

Applications of the anarchic society optimization (ASO) algorithm for optimizing operations of single and continuous multi-reservoir systems

Atiyeh Bozorgi, Omid Bozorg-Haddad, Mohammad-Mehdi Rajabi, Milad Latifi and Xuefeng Chu

ABSTRACT

To address water resources-related management issues, different evolutionary and heuristic algorithms have been developed in recent years. In this paper, a new algorithm is introduced for optimizing the operation of reservoir systems. Specifically, the anarchic society optimization (ASO) algorithm is applied to solve water resources management problems for the first time. In these problems, the operations of a single-reservoir hydropower system (Karun-4 reservoir) and a four-reservoir system with the objective of maximizing the profits of releases are optimized. The objective function values of the ASO algorithm and the genetic algorithm (GA) for Karun-4 reservoir are 1.254 and 1.535, respectively. The objective function value from the ASO algorithm is very close to the global optimum (1.213) from the non-linear programming (NLP). The optimal solution of the ASO algorithm for the four-reservoir system covers 93.88% of the NLP value, while the GA model only accounts for 91.86% of the global optimum, indicating that the ASO algorithm does have better performance. The applications of the ASO algorithm for three benchmark mathematical functions and the single- and multi-reservoir systems and the comparisons with the GA method demonstrate its acceptable performance and applicability to real, complex engineering problems.

Key words | anarchic society optimization (ASO) algorithm, genetic algorithm, optimization, reservoir operation

Atiyeh Bozorgi

Omid Bozorg-Haddad (corresponding author)

Department of Irrigation & Reclamation

Engineering, Faculty of Agriculture Engineering & Technology, College of Agriculture & Natural Resources,

University of Tehran,

Karaj,

Alborz,

Iran

E-mail: obhaddad@ut.ac.ir

Mohammad-Mehdi Rajabi

Milad Latifi

School of Civil Engineering, College of Engineering, University of Tehran,

Tehran,

Iran

Xuefeng Chu

Department of Civil & Environmental Engineering,

North Dakota State University,

Dept 2470,

Fargo,

ND 58108-6050,

USA

INTRODUCTION

Water supply is one of the most important goals in water resources management and planning. In this regard, dams can be used in order to achieve goals such as water supply, hydropower generation, and flood damage control and reduction. So the operation of dams requires an optimization algorithm as a tool to resolve these issues. Due to the complexity of optimal operation of a reservoir system and reducing usage of the classical optimization techniques, various evolutionary and heuristic algorithms have been developed and used.

Generally, the studies that have been carried out about operation of reservoir systems often require the use of optimization methods. These methods can be divided into classical and evolutionary and heuristic algorithms, which

are detailed as follows. First, classical methods are discussed. Murray & Yakowitz (1979) described a modified differential dynamic programming (DDP) technique which was used in multi-reservoir control problems. It was the most efficient procedure available for solving hydrology-related problems. Compared with other methods, an important feature of the DDP method is that the required computations do not grow exponentially with the dimension of the state or decision variables. Tauxe *et al.* (1979) used multi-objective dynamic programming (DP) for optimizing a multi-objective reservoir system by considering three objective functions, and evaluated the releases of the reservoir. Lall & Miller (1988) applied nonlinear programming

(NLP) methods to solve the problem of multi-objective reservoir operation and achieved satisfactory results. [Momtahan & Dariane \(2007\)](#) mentioned that linear programming (LP), DP, and stochastic dynamic programming (SDP) optimization models are useful tools in solving reservoir operation problems, but these models have some computational assumptions which reduce their efficiency and flexibility. As pointed out by [Rani & Moreira \(2010\)](#), the main limitation in the use of LP is that all objective functions and constraints should be linear. The limitations of the DP and SDP models are that they are subject to the curse of dimensionality and discretization of reservoir storage and inflow. [Sharif & Swamy \(2014\)](#) used the Lingo optimization modeling tool for two four-reservoir problems to evaluate its effectiveness. First, they solved the two four-reservoir problems with a linear objective function. Then, they developed their models for the two four-reservoir problems with a nonlinear objective function, and these were solved by using the discrete differential dynamic programming (DDDP) method. The results illustrated that Lingo was better than DDDP in the run time. [Soleimani *et al.* \(2016a, 2016b\)](#) used the SDP method to determine optimal reservoir operation rules. They used time series of reservoir inflows and water demands for the Aydoghmoush reservoir in Iran. Their results showed the advantage of considering uncertainties in the reservoir inflows and water demands in the SDP model. However, the classical optimization methods have some limitations, such as failure to gain the global optimum, tendency to become stuck in a local optimum, and curse of dimensionality ([Asgari *et al.* \(2015\)](#)).

Despite the availability of the classical methods, their limitations led the researchers to use evolutionary and heuristic algorithms for solving optimization problems, and the related algorithms are discussed herein ([Farhangi *et al.* \(2012\)](#); [Akbari-Alashti *et al.* \(2014\)](#); [Aboutalebi *et al.* \(2015, 2016\)](#); [Bozorg-Haddad *et al.* \(2016\)](#); [Soleimani *et al.* \(2016a\)](#); [Solgi *et al.* \(2016\)](#)). The best-known evolutionary algorithm is the genetic algorithm (GA), which uses the principles of biology. GA was first introduced by [Holland \(1975\)](#). [Jothiprakash & Shanthi \(2006\)](#) used GA in the optimization of reservoir operation. Particle swarm optimization (PSO) was introduced by [Eberhart & Kennedy \(1995\)](#), which was in the field of swarm intelligence. [Ghimire & Reddy \(2013\)](#) used the PSO algorithm to minimize the total annual square of

difference between the evaluated hydropower and the desirable one in a single hydropower reservoir system. [Bozorg-Haddad *et al.* \(2006\)](#) introduced the honey-bee mating optimization (HBMO) algorithm for the first time and examined its performance in optimization of the operation of multiple hydropower reservoirs ([Bozorg-Haddad *et al.* \(2008\)](#)). [Shah-Hosseini \(2007\)](#) developed the intelligent water drops (IWD) algorithm by considering the behavior of water droplets that always chose the optimal path to reach a river. The biography-based optimization (BBO) algorithm was introduced by [Simon \(2008\)](#). [Bozorg-Haddad *et al.* \(2015\)](#) used the BBO algorithm to optimize the operation of a single hydropower reservoir and a continuous four-reservoir system. Comparison with the GA algorithm demonstrated the superior performance of BBO. [Yang \(2008\)](#) developed the firefly algorithm (FA). [Yang \(2009\)](#) showed that the FA had a high ability in solving multi-optimization problems. [Garousi-Nejad *et al.* \(2016a\)](#) used the FA for optimal operation of a reservoir system for both irrigation and hydropower purposes, compared with the GA, and proved that FA had better performance. [Garousi-Nejad *et al.* \(2016b\)](#) applied the FA for maximizing release profits for discrete and continuous multi-reservoir systems. The results indicated the superiority of FA performance compared with the modified firefly algorithm (MFA). [Eskandar *et al.* \(2012\)](#) proposed the water cycle algorithm (WCA) and [Bozorg-Haddad *et al.* \(2014\)](#) used WCA and GA for the optimization of reservoir operation and showed the superiority of WCA in comparison with GA. Generally, evolutionary and heuristic algorithms can obtain the solution that has the closest distance to the global optimum. Depending on the problem, this best solution can be a local optimum or near optimum. However, these algorithms are able to achieve an acceptable solution without being stuck in the local optimum by searching the entire feasible solution space. Also, they do not have the limitations of the classical methods and they are able to solve complex engineering problems. In addition, the evolutionary algorithms have the potential to solve multi-objective problems.

This study involves the application of the anarchic society optimization (ASO) algorithm, a new approach in the category of evolutionary algorithms, for optimal operation of reservoir systems. [Ahmadi-Javid \(2011\)](#) proposed the ASO algorithm, inspired by a society in which members

behave anarchically and adventurously, and dislike stability. The ASO algorithm was used for discrete problems to minimize the objective function. Comparison with the GA and PSO algorithms indicated that the ASO algorithm performed better than the other two algorithms. [Shayeghi & Dadashpour \(2012\)](#) applied the ASO algorithm for optimizing a problem in the electronic field, compared with other methods such as PSO, and proved that the ASO algorithm was superior in convergence and computation efficiency. [Ahmadi-Javid & Hooshangi-Tabrizi \(2012\)](#) used the ASO algorithm for optimal problems with two objective functions in the field of industrial engineering. They demonstrated the superiority of this algorithm to PSO. [Ahmadi-Javid & Hooshangi-Tabrizi \(2015\)](#) applied the ASO algorithm for permutation scheduling flow problems with a linear objective function, showing the effectiveness of the ASO algorithm in such an application.

Generally, the methods used to solve optimization of reservoir systems are divided into two groups: classical methods, and evolutionary and heuristic algorithms. In recent years, the evolutionary and heuristic algorithms have been widely used to determine the optimal operation of reservoir systems. Many applications of the evolutionary and heuristic algorithms are based on natural processes of living organisms. Unlike others, the ASO algorithm is based on human societies and is used in this study to optimize (1) a single-reservoir system, Karun-4, with an objective function of minimizing the amount of shortage and (2) a four-reservoir system, with the aim of maximizing the profits from releases. To the best of our knowledge, it is the first time this ASO algorithm has been used to solve water resources management problems. The structure of the ASO algorithm used in this study is the same as the one proposed by [Ahmadi-Javid \(2011\)](#). The algorithm is easy to program and it also requires fewer parameters. The performance of the ASO algorithm is validated by solving the mathematical benchmark functions. The ASO algorithm is also further investigated and compared with GA in water resources applications.

