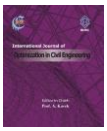
INTERNATIONAL JOURNAL OF OPTIMIZATION IN CIVIL ENGINEERING
Int. J. Optim. Civil Eng., 2016; 6(4):493-504



MULTI-OBJECTIVE OPTIMIZATION OF ARCH DAMS USING DIFFERENTIAL EVOLUTION METHODS

S. Talatahari[*, †], M.T. Aalami and R. Parsiavash
Department of Civil Engineering, University of Tabriz, Tabriz, Iran

ABSTRACT

For optimization of real-world arch dams, it is unavoidable to consider two or more conflicting objectives. This paper employs two multi-objective differential evolution algorithms (MoDE) in combination of a parallel working MATLAB-APDL code to obtain a set of Pareto solutions for optimal shape of arch dams. Full dam-reservoir interaction subjected to seismic loading is considered. A benchmark arch dam is then examined as the numerical example. The numerical results are compared to show the performance of the MoDE methods.

Keywords: multi-objective differential evolution algorithm; double curvature arch dam; optimum design.

Received: 20 November 2015; Accepted: 15 March 2016

1. INTRODUCTION

One of the most significant current matter in the structural optimization field is finding an optimal design of arch dams. Due to high construction costs and importance of their safety during an earthquake, it seems necessary to find an optimal shape for dams by considering seismic loading and fluid-structure interaction in analyzing and designing of these geometrically complicated structures. Because of reducing the complexity of problem, some simplifications are involved in literature. In number of studies the arch dam is considered with empty reservoir [1, 2] and in some others reservoir's effect is simplified by added mass approach [3, 4] which overestimates the hydrodynamic effects in dam body [5]. In addition, aside from structural point of view, finding an appropriate method using optimization techniques is of main concern in the field of arch dam optimization.

Recently, researchers have shown an increased interest in employing metaheuristic approaches in finding optimal shape of arch dams. However, many studies in this field have

[*]Corresponding author: Department of Civil Engineering, University of Tabriz, Tabriz, Iran
[†]E-mail address: Talatahari@tabrizu.ac.ir (S. Talatahari)

considered this problem as a single objective optimization, which yields only one single design as the answer [6, 7]; actually for this problem, there are some conflicting objectives that should be optimized simultaneously. The concrete volume and stress state of the dam body are two contrasting objectives that should be minimized. To find a single design as the best optimal shape, a decision maker method can be used that rank the Pareto solutions to rate them according to their preferences.

This paper represents the multi-objective formulation of dam optimization problem with employing multi-objective differential evolution (MoDE) algorithm for optimizing arch dams considering dam-reservoir interaction subjected to seismic loading. The single objective DE is a vector-based evolutionary algorithm which successful application of it for structural optimization has been reported in the literature [8]. A parallel working APDL-MATLAB code for modeling, analyzing and obtaining fitness functions, is developed for interfacing simultaneously with the MoDE to find the Pareto solution. The performance of proposed methodology is evaluated for a benchmark real-world arch dam.

2. FORMULATION OF STRESS-VOLUME MULTI-OBJECTIVE ARCH DAMS

The solution of problems with multiple objective is a set of multiple sub-solutions, which optimizes simultaneously these objectives. Multi-objective optimization problems arise when optimal decisions need to be taken in the presence of trade-offs between two or more conflicting objectives. Typically, there does not exist a single solution that simultaneously optimizes each objective. Instead, there exists a (possibly infinite) set of Pareto optimal solutions. A solution is called *non dominated* or *Pareto optimal* if none of the objective functions can be improved in value without degrading one or more of the other objective values.. In mathematical terms, this problem can be formulated as follows:

Minimize:
$$\{fit_1(\boldsymbol{x}), fit_2(\boldsymbol{x}), ..., fit_k(\boldsymbol{x}), ..., fit_N(\boldsymbol{x})\} \quad \boldsymbol{x} \in \boldsymbol{X} \tag{1}$$

