An ISaDE algorithm combined with support vector regression for estimating discharge coefficient of W-planform weirs

Somayeh Emami, Javad Parsa, Hojjat Emami and Akram Abbaspour

ABSTRACT

Various shapes of weirs, such as rectangular, trapezoidal, circular, and triangular plan forms, are used to adjust and measure the flow rate in irrigation networks. The discharge coefficient \( (C_d) \) of weirs, as the key hydraulic parameter, involves the combined effects of the geometric and hydraulic parameters. It is used to compute the flow rate over the weirs. For this purpose, a hybrid ISaDE-SVR method is proposed as a hybrid model to estimate the \( C_d \) of sharp-crested W-planform weirs. ISaDE is a high-performance algorithm among other optimization algorithms in estimating the nonlinear parameters in different phenomena. The ISaDE algorithm is used to improve the performance of SVR by finding optimal values for the SVR’s parameters. To test and validate the proposed model, the experimental datasets of Kumar et al. and Ghodsian were utilized. Six different input scenarios are presented to estimate the \( C_d \). Based on the modeling results, the proposed hybrid method estimates the \( C_d \) in terms of \( H/P, L_w/W_{mc} \), and \( L_c/W_c \). For the superior method, \( R^2 \), RMSE, MAPE, and \( \delta \) are obtained as 0.982, 0.006, 0.612, and 0.843, respectively. The amount of improvement in comparison with GMDH, ANFIS and SVR is 3.6%, 1.2% and 1.5% in terms of \( R^2 \).

Key words | discharge coefficient, improved SaDE algorithm, labyrinth weirs, support vector regression

HIGHLIGHTS

- A novel hybrid ISaDE method was used to predict the \( C_d \) of W-planform weirs.
- More discharge over W-planform is their major advantage against other types of weirs.
- ISaDE algorithm is used to improve the capability of SVR by finding optimal values for the SVR’s parameters.
- The model which includes parameters \((H/P, L_w/W_{mc}, L_c/W_c)\) has the best results.
- The \( H/P \) parameter was the most important input parameter.
INTRODUCTION

Weirs are among the most essential components of water transmission networks, due to the necessity of determining the flow rate in channels and the allocated amount for consumers. These are essential hydraulic structures for controlling the flow and water level, which can be utilized to increase the height of the water surface level and thereby provide the required water heights to divert the flow to lateral channels. Also, these structures are utilized as flow-measuring devices in crucial applications in rivers and open channels for their safe operation (Emami et al. 2018).

The length and shape of the weir crest are among the effective parameters in the flow rates over the weir, and numerous studies have been carried out on the effect of geometric and hydraulic parameters of weirs on the discharge coefficient ($C_d$). The use of non-linear planar weirs with a specific shape such as triangular, trapezoidal, piano key, circular and parabolic, which are known as labyrinth weirs, is one of the effective ways to increase the flow over a specified width.

Some notable approaches recently proposed to predict the $C_d$ of different plans of labyrinth weirs are shown in Figure 1.

An important type of triangular weir is the sharp-crested W-planform weir. A few attempts have been made to define the $C_d$ of W-planform weirs. More discharge over them is their major advantage against other types of weirs. The first experimental study on labyrinth weirs with different plans was conducted by Taylor (Dehdar-Bezbahani & Parsaei 2016). Besides the development of experimental models, numerical methods including computational fluid dynamic (CFD) techniques (Parsaei et al. 2015; Dehdar-Bezbahani & Parsaei 2016), and artificial intelligent techniques have emerged as distinctive and complementary tools in the subject of hydraulic field (Parsaei et al. 2015; Haghiabi et al. 2018; Parsaei et al. 2018a). Meta-heuristic algorithms as an important branch of artificial intelligence have been widely employed in solving various problems because of their accuracy and high performance. Aydin (2012), Aydin & Emiroglu (2013), Robertson (2014) and Crookston and Tullis (Crookston 2010) used CFD software to model labyrinth weirs and labyrinth side weirs.

Researchers have used neural networks (ANN), genetic programming (GP), support vector machine (SVM) and M5 model tree, group method of data handling (GMDH), adaptive neuro-fuzzy inference system (ANFIS), and meta-heuristic algorithms to optimize and predict the characteristics and geometry of labyrinth weirs with different plan forms including rectangular, trapezoidal, triangular, and a combination of these plans as well and their $C_d$ (Emin Emiroglu et al. 2010; Zaji & Bonakdari 2014; Ebtahaj et al. 2015; Azamathulla et al. 2016; Najafzadeh et al. 2016; Haghiabi et al. 2017; Parsaei et al. 2018b; Bilhan et al. 2018; Roushangar et al. 2018).

Karami et al. (2016) predicted the $C_d$ of the triangular labyrinth weir using SVM, ANN, and GP methods. SVM
is a machine-learning technique that is widely used for classification and regression purposes by means of a separating hyperplane. The benefits of SVM, especially its significant accuracy in classification utilizing the tuning parameters, associated with its wide real-world applications has led to the ever-increasing popularity of SVM in the last decade. Artificial neural network (ANN), as a famous computing technique, is inspired by the biological neural system which is utilized in solving complicated problems. ANNs can enjoy self-learning capabilities with task-specific rules and they reach better results by more available data. Genetic programming (GP) is a subset of machine learning and like all evolutionary algorithms (EAs) works by an iterative process. EAs are generally used to discover solutions to challenging real-world problems. GP starts by setting a goal function and then uses the Darwin evolution principle to generate a set of candidate solutions. The results obtained by Karami et al. (2016) showed that the SVM method was superior to the other two methods with RMSE = 0.0059. Haghiabi et al. (2017) obtained the $C_d$ of the triangular labyrinth weir by using the ANFIS and multi-layer perceptron (MLP) models. An adaptive neuro-fuzzy inference system (ANFIS) is based upon a Takagi–Sugeno fuzzy inference system. This model was first introduced in the early 1990s. It enjoys both neural network and fuzzy logic advantages and it is capable of capturing the benefits of both techniques in a single framework. A multilayer perceptron (MLP) is a subset of the feedforward artificial neural network and is utilized as a powerful ANN model for solving complex systems. MLP is generally used as a nonlinear mapping technique to relate the output vector to the input vector of the system. The results of the two models suggest that both models perform well in predicting the $C_d$. Roushangar et al. (2017) examined the $C_d$ of the arched labyrinth weir using the SVM method and concluded that this method has high accuracy in predicting the $C_d$ of arched weirs. Bonakdari & Zaji (2018) utilized three novel hybrid soft computing approaches, including neuro-fuzzy-differential evaluation (ANFIS-DE), neuro-fuzzy-genetic algorithm (ANFIS-GA) and neuro-fuzzy-particle swarm optimization (ANFIS-PSO), to simulate the $C_d$ of side weirs. The used methods, as the hybrid ones, enjoy the advantages of both employed algorithms. The differential evolution (DE) algorithm, which is utilized together with ANFIS in ANFIS-DE, is a population-based direct search method that uses several vectors. The way of interaction among the vectors is determined by the evolution of living species. This heuristic global optimization algorithm is easy to comprehend, simple, and fast to implement. Parsaie et al. (2018a) applied ANN, SVM and ANFIS models to predict the $C_d$ of a weir-gate system. Performance evaluation of the three models showed that the SVM model had the best performance ($R^2 = 0.94$ and RMSE = 0.008). Bonakdari et al. (2020) employed gene expression programming (GEP) in predicting the $C_d$ of the triangular labyrinth weir and concluded that the proposed method is better than the nearest-living-relative (NLR) method for predicting the $C_d$. Gene expression programming (GEP) is a learning algorithm to find out the relationship between inputs and output of the unknown system. The main advantage of GEP is in realizing the system inter-relationship by a clear mathematical equation that needs to identify a set of pre-defined mathematical operators. Shafiei et al. (2020) applied the ANFIS-FFA method to obtain the $C_d$ of the labyrinth weirs. A comparison of ANFIS-FFA model results with the ANFIS model showed that the proposed model has a good performance in predicting the $C_d$. The firefly algorithm (FFA), which is used together with ANFIS in the study, is a swarm-based meta-heuristic algorithm that mimics the flashing light behavior of fireflies. The algorithm works by mimicking how attracted a firefly is by the flashing light of any other firefly. According to the fireflies’ behavior, attractiveness is specified by the brightness of the flashing light and its distance, which is the foundation of the technique.