METHODOLOGY

Anarchism is a political philosophy that advocates stateless societies based on voluntary institutions. Anarchists

believe that the state is unnecessary and harmful. These societies are self-governed and managed by volunteer institutions. Anarchists are not in favor of chaos and disorder, but are opposed to concentrating power in a part of society. They believe that in a free society without having a superior power, the economy will grow better and volunteer institutions can provide better services than the common governments. Anarchists are generally opposed to the rule of any state and they consider democracy as the tyranny of the majority ([Woodcock 2004](#)). According to this vision, society does not need a central government, and all society members relying solely on their personal views or experiences of other groups will be able to choose the right path. Although this vision was adopted in some countries in some historical periods, it did not have successful results. However, anarchism can be inspiring in applications in the fields of engineering and sciences.

The ASO algorithm is inspired by human behavior in a society based on anarchism ([Ahmadi-Javid 2011](#)). In this algorithm, all members of the society are behaving in an anarchic way to improve their positions in the society. The main feature of the behavior of members in the community is that the members are irregular, unstable, and anarchic, so that in some situations they move to random (or worst) positions instead of moving to better ones. Also, an increase in the level of members increases the anarchic level of members' behavior.

One feature of this algorithm is that each member selects the next position based on individual experiences, group experiences, or historical experiences. Finally, after some movements, at least one of the members can reach the near optimal solution. The ASO algorithm is able to search the whole solution space and avoid falling into local optima.

Formulation

Assume that S is the solution space and $f:S \rightarrow R$ (refers to real numbers) is a function which should be minimized in the space of S . Also, assume that a society with N members is searching for the best location of living (global minimum of function f in the S) in an unknown land (solution space).

$X_i(k)$ denotes the position of the i th member in the k th iteration. The best position experienced by all members in

the k th iteration is $X^*(k)$. The best position visited by the i th member during the first k iterations is $P_i(k)$. The best position visited by all members during the first k iterations is $G(k)$. The elements of the algorithm according to movement policies are detailed as follows.

Planning procedure for movement

As shown in Figure 1, some society members are randomly selected from the solution space and their fitness is evaluated. Based on the computed fitness, comparison with $X^*(k)$, $P_i(k)$, and $G(k)$, the movement policy and the new position of the member are selected. After sufficient iterations, at least one of the members reaches the near optimum.

Selection of movement policy based on the current position

The first movement policy for the i th member in the k th iteration is denoted by $MP_i^{current}(k)$ and is chosen by its current position. To choose this movement policy, a Fickleness Index (FI), a measure of a member's dissatisfaction with its situation, is introduced and compared with other members. If the objective function f is positive on S , the fickleness index can be defined as one of the following (Ahmadi-Javid 2011):

$$FI_i(k) = 1 - \alpha_i \frac{f(X^*(k))}{f(X_i(k))} - (1 - \alpha_i) \frac{f(P_i(k))}{f(X_i(k))} \quad (1)$$

$$FI_i(k) = 1 - \alpha_i \frac{f(G(k))}{f(X_i(k))} - (1 - \alpha_i) \frac{f(P_i(k))}{f(X_i(k))} \quad (2)$$

where α_i is a non-negative number in the interval $[0, 1]$. The fickleness index is a number in the interval $[0, 1]$. Depending on the value of the fickleness index, the i th member is able to choose its next position. A low value of $FI_i(k)$ means that the i th member is satisfied with its position; and thus, choosing the movement policy according to $X^*(k)$ is reasonable. Otherwise, the i th member is not satisfied with its position and will select an unexpected movement. Hence, depending on the value of $FI_i(k)$, the following movement policy is

proposed for the i th member:

$$MP_i^{current}(k) = \begin{cases} \text{moving towards } X^*(k) & 0 \leq FI_i(k) \leq \alpha_i \\ \text{moving towards a random } X_i(k) & \alpha_i \leq FI_i(k) \leq 1 \end{cases} \quad (3)$$

Selection of movement policy based on other members' positions

The second movement policy for the i th member in the k th iteration is denoted by $MP_i^{society}(k)$ and is chosen according to other members' positions. The closer the members get to the $G(k)$ or the closer the members get together, the more reasonable their behaviors are. Although any members should move towards $G(k)$, according to their anarchic natures, their movement is not predictable. The External Irregularity index, $EI_i(k)$, for the i th member in the k th iteration can be expressed as one of the following (Ahmadi-Javid 2011):

$$EI_i(k) = 1 - e^{-\theta_i [f(X_i(k)) - f(G(k))]} \quad (4)$$

$$EI_i(k) = 1 - e^{-\delta_i D(k)} \quad (5)$$

where θ_i and δ_i = positive numbers; and $D(k)$ = a suitable dispersion measure (e.g. coefficient of variations $CV(k)$). Equation (4) is a measure of society members' distance from $G(k)$. The closer the members are to the $G(k)$, the more rational their behaviors are. Otherwise, the members show an anarchic behavior. Equation (5) shows a diversity index in the society, which is directly related to the dispersion of society members. Also, using this index, it is expected that if society members have less diversity, they behave more regularly. Then, by selecting a threshold for $EI_i(k)$, the movement policy based on the other members' positions can be written as:

$$MP_i^{society}(k) = \begin{cases} \text{moving towards } G(k) & 0 \leq EI_i(k) \leq \text{threshold} \\ \text{moving towards a random } X_i(k) & \text{threshold} \leq EI_i(k) \leq 1 \end{cases} \quad (6)$$

The closer the $EI_i(k)$ is to zero, the more irregular behaviors of the members are. As the $EI_i(k)$ approaches unity, the members show more regular behaviors.

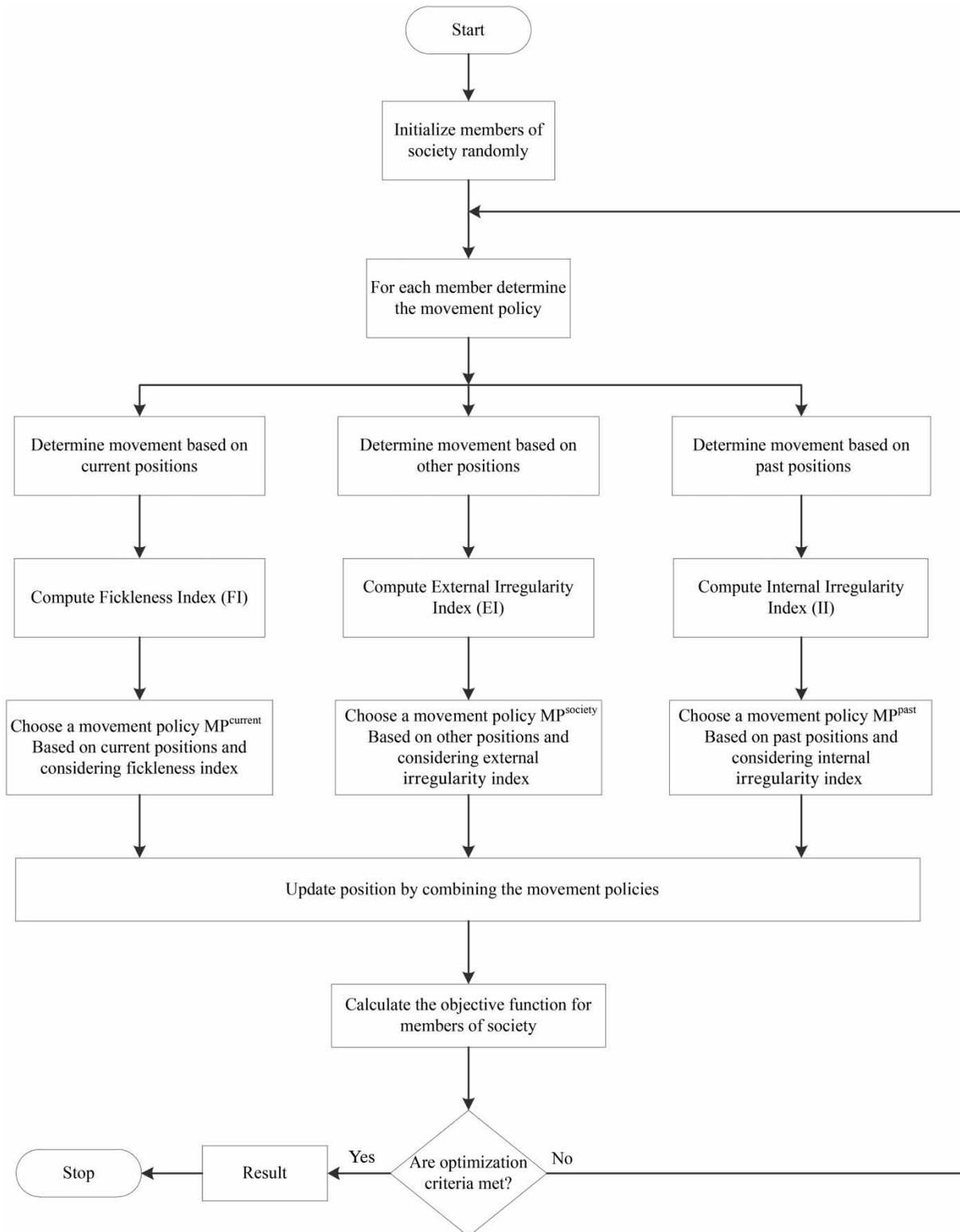


Figure 1 | Flowchart of the ASO algorithm.

