Discussions and Closures

Closure to “Improved Particle Swarm Optimization–Based Artificial Neural Network for Rainfall-Runoff Modeling” by Mohsen Asadnia, Lloyd H. C. Chua, X. S. Qin, and Amin Talei

DOI: 10.1061/(ASCE)HE.1943-5584.0000927

Mohsen Asadnia1; Lloyd H. C. Chua2; Amin Talei3; and X. S. Qin, A.M.ASCE

1Research Associate, DHI–NTU Water and Environment Research Center and Education Hub, NTU, 50 Nanyang Ave., Singapore 639798, Singapore.
2Associate Professor, Civil Engineering, School of Engineering, Deakin Univ., Waurn Ponds, Geelong, VIC 3216, Australia.
3Lecturer, School of Engineering, Monash Univ. Malaysia, Sunway Campus, Jalan Lagoon Selatan, Bandar Sunway, 47500 Selangor Darul Ehsan, Malaysia (corresponding author). E-mail: amin.talei@monash.edu; amin.taleyee@gmail.com
4Assistant Professor, School of Civil and Environmental Engineering, Nanyang Technological Univ., 50 Nanyang Ave., Singapore 639798, Singapore.

The writers wish to thank the discussers for comments and discussion. The aim of the original paper as highlighted in the Abstract and Introduction was to introduce a new technique in improving particle swarm optimization (PSO) for training an artificial neural network (ANN). Three well-known learning algorithms, namely (1) conjugate gradient (CG), (2) gradient descent (GD), and (3) Levenberg–Marquardt (LM), were adopted and the best performing one on the data was then compared with conventional PSO–neural network (NN) and improved PSO–NN. The response to the points raised by the discussers is as follows:

• The general thrust of the discussers’ question focuses on the selection of inputs. This is indeed pertinent for the ANN and other data-driven models such as neurofuzzy systems (NFs) and several publications have addressed this issue (Maier and Dandy 1997; Coulibaly et al. 2000; Sudheer et al. 2002; Nayak et al. 2007; Talei and Chua 2012). In the original paper however, the focus was on the comparison across ANN models adopting different training methods. Therefore, a less-than-rigorous approach was used in the selection of inputs and adopted correlation analysis to select a common set of inputs for the models. It is the writers’ view that deficiency in the less-than-optimal selection of inputs is not significant to the results, since this will be common to all the models considered. In Fig. 3 of the original paper, the cross-correlation analysis between rainfall lags and water level as can be seen from t-6 onwards the correlation coefficient is almost similar. The initial analyses showed that considering rainfall lags up to t-7 would be sufficient and considering more lags could not improve the models practically. To respond to the issue raised by discussers on using \( H(t-1) \) as the only water level lag, the decision was made based on the auto-correlation analysis between \( H(t) \) and its lags up to \( H(t-10) \). In this analysis, the correlation coefficient (CC) values between \( H(t-1), H(t-2), \) and \( H(t-3) \) with \( H(t) \) were 0.867, 0.734, and 0.658, respectively. The CC value was below 0.6 from \( H(t-4) \) onwards. Initial results showed that adding more water level lags will not improve model performance. This was attributed to the fact that the mutual information between highly correlated inputs can deteriorate the performance in data-driven models such as ANN and NFS (Talei and Chua 2012; Talei et al. 2013). Therefore, using \( H(t-1) \) was found to be sufficient for the models of the original paper. Moreover, it was revealed that using water level lags alone as input is not sufficient and adding rainfall lags will improve model performance.

• It is well-understood that the network architecture plays an essential role in performance of any ANN algorithm. In the original paper, p. 1324, Paragraph 2, the importance of choosing proper architectures for ANN was briefly discussed. It is also explained that the trial and error was the adopted method to determine the best architectures for all ANN algorithms. The approach in selecting an ANN algorithm is to achieve a simple architecture without compromising on accuracy of the network. Furthermore, although a single layer is widely used, the structure adopted is also data dependent. In the case of the original paper, for instance, in Table 2 of the original paper, it was reported that the best architecture for Model 4 for LM algorithm is a network with 5–8–4–1 architecture. By changing the architecture to one hidden layer (the best architecture was 5–8–1) the mean squared error (MSE) for testing phase will be 0.097 (m²) while by considering architecture 5–8–2–1 and 5–8–4–1 the MSE value will be reduced to 0.074 and 0.042 (m²), respectively.

• The idea of comparing traditional ANN algorithms such as CG, GD, and LM was to find the best network with the lesser error and compare it with the developed PSO-based ANN algorithms. On p. 1324, Paragraph 2, it has been mentioned that for all the models, two stopping criteria were defined which were (1) the maximum number of iterations \( N = 1,500 \), and (2) the error tolerance of \( \varepsilon = 0.001 \). The number of iterations should be considered large enough to enable particles in finding a solution with an error less than the error tolerance. The results showed that the tolerance condition for CG was satisfied at number of epochs of 1,149, while this number for GD was 749. This means that the maximum number of iteration of 1,500 is never achieved as the error tolerance criterion has been satisfied before reaching that maximum iteration. Moreover, Table 2 of the original paper shows that for both Models 4 and 9, GD has smaller training and testing errors as compared to CG.

References