According to the above-mentioned studies, different methods of artificial intelligence have been employed as appropriate optimization or simulation tools to determine the $C_d$ of these weirs. In this paper, a hybrid predictive model based on SVR and improved self-adaptive evolutionary optimization algorithm (ISADE) is proposed to study the effect of geometric parameters on the $C_d$ of the sharp-crested W-planform labyrinth weir. The proposed method was evaluated with six different combinations of the effective parameters to determine the most appropriate combination. To train and test the proposed hybrid ISADE-SVR model, the experimental studies of Kumar et al. (2012) and Ghodsian (2009) have been used. Also, the results of the ISADE-SVR method are compared with the previous findings.
The remaining parts are organized as follows. Problem Definition defines the problem that this paper is focused on. Methodology describes the proposed methodology. The experiments are given in the fourth section. Discussion of results is provided in the fifth section. Finally, the sixth section concludes the paper and lists some future interesting directions.

**PROBLEM DEFINITION**

The purpose of determining the $C_d$ is to investigate the performance of the weir by estimating the flow characteristics passing over the weir. The $C_d$ of the W-planform labyrinth weir is a function of six basic parameters, which are: upstream total head of flow ($H$), effective length of weir ($L_e$), weir height ($P$), weir width ($W$), vertex angle ($\theta$) and flow depth ($y$).

The following equation is used to calculate $C_d$ in labyrinth weirs (Bagheri & Heidarpour 2010):

$$Q = \frac{2}{3} C_d \sqrt{2gL_e W H^1.5}$$

(1)

The discharge coefficient of the W-planform weir formula and its effective parameters can be given by Equation (2):

$$C_d = f(H, W_{mc}, W_c, L_w, L_e, P, g, \sigma, \mu, \rho)$$

(2)

where $W_{mc}$ is the main channel width, $W_c$ is the width of one cycle of the weir, $L_w$ is the total length of the weir, $L_e$ is the length of the one cycle, $g$ indicates the acceleration effected by gravity, $\sigma$ is the surface tension, $\mu$ is the dynamic viscosity of the fluid and $\rho$ is the specific mass. Using the dimensional analysis for discharge coefficient, the functional relationship as Equation (3) is derived; as the flow over the weir is turbulent, the Reynolds number and Weber number can be omitted (Gupta et al. 2015):

$$C_d = f\left(\frac{H}{P}, \frac{L_w}{W_{mc}}, \frac{L_e}{W_c}, \frac{L_w}{W_c}, \frac{H}{W_c}\right)$$

(3)

**METHODOLOGY**

**Support vector regression**

Support vector regression (SVR) is a supervised machine-learning method equipped with association learning algorithms (Cortes & Vapnic 1995; Drucker et al. 1997; Zaji et al. 2016). The main principle behind the SVR is the same as that of SVM. For a dataset $D = \{O_i, y_i\}_{i=1}^N$, where $N$ indicates the number of data objects, $O_i = \{a_{i1}, a_{i2}, \ldots, a_{im}\}$ is a defined $m$-dimensional data object and $y_i$ is the label that is assigned to $O_i$. In the SVR algorithm, each data object $O_i \in D$ is considered as a point in $m$-dimensional space. The goal is to create a prediction model using some training data to separate data objects through finding a hyperplane that differentiates the data objects into some separate groups. This hyperplane is calculated based on a few data points, known as support vectors. In other words, SVR aims to maximize the minimum distance of data points from a separator hyperplane by solving the following equation:

$$f(o) = \sum_{i=1}^{N} (a_i - a_i^o) K(o, o_i) + b$$

(4)

where $K(o, o_i)$ denotes the kernel function, which is computed by multiplying the two inner vectors $o$ and $o_i$ in the feature space $\phi(o)$ and $\phi(o_i)$, respectively. Three well-known kernel functions are the radial basis function (RBF), sigmoid, and polynomial basis function. In this paper, the RBF kernel function is used in the SVR due to its high performance and easy configuration compared with other kernel functions. The precise tuning of $C$, $\gamma$ and $\varepsilon$ is an important task to increase the prediction performance of the SVR with RBF kernel. To optimize these parameters, we introduced the ISaDE algorithm, which is described in the next section.

**Self-adaptive differential evolution algorithm**

Self-adaptive differential evolution (SaDE) is a well-known iterative optimization algorithm (Brest et al. 2006). SaDE is an evolutionary and multi-agent algorithm, in which each agent updates its position by the three operators of selection, mutation, and crossover. The flowchart of the SaDE algorithm is shown in Figure 2.