Selection of movement policy based on the past positions

The third movement policy for the i th member in the k th iteration is denoted by $MP_i^{past}(k)$ and is chosen based on past positions of the members. To choose this movement policy, the position of the i th member in the k th iteration is compared with $P_i(k)$. The closer the position of $X_i(k)$ is to the $P_i(k)$, the more regular behavior the member shows. Otherwise, the member shows a random behavior. To select the movement policy based on the past positions, the Internal Irregularity index ($II_i(k)$) for the i th member in the k th iteration is defined as:

$$II_i(k) = 1 - e^{-\beta_i [f(X_i(k)) - f(P_i(k))]} \quad (7)$$

where β_i = a positive number. Similarly to the previous policy, by selecting a threshold for $II_i(k)$, the movement policy based on past positions is written as:

$$MP_i^{past}(k) = \begin{cases} \text{moving towards } P_i(k) & 0 \leq II_i(k) \leq \text{threshold} \\ \text{moving towards a random } X_i(k) & \text{threshold} \leq II_i(k) \leq 1 \end{cases} \quad (8)$$

As the $II_i(k)$ approaches zero, the members show more rational behaviors, while the members behave more regularly if the $II_i(k)$ approaches unity.

Combination of movement policies

To select the overall movement policy, the three above movements should be combined. For this purpose, the following methods can be used:

- elitism
- sequential; $MP^{current}(k) \rightarrow MP^{past}(k) \rightarrow MP^{society}(k)$
- crossover

Also, it is possible to crossover the movement policies sequentially. In this research, this method is used to combine the movement policies. [Ahmadi-Javid \(2011\)](#) showed that using the above definitions, ASO algorithm is a general case of PSO.

The steps of this modeling study are introduced in the flowchart in [Figure 2](#). First, the results of the ASO algorithm are verified with the GA's results in solving the benchmark problems. Then, the ASO algorithm is used for optimizing the reservoir operation problems and its capability is compared with that of the GA.

CASE STUDY

Mathematical test functions

In general, different test functions are used to evaluate an algorithm for various engineering problems that have clear relationships and constraints. In this way, the performance of the algorithm for the benchmark problems is determined. The mathematical functions for the ASO algorithm have been coded in the MATLAB environment and also the GA MATLAB toolbox has been used. The algorithms were run with an Intel Core i7 computer (2.93 GHz, 6GB DDR3 SDRAM, 1TB HDD). It took a couple of seconds to solve each optimization problem.

The mathematical functions used in this context are Ackley, Styblinski-Tang, and Holder table, which are described by Equations (9)–(11), respectively.

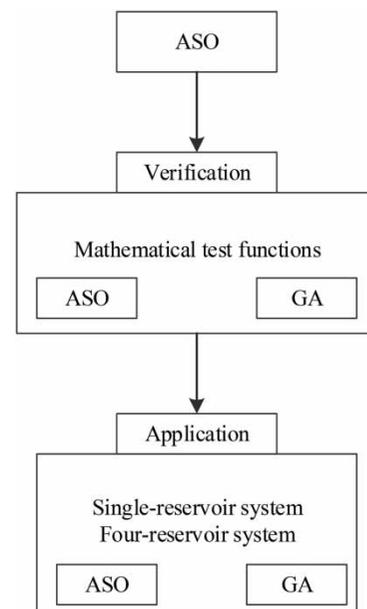


Figure 2 | Flowchart of different steps of research.

$$f(x_1, x_2) = -20 \exp(-0.2 \sqrt{0.5(x_1^2 + x_2^2)}) - \exp(0.5(\cos(2\pi x_1) + \cos(2\pi x_2))) + e + 20 - 5 \leq x_1, x_2 \leq 5 \quad (9)$$

$$f(x) = \frac{\sum_{i=1}^n x_i^4 - 16x_i^2 + 5x_i}{2} - 5 \leq x_i \leq 5, 1 \leq i \leq n \quad (10)$$

$$f(x_1, x_2) = - \left| \sin(x_1^2) \cos(x_2^2) \exp \left(\left| 1 - \frac{\sqrt{x_1^2 + x_2^2}}{\pi} \right| \right) \right| - 10 \leq x_1, x_2 \leq 10 \quad (11)$$

in which $f()$ = mathematical function; x = independent variable; $i = x$ index; and n = total number of x variables.

The Ackley function is a continuous one consisting of linear, exponential, and cosine functions. Its objective function is minimization with zero global optimum at point (0,0) in two dimensions. As illustrated in Figure 3(a), the Ackley function has many local optimum points and a global optimum point at the center.

The Styblinski-Tang function is a continuous and non-convex function and it has a minimization objective function. The global optimum solution is equal to -332.78 at point $(-903.2, -903.2)$. Figure 3(b) shows the two-dimensional plot of this function, which has three local minima and one global optimum. The ASO algorithm can achieve the global optimum in all runs of the program. However, the Holder table function is a multi-peak function that is defined only for two dimensions and it has a minimization objective function. As shown in Figure 3(c), this function has many local minima and four global minima with an amount of 208.19 at four points $(\pm 8.055, \pm 8.055)$. As indicated in Table 2, the ASO algorithm reached the global optimum at all runs, while GA did not reach the global optimal value in any run of the program.

Reservoir operation model with hydropower purpose

The objective function defined for operation of the reservoir system is to maximize the benefits of releasing, which can be expressed as:

$$B = \sum_{i=1}^n \sum_{t=1}^T b_i(t) R_i(t) \quad (12)$$

where B = total value of the objective function; t = time period; i = reservoir number; T = total number of operation periods; n = total number of reservoirs; $b_i(t)$ = benefit function of the i th reservoir in period t , and $R_i(t)$ = releasing amount of the i th reservoir during period t .

The most important constraint governing the operation of the reservoir system is the continuity equation, which is given by:

$$S_{(i,t+1)} = S_{(i,t)} + Q_{(i,t)} + M_{(i,j)} \cdot R_{(j,t)} + M_{(i,j)} \cdot Sp_{(j,t)} - Loss_{(i,t)} \quad (13)$$

for $j = 1, \dots, n \quad i = 1, \dots, n \quad t = 1, \dots, T$

in which $S_{(i,t)}$ and $S_{(i,t+1)}$ = storage volumes of the i th reservoir in time periods t and $t + 1$, respectively; $Q_{(i,t)}$ = inflow volume of the i th reservoir in time period t ; $M_{(i,j)}$ = matrix of the input-output connectivity among reservoirs; $Sp_{(i,t)}$ = overflow volume from the i th reservoir in time period t ; and $Loss_{(i,t)}$ = evaporation from the i th reservoir in time period t .

The evaporation loss is given by:

$$Loss_{(i,t)} = Ev_{(i,t)} \cdot \bar{A}_{(i,t)} \quad \text{for } i = 1, \dots, n \quad t = 1, \dots, T \quad (14)$$

$$\bar{A} = \frac{A_{(i,t)} + A_{(i,t+1)}}{2} \quad \text{for } i = 1, \dots, n, \quad t = 1, \dots, T \quad (15)$$

where $Ev_{(i,t)}$ = net evaporation (evaporation minus precipitation) for the i th reservoir in time period t ; $\bar{A}_{(i,t)}$ = average water area of the i th reservoir in time period t ; and $A_{(i,t)}$ and $A_{(i,t+1)}$ = water areas of the i th reservoir at the beginning and the end of time period t and $t + 1$, respectively.

The overflow volume from the reservoir is calculated by:

$$Sp_{(i,t)} = \max(s(i, t + 1) - s_{\max(i)}, 0) \quad (16)$$

for $i = 1, \dots, n \quad t = 1, \dots, T$

in which $s_{\max(i)}$ = maximum storage volume of the i th reservoir.

The constraints of releases and storage volumes of the reservoirs can be expressed as:

