

Optimal Design of Reinforced Concrete Cantilever Retaining Walls Utilizing Eleven Meta-Heuristic Algorithms: A Comparative Study

Ali Kaveh^{1*}, Kiarash Biabani Hamedani¹, Taha Bakhshpoori²

¹ Centre of Excellence for Fundamental Studies in Structural Engineering, School of Civil Engineering, Iran University of Science and Technology, Tehran-16, Iran

² Faculty of Technology and Engineering, Department of Civil Engineering, East of Guilan, University of Guilan, Rudsar-Vajargah, Iran

* Corresponding author, e-mail: alikaveh@iust.ac.ir

Received: 01 November 2019, Accepted: 16 November 2019, Published online: 17 January 2020

Abstract

In this paper, optimum design of reinforced concrete cantilever retaining walls is performed under static and dynamic loading conditions utilizing eleven population-based meta-heuristic algorithms. These algorithms consist of Artificial Bee Colony algorithm, Big Bang-Big Crunch algorithm, Teaching-Learning-Based Optimization algorithm, Imperialist Competitive Algorithm, Cuckoo Search algorithm, Charged System Search algorithm, Ray Optimization algorithm, Tug of War Optimization algorithm, Water Evaporation Optimization algorithm, Vibrating Particles System algorithm, and Cyclical Parthenogenesis Algorithm. Two well-known methods consisting of the Rankine and Coulomb methods are used to determine lateral earth pressures acting on cantilever retaining wall under static loading condition. In addition, Mononobe-Okabe method is employed for dynamic loading condition. The design is based on ACI 318-05 and the goal of optimization is to minimize the cost function of the cantilever retaining wall. The performance of the utilized algorithms is investigated through an optimization example of cantilever retaining wall. In addition, convergence histories of the algorithms are provided for better understanding of their performance.

Keywords

structural optimization, cantilever retaining wall, meta-heuristic algorithms, Mononobe-Okabe method, Rankine method, Coulomb method

1 Introduction

Retaining walls are structures designed and constructed in order to support vertical or near-vertical slopes of soil or other loose materials. Retaining wall structures are used for various design situations, such as railroads, highways, bridge abutments, culverts, etc. Therefore, low-cost design of the retaining wall structures can be considered as an important problem to be solved. Optimization approaches can be employed for this purpose. There are different types of retaining wall structures used for various purposes, but they can generally be grouped into two main categories of "gravity retaining walls" and "cantilever retaining walls". Gravity retaining walls are usually constructed with plain concrete and derive their stability against sliding and overturning depends by their self weight. This type of retaining wall is not an economical choice for high walls. Cantilever retaining walls are constructed of reinforced concrete. A typical cantilever retaining wall consists of three components,

including a vertical stem, a base slab, and a shear key. The base slab is composed of a toe slab at the front of the wall and a heel slab at the backfill side. There are various forces acting on a retaining wall, such as surcharge loads, lateral pressures of soil and surcharge, soil pressures on the footing, weight of soil above the base, self-weight of the retaining wall, etc. The design process of a cantilever retaining wall can be divided into two phases. In the first phase of design, the lateral earth pressures are calculated and the cantilever retaining wall is checked for stability against sliding, overturning, and bearing capacity failure. In the second phase, each element of the wall is controlled for strength and the reinforcement of each element is calculated. Therefore, proper design of retaining wall structures requires a satisfying estimation of the lateral earth pressures. There are various theories for estimating lateral earth pressure, such as Rankine method, Coulomb method, Mononobe-Okabe method.