For an $m$-dimensional optimization problem, the initial population is an $N_{pop} \times m$ matrix defined as follows:

$$P = [X_1, X_2, \ldots, X_{N_{pop}}]^T$$

(5)

where each individual $X_i \in P$ is an $m$-dimensional vector defined as follows:

$$X_i = [x_{i1}, x_{i2}, \ldots, x_{im}]$$

(6)
In the mutation step, a mutant individual $M_i$ is produced for $X_i$ as follows:

$$M_i = X_{r_1} + F_i (X_{r_2} - X_{r_3}), \quad \text{where } i \neq r_1 \neq r_2 \neq r_3$$

in which $r_1, r_2, r_3 \in [1, N_{\text{pop}}]$ are random numbers used for identifying individuals that take part in mutation, and $F \in [0, 1]$ indicates a real number that adjusts the amplification of the difference between $X_{r_2}$ and $X_{r_3}$. In standard SaDE, $F_i$ is defined as

$$F_i = \begin{cases} F_{i_{\text{lo}}} + \zeta \times F_{i_{\text{hi}}} & \text{if } \zeta < \Delta_1 \\ F_i & \text{otherwise} \end{cases}$$

where $\zeta \in [0, 1]$ is a uniform random variable, and $\Delta_1$ is a probability value adjusting the control factor $F$. In the simulations, $F_{i_{\text{lo}}} = 0.1$, $F_{i_{\text{hi}}} = 0.9$, and $\Delta_1 = 0.1$. $F$ is a random value in the range $[0, 1]$.

**Crossover**

In the crossover phase, for each individual $X_i$, the algorithm computes a trial vector $T_i$, defined as follows:

$$T_i = [t_{i,1}, t_{i,2}, \ldots, t_{i,D}]$$

where $t_{ij} \in T_i$ is defined as

$$t_{ij} = \begin{cases} m_{ij} & \text{if } r(j) \leq CR \text{ or } j = m(i) \\ x_{ij} & \text{if } r(j) > CR \text{ or } j \neq m(i) \end{cases}$$

in which $r(j) \in [0, 1]$ is a uniform random variable, and $m(i) \in [1, D]$ is a random variable guarantee that $T_i$ gets at least one element from vector $M_i$. $CR \in [0, 1]$ indicates the crossover constant, which is defined as

$$CR_i = \begin{cases} \zeta_3 & \text{if } \zeta_4 < \Delta_2 \\ CR_i & \text{otherwise} \end{cases}$$

where $\zeta_3$ is a uniform random variable in the interval $[0, 1]$, and $\Delta_2$ is a probability value adjusting the crossover constant $CR$. In the simulations, $\Delta_2$ is set to 0.1.

**Selection**

To obtain the next generation, each individual $X_i$ is updated as follows:

$$X_i^{g+1} = \begin{cases} T_i & \text{if } f(T_i) > f(X_i) \\ X_i & \text{otherwise} \end{cases}$$

If the trial vector $T_i$ obtains a better fitness than $X_i$, then $X_i$ is set to $T_i$; otherwise the old value $X_i$ is maintained.

**Improved SaDE (ISaDE)**

As mutation controls the algorithm exploration, it is the core operator of SaDE. Improving this parameter increases the algorithm search capability. In this study, an improved version of the mutation operator is proposed to improve the
optimization ability of standard SaDE. The new algorithm is named the improved SaDE (ISaDE) algorithm. The rest of the parts are the same as in the original SaDE algorithm except for a mutation phase which is introduced here. The result is much easier to implement compared with other extensions of the SaDE algorithm.

The new mutation operator is defined as follows:

\[
M_i = X_{c_1} + \left( X_{c_2} - X_{c_3} \right) \times 3e^{-\left( \frac{k}{I} \right)^2},
\]

where \(i \neq c_1 \neq c_2 \neq c_3\) (13)

in which \(k\) is the current iteration, and \(I\) is the maximum number of iterations; \(c_i\) is the \(i\)th chaotic variable generated based on the Logistic chaotic map. The reason for using a Logistic map is that it shows good chaotic characteristics, represents better randomness than other maps, and helps the algorithm to explore the points that are scattered around the search space as far as possible. The variable \(c_i\) is computed as follows:

\[
c_i^{k+1} = ac_i^k (1 - c_i^k)
\]

where \(a\) is a scaler value, \(c_i^k\) is the \(k\)th chaotic number in the chaotic sequence, and \(k\) is the index of the chaotic sequence. The initial value of \(c_i^0\) is in the range of \((0, 1)\), provided that \(c_i^0 \notin \{0, 0.25, 0.5, 0.75, 1\}\). In this work, \(a = 4\) is used. Equation (14) increases the exploration power of the algorithm such that different points of the search space are explored. The crossover and selection operators are left without change. Figure 3 illustrates the flowchart of the ISaDE algorithm.

The proposed ISaDE-SVR method

As mentioned before, the prediction performance of the SVR with RBF kernel is highly dependent on the three parameters \(C, \gamma,\) and \(\varepsilon\). In other words, selection of optimal values for these parameters improves the speed of training and increases the performance of classification as well. To find the optimal values of the parameters and optimum feature selection, we used the ISaDE optimization algorithm. Figure 4 shows the proposed ISaDE-SVR approach for SVM parameter selection and feature extraction. This algorithm provides a better training effect and improves the prediction accuracy.

A detailed description of the ISaDE-SVR follows.

Generate initial population

Since the feature subset selection and the SVR’s parameter optimization should be addressed simultaneously, each candidate solution in the population is composed of a feature permutation and parameter combination, as follows

\[
X_i = \{p_1, p_2, p_3, f_1, f_2, \ldots, f_6\}
\]

where \(p_1, p_2, p_3\) are float numbers, which are candidate values for the three parameters \(C, \sigma,\) and \(\varepsilon\). These values are generated randomly. The range of values for \(p_1\) is \([0, 100]\), and for \(p_2\) and \(p_3\) it is \([0, 1]\). Each feature \(f_i\) is a
binary variable and its value is 1 when it is considered in the model, and 0 when the feature is ignored.

Computing the fitness function

The fitness of each solution is evaluated using the mean squared error (MSE) of five-fold cross-validation for the SVR. The fitness function is defined as

\[
f(x) = \frac{1}{n} \sum_{i=1}^{n} (\hat{X}_i - X_i)^2
\]  

where \(X_i\) is the observed value, \(\hat{X}_i\) is the predicted value, and \(n\) is the total number of data in the dataset.

The individual with a smaller value of MSE is more preferable. The idea behind using the cross-validation strategy is to prevent under-fitting or over-fitting effects in the ISaDE-SVR approach. In \(n\)-fold cross validation, the training set is divided into \(n\) equal subsets. At each iteration, one of the subsets is taken as the testing set in turn, and \(n-1\) subsets are considered as training sets in the SVR method, then the above procedure is iterated until each subset is validated once.

Until the termination conditions are met, three operators including mutation, crossover and selection are iteratively carried out to update the population.