$$R_{\min(i,t)} \leq R_{(i,t)} \leq R_{\max(i)} \quad (17)$$

for $i = 1, \dots, n \quad t = 1, \dots, T$

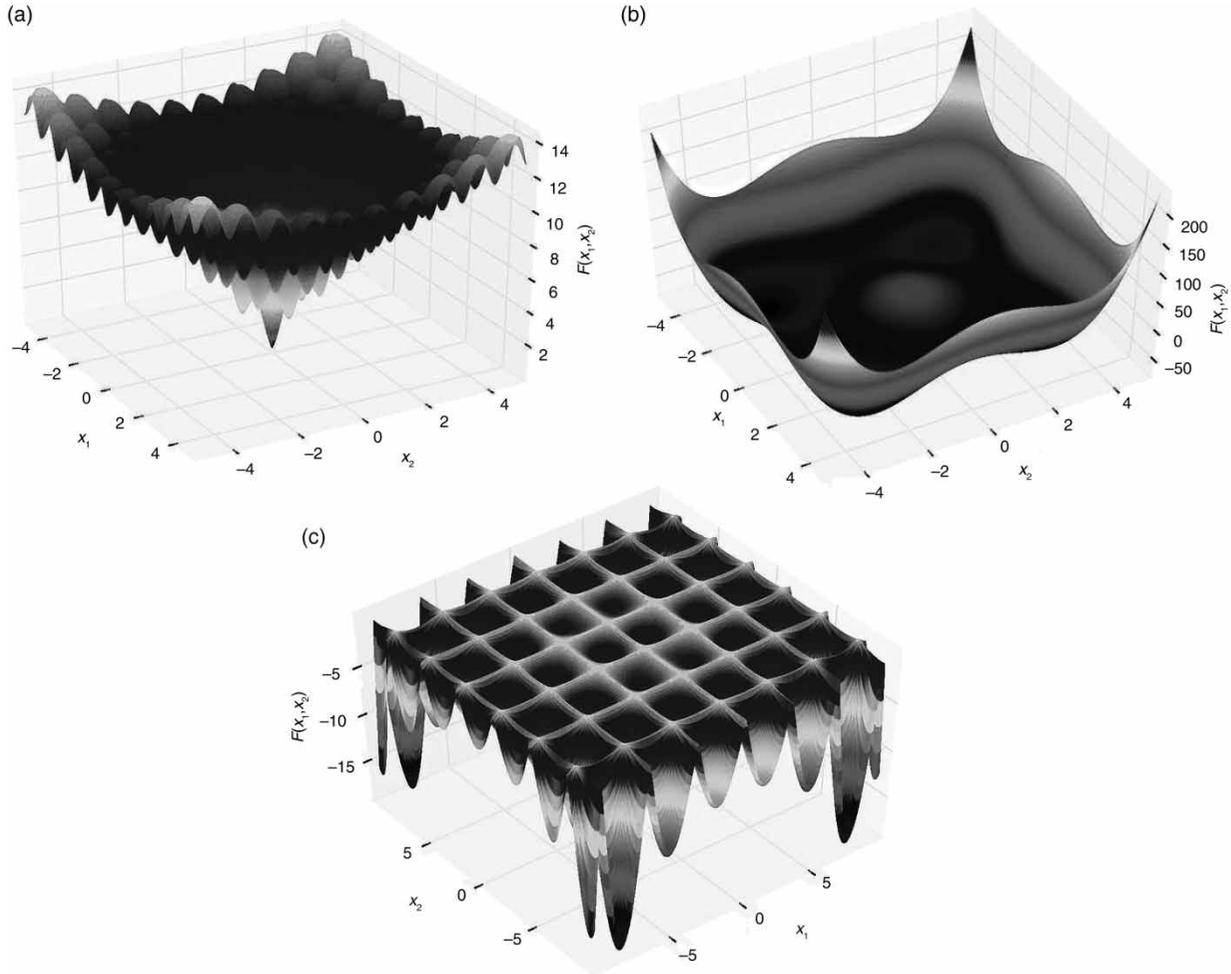


Figure 3 | Mathematical test functions: (a) Ackley; (b) Styblinski-Tang; (c) Holder table.

$$S_{\min(i)} \leq S_{(i,t)} \leq S_{\max(i)} \quad (18)$$

for $i = 1, \dots, n \quad t = 1, \dots, T$

where $R_{\min(i,t)}$ and $R_{\max(i,t)}$ = minimum and maximum releases for reservoir i in time period t , respectively; and $S_{\min(i,t)}$ = minimum limit of the storage volume for the i th reservoir in time period t .

The final constraint of the optimal reservoir operation model, which is related to the initial and desired volumes of the reservoirs in the last operation period, can be written as:

$$S_{(i,1)} = S_{(i,T+1)} \quad \text{for } i = 1, \dots, n \quad (19)$$

where $S_{(i,T+1)}$ = target storage volume of the i th reservoir at the end of the entire operation period.

Karun-4 reservoir

Karun-4 reservoir is located in Chaharmahal and Bakhtiari (geographic coordinates: 35°31'N, 50°24'E). The reservoir was built for hydroelectric power generation. Figure 4 shows the area-volume-elevation relationships of Karun-4 reservoir. The mathematical expressions for the volume-area and volume-elevation relationships used in the hydro-power reservoir problems are given by:

$$A_t = 9 \times 10^{-7} S_t^2 + 0.0057691 S_t + 13.0504272 \quad (20)$$

for $t = 1, \dots, T$

$$H_t = -6 \times 10^{-6} S_t^2 + 0.058067 S_t + 931.713958 \quad (21)$$

for $t = 1, \dots, T$

in which H_t = water level of the reservoir in time period t .

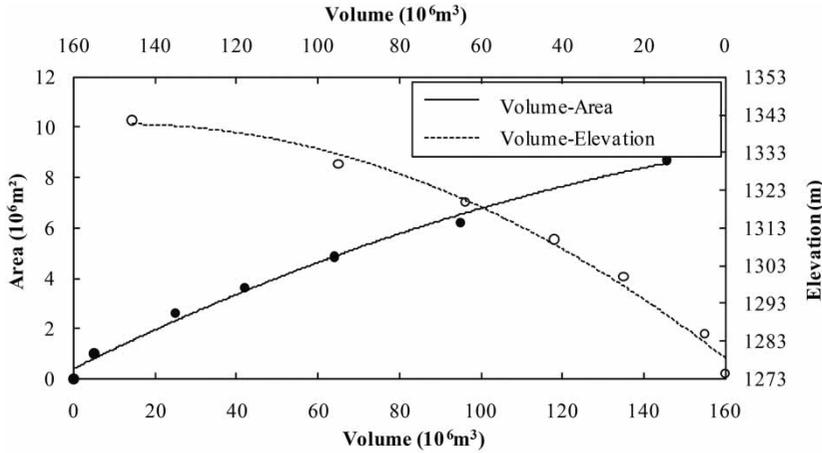


Figure 4 | Area-volume-elevation curves for Karun-4 reservoir.

The minimum and maximum volumes of the reservoir are 1.141×10^6 and $190 \times 10^6 \text{ m}^3$, respectively. The power plant capacity (PPC) is equal to $1.0 \times 10^9 \text{ W}$. Optimization of the Karun-4 reservoir operation was performed by using the ASO algorithm and GA on monthly time steps for a five-year period. The amount of generated power is calculated by:

$$P_{(t)} = ge \frac{Rp_{(t)}}{PF.Mul_{(t)}} \frac{[\bar{H}_{(t)} - Tw_{(t)}]}{1000} \quad \text{for } t = 1, \dots, T \quad (22)$$

where $P_{(t)}$ = power generated at time period t (10^6 W); g = gravity acceleration (m/s^2); e = efficiency of the power plant; $Rp_{(t)}$ = release of water from the power plant in time period t (10^6 m^3); PF = functional coefficient; $Mul_{(t)} = 10^6$ times the number of seconds in time period t ; $\bar{H}_{(t)}$ = average water level of the reservoir in time period t ; and $Tw_{(t)}$ = tail-water level of the reservoir in time period t (m).

In addition, the following constraints were also used in the optimization modeling:

$$Rps_{(t)} = R_{(t)} - Rp_{(t)} \quad \text{for } t = 1, \dots, T \quad (23)$$

$$\bar{H}_{(t)} = \frac{H_{(t)} + H_{(t+1)}}{2} \quad \text{for } t = 1, \dots, T \quad (24)$$

$$0 \leq P_{(t)} \leq PPC \quad \text{for } t = 1, \dots, T \quad (25)$$

where $Rps_{(t)}$ = overflow of the power plant after hydropower generating at time period t ; $H_{(t)}$ and $H_{(t+1)}$ = water levels of

the reservoir at the beginning and the end of the operation period; and PPC = power plant capacity (10^6 W).

The objective function used in this optimization problem is given by:

$$\text{Min}Z = \sum_{t=1}^T \left[1 - \frac{P_{(t)}}{PPC} \right]^2 \quad (26)$$

in which Z = lack of productivity.

Four-reservoir system

Chow & Cortes Rivera (1974) first introduced a hypothetical four-reservoir system with the objective function of maximizing benefits for a 12-month operation period (Figure 5). The data required for solving this problem can be obtained from Murray & Yakowitz (1979).

The hypothetical case study consists of four reservoirs connected in parallel and series (Figure 5). Each of the three upstream reservoirs (Reservoirs 1–3) is used for hydropower generating and the last one (Reservoir 4) is used for irrigation purposes.

The coefficient matrix of the four reservoirs can be expressed as:

$$M = \begin{bmatrix} -1 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 \\ 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & -1 \end{bmatrix} \quad (27)$$

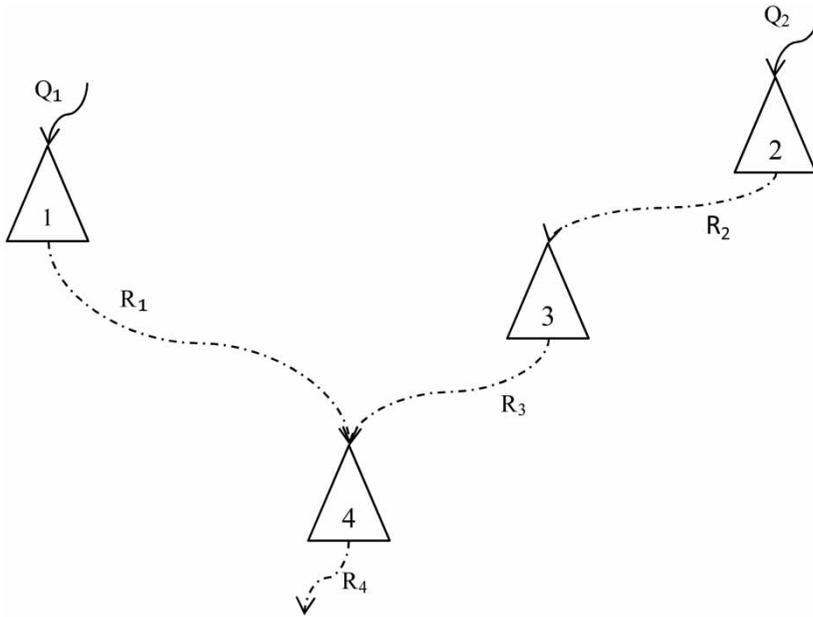


Figure 5 | Schematic of the four-reservoir system.

