

Article

Optimization of Water-Supply and Hydropower Reservoir Operation Using the Charged System Search Algorithm

Behzad Asadieh ^{1,*} and Abbas Afshar ²

¹ Department of Earth and Environmental Science, University of Pennsylvania, Philadelphia, PA 19104, USA

² Civil Engineering Department, Iran University of Science and Technology, Tehran 1417466191, Iran;
a_afshar@iust.ac.ir

* Correspondence: basadieh@sas.upenn.edu

Received: 17 December 2018; Accepted: 3 January 2019; Published: 8 January 2019



Abstract: The Charged System Search (CSS) metaheuristic algorithm is introduced to the field of water resources management and applied to derive water-supply and hydro-power operating policies for a large-scale real-world reservoir system. The optimum algorithm parameters for each reservoir operation problems are also obtained via a tuning procedure. The CSS algorithm is a metaheuristic optimization method inspired by the governing laws of electrostatics in physics and motion from the Newtonian mechanics. In this study, the CSS algorithm's performance has been tested with benchmark problems, consisting of highly non-linear constrained and/or unconstrained real-valued mathematical models, such as the Ackley's function and Fletcher–Powell function. The CSS algorithm is then used to optimally solve the water-supply and hydropower operation of “Dez” reservoir in southern Iran over three different operation periods of 60, 240, and 480 months, and the results are presented and compared with those obtained by other available optimization approaches including Genetic Algorithm (GA), Ant Colony Optimization (ACO), Particle Swarm Optimization (PSO) and Constrained Big Bang–Big Crunch (CBB–BC) algorithm, as well as those obtained by gradient-based Non-Linear Programming (NLP) approach. The results demonstrate the robustness and superiority of the CSS algorithm in solving long term reservoir operation problems, compared to alternative methods. The CSS algorithm is used for the first time in the field of water resources management, and proves to be a robust, accurate, and fast convergent method in handling complex problems in this field. The application of this approach in other water management problems such as multi-reservoir operation and conjunctive surface/ground water resources management remains to be studied.

Keywords: charged system search (css); water resources management; reservoir operation; water-supply; hydropower; metaheuristic optimization

1. Introduction

Optimal utilization of available fresh-water resources requires development and application of robust and effective methods to plan and operate current reservoirs [1,2]. Effective reservoir operation requires policies that optimize releases from the reservoir or storage volume, in order to achieve desired objectives such as maximizing power generation or minimizing water deficit, flood risk, and operation costs. Adoption of these policies are a must for the operators to make decisions based on the current and past conditions of the reservoir storage and river inflow, in order to manage the upcoming flood risks and water shortages [1–3].

Effective management of current and planned reservoirs will become even more challenging as the Earth's climate has changed in the past and is projected to change even more in the future decades,

due to anthropogenic global warming and climate change [4–10]. Climate change will change the distribution, intensity, and frequency of precipitation events, and consequently change river flows and water resources reliability in global and regional scales [11–18]. Regional studies show future changes in reliability of reservoirs due to the changes in climate, projected by climate models [19–22]. The complexity of the water resources management problems, climate uncertainty, and the large number of decision variables requires development and utilization of novel optimization methods with high accuracy and convergence speed for optimization of such hydro-systems.

Reservoir operation problems are considered as Dynamic Optimization problems, since the water release at each time step is affected by releases in the previous time steps. Dynamic Optimization problems are more complex and non-linear, and require higher computational cost compared to Static Optimization problems. Earlier studies utilized Linear Programming (LP), Dynamic Programming (DP) and Nonlinear Programming (NLP) for the optimal solution of reservoir operation problems [23,24]. The linear programming may not be a suitable approach for the reservoir operation, because of the complexity and nonlinearity in the objective function of these problems. DP has long been used for solving reservoir operation problems, however, dimensionality issues limit the application of this method in exploring an optimal operation for complex reservoir systems and multi-reservoir systems. The Nonlinear Programming has a slow rate of convergence, resulting in high computational time and cost, as well as the tendency of convergence to local optima [23]. The conventional mathematical optimization methods have had challenging problems in solution of the water resources management problems, which usually involve non-linear and non-convex objective functions and constraints, and multiple decision variables.

Developments in fast computing machines the recent decades have led to the development of novel algorithms that exploit a large number of iterative processes to model an intelligent search procedure that leads to finding optimal solutions. Metaheuristic algorithms are among those methods that combine rules and randomness, to model natural phenomena. Metaheuristics have been developed in the recent decades as probing implements in a variety of fields. Robustness, ease of use and broad applicability of these methods have led to their increased application in water resources management problems [25–33]. They also facilitate handling the nonlinear and non-convex relationships of the formulated model, which have reportedly overcome most limitations of conventional numerical methods [34–37]. Genetic Algorithms (GAs) [38], Ant Colony Optimization (ACO) [39], and particle swarm optimization (PSO) [40] are among the most frequently used techniques in this category.

The Charged System Search (CSS) Algorithm, proposed by Kaveh and Talatahari in 2010 [41], is a novel metaheuristic optimization method inspired by the governing laws of electrostatics in physics and motion from the Newtonian mechanics. The CSS algorithm has successfully been applied to various structural engineering problems [42,43] and water distribution network design [44]. Comparison of the results with those of the few other metaheuristic algorithms demonstrates its efficiency and robustness in finding the optimum solutions in such problems [42,43]. Although its performance in few structural problems has been demonstrated, its efficiency in other civil engineering problems such as water resources management remains to be tested. The CSS algorithm is already tested with some mathematical test functions in the original reference [41]. However, water resources management problems are more complex than structural problems, and hence, evaluation of optimization methods in this field requires higher level non-linearity and non-convexity in the test functions.

In the present study, the CSS optimization algorithm is introduced to the field of water resources management and is applied to the optimization of a real-world large-scale reservoir operation problem. For pre-evaluation of its capability of coping complex and nonlinear problems in the field of water resources management, its performance is tested using four well-known and highly nonlinear benchmark mathematical functions, as suggested by earlier studies. The CSS algorithm is then applied to the optimal water-supply and hydropower operation of “Dez” reservoir in southern Iran for three different monthly operation periods of 60, 240, and 480 months, and the results are presented and compared with those obtained by other optimization approaches.

2. Materials and Methods

2.1. Charged System Search (CSS) Algorithm

The CSS algorithm is developed based on the governing laws of electrostatics in physics and motion from the Newtonian mechanics [41]. The CSS utilizes a number of agents or solution candidates, which are called charged particles (CPs). Each CP is considered to be a charged sphere which can impose electrical forces on the other CPs according to the Coulomb and Gauss laws of electrostatics. Then, the Newton's law is utilized to calculate the acceleration value based on the resultant force acting on each CP. Finally, utilizing the Newtonian mechanics, the position of each CP is determined at any time step based on its previous position, velocity and acceleration in the search space [41]. Each CP is considered as a charged sphere with radius a , which has a uniform volume charge density (q_i) equal to:

$$q_i = \frac{fit(i) - fit_{worst}}{fit_{best} - fit_{worst}}, i = 1, 2, \dots, N \quad (1)$$

where fit_{best} and fit_{worst} are the best and the worst fitness values of all the particles, and $fit(i)$ is the fitness of the particle i , and N is the total number of CPs. The initial positions of CPs are assigned randomly in the search space, within the boundaries determined by the problem. The initial velocities of the CPs are taken as zero.