Where the integer $N \geq 2$ is the number of objectives, $\boldsymbol{X}$ is the feasible set of the decision vector. The feasible set is typically defined by some constraint functions:

$$\Phi_i(\mathrm{x}) > 0 \quad (i = 1, 2, 3, \ldots, p) \tag{2}$$

in which $\Phi_i(\boldsymbol{x})$ and p denote the constraint functions and number of constraints, respectively. In addition, the vector-valued objective function is often defined as:

$$fit : \boldsymbol{X} \rightarrow R^N, fit(\boldsymbol{x}) = \left(fit_1(\boldsymbol{x}), fit_2(\boldsymbol{x}), ..., fit_k(\boldsymbol{x}), ..., fit_N(\boldsymbol{x})\right)^T \tag{3}$$

A feasible solution, $\boldsymbol{x}_1 \in \boldsymbol{X}$, is said to dominate another solution, $\boldsymbol{x}_2 \in \boldsymbol{X}$, if both following equations are satisfied:

$$fit_i(\boldsymbol{x}_1) \le fit_i(\boldsymbol{x}_2) \quad for\ all\ indices \quad i \in \{1,2,...,\mathrm{N}\} \tag{4}$$

$$fit_j(\boldsymbol{x}_1) < fit_j(\boldsymbol{x}_2) \quad for\ at\ least\ one\ index \quad j \in \{1,2,...,N\} \tag{5}$$

A solution, $\boldsymbol{x}_1 \in \boldsymbol{X}$, is called the Pareto optimal if there does not exist another solution that dominates it.

Considering these definitions here, two types of objectives are considered:

Objective 1: The concrete volume of the arch dam that should be minimized. It can be determined by integration on the dam surfaces:

Minimize:
$$fit_1(\boldsymbol{X}) = \iint_A \left| y_u(x,z) - y_d(x,z) \right| dA \tag{6}$$

in which, *A* is an area produced by projecting the dam body on a xz plan; $y_u(x,z)$ and $y_d(x,z)$ are parabolas of upstream and downstream surfaces of the arch dam, respectively.

Objective 2: Failure criterion function of Willam and Warnke. For concrete structures, it is defined as [9]:

Minimize:
$$fit_2(\boldsymbol{X}) = \left(\frac{F}{f_c} - \frac{S}{s_f} \right)_{n,t} \quad n = 1,2,\ldots,n_d \quad t = 1,2,\ldots,T \tag{7}$$

where, *F* is the function of principal stress state ($\sigma_{xp}, \sigma_{yp}, \sigma_{zp}$) in which, $\sigma_{xp}, \sigma_{yp}\ and\ \sigma_{zp}$ are principal stresses in principal directions $x, y\ and\ z$, respectively. *T* is the earthquake duration. n_d is the total number of nodes in the finite element model. s_f is safety factor which for the earthquake loading may be chosen as $s_f = 1$, [10]. *S* is failure surface expressed in terms of principal stresses.

3. DESIGN CONSTRAINTS

Two constraints are considered in this study. The first one is required to ensure that upstream and downstream faces of the dam do not pass through each other (Eq. 8). The second one is for constructing facilities and having smooth cantilevers over the height of the dam. For this purpose, the slope of overhang at the upstream and downstream faces of the dam should satisfy Eq. 9:

$$r_{di} \le r_{ui} \Rightarrow \frac{r_{di}}{r_{ui}} - 1 \le 0, i = 1,2,..6 \tag{8}$$

$$\gamma \le \gamma_{alw} \Rightarrow \frac{\gamma}{\gamma_{alw}} - 1 \le 0 \tag{9}$$

where r_{di} and r_{ui} are the radius of curvature at ith level in the z direction and $\gamma = \cot\alpha$ is the slope of overhang at the downstream and upstream faces of the dam. γ_{alw} is the allowable absolute value for the aforementioned parameter. To ensure the sliding stability of the dam, following equation should be satisfied:

$$\varphi_l \le \varphi \le \varphi_u \tag{10}$$

where, φ is the central angle of arch dam at ith level in the z direction. φ_l and φ_u are allowable lower and upper bounds of the central angle. For practical purposes φ usually varies from 90° to 130° throughout the dam height.