As shown in Equation (12), the objective function for the four-reservoir system is to maximize the benefits of releases. The constraints for operation of the four reservoirs are described in Equations (13)–(19).

In the evolutionary and heuristic algorithms, if constraints are not met, the penalty functions are defined so that they affect the amount of objective function to achieve a desired response. The penalties for violation of storage control, including the minimum, maximum, and target storages at the end of operation time, can be respectively expressed as:

$$P1_{(i,t)} = \begin{cases} 0 & \text{if } S_{(i,t+1)} > S_{\min(i)} \\ K_1[S_{\min(i)} - S_{(i,t+1)}]^2 & \text{otherwise} \end{cases} \quad (28)$$

$$P2_{(i,t)} = \begin{cases} 0 & \text{if } S_{(i,t+1)} > S_{\max(i)} \\ K_2[S_{(i,t+1)} - S_{\max(i)}]^2 & \text{otherwise} \end{cases} \quad (29)$$

$$P3_{(i,t)} = \begin{cases} 0 & \text{if } S_{(i,T+1)} > S_{(i,target)} \\ K_3[S_{(i,T+1)} - S_{(i,target)}]^2 & \text{otherwise} \end{cases} \quad (30)$$

where $S_{(i,target)}$ = target volume of the i th reservoir in the last time period of operation; $P1_{(i,t)}$, $P2_{(i,t)}$ and $P3_{(i,t)}$ =

penalty functions due to violation of the minimum and maximum storage volumes, and the target volume for the last time period, respectively, for reservoir i in time period t ; and K_1 , K_2 and K_3 = penalty coefficients due to violation of the minimum and maximum storage volumes, and the target volume for the last time period, respectively. In the modeling, the constants of K_1 , K_2 , and K_3 were equal to 60, 40, and 40, respectively.

Taking into account the penalty functions (Equations (28)–(30)), the modified objective function can be expressed as:

$$MaxB = \sum_{i=1}^n \sum_{t=1}^T [b_{(i,t)} \cdot R_{(i,t)} - [P1_{(i,t)} + P2_{(i,t)} + P3_{(i,t)}]] \quad (31)$$

RESULTS AND DISCUSSION

The modeling results obtained for (1) the three benchmark mathematical functions, (2) operation of the single-reservoir (Karun-4) system, and (3) operation of the multiple-reservoir system are detailed as follows.

Table 1 | Characteristics of ASO algorithm and GA used for three benchmark functions

	Characteristic	Ackley	Styblinski-Tang	Holder table
ASO	Population	7	7	7
	Number of evaluation	7,000	7,000	7,000
	α – Fickleness	0.01	0.01	0.9
	θ – External irregularity	0.10	0.10	0.01
	β – Internal irregularity	0.80	0.80	0.80
	Number of dimensions (variables)	2	2	2
GA	Population	7	7	7
	Number of evaluation	7,000	7,000	7,000
	Mutation function	Uniform	Uniform	Uniform
	Mutation rate	0.05	0.05	0.05
	Crossover function	Two-point crossover	Two-point crossover	Two-point crossover
	Crossover fraction	0.60	0.60	0.60
	Selection function	Roulette wheel	Roulette wheel	Roulette wheel

Results for three mathematical functions

The parameters used in the ASO algorithm and GA for the Ackley, Styblinski-Tang, and Holder table functions are listed in Table 1. As shown in Table 1, the GA and ASO algorithms were implemented with seven chromosomes and 1,000 generations. The modeling results of 10 runs of

GA and the ASO algorithm for the three benchmark functions are shown in Table 2.

The best solutions of the ASO algorithm for the Ackley, Styblinski-Tang, and Holder table benchmark functions from the 10 runs are 6.53×10^{-6} , -78.33 and -19.208 , respectively. The corresponding best solutions of the GA model for the three benchmark test functions

Table 2 | Values of objective function calculated with ASO algorithm and GA from 10 different runs for three benchmark functions

Run number	Ackley		Styblinski-Tang		Holder table	
	GA	ASO	GA	ASO	GA	ASO
1	7.83×10^{-3}	1.52×10^{-5}	-77.34	-78.33	-8.715	-19.208
2	2.17×10^{-2}	1.10×10^{-5}	-76.97	-78.33	-11.012	-19.208
3	1.53×10^{-2}	8.19×10^{-6}	-71.83	-78.33	-19.211	-19.208
4	1.97×10^{-2}	8.27×10^{-6}	-64.19	-78.33	-5.712	-19.208
5	4.27×10^{-2}	6.76×10^{-6}	-75.77	-78.33	-11.072	-19.208
6	3.61×10^{-2}	1.01×10^{-5}	-64.21	-78.33	-6.259	-19.208
7	2.34×10^{-2}	1.65×10^{-5}	-78.33	-78.33	-4.712	-19.208
8	1.50×10^{-2}	6.53×10^{-6}	-67.44	-78.33	-4.712	-19.208
9	1.58×10^{-2}	6.82×10^{-6}	-48.93	-78.33	-9.363	-19.208
10	1.01×10^{-2}	9.56×10^{-6}	-78.32	-78.33	-19.149	-19.208
Best	7.83×10^{-3}	6.53×10^{-6}	-78.33	-78.33	-19.211	-19.208
Average	2.08×10^{-2}	9.89×10^{-6}	-70.33	-78.33	-9.989	-19.208
Worst	4.27×10^{-2}	1.65×10^{-5}	-48.93	-78.33	-4.712	-19.208
Standard deviation	1.10×10^{-2}	3.49×10^{-6}	9.39	9.91×10^{-9}	5.38	1.40×10^{-4}
Coefficient of variation	5.31×10^{-1}	3.52×10^{-1}	-1.34×10^{-1}	-1.26×10^{-10}	-5.39×10^{-1}	-7.30×10^{-6}
Global	0		-78.33		-19.208	

are 7.83×10^{-3} , -78.33 and -19.21 , respectively. As shown in Table 2, the ASO algorithm results are closer to the global optima for all three test functions. Also, the ASO

algorithm resulted in better average and worst answers in the 10 consequent runs than the GA method. For the two test functions of Styblinski–Tang and Holder table,

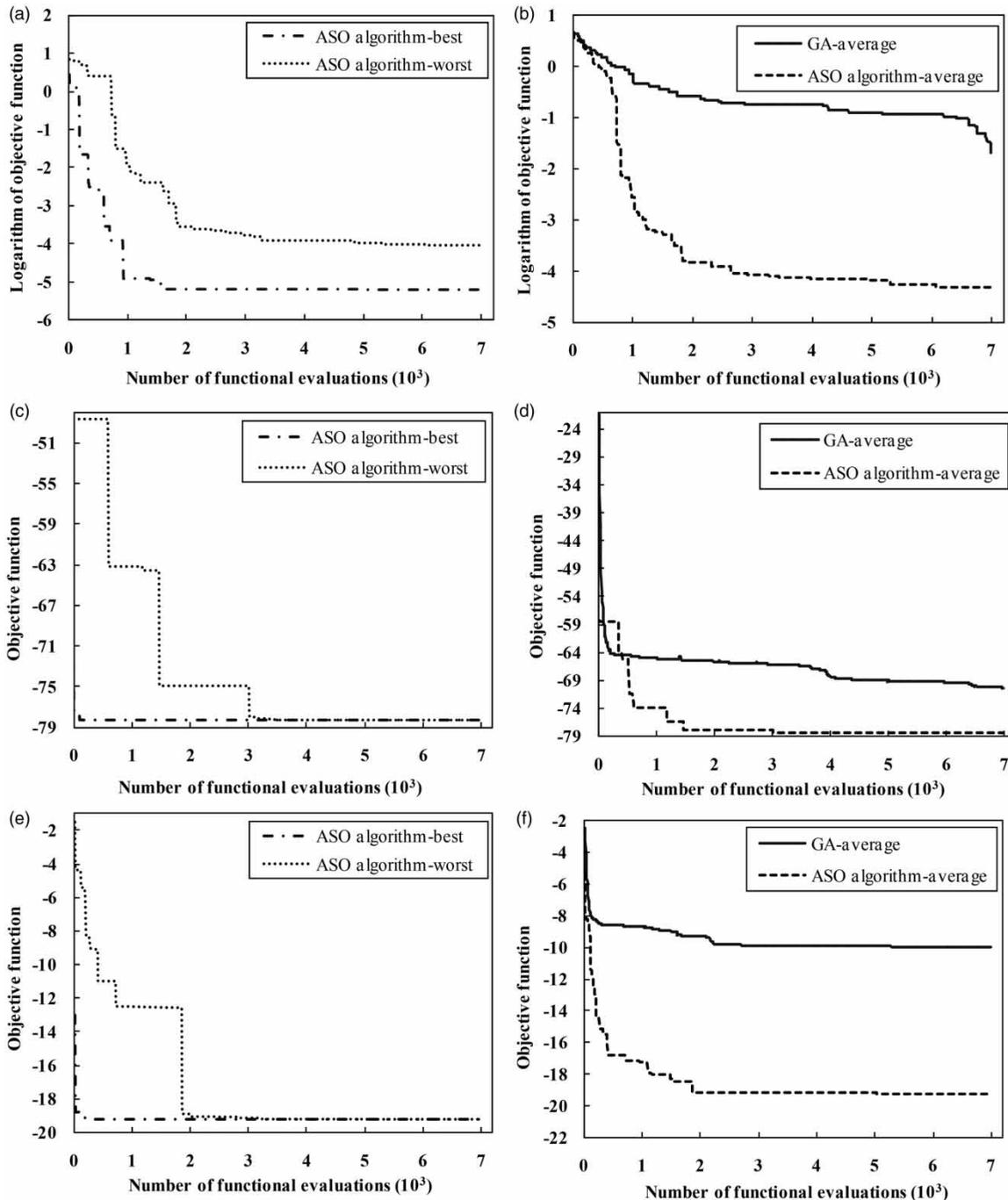


Figure 6 | Best and worst convergence of ASO for (a) Ackley, (c) Styblinski–Tang, and (e) Holder table functions; and average convergence of ASO and GA for (b) Ackley, (d) Styblinski–Tang, and (f) Holder table functions.