The CPs are scattered in the search space and can impose electric forces on the others. The magnitude of the force for the CP located inside or outside of the sphere are determined differently. The resultant electrical force acting on CPs inside or outside of the sphere is determined using:

$$F_j = q_j \sum_{i, i \neq j} \left(\frac{q_i}{a^3} r_{ij} i_1 + \frac{q_i}{r_{ij}^2} i_2 \right) p_{ij} (X_i - X_j) \begin{cases} j = 1, 2, \dots, N \\ i_1 = 1, i_2 = 0 \Leftrightarrow r_{ij} < a \\ i_1 = 0, i_2 = 1 \Leftrightarrow r_{ij} \geq a \end{cases} \quad (2)$$

where F_j is the resultant force acting on the j th CP. r_{ij} is the separation distance between two particles defined as:

$$r_{ij} = \frac{\|X_i - X_j\|}{\|(X_i + X_j)/2 - X_{best}\| + \varepsilon} \quad (3)$$

where X_i and X_j are the positions of the i th and j th CPs, respectively; X_{best} is the position of the best current CP. ε here is a small positive number to avoid singularity. The p_{ij} determines the probability of moving each CP toward the others as:

$$p_{ij} = \begin{cases} 1 & \frac{fit(i) - fit_{best}}{fit(j) - fit(i)} > rand \text{ or } fit(j) > fit(i) \\ 0 & \text{otherwise} \end{cases} \quad (4)$$

As shown in Equation (2), the force imposed on a CP inside the sphere is proportional to the separation distance between particles. However, for the CPs located outside the sphere, it is inversely proportional to the square of the separation distance. The new locations of the CPs are calculated based on the resultant forces and the laws of the motion. At this step, each CP moves towards its new position according to the resultant forces and its previous velocity as:

$$X_{j,new} = rand_{j1} \cdot k_a \cdot \frac{F_j}{m_j} \cdot \Delta t^2 + rand_{j2} \cdot k_v \cdot V_{j,old} \cdot \Delta t + X_{j,old} \quad (5)$$

$$V_{j,new} = \frac{X_{j,new} - X_{j,old}}{\Delta t} \quad (6)$$

where $rand_{j1}$ and $rand_{j2}$ are two random numbers uniformly distributed in the range (0,1). Here, m_j is the mass of the j th CP, which is set to be equal to q_j . Δt is the time step and is set to unity. k_a is the

acceleration coefficient; k_v is the velocity coefficient that controls the influence of the previous velocity, which may either be kept constant or let it vary in the next time steps:

$$k_a = \alpha \times (1 + \text{iter}/\text{iter}_{\max}), k_v = \beta \times (1 - \text{iter}/\text{iter}_{\max}) \quad (7)$$

where iter is the current iteration number and iter_{\max} is the maximum number of iterations set for the algorithm run. According to this equation, k_v decreases linearly to zero while k_a increases to 2α as the number of iterations increases, which preserves the balance between the exploration and the speed of convergence [41]. It is noteworthy that the parameters α and β in Equation (7) are tunable and defining these parameters result in definition of the acceleration and velocity coefficients (k_a and k_v). Value of 0.5 for both parameters α and β has been recommended in the reference paper of the CSS algorithm [41]. Substituting for k_a and k_v from Equation (7), Equations (5) and (6) can be rewritten as:

$$X_{j,\text{new}} = \alpha \times \text{rand}_{j1} \cdot (1 + \text{iter}/\text{iter}_{\max}) \cdot \sum_{i,i \neq j} \left(\frac{q_i}{a^3} r_{ij} i_1 + \frac{q_i}{r_{ij}^2} i_2 \right) p_{ij} (X_i - X_j) + \beta \times \text{rand}_{j2} \cdot (1 - \text{iter}/\text{iter}_{\max}) \cdot V_{j,\text{old}} + X_{j,\text{old}} \quad (8)$$

$$V_{j,\text{new}} = X_{j,\text{new}} - X_{j,\text{old}} \quad (9)$$

In addition, to save the best results, a memory, known as the Charged Memory (CM), is recommended [41]. If each CP moves out of the search space, its position is corrected using the harmony search-based handling approach, in which a new value is produced or selected from the CM, on a probabilistic basis. It is highly recommended to refer to the main reference [41] for better understanding the concepts and structure of the CSS algorithm, as some concepts in the present paper might not be described as detailed.

In the original CSS algorithm, when the calculations of the amount of forces are completed for all CPs, the new locations of the CPs are determined and also CM updating is fulfilled. In other words, the new location for each CP is determined after completion of an iteration and before commencement of the new iteration. Kaveh and Talatahari [45], ignoring this assumption, proposed the Enhanced CSS algorithm in which after evaluation of each CP, all updating processes are performed. Using this method of updating in the CSS algorithm, the new position of each agent can affect the moving process of the subsequent CPs while in the standard CSS, unless an iteration is completed, the new positions are not utilized. This enhanced algorithm, compared to the original CSS, while not requiring additional computational time, improves the performance of the algorithm by using the information obtained by CPs instantly. In a detailed investigation, considering the i th CP in the original CSS, although the solutions obtained by the CPs with a number less than i are created before the selected agent is used, however, these new designs cannot be employed to direct the i th CP in the current iteration. On the other hand, the original CSS archives the information obtained by the agents until a pre-determined time and this results in a break in the optimization process, while in the enhanced CSS algorithm the information of the new position of each agent is utilized in the subsequent search process, and this procedure improves the optimization abilities of the algorithm and also increases the convergence speed [45].

2.2. Water-Supply and Hydropower Reservoir Operation

In a water-supply reservoir operation, the objective is to obtain a set of releases from the reservoir (or a set of reservoir storage volumes) for the operation period with given inflow such that a predefined pattern of demands is met. In the other words, the objective is to set the released flow as close as possible to the demand and decrease the unnecessary overflows from the reservoirs, and hence, minimize the water deficit. Therefore, optimal operation of a water supply reservoir can be stated mathematically as [34,37]:

$$\text{Minimize } F = \sum_{t=1}^{NT} \left(\frac{D(t) - R(t)}{D_{\max}} \right)^2 \quad (10)$$

subject to continuity equations at each time step:

$$S(t+1) = S(t) + I(t) - R(t) - \text{Loss}(t) \quad (11)$$

$$S_{\min} \leq S(t) \leq S_{\max} \quad (12)$$

$$R_{\min} \leq R(t) \leq R_{\max} \quad (13)$$

$$\text{Loss}(t) = \text{Ev}(t) \times A(t) / 1000 \quad (14)$$

$$A(t) = x_0 + x_1 \times S(t) + x_2 \times S(t)^2 + x_3 \times S(t)^3 \quad (15)$$

where NT is the number of time steps, $D(t)$ is water demand in time step t in million cubic meters (MCM), $R(t)$ is release from the reservoir in time step t (MCM), D_{\max} is maximum demand (MCM), $S(t)$ is storage at the start of time step t (MCM), $I(t)$ is inflow in time step t (MCM), S_{\min} and S_{\max} are minimum and maximum storage of reservoir (MCM), respectively and R_{\min} and R_{\max} are minimum and maximum allowed release from reservoir (MCM), respectively. $\text{Loss}(t)$ is net amount of gain and loss of the reservoir resulting from precipitation and evaporation in time step t . $\text{Ev}(t)$ is the evaporation height during the time step t , and x_0, x_1, x_2 and x_3 are constants that can be obtained by fitting Equation (15) to the existing data. In cases where the evaporation loss is not considered in the formulations, $\text{Loss}(t)$ is excluded from the Equation (11).