EXPERIMENT AND DISCUSSION

Evaluation of ISaDE algorithm

To evaluate the performance of the proposed ISaDE algorithm, 14 test problems were used. These functions include nine unimodal and four multimodal functions. Table 1 lists the characteristics of the benchmark problems. A detailed description of the test functions is available in Civicioglu (2013) and Karaboga & Akay (2009).

The ISaDE algorithm is compared with five algorithms including particle swarm optimization (PSO) (Vesterstrom & Thomsen 2004), artificial bee colony (ABC) (Karaboga & Basturk 2007), SaDE (Civicioglu 2013), grey wolf optimization (GWO) (Mirjalili et al. 2014), and salp swarm optimization (SSA) (Mirjalili et al. 2017). The PSO algorithm simulates the swarm intelligence behavior of bird flocking and schooling in nature. In every iteration of PSO, the positions of search agents are updated using the local best knowledge found by each agent and the global knowledge found by the swarm. The ABC simulates the cooperation among honey bees in finding food sources. SaDE is an extension of the differential evolution algorithm that utilizes a self-adaptive approach to tune the parameters. The GWO algorithm simulates the hunting behavior of grey wolves. In GWO, there are four kinds of wolves: alpha, beta, delta and omega. The algorithm updates the positions of the wolves using three main operators: finding prey, encircling prey and attacking prey. These operators hopefully cause the greys to converge to the global optimum point in solution space. The SSA algorithm mimics the swarming mechanism of salps in foraging and navigating in
Table 1 | Test functions used in the test

| Function   | Formulation                                                                 | Dimension | Range           | Minimum |
|------------|-----------------------------------------------------------------------------|-----------|-----------------|---------|
| Easom      | $f_1(x) = -\cos(x_1)\cos(x_2)\exp(-((x_1-\pi)^2 + (x_2-\pi)^2))$            | 2         | $[-100, 100]$   | -1      |
|            | $f_2(x) = 100(x_1^2 - x_2^2) + (x_1 - 1)^2 + (x_2 - 1)^2 + 90(x_2^2 - x_2)$ |           |                 |         |
|            | $f_3(x) = 10.1((x_2 - 1)^2 + (x_4 - 1)^2) + 19.8(x_2 - 1)(x_4 - 1)$          | 4         | $[-10, 10]$     | 0       |
| Beale      | $f_5(x) = (1.5 - x_1 + x_1x_2)^2 + (2.25 - x_1 + x_1x_2)^2 + (2.625 - x_1 + x_1x_2)^2$ | 5         | $[-4.5, 4.5]$   | 0       |
| Trid 6     | $f_6(x) = \sum_{i=1}^n (x_i - 1)^2 - \sum_{i=1}^n x_i x_{i-1}$             | 6         | $[-36, 36]$     | -50     |
| Quartic    | $f_7(x) = \sum_{i=1}^n x_i^2 + \text{random}[0, 1]$                        | 30        | $[-1.28, 1.28]$ | 0       |
| Zakharov   | $f_8(x) = \sum_{i=1}^n x_i^2 + (\sum_{i=1}^n 0.5ix_i^2) + (\sum_{i=1}^n 0.5ix_i)^4$ | 10        | $[-5, 10]$     | 0       |
| Powell     | $f_9(x) = \sum_{i=1}^n x_i^2 + (\sum_{i=1}^n 0.5ix_i^2) + (\sum_{i=1}^n 0.5ix_i)^4$ | 24        | $[-4, 5]$      | 0       |
| Schwefel 2.22 | $f_{10}(x) = \sum_{i=1}^n |x_i| + \prod_{i=1}^n |x_i|$                 | 30        | $[-10, 10]$    | 0       |
| Rosenbrock | $f_{11}(x) = \sum_{i=1}^n [100(x_i - x_i)^2 + (x_i - 1)^2]$                | 30        | $[-30, 30]$    | 0       |
| Michalewicz10 | $f_{12}(x) = -\sum_{i=1}^n \sin(x_i)\sin((ix_i^2/\pi)^m)$, $m = 10$       | 10        | $[0, \pi]$     | -9.66   |
|            | $f_{13}(x) = \sum_{i=1}^n (A_i - B_i)^2$                                   |           |                 |         |
| FletcherPowell 10 | $A_i = \sum_{j=1}^n (a_{ij}\sin a_j + b_{ij}\cos a_j)$                 | 10        | $[-\pi, \pi]$  | 0       |
|            | $B_i = \sum_{j=1}^n (a_{ij}\sin x_j + b_{ij}\cos x_j)$                    |           |                 |         |
| Rastrigin  | $f_{14}(x) = 10n + \sum_{i=1}^n (x_i^2 - 10\cos(2\pi x_i))$                | 30        | $[-5.12, 5.12]$ | 0       |
| Griewank   | $f_{15}(x) = \frac{1}{4000}\left(\sum_{i=1}^n x_i^2\right) - \left(\prod_{i=1}^n \cos\left(\frac{x_i}{\sqrt{i}}\right) + 1\right)$ | 30        | $[-600, 600]$  | 0       |
| Ackley     | $f_{16}(x) = -20\exp\left(-0.2\sqrt{\sum_{i=1}^n x_i^2}\right) - \exp\left(\sum_{i=1}^n \cos(2\pi x_i)\right) + 20 + e$ | 30        | $[-32, 32]$    | 0       |

Oceans. This algorithm updates the positions of salps using their local and global knowledge. Table 2 shows the multi-problem-based pairwise comparison among the algorithms using the Wilcoxon signed-rank test (Derrac et al. 2011). The comparison is performed based on the mean cost values obtained over 30 simulation runs. This test is considered to statistically evaluate the performance of the algorithm when solving several benchmark functions. The results testify that the ISaDE is more successful than its counterparts in converging to the global optimum of the benchmark functions, with a significance value $\alpha = 0.05$. Overall, the ISaDE outperforms the other algorithms in terms of convergence rate and solution quality. Due to the excellent performance of the ISaDE in solving numerical optimization problems, we used it to predict the discharge coefficient of labyrinth weirs.