the results from the ASO algorithm are equal to the global optima. Also, the standard deviation and the coefficient of variation for the ASO algorithm are smaller than those for the GA algorithm, indicating better performance of the ASO algorithm.

Figure 6 shows the maximum and minimum objective functions calculated by the ASO algorithm and the average rates of convergence of ASO algorithm and GA for the three benchmark functions. As shown in Figure 6, the rate of convergence in the ASO algorithm is higher than that of GA. Also, the best result of the ASO algorithm is closer to the global optimum than that of GA. For all three mathematical test functions, convexity of the ASO algorithm curves is greater than that of the GA curves, showing that the ASO algorithm has better performance.

Results for the single-reservoir (Karun-4) system operation

The optimal operation of Karun-4 reservoir with the objective function of hydropower was determined by the ASO algorithm and the results were compared with those from the GA and NLP methods. The ASO algorithm and GA optimization models were coded by using MATLAB 2014. Also, Lingo 11.0 was used to attain the global optima.

Table 3 shows the parameters of the ASO algorithm and GA used for optimal operations of Karun-4 reservoir and the four-reservoir system. Table 4 displays a summary of the results with 10 different runs of the GA and ASO algorithm models for the single- and multi-reservoir systems. The global optimum obtained from the NLP model is 1.213. In the GA and ASO algorithm models for Karun-4 reservoir, 1,000 generations and 70 chromosomes for each generation were considered. The best solution of GA from the 10 random runs is 1.535, whereas the best solution of ASO algorithm is 1.254. Also, the average values of all results for the GA and ASO algorithm models are 1.292 and 1.746, respectively, which proves the better performance of the ASO algorithm.

As shown in Table 4, the standard deviations of the 10 runs of the GA and ASO algorithm models for Karun-4 reservoir are 0.162 and 0.022, respectively. The ASO

Table 3 | Parameters of the ASO algorithm and GA used for the reservoir system problems

Parameter		Karun-4	Four-reservoir
ASO	Population	70	40
	Iterations	1,000	10,000
	Number of evaluations	70,000	400,000
	α – Fickleness	0.90	0.90
	θ – External irregularity	0.90	0.01
	β – Internal irregularity	0.50	0.90
	Number of dimensions (variables)	60	48
GA	Population	70	40
	Iterations	1,000	10,000
	Number of evaluations	70,000	400,000
	Mutation function	Uniform	Uniform
	Mutation rate	0.02	0.03
	Crossover function	One-point crossover	One-point crossover
	Crossover fraction	0.60	0.50
	Selection function	Roulette wheel	Roulette wheel

algorithm yielded very stable results. The coefficient of variation for the ASO algorithm is about five times smaller than that from the GA model. Figure 7(a) shows the best and

Table 4 | Statistical outputs from 10 runs of the ASO algorithm and GA for the reservoir system problems

Run number	Karun-4		Four-reservoir	
	GA	ASO	GA	ASO
1	1.673	1.294	278.55	285.64
2	1.549	1.271	280.85	287.32
3	1.865	1.304	272.52	285.81
4	1.752	1.319	276.98	285.76
5	1.987	1.278	283.21	288.92
6	1.753	1.273	276.82	284.73
7	1.931	1.254	274.83	288.53
8	1.570	1.325	278.64	287.30
9	1.842	1.300	272.80	289.43
10	1.535	1.299	280.88	286.18
Best	1.535	1.254	283.21	289.43
Average	1.746	1.292	277.61	286.96
Worst	1.987	1.325	272.52	284.73
Standard deviation	0.162	0.0224	3.53	1.59
Coefficient of variation	0.093	0.0173	0.0127	0.0055
NLP	1.213		308.29	

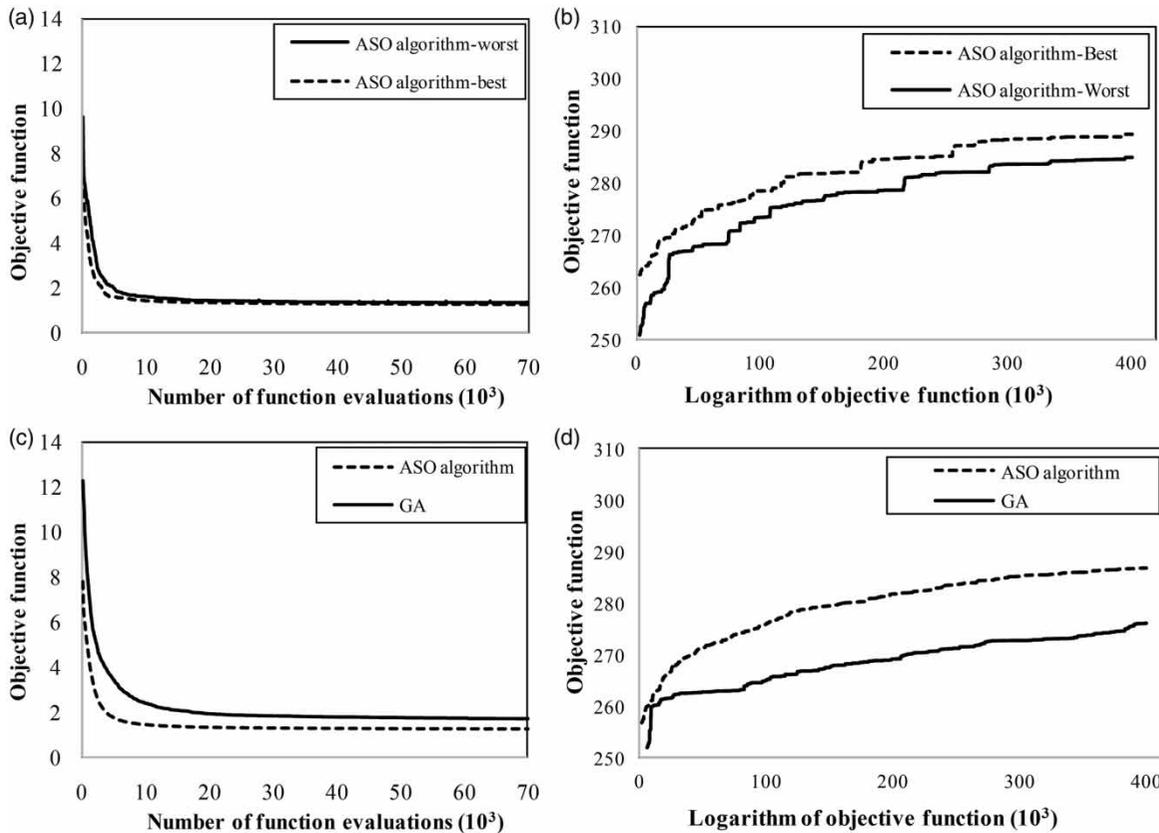


Figure 7 | Minimum (best) and maximum (worst) rates of convergence of the ASO algorithm over 10 runs for (a) Karun-4 reservoir and (b) four-reservoir system; and average rates of convergence of the GA and ASO methods from 10 runs for (c) Karun-4 reservoir and (d) four-reservoir system.

worst results of the ASO algorithm for all generations for Karun-4 reservoir. As shown in Figure 7(c), comparison of the rates of convergence and the convexity of the ASO algorithm and GA curves demonstrates that the ASO algorithm reached the global optimum faster.

Figure 8 displays the temporal changes in the release volumes from Karun-4 reservoir from the NLP, ASO algorithm, and GA methods in the five-year running period. A good likeness can be observed between the ASO algorithm, GA and global optimum curves despite certain variations (Figure 8). To highlight the main goal for operation of a hydropower system (i.e. to maximize the power production in each period), temporal changes in the reservoir storage and the power generated in each period are shown in Figures 9 and 10. The detailed variations in monthly storage of Karun-4 reservoir during the operation period within the upper and lower permissible ranges can be observed in Figure 9. As shown in Figure 10, power generation

calculated by the ASO algorithm in the operation duration is close to the NLP power production, showing high performance of the ASO algorithm method.

Results for multiple-reservoir system operation

Table 3 also lists the optimization parameters of the ASO algorithm for the four-reservoir problem. These parameters were obtained by using the trial and error method. Similarly to the Karun-4 reservoir model, the ASO algorithm and GA were executed 10 times. In this four-reservoir system model, each population has 40 chromosomes and the maximum generation is 10,000. Thus, the number of evaluations (NFE) is 400,000 (Table 3). The global optimum from the NLP model is 308.29.