In a hydropower reservoir operation, the objective is to obtain a set of releases from the reservoir (or a set of reservoir storage volumes) such that the power generation from the reservoir is maximum, or as close as possible to the installed capacity of the hydro-electric plant. Hydropower operation of a single reservoir may be defined as [34,46]:

$$\text{Minimize } F = \sum_{t=1}^{NT} \left(1 - \frac{p(t)}{\text{power}} \right) \quad (16)$$

subject to the continuity constraints defined by equations 11 to 15 defined for the simple operation problem. Here $p(t)$ is power generated in megawatts (MW) in time step t , power is the installed capacity of hydro-electric plant (MW), and other parameters are defined as before. The power generated in time step t can be stated as follow:

$$p(t) = \min \left[\left(\frac{g \times \eta \times r(t)}{PF} \right) \times \left(\frac{h(t)}{1000} \right), \text{power} \right] \quad (17)$$

in which $h(t)$ is the effective head of the hydroelectric plant as defined by Equation (18):

$$h(t) = \left(\frac{H(t) + H(t+1)}{2} \right) - \text{TWL} \quad (18)$$

$H(t)$ is the elevation of water in reservoir at time step t which may be defined as a function of storage in the reservoir as:

$$H(t) = a + b \times S(t) + c \times S(t)^2 + d \times S(t)^3 \quad (19)$$

where g is the Earth's gravity acceleration, η is the efficiency of the hydroelectric plant, $r(t)$ is release from reservoir (m^3/s), PF is the plant factor, TWL (tail water level) is the downstream elevation of the hydroelectric plant (m), a, b, c and d are constants that can be obtained by fitting Equation (19) to the reservoir's data.

3. Results and Discussion

3.1. Model Performance Evaluation: Mathematical Test Functions

The CSS algorithm has previously been tested with some mathematical tests and structural optimization problems in the original reference [41]. However, water resources management problems are more complex than structural problems and require a higher level of robustness in utilized approaches. In order to assess the performance of the proposed algorithm in complex and nonlinear problems in the field of water resources management, it is first applied to benchmark constrained and unconstrained mathematical optimization functions. For evaluation of robustness of the standard CSS algorithm and also assess the impact of utilizing the new method of updating on performance of the algorithm (called Enhanced CSS or ECSS in this section) [45], the functions are optimized with both standard CSS and ECSS.

Although value of 0.5 for both tunable coefficients in Equation (7) (i.e., α and β) have been recommended in the reference paper of the CSS algorithm [41], different values may result in better solutions. A sensitivity analysis for mathematical test functions shows that the value of 0.8 for these coefficients results in better solutions and increases the convergence rate of the algorithm. The best, worst, mean and standard deviation of the results are reported for 10 different runs of the algorithm for each of the mathematical test functions.

The first optimization problem is the Ackley's function [47], a continuous and multi-modal function defined by modulating an exponential function with a cosine wave of moderate amplitude. Ackley's function is defined as:

$$\text{Min}f(x) = c_1 + \exp(1) - c_1 \exp\left(-c_2 \sqrt{\frac{1}{n} \sum_{i=1}^n x_i^2}\right) - \exp\left(\frac{1}{n} \sum_{i=1}^n \cos(c_3 x_i)\right) \quad (20)$$

$$-5 \leq x_i \leq 5 (i = 1, 2, \dots, n) \quad (21)$$

where $c_1 = 20$, $c_2 = 0.2$, $c_3 = 2\pi$ and n is taken equal to 2 here. The functions surface is a nearly flat outer region with moderate fluctuations converging to a hole in the middle. Multiple hills and valleys on the surface cause moderate complexity for optimization methods, as the search algorithms performing based on hill-climbing techniques are most likely to be trapped in local optima (Figure 1). An algorithm with a large scanning span that searches a wider neighborhood would be able to avoid the valleys and located better optima. Therefore, Ackley's function provides one of the reasonable test cases for the CSS algorithm. The results obtained by the CSS and ECSS algorithms as well as those obtained by Genetic Algorithm (GA) [48], Honey-Bees Mating Optimization algorithm (HBMO) [49] and Particle Swarm Optimization algorithm (PSO) [50] are presented in Table 1. The results for CSS and ECSS shown in Table 1 are best out of ten runs of the algorithms. The ECSS algorithm with 10 CPs, reaches the fitness value of 2×10^6 after 440 function evaluations (44 iterations with 10 CPs) and the best fitness value of 0 (global optimum) after 930 function evaluations in the best run of the algorithm. Results show that all ten executions of the algorithms reach quite close to the global optimum value of the objective function, where the standard deviation of the objective function value over ten runs is 1.6×10^{-15} for the CSS and 1.4×10^{-15} for the ECSS. Table 1 demonstrates the impact of using the new method of updating in the structure of the algorithm on the convergence rate of the algorithm, as ECSS shows higher convergence speed compared to the standard CSS. The table denotes that the CSS algorithm obtains more accurate values in smaller number function evaluations in comparison with the other metaheuristic rivals, in terms of accuracy and convergence speed.

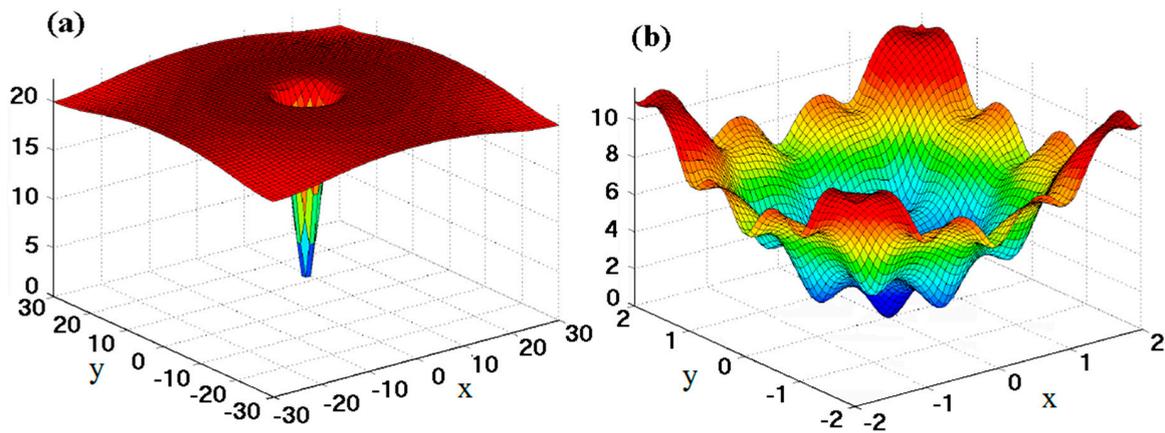


Figure 1. Surface of the Ackley’s function for (a) $-30 \leq x_1, x_2 \leq 30$ and (b) $-2 \leq x_1, x_2 \leq 2$.

Table 1. Results of the standard Charged System Search (CSS) and Enhanced Charged System Search (ECSS) algorithm, in comparison with the other rival metaheuristics algorithms for Ackley’s function.

-	CSS	ECSS	PSO	HBMO	GA	
Value of objective function	8×10^{-16}	8×10^{-16}	0.000002	0.000002	-0.0054618	-0.005456
No. of function evaluations	1040	930	440	500	174,000	-

The second numerical example is an unconstrained sine function defined as [48]:

$$Maxf(x_1, x_2) = 21.5 + x_1 \cdot \sin(4\pi x_1) + x_2 \cdot \sin(4\pi x_2) \tag{22}$$

$$- 3.0 \leq x_1 \leq 12.1 \tag{23}$$

$$4.1 \leq x_2 \leq 5.8 \tag{24}$$

As seen in Figure 2a, the search space for this function is a highly non-linear and multi-modal surface. The ECSS algorithm with 30 CPs reached the best fitness value of 38.85029 after 1590 function evaluations (53 iterations with 30 CPs). The results obtained by the CSS and ECSS algorithms as well as those obtained by GA [48], HBMO [49] and PSO [50] are presented in Table 2. The results for CSS and ECSS shown in Table 2 are best out of ten runs of the algorithms. Results show that the standard deviation of objective function value is approximately zero, indicating that all 10 runs have converged to approximately one single solution. As seen from the Table 2, the CSS algorithm performs considerably faster comparing to the other rival approaches, where the CSS and ECSS algorithms locate the optima with 1950 and 1590 times function evaluations, respectively.