Table 2 | Multi-problem-based statistical pairwise comparison of ISaDE and comparison algorithms using two-sided Wilcoxon signed-rank test ($\alpha = 0.05$)

| Comparison      | $T_+$ | $T_-$ | $p$-value | Winner |
|-----------------|-------|-------|-----------|--------|
| ISaDE vs PSO    | 57    | 9     | 0.03318   | ISaDE  |
| ISaDE vs ABC    | 76    | 15    | 0.03138   | ISaDE  |
| ISaDE vs SaDE   | 21    | 0     | N/A       | ISaDE  |
| ISaDE vs GWO    | 40    | 5     | 0.02202   | ISaDE  |
| ISaDE vs SSA    | 96    | 9     | 0.00634   | ISaDE  |
Benchmark

To examine the capabilities of the proposed method in predicting the $C_d$ of sharp-crested W-planform weirs, Kumar et al. (2012) and Ghodsian (2009) are used. This dataset contains 223 records, each one consisting of possible values for six hydraulic and geometric parameters of the weir and a measured value for $C_d$. This dataset was gathered in a horizontal rectangular tilting flume made of glass walls with a width of 0.245 m, length of 5.360 m, and a depth of 0.450 m. The sharp-crested W-planform labyrinth weirs were placed 5.150 m away from the flume entrance. The experiments were done with a wide range of W-planform weir parameters including the total length of the weir ($L_{w}$), length of one cycle of the weir ($L_{c}$), the width of one cycle of the weir ($W_{c}$), vertex angle ($\theta$) and flow depth ($y$). The schematic of the flume and the position of the weir used in the experiments are shown in Figure 5. The parameters of the dataset are given in Table 3. We split the dataset into two sets: training and test set. The number of 180 records is randomly selected to be in the training set and the remainder form the test set.

![Figure 5 | Schematic of experimental setup (Kumar et al. 2012).](http://iwaponline.com/ws/article-pdf/21/7/3459/960753/ws021073459.pdf)

**Table 3 | Range of data in the experiments**

| Parameter | Description | Min     | Max     | Avg     | STDEV |
|-----------|-------------|---------|---------|---------|-------|
| $W_{mc}$  | Channel width (m) | 0.245   | 0.300   | 0.271   | 0.019 |
| $W_{c}$   | Cycle width (m)   | 0.123   | 0.280   | 0.213   | 0.075 |
| $L_{w}$   | Weir length (m)   | 0.245   | 1.200   | 0.475   | 0.282 |
| $L_{c}$   | Cycle length (m)  | 0.123   | 1.082   | 0.373   | 0.263 |
| $P$       | Weir height (m)   | 0.092   | 0.170   | 0.110   | 0.024 |
| $H$       | Total head (m)    | 0.007   | 0.145   | 0.046   | 0.024 |
| $C_d$     | Discharge Coefficient (--) | 0.148   | 0.906   | 0.595   | 0.172 |

**Input parameters**

To investigate the most appropriate input parameters, six different input combinations were evaluated. To select the most effective input parameters, first, all input parameters are considered for the development of the ISaDE-SVR model and then one of the input parameters is removed from the inputs and the model is re-trained and tested with the same structure. Table 4 presents the different input combinations. The input parameters are non-dimensional ones that have been used in several other investigations (Parsaie & Haghiabi 2017; Haghiabi et al. 2018; Haghiabi et al. 2019).

**Performance criteria**

Statistical criteria of correlation coefficient ($R^2$), root mean square error (RMSE), mean absolute percentage error (MAPE) and averaged absolute deviation ($\bar{\delta}$) were used to evaluate the efficiency of the proposed method along with other models. RMSE computes the model error at the same scale as the output parameters, but MAPE and $\bar{\delta}$ compute the non-dimensional error of the model. These criteria are defined as follows:

\[
R^2 = \frac{\sum_{i=1}^{n} (e_i - \bar{e})(p_i - \bar{p})^2}{\sum_{i=1}^{n} (e_i - \bar{e})^2 \sum_{i=1}^{n} (p_i - \bar{p})^2}
\]

\[
RMSE = \sqrt{\frac{1}{n} \sum_{i=1}^{n} (e_i - p_i)^2}
\]

**Table 4 | Input scenario**

| Model | Input combinations | Absent |
|-------|--------------------|--------|
| $\Pi_1$ | $H/P, L_{w}/W_{mc}, L_{c}/W_{c}, L_{w}/W_{c}, H/W_{c}$ | $-$ |
| $\Pi_2$ | $H/P, L_{w}/W_{mc}, L_{c}/W_{c}, H/W_{c}$ | $L_{w}/W_{c}$ |
| $\Pi_3$ | $L_{w}/W_{mc}, L_{c}/W_{c}, H/W_{c}$ | $H/P$ |
| $\Pi_4$ | $H/P, L_{c}/W_{c}, H/W_{c}$ | $L_{w}/W_{mc}, L_{w}/W_{c}$ |
| $\Pi_5$ | $H/P, L_{w}/W_{mc}, H/W_{c}$ | $L_{c}/W_{c}, L_{w}/W_{c}$ |
| $\Pi_6$ | $H/P, L_{w}/W_{mc}, L_{c}/W_{c}$ | $L_{w}/W_{c}, H/W_{c}$ |

Downloaded from http://iwaponline.com/ws/article-pdf/21/7/3459/960753/ws021073459.pdf by guest.
Table 5 | Performance evaluation of the ISaDE-SVR method for different input combinations

| Model | Train  | Test  | SVM parameters |
|-------|--------|-------|----------------|
|       | $R^2$ | RMSE | MAPE | $\delta$ | $R^2$ | RMSE | MAPE | $\delta$ | C | $\varepsilon$ | $\gamma$ |
| $\Pi_1$ | 0.884 | 0.009 | 1.202 | 1.290 | 0.845 | 0.011 | 1.343 | 1.352 | 100 | 0.5 | 1 |
| $\Pi_2$ | 0.991 | 0.006 | 0.802 | 0.810 | 0.972 | 0.007 | 0.879 | 0.880 | 10 | 1 | 1 |
| $\Pi_3$ | 0.834 | 0.011 | 1.225 | 1.150 | 0.825 | 0.013 | 1.324 | 1.162 | 10 | 0.5 | 1 |
| $\Pi_4$ | 0.980 | 0.008 | 0.920 | 0.941 | 0.965 | 0.009 | 0.963 | 1.102 | 10 | 1 | 1 |
| $\Pi_5$ | 0.778 | 0.014 | 1.403 | 1.221 | 0.759 | 0.016 | 1.530 | 1.422 | 100 | 0.5 | 1 |
| $\Pi_6$ | 0.996 | 0.005 | 0.521 | 0.792 | 0.982 | 0.006 | 0.612 | 0.843 | 10 | 1 | 1 |

Figure 6 | Comparison of observed $C_d$ and the predicted $C_d$ obtained by SVR on the training stage.
**RESULTS**

**Investigating the effect of input scenarios**

The performance of the proposed hybrid method was performed with six different input scenarios to obtain the $C_d$ of the W-planform weir. Table 5 presents the performance of the ISaDE-SVR method with the different input combinations and the optimized parameters of the SVR.