The best solutions of the GA and ASO algorithm models from the 10 runs are 283.21 and 289.43, respectively. In addition to the best solution, the average solutions from

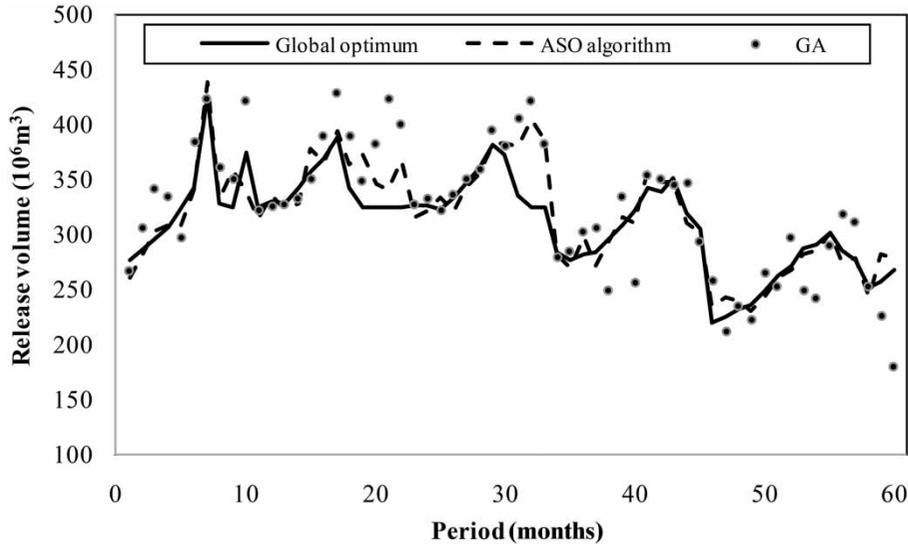


Figure 8 | Optimal releases of Karun-4 reservoir during the operation period.

the 10 runs for the GA and ASO algorithms are 277.61 and 286.96, respectively. The ASO algorithm results are closer to the NLP global optimum. The standard deviation of the ASO algorithm results is approximately two times smaller than that of GA, and the coefficient of variation for the ASO algorithm is also much lower than that for GA (Table 4). Therefore, the ASO algorithm has better stability, which proves its reliability.

Figure 7(b) shows the convergence of the best and worst solutions of the ASO algorithm; Figure 7(d) presents the

average convergence curves from all runs of the ASO algorithm and GA. The ASO algorithm result shows a higher rate of convergence to the global optimum. Figure 11 shows the comparison of the monthly releases of the four reservoirs (decision variables) and the global optima from the NLP model for different operation periods. Also, Figure 12 shows the comparison of the storage volumes of the four reservoirs (state variables) and the global optima from the NLP model. The similarity of these two curves proves the high performance of the ASO algorithm.

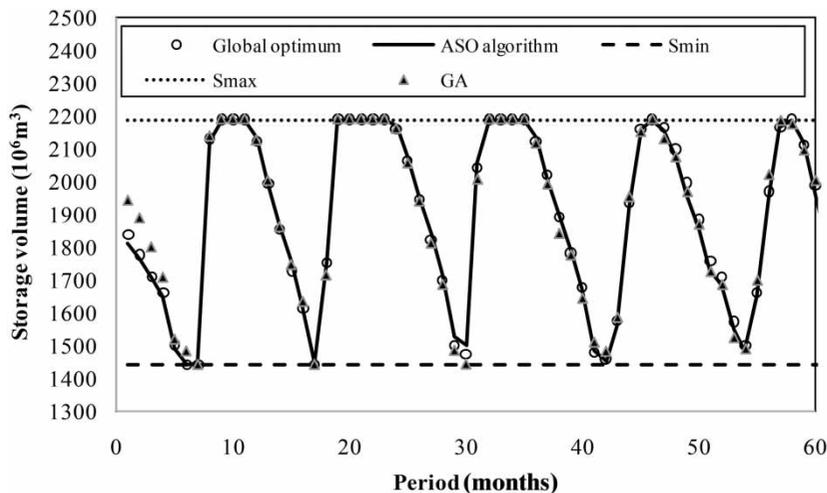


Figure 9 | Variations of optimal storage volumes of Karun-4 reservoir during the operation period.

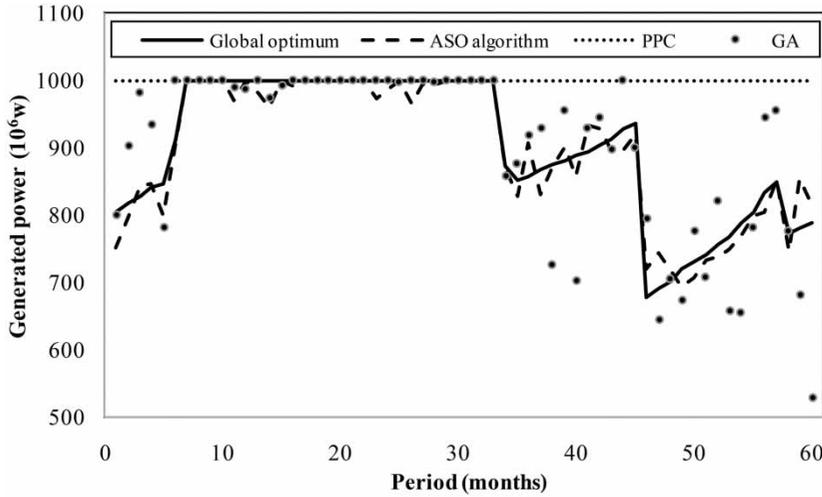


Figure 10 | Optimal power generation during the Karun-4 operation period.

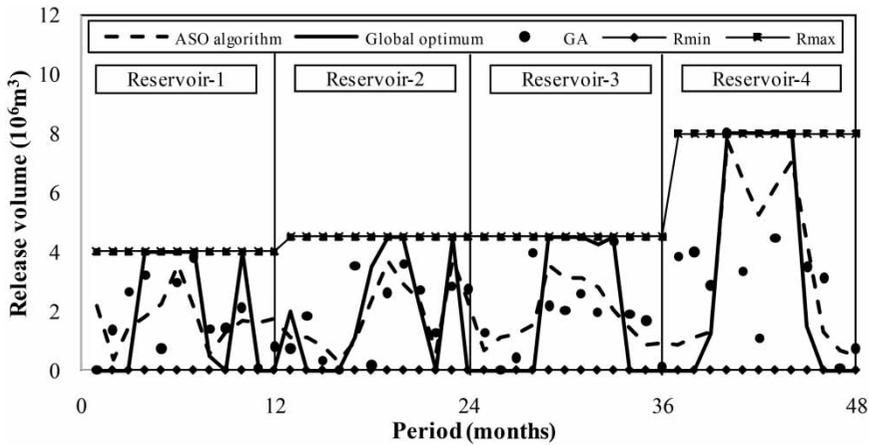


Figure 11 | Monthly releases of the four-reservoir system optimized by the ASO algorithm and GA.

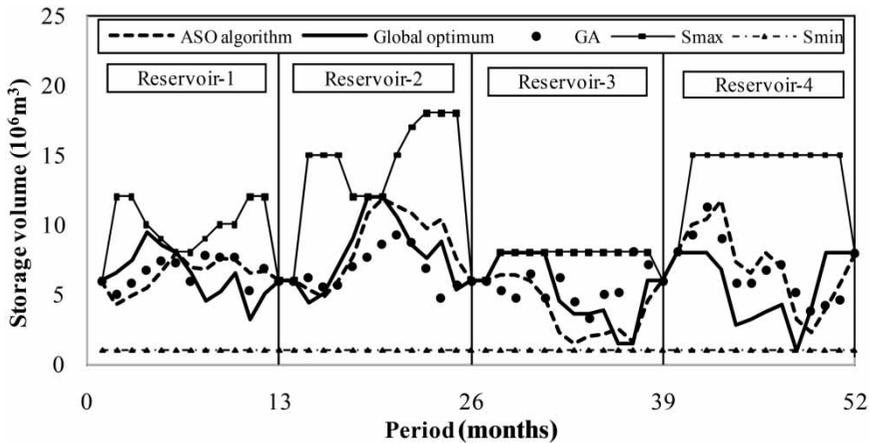


Figure 12 | Temporal variations in storage volume of the four-reservoir system optimized by the ASO algorithm and GA.

CONCLUSIONS

The evolutionary and heuristic algorithms use a general structure to search for the global optimum solution for a system. However, they employ different searching ways. Due to the special condition of each problem, it is better to modify the algorithm to improve its applicability and performance. These algorithms have been widely used. For applications of the ASO algorithm for optimal operation of a reservoir system, however, it is still at the initial stage. In this study, to demonstrate the performance of the ASO algorithm, three mathematical benchmark functions were solved with the ASO algorithm, and the results were compared with those from the GA method, indicating that the ASO algorithm results were closer to the optimal solutions.

Furthermore, the ASO algorithm was used for optimal operation of Karun-4 reservoir and a multiple-reservoir system with the goal to maximize the benefit of releases. In each case study, the ASO algorithm and GA were run 10 times. For the single-reservoir system, the best solutions of the ASO algorithm and GA were 1.254 and 1.535, respectively. The ASO algorithm model resulted in a much smaller coefficient of variation (0.0173) than the GA model (0.093). In addition, the average solutions from 10 runs of the ASO algorithm and GA were 1.292 and 1.746, respectively. These modeling results proved the high performance of the ASO algorithm in finding the global optimum for a single-reservoir system.