To assess the performance of the CSS algorithm in handling constrained problems, a two-variable, two-constraint constrained exponential function is considered [49], defined as (Figure 2b):

$$Minf(x_1, x_2) = (x_1^2 + x_2 - 11)^2 + (x_1 + x_2^2 - 7)^2 \tag{25}$$

Subject to:

$$g_1(x) = 5.059 - x_1^2 - (x_2 - 2.5)^2 \geq 0 \tag{26}$$

$$g_2(x) = (x_1 - 0.05)^2 + (x_2 - 2.5)^2 - 4.84 \geq 0 \tag{27}$$

$$0 \leq x_1 \leq 6, 0 \leq x_2 \leq 6 \tag{28}$$

The unconstrained objective function $f(x_1, x_2)$ has a minimum solution at (3, 2) with a function value equal to zero. However, due to multiple constraints imposed to the function, this solution is not feasible and the constrained optimal solution is $x = (2.2461, 2.3815)$ with a function value equal to

$f = 13.61227$. The feasible region is only approximately 0.7% of the total search space, which is a narrow crescent-shaped region, and the optimum solution is lying on the second constraint.

Employing the ECSS algorithm with 20 CPs, the best obtained fitness value is 13.59087 at $x = (2.23809, 2.24677)$ after 600 function evaluations (30 iterations with 20 CPs). Results obtained by the CSS and ECSS algorithms as well as those obtained by GA [48], HBMO [49] and PSO [50] are presented in Table 3. The results for CSS and ECSS shown in Table 3 are the best out of ten runs of the algorithms. All ten runs show a very small discrepancy with the global result as indicated by a very small value of the standard deviation. However, the ECSS reaches the global optima with less function evaluations. As seen in Table 3, the ECSS algorithm finds the optima with considerably less function evaluations comparing the other metaheuristic approaches.

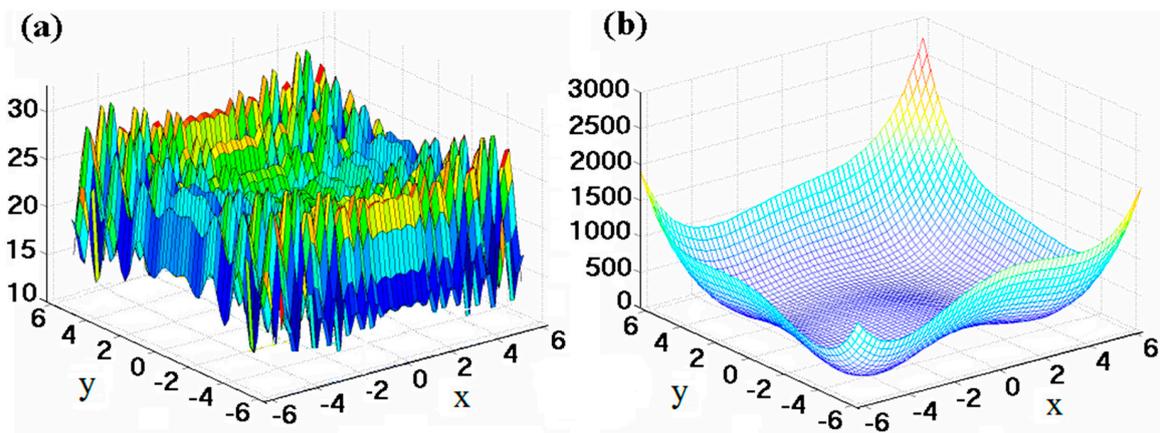


Figure 2. Surface of the unconstrained sine function (a) and the constrained exponential function (b).

Table 2. Results of the standard CSS and ECSS algorithms in comparison with the other heuristics for the unconstrained sine function.

	Standard CSS	ECSS	PSO	HBMO	GA
Value of objective function	38.8502945	38.8502945	38.8502939	38.8502945	38.850295
No. of function evaluations	1950	1590	15,000	174,000	-

Table 3. Results of the CSS and ECSS algorithms in comparison with the other heuristics for the constrained exponential function.

	Standard CSS	ECSS	PSO	HBMO
Value of objective function	13.59088	13.59087	13.60821	13.62305
No. of function evaluations	670	600	2500	14,300,000

The last test function used for investigation of performance of the CSS algorithm facing highly non-linear multi-variable problems is the well-known multimodal and continuous Fletcher–Powell function [47], which is a non-separable, non-linear, and irregular function, described as:

$$\begin{aligned}
 f(x) &= \sum_{i=1}^n (A_i - B_i)^2 \\
 A_i &= \sum_{j=1}^n (a_{ij} \sin \alpha_j + b_{ij} \cos \alpha_j) \\
 B_i &= \sum_{j=1}^n (a_{ij} \sin x_j + b_{ij} \cos x_j) \\
 -\pi &\leq x_j \leq \pi (i = 1, 2, \dots, n)
 \end{aligned}
 \tag{29}$$

where x_j is the decision variable, α_j is a random coefficient within the range of $-\pi$ and π , a_{ij} and b_{ij} are random coefficients within the range of -100 and 100 , and n is the dimension of the function (Figure 3). The optimum point of the function is at $x_j = \alpha_j$ where the objection function value will be equal to zero. A 30-variable Fletcher–Powell function is chosen here to be optimized utilizing the CSS algorithm. The ECSS algorithm with 20 CPs, reaches the fitness value of 1246.37 after 2E5 function evaluations and the best fitness value of 440.29 after 2E6 function evaluations in the best run of the algorithm. The results obtained by ECSS algorithm as well as those obtained by the GA [48], HBMO [49], PSO [50] and Nonlinear Programing (NLP) using LINGO 8.0 software [49] are presented in Table 4. The ECSS algorithm obtains more optimum values in smaller number function evaluations in comparison with the other metaheuristic rivals. However, due to complexity of the Fletcher–Powell function, the algorithm does not converge to optimal values at every attempt, as shown by a fairly large standard deviation value of 8805.65, for 10 runs of the algorithm. Multiple runs of the algorithm are needed to obtain the best optima achievable by the approach, as also seen in other algorithms [49].

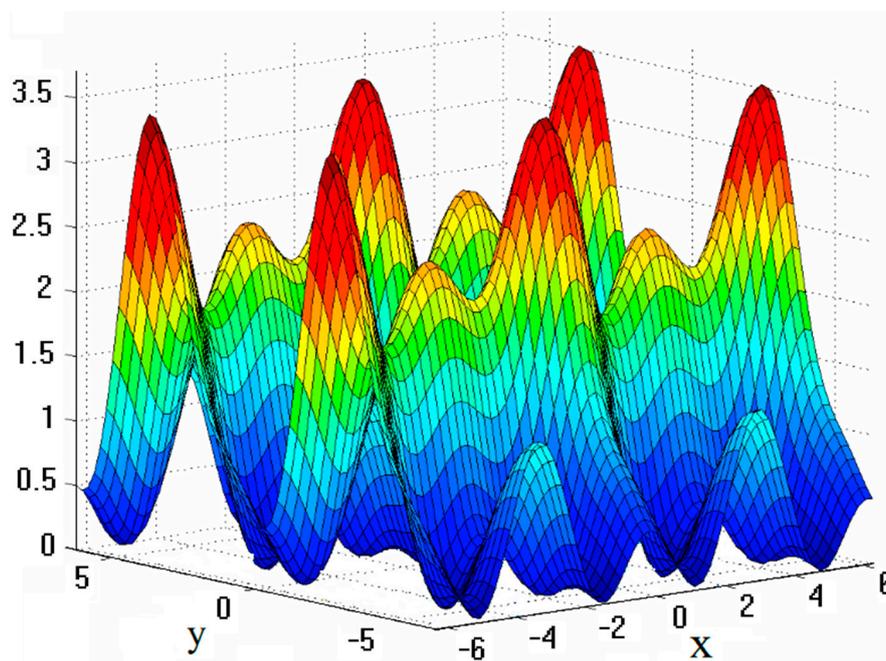


Figure 3. Surface of the 2-variable Fletcher–Powell function.

Table 4. Results of the ECSS algorithm in comparison with the other rival metaheuristics algorithms for 30-variable Fletcher–Powell function.

