is useful in instances where known hydrographic data are at a point other than the point of interest. This is also true in those instances where the channel profile or plan is changed in such a way as to alter the natural velocity or channel storage characteristics. During this process the flood inflow hydrograph changes its shape: its peak is usually lowered and its base extended, *i.e.* the flood subsides. The usual task is to determine the peak reduction – attenuation of the flood between inflow and outflow – and the time lag between the peaks [5].

The Muskingum Method combines continuity, a prism component of storage, and a wedge component to describe the total storage in the reach as:

$$\begin{aligned}\frac{dS_t}{dt} &\approx \frac{\Delta S_t}{\Delta t} = I_t - O_t \\ S_t &= K[xI_t + (1-x)O_t]\end{aligned}\quad (1)$$

where  $S_t$ ,  $I_t$ , and  $O_t$  denote the instantaneous amounts of storage, inflow, and outflow, respectively, at time  $t$ ;  $K$  is the storage-time constant for the river reach, which has a value reasonably close to the flow travel time through the river reach; and  $X$  is the weighting factor usually varying between 0 and 0.5. When  $X = 0.5$ , the storage is described as a full wedge. In natural streams the weighting factor  $X$  ranges from 0 to 0.3.

The parameters  $K$  and  $X$  are usually estimated using a graphical or least-squares procedure [6]. Stephenson [7] and Gill [8] proposed linear programming techniques. These methods can work well in the absence of lateral inflows or when such flows are small [9]. Choosing a suitable time interval for the routing period, the continuity equation in its finite difference form becomes

$$S_{i+1} - S_i = \frac{(I_i + I_{i+1})\Delta t}{2} - \frac{(O_i + O_{i+1})\Delta t}{2} \quad (2)$$

using the linear Muskingum model,

$$O_{i+1} = C_1 I_i + C_2 I_{i+1} + C_3 O_i, \quad (i = 1, 2, 3, \dots) \quad (3)$$

Where

$$C_1 = \frac{2Kx + \Delta t}{\Delta t + 2K - 2Kx}, \quad C_2 = \frac{-2Kx + \Delta t}{\Delta t + 2K - 2Kx} \quad \text{and} \quad C_3 = \frac{2K - 2Kx - \Delta t}{\Delta t + 2K - 2Kx} \quad (4)$$

in which  $O_i$  and  $I_i$  represent the outflow and inflow discharges at time  $t_i$ , respectively; and  $C_1$ ,  $C_2$  and  $C_3$  are the Muskingum parameters where  $C_1 + C_2 + C_3 = 1$ . The routing procedure consists of the following steps:

- Step 1.** Assume the values of two parameters ( $K$  and  $x$ );
- Step 2.** Calculate the coefficients  $C_1$ ,  $C_2$ , and  $C_3$  using Eq. (4);
- Step 3.** Compute the magnitude of the outflow at the next time ( $O_{i+1}$ ) using Eq. (3)
- Step 4.** Repeat Step 3 for all times.

In this article  $C_1$ ,  $C_2$  and  $C_3$  are determined using optimization techniques by obtaining a good match between the observed and calculated discharge. The objective function to be minimized is the sum of the squared residuals (SSQs) between the observed and calculated outflows as follows

$$\min \text{SSQ} = \sum_{j=1}^{N_o} [O_{oj} - O_{cj}(C_1, C_2, C_3)]^2 \quad (5)$$

in which  $O_{oj}$  is observed value of the outflow at time  $t_j$ ;  $O_{cj}$  is the computed value of the

outflow at time  $t_j$ ;  $j$  is an index varying from one to  $N_o$ , where  $N_o$  is the number of hydrograph ordinates of the observed hydrograph.

### 3. IMPERIALIST COMPETITIVE ALGORITHM (ICA)

Imperialist competitive algorithm (ICA) is a new progressive algorithm for optimization. This algorithm starts with an initial population. Each population in ICA is called country. Some of the best countries in the population are selected to be the imperialists and the rest form the colonies of these imperialists. When the competition starts, the imperialists attempt to achieve more colonies and the colonies start to move toward their imperialists. So during the competition the powerful imperialists will survive and the weak ones will be collapsed. At the end just one imperialist will remain [10]. The main components of the basic ICA algorithm are described in detail below.