In ISaDE-SVR, the $\Pi_6$ model, which includes the most effective geometric parameters ($H/P$, $L_{w}/W_{mc}$, $\cdots$), $\cdots$,
$L_c/W_c)$, has the best results with $R^2 = 0.982$ and RMSE = 0.006 in the testing stage. As shown in Table 5, after model $\Pi_6$, models $\Pi_2$ and $\Pi_4$, which include the $(H/W_c)$ parameter in addition to the mentioned geometric parameters, have surprisingly appropriate and satisfactory results. However, their results are significantly weaker than the $\Pi_6$ model.

Figures 6–9 show the scatterplot of the proposed method for the $\Pi_6$ model in the training and testing stages. As indicated in these figures, by using the input combinations as $\Pi_6$, the performance of the ISaDE-SVR model is very close to the observed values. It can be concluded that the ISaDE-SVR method performs properly in predicting the $C_d$ of W-planform weirs.

The numerical results obtained by the proposed method and SVR method for the $\Pi_6$ combination are given in the training and testing stages based on the performance criteria in Tables 6 and 7.

![Scatterplot of the proposed method for the $\Pi_6$ model in the training and testing stages.](image)

**Figure 8** | Comparison of observed $C_d$ and the predicted $C_d$ obtained by SVR on the test stage.
In general, according to the results presented in Figures 6–9, and Tables 6 and 7, the capability of the proposed hybrid method is appropriate for predicting the $C_d$ of W-planform weirs.

**Figure 9** Comparison of observed $C_d$ and the predicted $C_d$ obtained by ISaDE-SVR on the test stage.

**Table 6** Performance of proposed methods in the training dataset

| Method   | $R^2$ | RMSE | MAPE | $\delta$ |
|----------|-------|------|------|----------|
| SVR      | 0.978 | 0.008| 0.745| 0.985    |
| ISaDE-SVR| 0.996 | 0.005| 0.521| 0.792    |

**Table 7** Performance of proposed methods in the test dataset

| Method   | $R^2$ | RMSE | MAPE | $\delta$ |
|----------|-------|------|------|----------|
| SVR      | 0.967 | 0.009| 0.802| 1.120    |
| ISaDE-SVR| 0.982 | 0.006| 0.612| 0.843    |
Figure 10 | The histogram of $\Omega$ index versus the $C_d$. 
To further examine the results of the ISaDE-SVR method, the $\Omega$ index was defined as the ratio of the predicted $C_d$ to the observed values to compare the accuracy of the results. This index helps to recognize the behavior of the models in overestimation or underestimation of the $C_d$. The proximity of the $\Omega$ index to unity indicates the acceptable performance of the proposed method compared with other methods. The histogram of $\Omega$ is plotted for the different input models in Figure 10.

The maximum, minimum, and average values of the $\Omega$ index for the six models are presented in Table 8. Among all the six models, the largest value of the $\Omega$ index is obtained for model $\Pi_5$. Therefore, models $\Pi_2$, $\Pi_4$, and $\Pi_6$ predict the discharge coefficient close to the observed values compared with the other combinations. The $H/P$, $L_w/W_{mc}$, and $L_c/W_c$ parameters were found to be the most effective parameters in the estimation of $C_d$.

Comparison of the results of ISaDE-SVR with previous studies

Given that limited studies have predicted the $C_d$ of W-planform weirs using soft computing techniques, in this section, the results of the ISaDE-SVR model with the input combination of $\Pi_6$ are compared with the prior studies of Haghiabi et al. (2017) and Haghiabi et al. (2019). Figure 11 and Table 9 show a comparison of the proposed method with the results of other mentioned studies. They show that the ISaDE-SVR method has a better performance with $R^2$ of 0.982 compared with ANFIS and GMDH models with $R^2$ of 0.97 and 0.94, respectively. The superiority of the proposed model over the others originates from two sources. The first reason is related to the good performance of the algorithms of the ISaDE and SVR models in recognition of the weight of each effective parameter, and the second reason is that the proposed model enjoys the benefits of both ISaDE and SVR models due to the hybrid property of the model.

**Table 8** | Maximum, minimum and average for the six input combination models

| Model | $\Omega_{max}$ | $\Omega_{min}$ | $\Omega_{ave}$ |
|-------|----------------|----------------|---------------|
| $\Pi_1$ | 1.021 | 0.881 | 0.980 |
| $\Pi_2$ | 1.023 | 0.906 | 0.999 |
| $\Pi_3$ | 1.019 | 0.889 | 0.978 |
| $\Pi_4$ | 1.012 | 0.901 | 0.990 |
| $\Pi_5$ | 1.128 | 0.822 | 1.003 |
| $\Pi_6$ | 1.022 | 0.903 | 0.999 |

**Table 9** | Evaluation of the performance of the ISaDE-SVR method and counterpart methods

| Model | $R^2$ | RMSE | MAPE | $\delta$ |
|-------|-------|------|------|---------|
| GMDH  | 0.946 | 0.055 | –    | –       |
| ANFIS | 0.970 | 0.045 | –    | –       |
| SVR   | 0.967 | 0.009 | 0.802 | 1.120   |
| ISaDE-SVR | **0.982** | **0.006** | **0.612** | **0.843** |

**CONCLUSION**

In this study, a novel hybrid ISaDE method, namely ISaDE-SVR, was used to predict the $C_d$ of W-planform weirs. For this purpose, six models were introduced by combining...
different effective input parameters to predict the $C_d$. Then, by analyzing the computational results, the superior model was determined. This model predicted the $C_d$ in terms of the ratio of total head over weir crest to the weir height ($H/P$), the ratio of the total length of the weir to the main channel width ($L_w/W_m$), and the ratio of the length of one cycle to the main channel width ($L_c/W_c$). Also, the $H/P$ parameter was the most important input parameter in predicting the $C_d$ of the W-planform weir using the proposed method. The results indicate that the proposed ISaDE-SVR model performs more efficiently than other previous studies. The capabilities of both algorithms of ISaDE and SVR models in recognition of the weight of each effective parameter and the combination of these capabilities in the proposed hybrid model are the reasons for the good performance of the model against the others. ISaDE-SVR does need parameter setting and it is a fast convergence algorithm. It outperformed the counterpart algorithms in finding the discharge coefficient. However, the performance of ISaDE-SVR is far from ideal. Therefore, interesting future work is to combine the ISaDE algorithm with artificial neural networks and neuro-fuzzy models to improve the performance of $C_d$ prediction. Another search direction is to apply the ISaDE-SVR method on other optimization problems to evaluate its potential and disadvantages.

ACKNOWLEDGEMENTS

The manuscript is original research in the field of water resources.

CONFLICT OF INTEREST

We (authors) declare that we have no conflict of interest.

DATA AVAILABILITY STATEMENT

All relevant data are included in the paper or its Supplementary Information.