-	ECSS	PSO	HBMO	GA	NLP (LINGO 8.0)
Value of objective function	440.29	1246.37	50,460.83	1934	9453
No. of function evaluations	2×10^6	2×10^5	2×10^5	4.4×10^6	-

Results demonstrate that the standard CSS algorithm without any improvements is a robust and fast convergent approach which outperformed its other metaheuristic rivals in optimization of complex and multimodal mathematical functions and seems to be capable of handling the highly nonlinear and non-convex problem of large-scale reservoir operation. Results also indicate that utilizing the new method of updating in the structure of algorithm, referred here as enhanced CSS or ECSS, improves its convergence speed. The remainder of this paper presents application of the new approach in water supply and hydropower reservoir operation problems. In all reservoir operation problems presented in the next sections, the enhanced CSS algorithm is utilized and will simply be called the CSS algorithm.

3.2. Reservoir Operation Case Study

To evaluate the performance of the CSS algorithm in solution of large-scale water resources management problems, water-supply and hydropower operation of “Dez” reservoir in southern Iran has been considered. Total storage capacity of “Dez” reservoir in pre-defined normal water level is 2510 MCM and the average inflow of the reservoir over 40 years period from 1970–2010 is 5900 MCM. The initial storage of the reservoir is taken equal to 1430 MCM. The maximum and minimum allowable storage volumes are set equal to 3340 and 830 MCM, respectively. The maximum and minimum monthly water release set equal to 1000 MCM and zero, respectively. The coefficients of the volume-elevation curve defined by Equation (19) are used as: $a = 249.83364$, $b = 0.58720$, $c = -1.37 \times 10^{-5}$ and $d = 1.526 \times 10^{-9}$. The total installed capacity of hydroelectric power plant of the Dez reservoir is 650 MW, being operated with plant factor of 0.417 and 90% efficiency. The tail water level in downstream is assumed constant at 172 m above sea level.

These problems are solved here using the CSS algorithm for optimal monthly operation over 5, 20 and 40-year time spans, which are 60, 240 and 480 monthly periods, respectively. The water-supply operation and hydropower operation problems are solved using the CSS algorithm, separately, and results are shown for each operation type. The parameters of acceleration (ka) and velocity (kv) coefficients in Equation (7) are taken as 0.3, 0.3 and 0.5 for the 60, 240 and 480-month operational period problems, respectively, obtained via a tuning procedure. Tests show that for best results in reservoir operation optimization utilizing the CSS algorithm, the charged sphere radius (a) in the Equation (2) should be set close to zero (1E9). All the results presented here are obtained using 40, 100 and 1000 CPs for the 60, 240 and 480-month problems, respectively. The number of objective function evaluations is limited to 400,000 for all executions of the algorithm.

The water-supply and hydropower operation of “Dez” reservoir is first solved disregarding the effect of evaporation from the reservoir utilizing the CSS. The results of 5 executions of the algorithm are presented in Table 5. Disregarding evaporation, the NLP solver (LINGO 9.0) produces the objective function values of 20.6 and 45.4 for the hydropower operation over 240, and 480 periods [37]. Disregarding evaporation, the value of 45.8 for 480 months hydropower operation of “Dez” reservoir, using the LINGO 9.0, was also reported by earlier studies [34]. The CSS approach results in optimal solution of 21.4936 and 50.3394 for 240 and 480 period hydropower problems, respectively.

Table 5. Results of “Dez” reservoir operation using CSS algorithm over 5—Disregarding evaporation loss.

Operation Method	Operation Period (Months)	Best Solution	Worst Solution	Average of Solutions	Standard Deviation
Water-Supply	60	0.654	0.667	0.659	0.003
	240	2.417	2.587	2.514	0.0391
	480	9.204	10.206	9.662	0.2265
Hydropower	60	7.377	7.989	7.595	0.089
	240	21.494	22.087	21.783	0.1327
	480	50.339	50.967	50.631	0.1415

The water-supply and hydropower operation of “Dez” reservoir is then solved considering the evaporation losses from the surface on the storage volume of the reservoir (Table 6). The inclusion of evaporation further increases the non-linearity of the model, in particular for the hydropower operation model for which the well-known LINGO NLP solver has failed to find a feasible solution for 240 and 480 operational periods, as previously reported [46]. The LINGO NLP solver (LINGO 9.0) has yielded optimum solutions of 0.732, 4.77, and 10.50 for the water-supply problem for 60, 240, and 480 months [46]. The LINGO NLP solver was able to locate a near-optimal solution of 7.37 for the hydropower operation over the shortest operation period of 60 months, but failed to yield any feasible solution for the longer operation periods of 240 and 480 months, which may be due to the non-convexity of the hydropower operation, which is even higher for the 240 and 480-month problem [46]. Failure in

finding a feasible solution for 480-period hydropower operation of “Dez” reservoir considering the evaporation from the reservoir, using the NLP method, was reported by other studies as well [34].

Gradient-based nonlinear programming (NLP) methods solve problems that do not involve high level of nonlinearity in objective function and constraints. However, in models with large number of decision variables and/or highly nonlinearity, these approaches tend to fail in locating feasible solutions, or converge to local optima [51]. In long-period reservoir operations, or in models with evaporation losses consideration, the non-linearity and non-convexity of the model rises and gradient-based NLP solvers may not be a suitable choice since they may either produce local suboptimal solutions and/or may even fail to locate any feasible solutions.

It can be seen from Table 6 that in all short and long operation period of 60, 240 and 480 months, the CSS algorithm can yield the near-optimal solutions in both water-supply and hydropower operation models. In longer operation periods of 240 and 480 months for the water-supply operation model, the CSS algorithm results in objective function values of 2.7302 and 9.4263 respectively. Comparing the results of the CSS algorithm with those obtained by LINGO 9.0 NLP solver demonstrates that the CSS approach outperforms the NLP method with the solutions of 4.77 and 10.5 for 240 and 480-month operation periods, respectively. In 240 and 480-month hydropower operation, the CSS results in near optimal solutions with objective function values of 22.9670 and 54.3472, respectively. The NLP method fails in producing any feasible solution for these problems, as noted earlier. These results indicate robustness of the CSS algorithm to solve both convex and non-convex, small-scale and large-scale, reservoir-operation problems.

Table 6. Results of “Dez” reservoir operation using CSS algorithm over 10 runs—Considering evaporation loss.

Operation Method	Operation Period (Months)	Best Solution	Worst Solution	Average of Solutions	Standard Deviation
Water-Supply	60	0.734	0.740	0.738	0.002
	240	2.730	2.911	2.818	0.058
	480	9.426	10.848	10.209	0.336
Hydropower	60	7.353	7.664	7.545	0.155
	240	22.967	23.871	23.455	0.272
	480	54.347	55.221	54.752	0.210

Table 7 presents the results obtained by genetic algorithm (GA), Particle Swarm Optimization (PSO), Ant Colony Optimization (ACO) and Big Bang-Big Crunch (BB-BC) algorithm for the current Dez reservoir operation problem [37,46,52]. Comparing the results of Tables 6 and 7 shows that in all operation problems, these algorithm yield solutions inferior to those obtained by the CSS approach, with the difference being more significant for the longer operation periods. For the 480-month operations, the solutions yielded by the GA and PSO algorithms are significantly far from the optimal solution.

All 10 runs of the CSS algorithm for simple and hydropower operations have resulted in feasible solutions; while some of the other metaheuristic approaches failed in locating feasible solutions for all executions. These results can be compared with those obtained by the conventional Ant Colony Optimization Algorithm (ACO) [52]. The results show that ACO was capable of producing 10 feasible solutions for the simplest case of water-supply operation over 60 monthly periods and 8 feasible solutions for the hydropower operation over 60 monthly periods. In longer operation periods, i.e., 240 and 480 monthly periods, the number of runs with a feasible solution decrease. For 240 monthly periods, only 8 and 7 feasible solutions were created for water-supply and hydropower operations, respectively while for 480 monthly periods, ACO was only capable of producing one feasible solution for both water-supply and hydropower operation. Table 7 shows that while GA was unable to find a feasible solution for the longest operation period, the PSO algorithm could produce feasible solution only for the shortest operation period [37,52].