4. SINGLE OBJECTIVE DIFFERENTIAL EVOLUTION

The single objective Differential evolution (DE) is a vector-based evolutionary algorithm developed by Storn and Price, designed for optimization problems in continuous search space [11]. It is a stochastic search algorithm with self-organizing tendency and does not use the information of derivatives. Thus, it is a population-based, derivative-free method. Although DE can be considered as a further development to genetic algorithms, in contrast with GA, it treats solutions as real-number strings, thus no encoding and decoding is needed.

DE has its own evolutionary strategies: mutation, crossover and selection. Different from genetic algorithm, mutation is the key operator of DE. To illustrate, in genetic algorithms, mutation is carried out at one site or multiple sites of a chromosome, while in differential evolution, a difference vector of two randomly chosen vectors is used to perturb an existing vector. If three distinct individuals x_{r1}^{g} , x_{r2}^{g} and x_{r3}^{g} are randomly selected from the current population, the mutation operator is mathematically described as:

$$m_i^{g+1} = x_{r1}^{g} + F.(x_{r2}^{g} - x_{r3}^{g}) \tag{11}$$

in which, the m_i^{g+1} is the mutant vector, r_1 , r_2 and r_3 are three mutually exclusive integers different from index i . $F \in [0; 2]$ is a parameter, often referred to as the differential weight. In principle, $F \in [0; 2]$, but in practice, a scheme with $F \in [0; 1]$ is more efficient and stable. The trial vector v_i^{g+1} is obtained by applying the crossover operator on mutant vector m_i^{g+1} as follows:

$$v_{ij}^{g+1} = \begin{cases} m_{ij}^{g+1}, \; If \;\; (rand \le CR) \; or \; j = sn \\ x_{ij}^{g} \;, otherwise \end{cases} \tag{12}$$

in which, $rand$ generates a random value in [0,1]; CR is the crossover constant in [0,1]; sn is an arbitrary number in (1,2,…,D) that ensure v_i^{g+1} gets at least one parameter from m_i^{g+1}.

The Selection is essentially the same as that used in genetic algorithms. It is to select the most fittest to pass onto the next generation. The better individual between v_i^{g+1} and x_i^g is survived into the next generation:

$$x_i^{g+1} = \begin{cases} v_i^{g+1}, & if\ v_i^{g+1} is\ better\ than\ \mathrm{x}_i^g . \\ x_i^g\ , & otherwise \end{cases} \tag{13}$$

DE repeats the above three operators until a termination criterion is satisfied.

5. MULTI-OBJECTIVE DIFFERENTIAL EVOLUTION

To date there are relatively some papers that propose ways of extending DE to handle the multi-objective optimization. Single objective Differential Evolution strategy with an aggregating function to solve bi-objective problems was proposed in [12]. Another MODE was proposed in [13]. This algorithm uses a variant of the original DE so that the best individual is adopted to create the offspring. The Pareto-Based DE was then proposed in [14]. In this algorithm, the DE is extended to multi-objective optimization by incorporating a non-dominated sorting and ranking selection procedure proposed by Deb et al. [15, 16]. Vector Evaluated Differential Evolution (VEDE) as a parallel, multi-population algorithm was proposed in [17]. Non-dominated Sorting Differential Evolution (NSDE) was proposed in [18]. Generalized Differential Evolution (GDE) was also proposed in [19]. GDE extends the selection operation of the basic DE algorithm for constrained multi-objective optimization. It is a simple modification of the NSGA-II [16]. In contrast with NSGA-II in this method, DE operators are used for generating new individuals. In current study, we utilize the MoDE-RMO method proposed by Chen et al. [20]. This algorithm combines the ranking-based mutation operator with MoDE to accelerate the convergence speed and enhance its performance.