REFERENCES

Akhbari, A., Hossein Zaji, A., Azimi, H. & Vafaeifard, M. 2017 Predicting the discharge coefficient of triangular plan form weirs using radial basis function and MS’ methods. J. Appl. Res. Water Wastewater 4 (1), 281–289.

Aydin, M. C. 2012 CFD simulation of free-surface flow over triangular labyrinth side weir. Adv. Eng. Softw. 45 (1), 159–166. doi:10.1016/j.advengsoft.2011.09.006.

Aydin, M. C. & Emiroglu, M. E. 2013 Determination of capacity of labyrinth side weir by CFD. Flow Meas. Instrum. 29, 1–8. doi:10.1016/j.flowmeasinst.2012.09.008.

Azamathulla, H. M., Haghabi, A. H. & Parsaie, A. 2016 Prediction of side weir discharge coefficient by support vector machine technique. Water Sci. Technol. Water Supply 16 (4), 1002–1016. doi:10.2166/ws.2016.014.

Azimi, A. H. & Hakim, S. S. 2019 Hydraulics of flow over rectangular labyrinth weirs. Irrig. Sci. 37 (2), 183–193. doi:10.1007/s00271-018-0616-6.

Bagheri, S. & Heidarpour, M. 2010 Application of free vortex theory to estimating discharge coefficient for sharp-crested weirs. Biosyst. Eng. 105 (3), 423–427. doi:10.1016/j.biosystemseng.2010.01.001.

Bilhan, O., Emiroglu, M. E. & Kisi, O. 2011 Use of artificial neural networks for prediction of discharge coefficient of triangular labyrinth side weir in curved channels. Adv. Eng. Softw. 42 (4), 208–214. doi:10.1016/j.advengsoft.2011.02.006.

Bilhan, O., Aydin, M. C., Emiroglu, M. E. & Miller, C. J. 2018 Experimental and CFD analysis of circular labyrinth weirs. J. Irrig. Drain. Eng. 144 (6), 04018007. doi:10.1061/(ASCE)IR.1943-7746(2018)144:6(04018007).

Bonakdari, H. & Zaji, A. H. 2018 New type side weir discharge coefficient simulation using three novel hybrid adaptive neuro-fuzzy inference systems. Appl. Water Sci. 8 (1), 10. doi:10.1007/s13201-018-0669-y.

Bonakdari, H., Ebtehaj, I., Gharabaghi, B., Sharifi, A. & Mosavi, A. 2020 Prediction of discharge capacity of labyrinth weir with gene expression programming. In: Intelligent Systems and Applications (K. Arai, S. Kapoor & R. Bhatia, eds), Springer, Cham, Switzerland, pp. 202–217. doi:10.20944/preprints202001.0513.v1.

Brest, J., Greiner, S., Boskovic, B., Mernik, M. & Zumer, V. 2006 Self-adapting control parameters in differential evolution: a comparative study on numerical benchmark problems. IEEE Trans. Evol. Comput. 10 (6), 646–657. doi:10.1109/TEVC.2006.872133.

Civicioglu, P. 2013 Backtracking search optimization algorithm for numerical optimization problems. Appl. Math. Comput. 219 (15), 8121–8144.

Cortes, C. & Vapnic, V. 1995 Support-vector networks. Mach. Learn. 20, 273–297.

Crookston, B. M. 2010 Labyrinth Weirs. PhD thesis, Utah State University, Logan, UT, USA.

Dehdar-behbahani, S. & Parsaie, A. 2016 Numerical modeling of flow pattern in dam spillway’s guide wall. Case study:
Balaroud dam, Iran. *Alexandria Eng. J.* **55** (1), 467–473. doi:10.1016/j.aej.2016.01.006.

Derrac, J., García, S., Molina, D. & Herrera, F. 2011 A practical tutorial on the use of nonparametric statistical tests as a methodology for comparing evolutionary and swarm intelligence algorithms. *Swarm Evol. Comput.* **1** (1), 3–18. doi:10.1016/j.swevo.2011.02.002.

Drucker, H., Burges, C. J. C., Kaufman, L., Smola, A. & Vapnik, V. 1997 Support vector regression machines. In: *Advances in Neural Information Processing Systems 9* (NIPS 1996) (M. C. Mozer, M. Jordan & T. Petsche, eds), MIT Press, Cambridge, MA, USA, pp. 155–161.

Eftekhari, I., Bonakdari, H., Zaji, A. H., Azimi, H. & Sharifi, A. 2015 Gene expression programming to predict the discharge coefficient in rectangular side weirs. *Appl. Soft Comput.* **35**, 618–628. doi:10.1016/j.asoc.2015.07.003.

Emami, S., Arvanaghi, H. & Parsa, J. 2018 Numerical investigation of geometric parameters effect of the labyrinth weir on the discharge coefficient. *J. Rehabil. Civ. Eng.* **6** (1), 1–9. doi:10.22075/jrce.2017.11428.1188.

Emin Emiroglu, M., Kisli, O. & Bilhan, O. 2010 Predicting discharge capacity of triangular labyrinth side weir located on a straight channel by using an adaptive neuro-fuzzy technique. *Adv. Eng. Softw.* **41** (2), 154–160. doi:10.1016/j.advengsoft.2009.09.006.

Ghodsian, M. 2009 Stage-discharge relationship for a triangular labyrinth spillway. *Proc. Inst. Civ. Eng. – Water Manag.* **162** (3), 173–178. doi:10.1680/wama.2009.00033.

Gupta, K. K., Kumar, S. & Ahmad, Z. 2015 Flow characteristics of sharp-crested W-planform weirs. In: *1st International Conference on Research in Science, Engineering & Management (IOCRESM 2014)*, International Journal of Advanced Technology & Engineering Research, pp. 176–180.

Haghiabi, A. H., Azamathulla, H. M. & Parsaie, A. 2017 Prediction of head loss on cascade weir using ANN and SVM. *ISH J. Hydraul. Eng.* **23** (1), 102–110. doi:10.1080/09715010.2016.1241724.

Haghiabi, A. H., Parsaie, A. & Ememgholizadeh, S. 2018 Prediction of discharge coefficient of triangular labyrinth weirs using Adaptive Neuro Fuzzy Inference System. *Alexandria Eng. J.* **57** (3), 1773–1782. doi:10.1016/j.aej.2017.05.005.

Haghiabi, A. H., Parsaie, A. & Shamshi, Z. 2019 Intelligent mathematical modeling of discharge coefficient of nonlinear weirs with triangular plan. *AUT J. Civ. Eng.* **3** (2), 149–156. doi:10.22060/ajce.2018.14833.5500.

Izadbaksh, M. & Hajiabadi, R. 2019 Modeling discharge capacity of labyrinth weirs through a learning machine approach. *J. Appl. Res. Water Wastewater* **6** (2), 100–108.