In the multi-reservoir case study, the best solutions of the ASO algorithm and GA were 289.43 and 283.21, respectively, showing that the ASO algorithm covered 93% of the global optimum, while GA covered 91.86% of the NLP result. The coefficients of variation for the ASO algorithm and GA were 0.0055 and 0.0127, respectively, which again demonstrated the high stability and reliability of the ASO algorithm. In this case, the ASO algorithm also provided better average solutions than GA. The results show that the ASO algorithm achieved 93.08% of the global solution, while GA reached 90.04% of the NLP result, which was 308.29. It thus can be concluded that the ASO algorithm is better than the GA method for determining optimal operation of a multi-reservoir system. The real applications of the ASO algorithm to optimizing the operations for both single and multiple reservoir systems and its good

performances demonstrate that the ASO algorithm has high potential for solving real and complex engineering problems.

REFERENCES

- Aboutalebi, M., Bozorg-Haddad, O. & Loáiciga, H. A. 2015 [Optimal monthly reservoir operation rules for hydropower generation derived with SVR-NSGAI](#). *J. Water Resour. Plan. Manage.* **141** (11), 04015029.
- Aboutalebi, M., Bozorg-Haddad, O. & Loáiciga, H. A. 2016 [Multi-objective design of water-quality monitoring networks in river-reservoir systems](#). *J. Environ. Eng.* **143** (1), 04016070.
- Ahmadi-Javid, A. 2011 Anarchic society optimization: a human-inspired method. In: *IEEE Congress on Evolutionary Computation (CEC)*, New Orleans, Los Angeles, USA, June 5–8.
- Ahmadi-Javid, A. & Hooshangi-Tabrizi, P. 2012 An anarchic society optimization algorithm for a flow-shop scheduling problem with multiple transporters between successive machines. In: *International Conference on Industrial Engineering and Operations Management (ICIEOM)*, Istanbul, Turkey, July 3–6.
- Ahmadi-Javid, A. & Hooshangi-Tabrizi, P. 2015 [A mathematical formulation and anarchic society optimization algorithms for integrated scheduling of processing and transportation operations in a flow-shop environment](#). *Int. J. Prod. Res.* **53** (19), 5988–6006.
- Akbari-Alashti, H., Bozorg-Haddad, O., Fallah-Mehdipour, E. & Mariño, M. A. 2014 [Multi-reservoir real-time operation rules: a new genetic programming approach](#). *Proc. Inst. Civil Eng. Water Manage.* **167** (10), 561–576.
- Asgari, H. R., Bozorg-Haddad, O., Pazoki, M. & Loáiciga, H. A. 2015 [Weed optimization algorithm for optimal reservoir operation](#). *J. Irrig. Drain. Eng.* **142** (2), 04015055.
- Bozorg-Haddad, O., Afshar, A. & Mariño, M. A. 2006 [Honey-bees mating optimization \(HBMO\) algorithm: a new heuristic approach for water resources optimization](#). *Water Resour. Manage.* **20** (5), 661–680.
- Bozorg-Haddad, O., Afshar, A. & Mariño, M. A. 2008 [Design-operation of multi-hydropower reservoirs: HBMO approach](#). *Water Resour. Manage.* **22** (12), 1709–1722.
- Bozorg-Haddad, O., Moravej, M. & Loáiciga, H. A. 2014 [Application of the water cycle algorithm to the optimal operation of reservoir systems](#). *J. Irrig. Drain. Eng.* **141** (5), 04014064.
- Bozorg-Haddad, O., Hosseini-Moghari, S. M. & Loáiciga, H. A. 2015 [Biogeography-based optimization algorithm for optimal operation of reservoir systems](#). *J. Water Resour. Plan. Manage.* **142** (1), 04015034.
- Bozorg-Haddad, O., Ghajarnia, N., Solgi, M., Loaiciga, H. A. & Marino, M. A. 2016 [A DSS-based honeybee mating optimization \(HBMO\) algorithm for single- and](#)

- multi-objective design of water distribution networks. In: *Metaheuristics and Optimization in Civil Engineering*, Vol. 7 (X.-S. Yang, G. Bekdas & S. M. Nigdeli, eds). Springer International Publishing, Switzerland, pp. 199–233.
- Chow, V. T. & Cortes-Rivera, G. 1974 Application of DDDP in water resources planning. *WRC Research Report*, No. 78, Project No. B-060-ILL, University of Illinois at Urbana-Champaign, Water Resources Center, Hydrosystems Laboratory.
- Eberhart, R. C. & Kennedy, J. 1995 A new optimizer using particle swarm theory. In: *Proceedings of the Sixth International Symposium on Micro Machine and Human Science*, Nagoya, Japan, October 4–6.
- Eskandar, H., Sadollah, A., Bahreininejad, A. & Hamdi, M. 2012 Water cycle algorithm – a novel metaheuristic optimization method for solving constrained engineering optimization problems. *Comput. Struct.* **110**, 151–166.
- Farhangi, M., Bozorg-Haddad, O. & Mariño, M. A. 2012 Evaluation of simulation and optimization models for WRP with performance indices. *Proc. Inst. Civil Eng. Water Manage.* **165** (5), 265–276.
- Garousi-Nejad, I., Bozorg-Haddad, O., Loáiciga, H. A. & Mariño, M. A. 2016a Application of the firefly algorithm to optimal operation of reservoirs with the purpose of irrigation supply and hydropower production. *J. Irrig. Drain. Eng.* **142** (10), 04016041.
- Garousi-Nejad, I., Bozorg-Haddad, O. & Loáiciga, H. A. 2016b Modified firefly algorithm for solving multireservoir operation in continuous and discrete domains. *J. Water Resour. Plan. Manage.* **142** (9), 04016029.
- Ghimire, B. N. & Reddy, M. J. 2013 Optimal reservoir operation for hydropower production using particle swarm optimization and sustainability analysis of hydropower. *ISH J. Hydr. Eng.* **19** (3), 196–210.
- Holland, J. H. 1975 *Adaptation in Natural and Artificial Systems*, 2nd edn (1992). University of Michigan Press, Ann Arbor, Cambridge, MA, USA.
- Jothiprakash, V. & Shanthi, G. 2006 Single reservoir operating policies using genetic algorithm. *Water Resour. Manage.* **20** (6), 917–929.
- Lall, U. & Miller, C. W. 1988 An optimization model for screening multipurpose reservoir systems. *Water Resour. Res.* **24** (7), 953–968.
- Momtahn, S. & Dariane, A. B. 2007 Direct search approaches using genetic algorithms for optimization of water reservoir operating policies. *J. Water Resour. Plan. Manage.* **133** (3), 202–209.
- Murray, D. M. & Yakowitz, A. J. 1979 Constrained differential dynamic programming and its application to multireservoir control. *Water Resour. Res.* **15** (5), 1017–1027.
- Rani, D. & Moreira, M. M. 2010 Simulation–optimization modeling: a survey and potential application in reservoir systems operation. *Water Resour. Manage.* **24** (6), 1107–1138.
- Shah-Hosseini, H. S. 2007 Problem solving by intelligent water drops. In: *Proceedings of IEEE Congress on Evolutionary Computation*, Singapore, September 25–28.
- Sharif, M. & Swamy, V. S. V. 2014 Development of LINGO-based optimization model for multi-reservoir systems operation. *Int. J. Hydrol. Sci. Technol.* **4** (2), 126–138.
- Shayeghi, H. & Dadashpour, J. 2012 Anarchic society optimization based PID control of an automatic voltage regulator (AVR) system. *Electric. Electron. Eng.* **2** (4), 199–207.
- Simon, D. 2008 Biogeography-based optimization. *Evol. Comput.* **12** (6), 702–713.
- Soleimani, S., Bozorg-Haddad, O., Saadatpour, M. & Loáiciga, H. A. 2016a Optimal selective withdrawal rules using a coupled data mining model and genetic algorithm. *J. Water Resour. Plan. Manage.* **142** (12), 04016064.
- Soleimani, S., Bozorg-Haddad, O. & Loáiciga, H. A. 2016b Reservoir operation rules with uncertainties in reservoir inflow and agricultural demand derived with stochastic dynamic programming. *J. Irrig. Drain. Eng.* **142** (11), 04016046.
- Solgi, M., Bozorg-Haddad, O. & Loáiciga, H. A. 2016 The enhanced honey-bee mating optimization algorithm for water resources optimization. *Water Resour. Manage.* **3** (31), 885–901.
- Tauxe, G. W., Inman, R. R. & Mades, D. M. 1979 Multiobjective dynamic programming with application to a reservoir. *Water Resour. Res.* **15** (6), 1403–1408.
- Woodcock, G. 2004 *Anarchism*. University of Toronto Press, Higher Education Division, Toronto, Canada.
- Yang, X. S. 2008 *Firefly Algorithm, Nature-Inspired Meta-Heuristic Algorithms*. Wiley Online Library, Singapore, pp. 79–90.
- Yang, X. S. 2009 Firefly algorithm for multimodel optimization. *Stoch. Algorithms Found. Appl.* **5792** (2), 169–178.

First received 16 December 2016; accepted in revised form 16 July 2017. Available online 1 September 2017