#### *Step 1: Initialization*

The primary locations of the countries are determined randomly in the interval  $[x_{\min}, x_{\max}]$  in which  $x_{i,\min}$  and  $x_{i,\max}$  are the minimum and the maximum allowable values for the variables.

For each country, the cost identifies its usefulness. The related cost of a country is found by evaluation of the cost function  $f_{\text{cost}}$  of the corresponding variables considering the related objective function. Total number of initial countries is set to  $N_{\text{country}}$  and the number of the most powerful countries to form the empires is taken as  $N_{\text{imp}}$ . The remaining  $N_{\text{col}}$  of the initial countries will be the colonies each of which belongs to an empire. In this paper, 10 percent of countries belong to empires and the remaining is used as colonies. To form the initial empires, the colonies are divided among imperialists based on their power. To fulfill this aim, the normalized cost of an imperialist is defined as

$$C_n = f_{\text{cost}}^{(\text{imp},n)} - \max_i (f_{\text{cost}}^{(\text{imp},i)}) \quad (6)$$

where  $f_{\text{cost}}^{(\text{imp},n)}$  is the cost of the  $n$ th imperialist and  $C_n$  is its normalized cost. The initial colonies are divided among empires based on their power or normalized cost, and for the  $n$ th empire it will be as follows

$$NC_j = \text{Round} \left( \frac{C_j}{\sum_{i=1}^{N_{\text{imp}}} C_i} \cdot N_{\text{col}} \right) \quad (7)$$

where  $NC_j$  is the initial number of the colonies related to the  $j$ th empire which are selected randomly among the colonies. These colonies together with the  $j$ th imperialist form the empire number  $j$ .

*Step 2: Colonies Movement*

In ICA, the assimilation policy, pursued by some of former imperialist states, is modelled by moving all the colonies toward the imperialist. According to this movement, a colony moves toward the imperialist by a random value that is uniformly distributed between 0 and  $\beta \times d$ :

$$\{x\}_{new} = \{x\}_{old} + U(0, \beta \times d) \times \{V_1\} \quad (8)$$

where  $\beta$  is a control parameter and  $d$  is the distance between colony and imperialist.  $\{V_1\}$  is a vector which its start point is the previous location of the colony and its direction is toward the imperialist locations. The length of this vector is set to unity.

In the original ICA, to increase the searching around the imperialist, a random amount of deviation,  $\theta$ , is added to the direction of movement.  $\theta$  is a random number with uniform distribution.

In order to improve the ICA performance, the orthogonal imperialist competitive algorithm was developed by Kaveh and Talatahari [4]. This algorithm not only uses different random values, but also utilizes the orthogonal colony-imperialistic contacting line instead of  $\theta$  for deviating the colony as follows

$$\begin{aligned} \{x\}_{new} &= \{x\}_{old} + \mathbf{b} \times d \times \{rand\} \otimes \{V_1\} + U(-1, +1) \times \tan(q) \times d \times \{V_2\}, \\ \{V_1\} \cdot \{V_2\} &= 0, \|\{V_2\}\| = 1 \end{aligned} \quad (9)$$

where  $\{V_2\}$  is perpendicular to  $\{V_1\}$ . Since this vector must be crossed the point obtained from the two first terms, we use a random value by using  $U(-1, +1)$  for the third term of the Eq. (9) which changes its value in addition to its direction by using negative values.

*Step 3: Imperialist Updating*

If the new position of the colony is better than that of its relevant imperialist (considering the cost function), the imperialist and the colony change their positions and the new location with lower cost becomes the imperialist. Then the other colonies move toward this new position.

*Step 4: Imperialistic Competition*

In this step, all empires try to take the possession of colonies of other empires and control them. The imperialistic competition is modelled by just picking some (usually one) of the weakest colonies of the weakest empires and making a competition among all empires to possess these (this) colonies. Based on their total power, in this competition, each of the empires will have a likelihood of taking possession of the mentioned colonies.