5.1 Ranking-based mutation operator

Ranking-based mutation operator (RMO) for DE in single-objective optimization in which, the parents are proportionally selected according to their rankings in the current population [21]. The higher ranking a parent obtains the more opportunity it will be selected. Integrating the proposed ranking-based mutation operator into some DE algorithms indicated that the ranking-based mutation operator is able to enhance the performance of the DE algorithms in single objective optimization [20]. The main challenge in extending this operator for the MoDE is that, in single objective DE, the population can be directly sorted from best to worst, while in multi-objective optimization there exist many solutions, which are non-dominated with each other. To deal this problem, a non-dominated sorting and crowding distance are incorporated into ranking-based mutation operator as presented in the following subsections.

5.1.1 Non-dominated sorting and crowding distance

In the non-dominated sorting procedure, two entities is calculated for each solution [20]. The first one is domination count n_p, that is the number of solutions which dominate the solution p and the second one is S_p (a set of solutions that the solution dominates). All solutions in the first non-dominated front will have their domination count as zero. Now, for each solution p with $np = 0$, we visit each member (q) of its set S_p and reduce its domination count by one. Then, if the domination count for any member becomes zero, we put it in a separate list, denoted by Q. These members are belong to the second non-dominated front. Now, the above procedure is continued for other members and the third front is identified. This process continues until all fronts are identified.

Crowding distance is used to get an estimate of the density of solutions surrounding a particular individual i in the population, and it calculates the average distance of two solutions on either side of solution i (i.e. $i + 1$ and $i - 1$) along each of the objectives [20]. This requires sorting the population according to each objective function value in an ascending order. Afterwards, for each objective function, the boundary solutions i.e. solutions with smallest and largest function values, are assigned an infinite distance value. All other intermediate solutions are assigned a distance value equal to the absolute normalized difference in the function values of two adjacent solutions. This calculation is continued for other objective functions, as well. The overall crowding-distance value is calculated as the sum of individual distance corresponding to the each objective. The objective values were normalized before calculating the crowding distance.

5.1.2 Ranking assignment and selection probability

The non-domination front number i_{front} and crowding distance i_{cd} are obtained for each solution i, to define a partial order for the whole population. If any of the two following conditions are satisfied then the solution i is better than solution j:

$$i_{front} < j_{front} \tag{14}$$

$$i_{front} = j_{front} \quad \wedge \quad i_{cd} > j_{cd} \tag{15}$$

Now, based on the partial order defined before, the population can be sorted in ascending order. The ranking of an individual is assigned as follows:

$$R_i = Np - i, \quad i = 1, 2, ..., Np \tag{16}$$

where Np is the population size. According to Eq. (16), the best individual in the current population will obtain the highest ranking. After assigning the ranking for each individual, the selection probability p_i of the ith individual is calculated as:

$$p_i = \frac{R_i}{Np}, \quad i = 1, 2, ..., Np \tag{17}$$

The ranking-based mutation operator of "DE/rand/1" for multi-objective optimization can be described as follows [20]:

Step1: calculate the non-domination front number of each individual through the non-dominated sorting procedure.

Step2: calculate the crowding distance of each individual.

Step3: sort the population in ascending order and assign the ranking and selection probability p_i for each individual.

Step4: while $rand > p_{r1}$ or $r1 = i$, select base vector index randomly as: $r1 \in \{1, Np\}$

Step5: while $rand > p_{r2}$ or $r2 = r1$ or $r2 = i$, select terminal vector index randomly as: $r2 \in \{1, Np\}$

Step6: while $r3 = r2$ or $r3 = r1$ or $r3 = i$, select starting vector index randomly as: $r3 \in \{1, Np\}$

According to this procedure, the individual with a higher ranking will have a larger probability to be selected as base vector or terminal vector in the mutation operator; therefore, it is beneficial for the propagation of good information in the population to the offspring [20]. In the ranking-based mutation operator for multi-objective optimization, the starting vector is randomly selected while the base and terminal vectors are selected based on their selection probabilities. The reason is clear; if the two vectors in the difference vector are both chosen from better vectors, then the search step-size of the difference vector may decrease quickly and may lead to premature convergence [21, 22].