Many researchers have applied meta-heuristic algorithms for optimum design of retaining wall structures. Yepes et al. [1] presented a parametric study of commonly-used earth-retaining walls by Simulated Annealing algorithm. Kaveh and Shakouri Mahmud Abadi [2] performed cost optimization of cantilever retaining walls by Harmony Search and Improved Harmony Search algorithms. Camp and Akin [3] applied a hybrid multiphase Big Bang-Big Crunch algorithm to the low-cost design of cantilever retaining walls under static loading condition. Khajezadeh et al. [4] presented the Gravitational Search Algorithms and applied it for optimum design of reinforced concrete cantilever retaining walls. Kaveh and Behnam [5] performed optimum cost design of reinforced concrete cantilever retaining walls by means of Charged System Search algorithm. Kaveh and Khayatazad [6] introduced a pseudo-dynamic approach for estimating seismic earth pressures behind retaining walls and employed Ray Optimization algorithm to optimum design of cantilever retaining walls. Kaveh and Soleimani [7] utilized Colliding Bodies Optimization and Democratic Particle Swarm Optimization algorithms to determine optimum design of reinforced concrete cantilever retaining walls under static and dynamic loading conditions. Kaveh and Farhoudi [8] applied Dolphin Echolocation Optimization algorithm to optimum design of cantilever retaining walls. Kaveh and Jafarpour Laien [9] employed three meta-heuristic algorithms, including Colliding Bodies Optimization, Enhanced Colliding Bodies Optimization, and Vibrating Particles System algorithms, to solve the problem of minimizing the cost of cantilever retaining walls under static and dynamic loading conditions. Some other applications of meta-heuristic algorithms can be found in [10–14].

In this research, eleven population-based meta-heuristic algorithms are utilized for optimum design of reinforced concrete cantilever retaining walls. These algorithms consist of Artificial Bee Colony, Big Bang-Big Crunch, Teaching-Learning-Based Optimization, Imperialist Competitive Algorithm, Cuckoo Search, Charged System Search, Ray Optimization, Tug of War Optimization, Water Evaporation Optimization, Vibrating Particles System, and Cyclical Parthenogenesis Algorithm. The codes for these algorithms are those of Kaveh and Bakhshpoori [15]. The design is based on the requirements of the American Concrete Institute (ACI 318-05 [16]) and is implemented for both static and dynamic loading conditions. Two theories known as the Coulomb and Rankine theories are

applied for calculating the lateral earth pressures under static loading condition. In addition, the Mononobe-Okabe method is utilized to calculate the lateral earth pressures under dynamic loading condition. The cost function of cantilever retaining wall is considered as the objective function. The cost function is minimized subject to stability and strength constraints. A numerical example is studied to demonstrate the effectiveness of the meta-heuristic algorithms and to compare their performance.

2 Materials and methods

2.1 Meta-heuristic algorithms

Eleven meta-heuristic algorithms are utilized to optimize cantilever retaining walls. These algorithms are as follows: 1) Artificial Bee Colony (ABC) algorithm, 2) Big Bang-Big Crunch (BB-BC) algorithm, 3) Teaching-Learning-Based Optimization (TLBO) algorithm, 4) Imperialist Competitive Algorithm (ICA), 5) Cuckoo Search (CS) algorithm, 6) Charged System Search (CSS) algorithm, 7) Ray Optimization (RO) algorithm, 8) Tug of War Optimization (TWO) algorithm, 9) Water Evaporation Optimization (WEO) algorithm, 10) Vibrating Particles System (VPS) algorithm, and 11) Cyclical Parthenogenesis Algorithm (CPA). Kaveh and Bakhshpoori [1] coded these algorithms and performed some experimental evaluations to assess the performance of the algorithms in both aspects of convergence rate and accuracy. The algorithms are introduced briefly in Subsections 2.1.1 to 2.1.11.

2.1.1 Artificial Bee Colony algorithm

The Artificial Bee Colony (ABC) algorithm, introduced by Karaboga in 2005 [17], is based on the foraging behavior of honey bees. In ABC algorithm each candidate solution is represented by a food source, and its nectar quality represents the objective function of that solution. These food sources are modified by honey bees in a repetitive process manner with the aim of reaching food sources with better nectar. In the ABC honey bees are considered as three types: employed or recruited, onlooker, and scout bees with different tasks in the colony. Bees perform modification with different strategies according to their task. Employed bees try to modify the food sources and share their information with onlooker bees. Onlooker bees select a food source based on the information from employed bees and attempt to modify it. Scout bees perform merely random search in the vicinity of the hive. Hence in each iteration, the ABC algorithm searches in three sequential phases.