Total power of an empire is affected by the power of imperialist country and the colonies of an empire as

$$TC_j = f_{\text{cost}}^{(\text{imp},j)} + \alpha \cdot \frac{\sum_{i=1}^{NC_j} f_{\text{cost}}^{(\text{col},i)}}{NC_j} \quad (10)$$

where  $TC_n$  is the total cost of the  $j$ th empire and  $\alpha$  is a positive number which is considered to be less than 1. Also, the normalized total cost is defined as

$$NTC_j = TC_j - \max_i (TC_i) \quad (11)$$

where  $NTC_j$  is the normalized total cost of the  $j$ th empire. Finally, the possession probability of each empire is evaluated by

$$P_j = \frac{NTC_j}{\sum_{i=1}^{N_{\text{imp}}} NTC_i} \quad (12)$$

#### Step 5: Implementation

When an empire loses all its colonies, it is assumed to be collapsed. In this model implementation, where the powerless empires collapse in the imperialistic competition, the corresponding colonies will be divided among the other empires.

#### Step 6: Terminating Criterion Control

Moving colonies toward imperialists are continued and imperialistic competition and implementations are performed during the search process. When the number of iterations reaches a pre-defined value or the amount of improvement in the best result reaches a pre-defined value, the searching process is stopped. The pseudo-code of the ICA algorithm is presented in Table 1. More details about this algorithm are presented in [2,3].

Table 1: Pseudo-code for the ICA

- 
- 1) Initialize the algorithm parameters.
  - 2) Move the colonies toward their relevant imperialist
  - 3) If there is a colony in an empire which has lower cost than that of imperialist, exchange the positions of that colony and the imperialist.
  - 4) Compute the total cost of all empires.
  - 5) Use imperialistic competition and pick the weakest colony from the weakest empire.
  - 6) Eliminate the powerless empires.
  - 7) If there is just one empire, stop, if not go to 2
-

#### 4. VERIFICATION

To investigate the performance of applying ICA to solve the parameter estimation problem of the linear Muskingum model, a typical problem is used as an example. In this paper, the model is applied to the south canal between Chenggou and Linqing rivers in China. The length of the river course –Chenggou to Linqing – of South Canal in Haihe Basin, reaches 83.8km. There is no significant effect of the branches. The time interval in the calculation is taken as 12hr. The data set from Ref [11] is considered for illustration purpose. This example has been studied previously by Wang *et al.* [12,13], Zahn and Xu [14], and Yang *et al.* [15] for testing different parameter estimation methodologies. Also Zhengxiang and Ling [16] applied and compared several intelligent algorithms on Muskingum routing model using this example. Therefore, the performance of the proposed algorithm can effectively be compared with the previous reported results obtained with this example.

In Table 2 and 3 the optimum parameters and the objective function of the solutions obtained by the LSM (Least Squares Method), GA, SA, PSO [16], and the proposed ICA are presented. The results show that the SSQ is attained using ICA. It has been demonstrated that ICA gets better results than other methods.

For the proposed algorithm, a population of 30 countries consisting of 3 empires and 27 colonies are used. Tuning the utilized parameters for a meta-heuristic algorithm such as ICA is a very important issue. In order to fulfil this, herein a sensitive study on two parameters of the algorithm is performed. For various values of  $\beta$  and  $\tan(\theta)$ , this example is solved several times (15 times for each value of  $\beta$  and  $\tan(\theta)$ ) and the average SSQ is shown in Figure 1. This figure shows that  $\beta > 1$  make the colonies to move closer to the imperialist state from both sides while a very close value to 1 for  $\beta$  reduces the search ability of the algorithm. As shown in the Figure,  $\beta = 2$  and  $\tan(\theta) = 1$  are suitable values for the ICA algorithm.

Table 2: Comparison of Best SSQs from Various Methods

	SSQ	$C_1$	$C_2$	$C_3$
LSM	1311.28	0.0843	0.4626	0.4531
GA	1046.69	0.05476	0.45241	0.49283
SA	1046.69	0.0547522	0.4524160	0.4928318
PSO	1046.83	0.0547561	0.4524157	0.4928282
ICA ( <i>present work</i> )	1031.28	0.0100000	0.4774254	0.5116732

Table 3: Comparison of the Observed and Computed Outflows (cms)

T (hr)	1	2	3	4	5	6	7	8	9	10
I	261	389	462	505	525	543	556	567	577	583
$Q_o$	228	300	382	444	490	513	528	543	553	564