5.2 Multi-objective differential evolution with RMO

The multi-objective differential evolution with RMO (MoDE-RMO) combines the advantages of standard DE with the mechanism of Pareto-based ranking and crowding distance sorting [20]. In MoDE-RMO crossover operator is conducted in the same way as that in single-objective optimization but because trial and target vectors are usually non-dominated with each other, selection needs to be redesigned as shown in Fig. 1. At the end of a generation, total size of the population is between Np and $2NP$. This population is truncated for the next step of the algorithm by non-dominated sorting and evaluating the individual of the same front with the crowding distance. The truncation procedure keeps only the best Np vectors in the population. The pseudo-code of MoDE-RMO is shown in Fig. 2.

1) **if** the trial vector dominates the target vector **then**
   Use the trial vector to replace the target vector
2) **if** the target vactor dominates the trial vector **then**
   The trial vector is discarded
3) **else**
   The trial vector is added to the population
   end if

Figure 1. Pseudo-code of selection operator for MoDE-RMO

initialize vectors at the population
$p^g = \{x_1^g, x_2^g, ..., x_{NP}^g\}$ with $x_i^g = \{x_{i1}^g, x_{i2}^g, ..., x_{iD}^g\} (i = 1, ..., NP)$
Set mutation scale factor *F*,
Set crossover constant *CR*,
Set maximum number of generations, *Maxgen*.
Evaluate the fitness value of each target vector x_i^g.
for i=0 to g **do**
  Repeat
  Determine the selected vector indexes $r1$, $r2$, and $r3$ through the RMO operator.
  Use crossover using DE scheme to generate the trial vector v_i^{g+1} for each target vector x_i^g.
  Use RMO to generate a mutant vector m_i^{g+1} corresponding to the target vector x_i^g.
  Evaluate the trial vector v_i^{g+1} and use the following selection operation
  if v_i^{g+1} dominates x_i^g, **then** $x_i^{g+1} = v_i^{g+1}$.
  if x_i^g dominates v_i^{g+1}, **then** v_i^{g+1} is rejected.
  if v_i^{g+1} and x_i^g are dominated with each other, **then** add v_i^{g+1} to the population.
  until population is completed
    Sort the population based on fast non-dominated sorting and crowding distance
    Add best NP individuals into the next generation
end for

Figure 2. Pseudo-code of the MoDE-RMO algorithm

6. NUMERICAL INVESTIGATION

The Morrow point arch dam is selected as the case study in this paper (Fig. 3). This double curvature thin-arch concrete structure that is located 263 km southwest of Denver, Colorado, has 142.65 m high and 220.68 m long along the crest and its thickness varies from 3.66 m at the crest to 15.85 m at the base level [23]. A finite element model utilizing 8-node solid elements for arch dam and 8-node fluid element for reservoir has been developed and verified as presented in Ref [24]. Since the Morrow point dam model is symmetry, designers can analysis the half of the dam's finite element with considering the proportional boundary constraints in the center of dam. However, it is worth to mention that, the natural frequencies of some modes are eliminated due to considering only half of the dam as shown in literatures. Thus, in order to perform an exact analysis, we consider the complete model of the dam, here. According to the results, a good conformity has been achieved between the results of the present work with those of the reported in the literature [24].

The dam body construction required 273600 m^3 of concrete. Material properties of both the dam body and water are presented in Table 1. To handle the optimization, The MoDE and MoDE-RMO are coded in MATLAB software and modeling and analyzing of the arch

dam are done using a combination of parallel working MATLAB and Ansys Parametric Design Language (MATLAB-APDL) codes. Reservoir is supposed to be full and the dam-reservoir interaction subjected to seismic loading is taken to account in this example. In order to construct the dam geometry, six controlling levels are considered so the dam can be modeled using twenty shape design variables. In this research, the N–S record of 1940 El Centro earthquake is selected to apply to the arch dam-reservoir system in the upstream–downstream direction [25]. Damping ratio is considered equal to 5%.