Table 7. Results obtained by alternative approaches for the optimal operation of “Dez” reservoir, disregarding evaporation losses [37,46,52].

Method	Operation Purpose	Operation Period (months)	Best Solution	Worst Solution	Average of Solutions
CBB-BC	Water-Supply	60	0.734	0.736	0.735
		240	4.882	4.927	4.902
		480	10.971	11.035	10.999
	Hydropower	60	7.473	7.599	7.547
		240	25.086	27.555	25.928
		480	62.136	64.469	63.359
ACO	Water-Supply	60	0.785	0.814	0.800
		240	10.3	13.3	0.120
		480	65.6	194	113
	Hydropower	60	7.91	8.06	8.00
		240	35.3	40.0	37.6
		480	105	275	178
GA	Water-Supply	60	0.775	0.936	0.870
		240	41.7	249	112
		480	7410	20,900	13,400
	Hydropower	60	8.08	9.10	8.48
		240	55.1	617	159
		480	27,300	61,700	40,000
PSO	Water-Supply	60	1.07	3.85	2.06
		240	126	1500	594
		480	8470	23,600	14,500
	Hydropower	60	9.26	14.3	11.3
		240	221	4320	1600
		480	25,100	70,400	41,800
NLP (Lingo 9.0)	Water-Supply	60	0.732		
		240	4.77		
		480	10.5		
	Hydropower	60	7.37		
		240	20.6		
		480	45.4		

Results from improved metaheuristics such as a version of Big Bang-Big Crunch (BB-BC) algorithm improved for reservoir operation problems called Constrained Big Bang-Big Crunch (CBB-BC) algorithm [37] as well as Fully Constrained Ant Colony Optimization Algorithm (FCACOA) [52] also show the ability to locate feasible solutions in all 10 runs of the algorithms, with inferior results compared to the CSS algorithm. It should be noted that the number of objective function evaluations was limited to 400,000 for the CBB-BC algorithm as well [37]. Figure 4a shows the best solution obtained by the CSS algorithm (monthly releases) for 60-month water-supply operation, considering evaporation losses, versus the monthly water demand defined by the problem. Figure 4b illustrates the storage volume of the reservoir at each time step, calculated based on the releases ruled by the CSS algorithm and given river inflow. The Figure 4 shows the storage volume at each time step is confined between maximum (S_{max}) and minimum (S_{min}) allowable storage, defined by the problem.

Figure 5 presents variation of the objective function value versus the number of function evaluations for the best solution obtained by the CSS algorithm for “Dez” reservoir water-supply and hydropower operation, over each operational period.

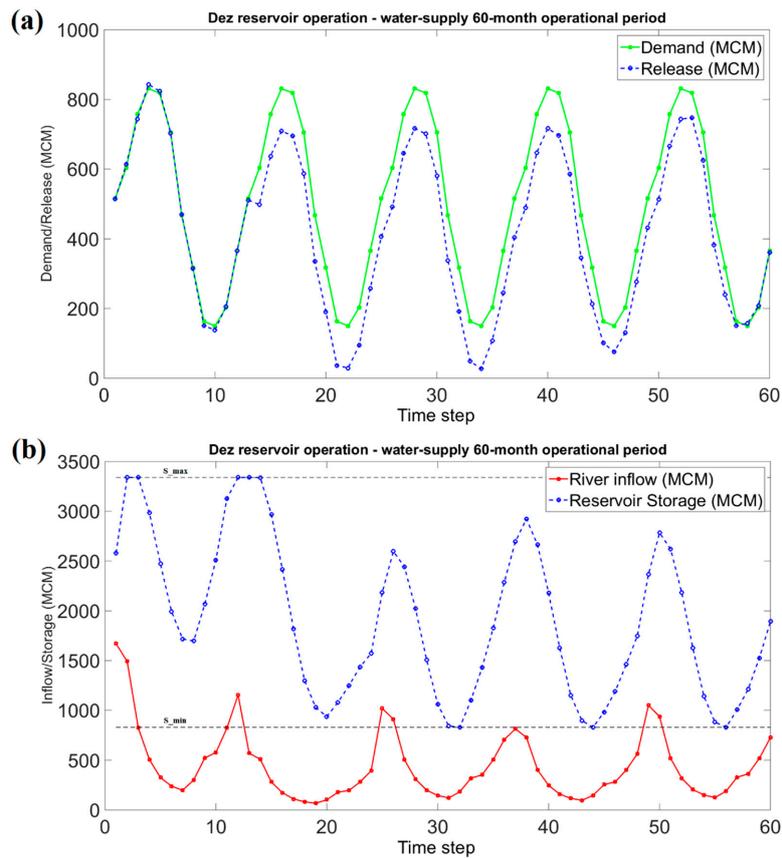


Figure 4. Best solution yielded by the CSS algorithm for 60-month water-supply operation considering evaporation losses: water demand versus release from the reservoir (a), and river inflow versus storage volume at each time step (b). As seen from the figure, the storage is confined between maximum (S_max) and minimum (S_min) allowable storage.

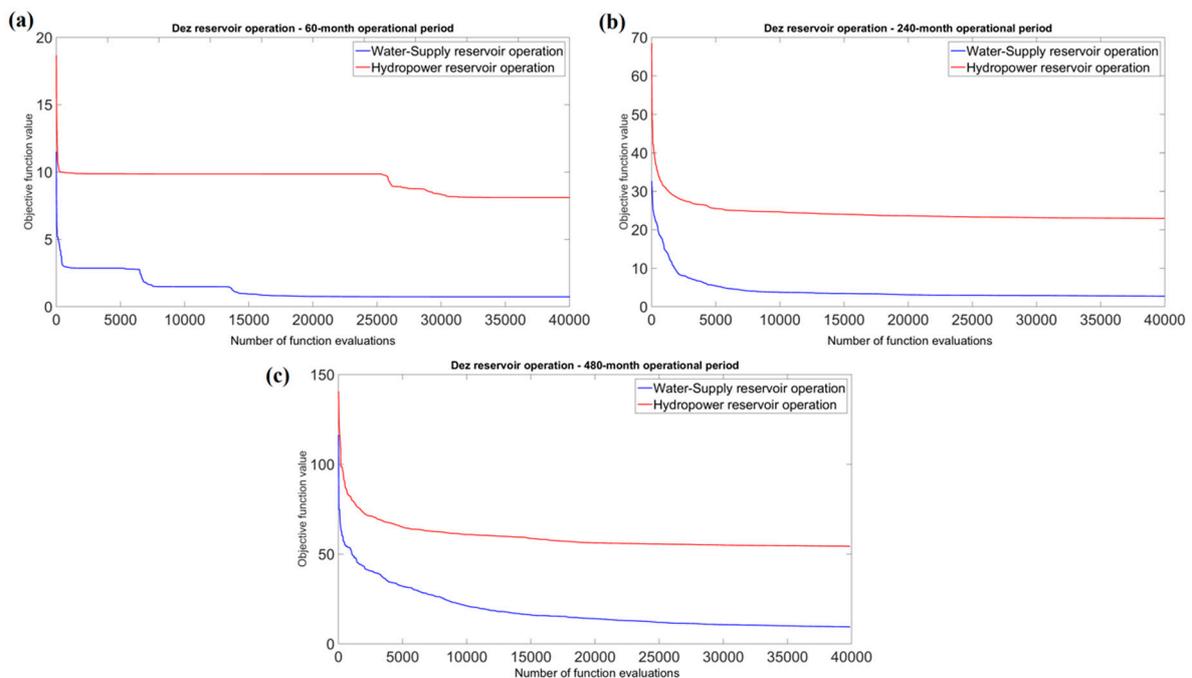


Figure 5. Convergence curve of the optimum solution obtained by the CSS algorithm for “Dez” reservoir water-supply and hydropower operation, considering evaporation losses, over 60 (a), 240 (b) and 480 (c) months periods.