$Q_c$	228	304.99	380.52	404.42	481.05	510.63	532.16	548.55	561.82	571.58
T (hr)	11	12	13	14	15	16	17	18	19	20
I	587	595	597	597	589	556	538	516	486	505
$Q_o$	573	581	588	594	592	584	566	550	520	504
$Q_c$	578.54	585.96	590.79	593.29	590.74	573.61	555.91	536.18	511.54	507.70
T (hr)	21	22	23	24	25	26	27	28	29	
I	477	429	379	320	263	220	182	167	152	
$Q_o$	483	461	420	368	318	271	234	193	178	
$Q_c$	492.56	461.61	421.43	372.20	319.21	271.99	227.75	198.08	175.59	

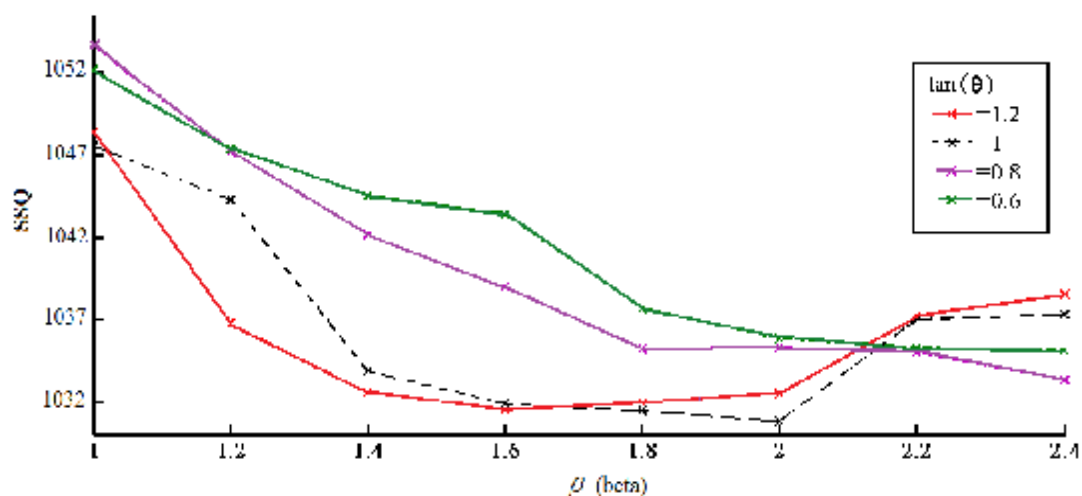


Figure 1. Sensitivity of SSQ to selected values of  $\beta$  and  $\tan(\theta)$

The computed outflow hydrograph together with the observed outflow hydrograph are shown in Figure 2. The plot depicts that the resulting outflow hydrograph, using the parameters estimated from ICA, closely follow the observed outflows.