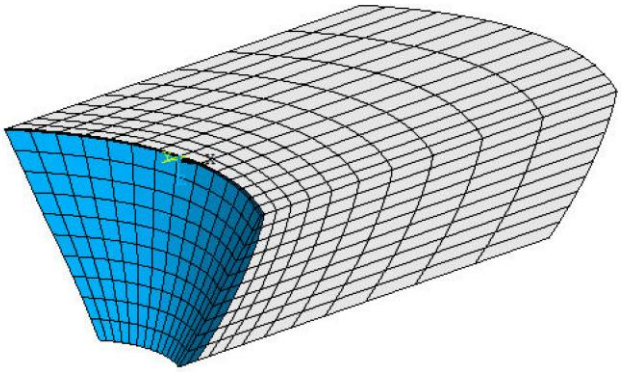


Figure 3. Finite element model of the morrow point dam

Table 1: Material property of the morrow point dam

Material	Property	Value	unit
Mass Concrete	Modulus of Elasticity	27.579	GP
	Poisson's Ratio	0.2	-
	Mass Density	2483	Kg/m^3
	Uniaxial compressive strength of concrete (f_c)	30	MPa
	Uniaxial tensile strength of concrete (f_t)	1.5	MPa
Water	Bulk Modulus	2.15	GP
	Mass Density	1000	Kg/m3
	Velocity of Pressure Waves	1438.66	m/s
	Wave reflection coefficient	1	-

For both algorithms presented in this paper, the number of agents is set to 100 and the number of iterations is limited to 200. In Table 2, the extreme values obtained by the MoDE algorithms are presented. It should be noted that the maximum allowable values for the dam's volume and Willam-Warnke failure criterion is limited to 3.3×10^5 and 0.8, respectively. According to this table, for those points which fit_2 is more important than fit_1, the MoDE-RMO yields 20% better than the MoDE. Since the accepted results should be negative for fit_2, so Table 2 presents best extreme value with negative fit_2, as well. The MoDE-RMO can find an acceptable design with 2.35×10^5 m^3 while the best acceptable results for the MoDE is 2.82×10^5 m^3. The values of variables for the best design are presented in Table 3. The Pareto fronts of these methods are presented in Fig. 4. According to this figure, the results of the MoDE-RMO are obviously better than the MoDE.

Table 2: Comparison of the extreme results

Optimization method	MoDE	MoDE-RMO
Obtained extreme values	[0.78, 2.49e5] [-0.45, 3.23e5]	[0.78, 2.01e5] [-0.55, 3.09e5]

Table 3: The best design obtained by the MoDE-RMO

Variable	Value	Variable	Value	Variable	Value
$\gamma(m/m)$	0.132	$\beta(m/m)$	0.536	$rd_1(m)$	107.990
$tc_1(m)$	5.514	$ru_1(m)$	108.824	$rd_2(m)$	92.778
$tc_2(m)$	10.947	$ru_2(m)$	93.807	$rd_3(m)$	79.680
$tc_3(m)$	13.586	$ru_3(m)$	79.860	$rd_4(m)$	66.365
$tc_4(m)$	15.470	$ru_4(m)$	68.384	$rd_5(m)$	53.872
$tc_5(m)$	14.618	$ru_5(m)$	54.010	$rd_6(m)$	40.009
$tc_6(m)$	14.946	$ru_6(m)$	40.229		