4. Conclusions

Water resources management problems, including optimal reservoir operation, are complex and nonlinear optimization problems with large number of decision variables, which require search for more robust methods with high convergence speed for their solution. In this study, a robust metaheuristic optimization algorithm called the Charged System Search algorithm was introduced to the field of water resources management, and was applied to a large-scale real-world reservoir operation problem for the first time. The efficiency of the CSS algorithm was pre-evaluated using four well defined and highly nonlinear benchmark mathematical functions. Results show that the CSS approach outperforms its metaheuristic rivals considered in this study, in terms of accuracy and convergence speed. The CSS was applied to the optimization of water-supply and hydropower operation of the “Dez” reservoir in Iran for operation periods of 60, 240, and 480 months, and the results compared with those derived by Genetic, PSO, ACO, and CBB-BC algorithms as well as those obtained by the NLP approach. The results suggest that the CSS algorithm solves large-scale reservoir-operation problems with higher accuracy and convergence speed compared to the other available heuristic-search methods considered in this study. The results also indicated superiority of the CSS algorithm to the LINGO 9.0 NLP solver in highly nonlinear problem of extracting operation policies for the long-term evaporation-included hydropower operation model of “Dez” reservoir, in which the NLP solver fails to yield a feasible solution. As the non-linearity of the problem was increased by adding evaporation and/or expanding the operation period, the CSS approach outperforms the NLP solver. The CSS algorithm proves to be a robust, accurate, and fast convergent approach in handling complex water resources problems, as the effectiveness of this algorithm was shown in the present study. However, performance of this algorithm in other water management problems, such as multi-reservoir operation problems and conjunctive surface/ground water resources management, is yet to be investigated. Additionally, the reservoir operation problem presented in this study was formulated based on known river inflow values to the reservoir. However, uncertainties in river flow may affect the water release policies. Inclusion of such uncertainties in reservoir operation formulations may further improve the results to achieve more reliable operation rules.

Author Contributions: B.A. and A.A. conceived and designed the experiment; B.A. performed the experiment and wrote the initial manuscript; B.A. and A.A. analyzed the results, wrote the paper, and made the conclusions.

Funding: This research received no external funding.

Conflicts of Interest: Authors declare no conflicts of interest.

References

1. Jothiprakash, V.; Shanthi, G. Single reservoir operating policies using genetic algorithm. *Water Resour. Manag.* **2006**, *20*, 917–929. [[CrossRef](#)]
2. Rani, D.; Moreira, M.M. Simulation-optimization modeling: A survey and potential application in reservoir systems operation. *Water Resour. Manag.* **2010**, *24*, 1107–1138. [[CrossRef](#)]
3. Kumar, D.N.; Baliarsingh, F.; Raju, K.S. Optimal reservoir operation for flood control using folded dynamic programming. *Water Resour. Manag.* **2010**, *24*, 1045–1064. [[CrossRef](#)]
4. Masson-Delmotte, V.; Zhai, P.; Pörtner, H.O.; Roberts, D.; Skea, J.; Shukla, P.R.; Pirani, A.; Chen, Y.; Connors, S.; Gomis, M.; et al. IPCC, 2018: Summary for Policymakers. In *Global Warming of 1.5 °C; An IPCC Special Report on the Impacts of Global Warming of 1.5 °C above Pre-Industrial Levels and Related Global Greenhouse Gas Emission Pathways, in the Context of Strengthening the Global; IPCC: Geneva, Switzerland, 2018.*
5. Asadieh, B.; Krakauer, N.Y. Global trends in extreme precipitation: Climate models versus observations. *Hydrol. Earth Syst. Sci.* **2015**, *19*, 877–891. [[CrossRef](#)]
6. Allen, M.R.; Ingram, W.J. Constraints on future changes in climate and the hydrologic cycle. *Nature* **2002**, *419*, 224–232. [[CrossRef](#)]
7. Held, I.M.; Soden, B.J. Robust responses of the hydrological cycle to global warming. *J. Clim.* **2006**, *19*, 5686–5699. [[CrossRef](#)]

8. Asadieh, B.; Krakauer, N.Y. Global change in streamflow extremes under climate change over the 21st century. *Hydrol. Earth Syst. Sci.* **2017**, *21*, 5863–5874. [[CrossRef](#)]
9. Stocker, T.F.; Qin, D.; Plattner, G.-K.; Tignor, M.M.B.; Allen, S.K.; Boschung, J.; Nauels, A.; Xia, Y.; Bex, V.; Midgley, P.M. *Climate Change 2013: The Physical Science Basis. Working Group I Contribution to the Fifth Assessment Report of the Intergovernmental Panel on Climate Change*; Cambridge University Press: Cambridge, UK, 2013.
10. Asadieh, B.; Krakauer, N.Y.; Fekete, B.M. Historical trends in mean and extreme runoff and streamflow based on observations and climate models. *Water* **2016**, *8*, 189. [[CrossRef](#)]
11. Asadieh, B.; Krakauer, N.Y. Impacts of Changes in Precipitation Amount and Distribution on Water Resources Studied Using a Model Rainwater Harvesting System. *J. Am. Water Resour. Assoc.* **2016**, *52*, 1450–1471. [[CrossRef](#)]
12. Armal, S.; Devineni, N.; Khanbilvardi, R. Trends in extreme rainfall frequency in the contiguous United States: Attribution to climate change and climate variability modes. *J. Clim.* **2018**, *31*, 369–385. [[CrossRef](#)]
13. Najjibi, N.; Devineni, N. Recent trends in the frequency and duration of global floods. *Earth Syst. Dyn.* **2018**, *9*, 757–783. [[CrossRef](#)]
14. Najafi, E.; Devineni, N.; Khanbilvardi, R.M.; Kogan, F. Understanding the Changes in Global Crop Yields through Changes in Climate and Technology. *Earth's Future* **2018**, *6*, 410–427. [[CrossRef](#)]
15. Asadieh, B. Implications of Climate Change for Extreme Precipitation and Stream-Flow, and Impacts on Water Resources Management. Ph.D. Thesis, City University of New York, City College of New York, New York, NY, USA, September 2017.
16. Asadieh, B.; Marinov, I.; Knutson, T.R. Land and Ocean Extreme Precipitation Response to Climate Change in CMIP5 and preliminary CMIP6 Model Outputs. In *American Geophysical Union Fall Meeting 2018, Washington, DC, USA, 10–14 December 2018*; AGU: Washington, DC, USA, 2018; Abstract #A23H-2950.
17. Asadieh, B.; Krakauer, N.Y. Projected Changes in Flood and Drought Risks under Climate Change in the 21st Century: A Multi-GCM Multi-GHM Approach. In *American Geophysical Union Fall Meeting 2016, San Francisco, CA, USA, 12–16 December 2016*; AGU: Washington, DC, USA, 2016; Abstract #H21G-1509.
18. Asadieh, B.; Krakauer, N.Y. Historical Changes in Global Extreme Precipitation in Climate Models and Observations. In *American Geophysical Union Fall Meeting 2014, San Francisco, CA, USA, 15–19 December 2014*; AGU: Washington, DC, USA, 2014; Abstract #GC33A-0491.
19. Raje, D.; Mujumdar, P.P. Reservoir performance under uncertainty in hydrologic impacts of climate change. *Adv. Water Resour.* **2010**, *33*, 312–326. [[CrossRef](#)]
20. Brekke, L.D.; Miller, N.L.; Bashford, K.E.; Quinn, N.W.T.; Dracup, J.A. Climate Change Impacts Uncertainty for Water Resources in the San Joaquin River Basin, California. *J. Am. Water Resour. Assoc.* **2004**, *40*, 149–164. [[CrossRef](#)]
21. Vicuna, S.; Maurer, E.P.; Joyce, B.; Dracup, J.A.; Purkey, D. The Sensitivity of California Water Resources to Climate Change Scenarios. *J. Am. Water Resour. Assoc.* **2007**, *43*, 482–498. [[CrossRef](#)]
22. Ehsani, N.; Vörösmarty, C.J.; Fekete, B.M.; Stakhiv, E.Z. Reservoir Operations Under Climate Change: Storage Capacity Options to Mitigate Risk. *J. Hydrol.* **2017**, *555*, 435–446. [[CrossRef](#)]
23. Yeh, W.W.-G. Reservoir Management and Operations Models: A State-of-the-Art Review. *Water Resour. Res.* **1985**, *21*, 1797–1818. [[CrossRef](#)]
24. Wurbs, R.A. Reservoir-System Simulation and Optimization Models. *J. Water Resour. Plan. Manag.* **1993**, *119*, 455–472. [[CrossRef](#)]
25. Ehsani, N.; Afshar, A. Application of NA-ACO in Multiobjective Contaminant Sensor Network Design for Water Distribution Systems. In *Proceedings of the Water Distribution Systems Analysis 2010, Tucson, AZ, USA, 12–15 September 2010*; pp. 327–337.
26. Ehsani, N.; Afshar, A. Optimization of Contaminant Sensor Placement in Water Distribution Networks: Multi-Objective Approach. *Water Distrib. Syst. Anal.* **2011**, *41203*, 338–346.
27. Afshar, A.; Najafi, E. Consequence management of chemical intrusion in water distribution networks under inexact scenarios. *J. Hydroinform.* **2014**, *16*, 178. [[CrossRef](#)]
28. Najafi, E.; Afshar, A. Consequences management of chemical intrusions in urban water distribution networks using the ant colony optimization algorithm. *J. Water Wastewater* **2013**, *3575*, 82–94.