Karaboga, D. & Akay, B. 2009 A comparative study of Artificial Bee Colony algorithm. *Appl. Math. Comput.* **214** (1), 108–132.

Karaboga, D. & Basturk, B. 2007 A powerful and efficient algorithm for numerical function optimization: artificial bee colony (ABC) algorithm. *J. Glob. Optim.* **39** (3), 459–471.

Karami, H., Karimi, S., Bonakdari, H. & Shamshirband, S. 2016 Predicting discharge coefficient of triangular labyrinth weir using extreme learning machine, artificial neural network and genetic programming. *Neural Comput. Appl.* **29**, 983–989. doi:10.1007/s00521-016-2588-x.

Khoshbin, F., Bonakdari, H., Ashraf Talez, S. H., Eftekhari, I., Zaji, A. H. & Azimi, H. 2016 Adaptive neuro-fuzzy inference system multi-objective optimization using the genetic algorithm/singular value decomposition method for modelling the discharge coefficient in rectangular sharp-crested side weirs. *Eng. Optim.* **48** (6), 935–948. doi:10.1080/0305215X.2015.1071807.

Kisi, O., Bilhan, O. & Emiroglu, M. E. 2013 Anfis to estimate discharge capacity of rectangular side weir. *Proc. Inst. Civ. Eng. – Water Manag.* **166** (9), 479–487. doi:10.1680/wama.2011.00095.

Kumar, S., Ahmad, Z., Mansoor, T. & Himanshu, S. K. 2012 Discharge characteristics of sharp crested weir of curved plan-form. *Res. J. Eng. Sci.* **1** (4), 16–20.

Mehri, Y., Soltani, J. & Khashechi, M. 2019 Predicting the coefficient of discharge for piano key side weirs using GMDH and DGMDH techniques. *Flow Meas. Instrum.* **65**, 1–6. doi:10.1016/j.flowmeasinst.2018.11.002.

Mirjalili, S., Gandomi, A. H., Mirjalili, S. Z., Saremi, S., Faris, H. & Mirjalili, S. M. 2017 Salp Swarm Algorithm: a bio-inspired optimizer for engineering design problems. *Adv. Eng. Softw.* **114**, 163–191. doi:10.1016/j.advengsoft.2017.07.002.

Mirjalili, S., Mirjalili, S. M. & Lewis, A. 2014 Grey Wolf Optimizer. *Adv. Eng. Softw.* **69**, 46–61.

Najafzadeh, M., Etemad-Shahidi, A. & Lim, S. Y. 2016 Scour prediction in long contractions using ANFIS and SVM. *Ocean Eng.* **111**, 128–135. doi:10.1016/j.oceaneng.2015.10.053.

Olyaie, E., Banejad, H. & Heydari, M. 2019 Estimating discharge coefficient of PK-Weir under subcritical conditions based on high-accuracy machine learning approaches. *Iran. J. Sci. Technol. – Trans. Civ. Eng.* **45**, 89–101. doi:10.1007/s40996-018-0150-z.

Parsaie, A. & Haghiahi, A. H. 2017 Support Vector Machine to predict the discharge coefficient of sharp crested w-planform weirs. *AUT J. Civ. Eng.* **1** (2), 195–204. doi:10.22060/cej.2017.13005.5309.

Parsaie, A., Haghiahi, A. H. & Moradinejad, A. 2013 CFD modeling of flow pattern in spillway's approach channel. *Sustain. Water Resour. Manag.* **1** (3), 245–251. doi:10.1007/s40996-015-0020-9.

Parsaie, A., Azamathulla, H. M. & Haghiahi, A. H. 2018 Prediction of discharge coefficient of cylindrical weir-gate using GMDH-PSO. *ISH J. Hydraul. Eng.* **24** (2), 116–123. doi:10.1080/09715010.2017.1372226.

Parsaie, A., Haghiahi, A. H., Saneie, M. & Torabi, H. 2018b Applications of soft computing techniques for prediction of energy dissipation on stepped spillways. *Neural Comput. Appl.* **29** (12), 1393–1409. doi:10.1007/s00521-016-2667-z.

Robertson, G. K. 2014 *Labyrinth Weir Hydraulics: Validation of CFD Modelling*. MSc thesis, Stellenbosch University, Stellenbosch, South Africa.

Roushanger, K. & Soleymanzadeh, A. 2016 Predicting trapezoidal weir discharge coefficient using evolutionary algorithm.
Roushangar, K., Alami, M. T., Asl, M. M. & Shiri, J. 2017 Modeling discharge coefficient of normal and inverted orientation labyrinth weirs using machine learning techniques. *ISH J. Hydraul. Eng.* 25 (3), 331–340. doi:10.1080/09715010.2017.1327333.

Roushangar, K., Alami, M. T., Shiri, J. & Asl, M. M. 2018 Determining discharge coefficient of labyrinth and arced labyrinth weirs using support vector machine. *Hydrol. Res.* 49 (3), 924–938. doi:10.2166/nh.2017.214.

Shafiei, S., Najarchi, M. & Shabanlou, S. 2020 A novel approach using CFD and neuro-fuzzy-firefly algorithm in predicting labyrinth weir discharge coefficient. *J. Brazilian Soc. Mech. Sci. Eng.* 42 (1), 44. doi:10.1007/s40430-019-2109-9.

Vesterstrom, J. & Thomsen, R. 2004 A comparative study of differential evolution, particle swarm optimization, and evolutionary algorithms on numerical benchmark problems.

In: *Proceedings of the 2004 Congress on Evolutionary Computation (CEC2004)*, IEEE, Piscataway, NJ, USA, pp. 1980–1987.

Zaji, A. H. & Bonakdari, H. 2014 Performance evaluation of two different neural network and particle swarm optimization methods for prediction of discharge capacity of modified triangular side weirs. *Flow Meas. Instrum.* 40, 149–156. doi:10.1016/j.flowmeasinst.2014.10.002.

Zaji, A. H., Bonakdari, H. & Karimi, S. 2015 Radial Basis Neural Network and Particle Swarm Optimization-based equations for predicting the discharge capacity of triangular labyrinth weirs. *Flow Meas. Instrum.* 45, 341–347. doi:10.1016/j.flowmeasinst.2015.08.002.

Zaji, A. H., Bonakdari, H., Khodashenas, S. R. & Shamshirband, S. 2016 Firefly optimization algorithm effect on support vector regression prediction improvement of a modified labyrinth side weir’s discharge coefficient. *Appl. Math. Comput.* 274, 14–19. doi:10.1016/j.amc.2015.10.070.

First received 30 August 2020; accepted in revised form 3 April 2021. Available online 16 April 2021