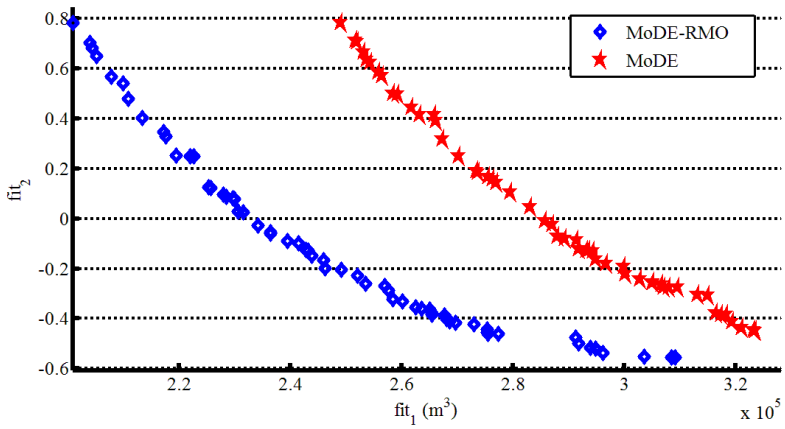


Figure 4. Pareto fronts of the MoDE-based methods

Table 4: Different possible scenarios for the morrow point arch dam with corresponding solutions

Scenario	Importance of criteria	Possible priority weights	Selected solution by MTDM					
			MoDE			MoDE-RMO		
			fit_1	fit_2	R_i	fit_1	fit_2	R_i
A	$C1 \gg C2$	[0.9,0.1]	291759.83	-0.122	0.8498	241519.88	-0.097	0.8499
B	$C1 > C2$	[0.7,0.3]	305005.00	-0.255	0.7368	253568.03	-0.260	0.7367
C	$C1 \approx C2$	[0.5,0.5]	317318.55	-0.384	0.7069	265417.17	-0.379	0.7071
D	$C1 < C2$	[0.3,0.7]	330957.78	-0.514	0.7368	275429.28	-0.455	0.7367
E	$C1 \ll C2$	[0.1,0.9]	399834.14	-0.609	0.8498	308460.05	-0.556	0.8499

For the process of decision making and finding the best solution, the decision maker (DM) should notify their preferences by considering all the information integrated in the

Pareto front. In order to show the wide range of possible solutions, five different scenarios are also considered. The numerical values of these scenarios for the algorithms are presented in Table 4. In this table, R is the global ranking function to performing the DM. As it can be seen, the all good results are found by the MoDE-RMO.

7. CONCLUSIONS

In this study, shape optimization of arch dams including the dam-reservoir interaction is presented as a multi-objective problem. The concrete volume and Willam-Warnke failure criterion are considered as objective functions. To solve this problem, two multi objective differential evolution algorithms are employed. To fulfill this aim, we developed an enhanced parallel-working APDL-MATLAB code. In order to examine the effectiveness of proposed methodology, Morrow Point arch dam optimization is performed. The resulted Pareto fronts of MoDE-RMO as well as extreme values are obtained and compared with the standard MoDE. According to the results, the utilized method can find much better results compared to its standard variant.

REFERENCES

1. Kaveh A, Mahdavi VR. Optimal design of arch dams for frequency limitations using charged sytem search and partical swarm optimization, *Int J Optim Civil Eng* 2011; **4**: 543-55.
2. Gholizadeh S, Seyedpoor SM. Optimum design of arch dams with frequency limitations, *Int J Optim Civil Eng* 2011; **1**: 1-14.
3. Kaveh A, Mahdavi VR. Colliding bodies optimization for design of arch dams with frequency limitations, *Int J Optim Civil Eng* 2014; **4**: 473-90.
4. Akbari J, Ahmadi MT, Moharrami H. Advances in concrete arch dams shape optimization, *Appl Math Model* 2011; **35**: 3316-33.
5. Kuo JSH. Fluid-structure interactions: Added mass computations for incompressible fluid, Earthquake Engineering Research Center, University of California, Berkeley, 1982, 126 p.
6. Tajalli F, Ahmadi MT, Moharrami H. A shape optimization algorithm for seismic design of a concrete arch dam, *Dam Eng* 2007; **18**: 139-62.
7. Seyedpoor SM, Salajegheh J, Salajegheh E, Gholizadeh S. Optimal design of arch dams subjected to earthquake loading by a combination of simultaneous perturbation stochastic approximation and particle swarm algorithms, *Appl Soft Comput* 2011; **11**: 39-48.
8. Talatahari S, Gandomi AH, Yang XS, Deb S. Optimum design of frame structures using the Eagle Strategy with Differential Evolution, *Eng Struct* 2015; **91**: 16-25.
9. Willam KJ, Warnke EP. Constitutive model for the triaxial behavior of concrete, *Int Assoc Bridge Struct Eng* 1975, pp. 1-30.
10. USBR, Design criteria for concrete arch and gravity dams, in: U.s.d.o.i.b.o. reclamation (Ed.), US Government Printing Office, Washington, 1977.