29. Bashi-Azghadi, S.N.; Afshar, A.; Afshar, M.H. Multi-period response management to contaminated water distribution networks: Dynamic programming versus genetic algorithms. *Eng. Optim.* **2017**, *50*, 415–429. [[CrossRef](#)]
30. Sangiorgio, M.; Guariso, G. NN-based implicit stochastic optimization of multi-reservoir systems management. *Water (Switzerland)* **2018**, *10*, 303. [[CrossRef](#)]
31. Fotovatikhah, F.; Herrera, M.; Shamshirband, S.; Chau, K.W.; Ardabili, S.F.; Piran, M.J. Survey of computational intelligence as basis to big flood management: Challenges, research directions and future work. *Eng. Appl. Comput. Fluid Mech.* **2018**, *12*, 411–437. [[CrossRef](#)]
32. Chau, K. Use of Meta-Heuristic Techniques in Rainfall-Runoff Modelling. *Water* **2017**, *9*, 186. [[CrossRef](#)]
33. Wang, W.C.; Xu, D.M.; Chau, K.W.; Chen, S.Y. Improved annual rainfall-runoff forecasting using PSO-SVM model based on EEMD. *J. Hydroinform.* **2013**, *15*, 1377–1390. [[CrossRef](#)]
34. Haddad, O.B.; Afshar, A.; Mariño, M.A. Optimization of non-convex water resource problems by honey-bee mating optimization (HBMO) algorithm. *Eng. Comput.* **2009**, *26*, 267–280. [[CrossRef](#)]
35. Cheng, C.-T.; Wang, W.-C.; Xu, D.-M.; Chau, K.W. Optimizing Hydropower Reservoir Operation Using Hybrid Genetic Algorithm and Chaos. *Water Resour. Manag.* **2008**, *22*, 895–909. [[CrossRef](#)]
36. Afshar, A.; Massoumi, F.; Afshar, A.; Mariño, M.A. State of the Art Review of Ant Colony Optimization Applications in Water Resource Management. *Water Resour. Manag.* **2015**, *29*, 3891–3904. [[CrossRef](#)]
37. Afshar, M.H.; Motaei, I. Constrained Big Bang-Big Crunch algorithm for optimal solution of large scale reservoir operation problem. *Int. J. Optim. Civ. Eng.* **2011**, *2*, 357–375.
38. Goldberg, D.E. *Genetic Algorithms in Search, Optimization and Machine Learning*, 1st ed.; Addison-Wesley Longman Publishing Co.: Toronto, ON, Canada, 1989; ISBN 0201157675.
39. Dorigo, M.; Birattari, M.; Stutzle, T. Ant colony optimization. *IEEE Comput. Intell. Mag.* **2006**, *1*, 28–39. [[CrossRef](#)]
40. Eberhart, R.; Kennedy, J. A new optimizer using particle swarm theory. In *MHS'95, Proceedings of the Sixth International Symposium on Micro Machine and Human Science, Nagoya, Japan, 4–6 October 1995*; IEEE: Piscataway, NJ, USA, 1995; pp. 39–43.
41. Kaveh, A.; Talatahari, S. A novel heuristic optimization method: Charged system search. *Acta Mech.* **2010**, *213*, 267–289. [[CrossRef](#)]
42. Kaveh, A.; Talatahari, S. Charged system search for optimum grillage system design using the LRFD-AISC code. *J. Constr. Steel Res.* **2010**, *66*, 767–771. [[CrossRef](#)]
43. Kaveh, A.; Talatahari, S. Optimal design of skeletal structures via the charged system search algorithm. *Struct. Multidiscip. Optim.* **2009**, *41*, 893–911. [[CrossRef](#)]
44. Sheikholeslami, R.; Kaveh, A.; Tahershamsi, A.; Talatahari, S. Application of Charged System Search Algorithm to Water Distribution Networks Optimization. *Int. J. Optim. Civ. Eng.* **2014**, *4*, 41–58.
45. Kaveh, A.; Talatahari, S. An enhanced charged system search for configuration optimization using the concept of fields of forces. *Struct. Multidiscip. Optim.* **2010**, *43*, 339–351. [[CrossRef](#)]
46. Afshar, M.H.; Shahidi, M. Optimal solution of large-scale reservoir-operation problems: Cellular-automata versus heuristic-search methods. *Eng. Optim.* **2009**, *41*, 275–293. [[CrossRef](#)]
47. Bäck, T. *Evolutionary Algorithms in Theory and Practice: Evolution Strategies, Evolutionary Programming, Genetic Algorithms*; Oxford University Press: Oxford, UK, 1995; ISBN 0-19-509971-0.
48. Gen, M.; Cheng, R. *Genetic Algorithms & Engineering Design*; Orchard, B., Yang, C., Ali, M., Eds.; Lecture Notes in Computer Science; John Wiley & Sons, Inc.: Hoboken, NJ, USA, 1997; Volume 3029, ISBN 0471127418.
49. Bozorg Haddad, O.; Afshar, A.; Mariño, M.A. Honey-Bees Mating Optimization (HBMO) Algorithm: A New Heuristic Approach for Water Resources Optimization. *Water Resour. Manag.* **2006**, *20*, 661–680. [[CrossRef](#)]
50. Meraji, H. Optimum Design of Flood Control Systems by Particle Swarm Optimization Algorithm. Master's Thesis, Iran University of Science and Technology, Tehran, Iran, July 2004.
51. Cai, X.; McKinney, D.C.; Lasdon, L.S. Solving nonlinear water management models using a combined genetic algorithm and linear programming approach. *Adv. Water Resour.* **2001**, *24*, 667–676. [[CrossRef](#)]
52. Afshar, M.H.; Moeini, R. Partially and Fully Constrained Ant Algorithms for the Optimal Solution of Large Scale Reservoir Operation Problems. *Water Resour. Manag.* **2008**, *22*, 1835–1857. [[CrossRef](#)]