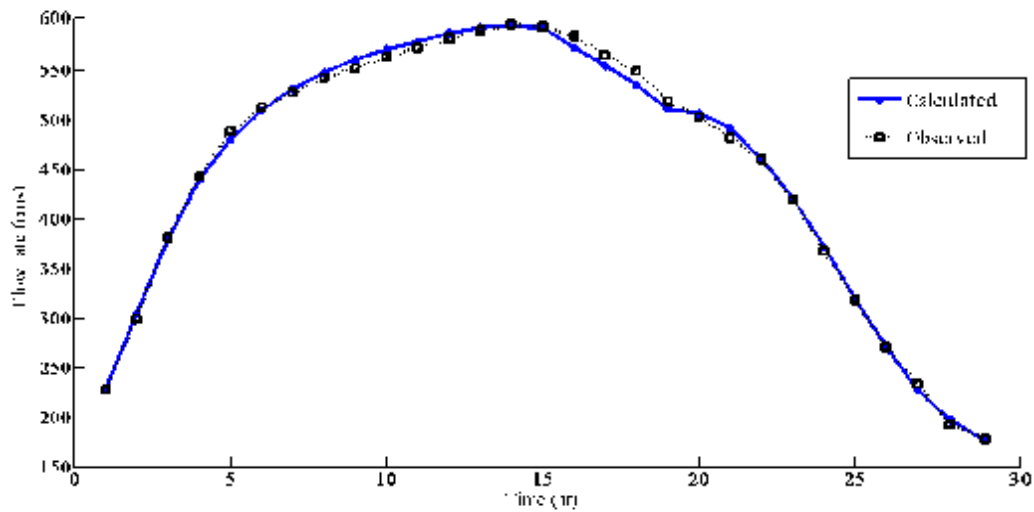


Figure 2. Observed and routed hydrographs using ICA

## 5. CONCLUSION

In the present study the newly developed meta-heuristic algorithm ICA has been proposed for estimating the hydrologic parameters of the Muskingum model. It found the best parameter values in terms of the minimal sum of the square deviation between the observed and routed outflows. The performance of this approach was compared with other common methods in Table 2. The results demonstrate that ICA can achieve a high degree of accuracy to estimate the parameters. Consequently, the model also shows robustness in forecasting outflow. The method presented may also be applicable to models other than the Muskingum model. ICA appears to offer good applicability in the hydrology field and further applications should be explored.

## REFERENCES

1. Kaveh A, Talatahari S, Sheikholeslami R. Optimum Seismic Design of Gravity Retaining Walls using the Heuristic Big Bang-Big Crunch Algorithm, in B.H.V. Topping, Y. Tsompanakis, (Editors), *Proceedings of the Second International Conference on Soft Computing Technology in Civil, Structural and Environmental Engineering*, Civil-Comp Press, Stirlingshire, Scotland, Paper 4, 2011, DOI:10.4203/ccp.97.4.
2. Talatahari S, Farahmand Azar B, Sheikholeslami R, Gandomi AH. Imperialist Competitive Algorithm Combined with Chaos for Global Optimization, *Commun Nonlinear Sci Numer Simulat* 2012; **17**: 1312-19.
3. Atashpaz-Gargari E, Lucas C. Imperialist competitive algorithm: an algorithm for optimization inspired by imperialistic competition, *Proceeding of IEE: congress on evolutionary computation*, Singapore, 2007, pp. 4661-67.

4. Kaveh A, Talatahari S. Optimum design of skeletal structures using imperialist competitive algorithm, *Comput Struct* 2010;**88**: 1220-29.
5. Novak P, Moffat AIB, Nalluri C, Narayanan R. Hydraulic Structures, Taylor & Francis, 4<sup>th</sup> ed, London and New York, 2007.
6. Singh VP. Elementary Hydrology, Prentice Hall, Englewood Cliffs, NJ, 1997.
7. Stephenson D. Direct optimization of Muskingum routing coefficients – an extension to the paper by M. A. Gill (1978): Flood routing by the Muskingum method, *J Hydrol* 1979; **41**: 161-5.
8. Gill MA. Flood routing by the Muskingum method, *J. Hydrol (Amsterdam)* 1978; **36**: 353-63.
9. Kshirsagar MM, Rajagopalan B, Lall U. Optimal parameter estimation for Muskingum routing with ungauged lateral inflow, *J Hydrol* 1995;**169**: 25-35.
10. Mozafari H, Abdi B, Amran A. Optimization of Composite Plates Based on Imperialist Competitive Algorithm, *Int J Comput Sci Eng* 2010;**2**(9): 2816-19.
11. Qu GJ. Parameter estimation for Muskingum models, *J Hydrol* 1977;**36**(3): 40–3.
12. Wang W, Xu Z, Qiu L, Xu D. Hybrid Chaotic Genetic Algorithms for Optimal Parameter Estimation of Muskingum Flood Routing Model, *Proceeding of IEEE: International Joint Conference on Computational Sciences and Optimization*, 2010, DOI 10.1109/CSO.2009.34.
13. Wang W, Kang Y, Qiu L. Optimal Parameter Estimation for Muskingum Model Using a Modified Particle Swarm Algorithm, *Proceeding of IEEE: International Joint Conference on Computational Sciences and Optimization*, 2010, DOI 10.1109/CSO.2010.143.
14. Zhan S, Xu J. Application of ant colony algorithm to parameter estimation of Muskingum Routing Model, *J Natural Disasters* 2006;**14**: 20-4.
15. Yang X, Jin J, Chen Z, Wei Y. A New Approach for Parameter Estimation of Muskingum Routing Model, *J Catastrophology* 1998;**13**: 1-6.
16. Zhengxiang Y, Ling K. Application and Comparison of Several Intelligent Algorithms on Muskingum Routing Model, *Proceeding of IEEE: 2<sup>nd</sup> International Conference on Digital Object Identifier*, 2010, DOI 10.1109/ICIFE.2010.5609501.