11. Price K, Storn R, Lampinen J. *Differential Evolution: A Practical Approach to Global Optimization*, Springer, 2005.
12. Babu BV, Mathew Leenus Jehanand M. *Differential Evolution for Multi-Objective Optimization, Congress on Evolutionary Computation (CEC'2003)*, IEEE Press, Canberra, Australia, 2003, pp. 2696–2703.
13. Xue F, Sanderson AC, Graves RJ. Pareto-*based Multi-Objective Differential Evolution, Congress on Evolutionary Computation (CEC'2003)*, IEEE Press, Canberra, Australia, 2003, pp. 862-869.
14. Madavan NK. *Multiobjective Optimization Using a Pareto Differential Evolution Approach, Evolutionary Computation (CEC'2002)*, IEEE Service Center, Piscataway, New Jersey, 2002, pp. 1145-1150.
15. Deb K, Agrawal S, Pratab A, Meyarivan T. *A Fast Elitist Non-Dominated Sorting Genetic Algorithm for Multi-Objective Optimization: NSGA-II, Parallel Problem Solving from Nature PPSN VI*, Springer, Paris, France, 2000, pp. 849-858.
16. Deb K, Pratap A, Agarwal S, Meyarivan T. A fast and elitist multi-objective genetic algorithms: NSGA-II, *IEEE Transactions on Evolutionary Computation* 2002; **6**: 182-97.
17. Parsopoulos KE, Taoulis DK, Pavlidis NG, Plagianakos VP, Vrahatis MN. Vector evaluated differential evolution for multiobjective optimization, *2004 Congress on Evolutionary Computation (CEC'2004)*, IEEE Service Center, Portland, Oregon, USA, 2004, pp. 204-211.
18. AW Iorio LX. *Solving Rotated Multi-objective Optimization Problems Using Differential Evolution, Advances in Artificial Intelligence*, Springer, Cairns, Australia, 2004, pp. 861-872.
19. Kukkonen S, Lampinen J. *An Extension of Generalized Differential Evolution for Multi-Objective Optimization with Constraints*, Parallel Problem Solving from Nature - PPSN VIII, Springer, Birmingham, UK, 2004, pp. 752–761.
20. Chen X, Du W, Qian F. Multi-objective differential evolution with ranking-based mutation operator and its application in chemical process optimization, *Chemometr Intell Lab Syst* 2014; **136**: 85-96.
21. Gong W, Cai Z. Differential evolution with ranking-based mutation operators, , *IEEE Trans on Cybernetics* 2013; **43**: 2066-81.
22. Gong W, Cai Z, Liang D. Engineering optimization by means of an improved constrained differential evolution, *Comput Methods Appl Mech Eng* 2014; 268: 884-904.
23. USACE, Earthquake Design and Evaluation of Concrete Hydraulic Structures, U. S. Army Corps of Engineers, Washington, DC, America, 2007.
24. Talatahari S, Aalami MT, Parsiavash R. Optimum design of double curvature arch dams using a quick hybrid charged system search algorithm, *Int J Optim Civil Eng* 2016; **6**(2): 227-43.
25. PEER, El Centro, 1940 Ground Motion Data, in: P.E.E.R. Centre (Ed.), 2009.