2.1.2 Big Bang-Big Crunch algorithm

The Big Bang-Big Crunch (BB-BC) algorithm was developed by Erol and Eksin [18]. BB-BC is taken from the prevailing evolutionary theory for the origin of universe: the Big Bang Theory. According to this theory, in the Big Bang phase, particles are drawn toward irregularity by losing energy, while in the Big Crunch phase, they converged toward a specific direction. Like other population-based meta-heuristics, BB-BC starts with a set of random initial candidate solutions, as the initial Big Bang. After each Big Bang phase, a Big Crunch phase should take place to determine a convergence operator by which particles will be drawn into an orderly fashion in the subsequent Big Bang phase. The convergence operator can be the weighted average of the positions of the candidate solutions or the position of the best candidate solution. These two contraction (Big Crunch) and dispersing (Big Bang) phases are repeated in the cyclic body of the algorithm in succession to satisfy a stopping criteria with the aim of steering the particles toward the global optimum.

2.1.3 Teaching-Learning-Based Optimization algorithm

Teaching-Learning-Based Optimization (TLBO) algorithm was developed in 2011 by Rao et al. [19] based on the classical school learning process. TLBO consists of two stages consisting of the effect of a teacher on learners, and the influence of learners on each other. In this algorithm the initial population is selected randomly comprising of students or learners. The smartest student with the best objective function is assigned as the teacher in each iteration. Students are updated iteratively to search the optimum within two phases: based on the knowledge transfer from the teacher (teacher phase) and from interaction with other students (learner phase). In TLBO the performance of the class in learning or the performance of teacher in teaching is considered as a normal distribution of marks obtained by the students. TLBO improves other students in the teacher phase by using the difference between the teacher's knowledge and the average knowledge of all the students. The knowledge of each student is obtained based on the position taken place by that student in the search space. In a class, students also improve themselves via interacting with each other after the teaching is completed by the teacher. In the learner phase, TLBO improves each student by the knowledge interaction between that student and another randomly selected one. Some modifications on TLBO can be found in Refs. [20–22].

2.1.4 Imperialist Competitive algorithm

The Imperialist Competitive Algorithm (ICA) is a socio-politically motivated optimization algorithm proposed by Atashpaz-Gargari and his co-workers [23]. Like other population-based meta-heuristics, ICA starts with a set of random initial candidate solutions, as the initial countries. A specific number of best countries are considered as emperors. The emperors take a number of remaining countries as their colonies based on competency. Therefore, the countries are categorized into emperors and colonies and collectively form empires. ICA has two main mechanisms which are the basis of the ICA: improving the colonies of each empire by intrinsic learning of colonies from their emperor and imperialistic competitions among empires. First one results in powering empires themselves. In this way each colony has opportunity to take the role of emperor of that empire. During imperialistic competitions among empires, weakest empires lose their weakest colonies, and powerful empires take possession of them until the weakest empire collapses. Colonies improvements and imperialistic competitions direct the search process toward the powerful imperialists or the optimum points. These mechanisms are repeated in the cyclic body of the algorithm in succession to satisfy a stopping criteria with the aim of collapsing all the empires except the most powerful one which will have all the countries under its control.

2.1.5 Cuckoo Search algorithm

Cuckoo Search (CS) algorithm was developed by Yang and Deb [24] as an efficient population-based meta-heuristic inspired by the behavior of some cuckoo species. Cuckoos are fascinating birds because of their aggressive reproduction strategy. These species lay their eggs in the nests of other host birds. The host takes care of the eggs presuming that the eggs are its own. However, some of host birds are able to combat with this parasites behavior of cuckoos and throw out the discovered alien eggs or build their new nests in new locations. All the nests or eggs whether they belong to the cuckoos or host birds represent the candidate solutions in the search space. Cuckoos and host birds try to breed their own generation. In the cyclic body of the algorithm, two sequential search phases are performed by cuckoos and host birds. Firstly, cuckoos produce the eggs. In this phase eggs are produced by guiding the current solutions toward the best known solution. Then these new eggs are intruded to the nests of host birds based on the replacement strategy. After cuckoo breeding, it turns to the host birds. If a cuckoo's egg is very similar to a host's

egg, then this cuckoo's egg is less likely to be discovered. In this phase host birds discover a fraction of alien eggs and update them by adding them a random permutation-based step size. Based on the replacement strategy, host bird replaces the produced egg with the current one. These two search phases are repeated in the cyclic body of the algorithm until reaching to a stopping criterion.

2.1.6 Charged System Search algorithm

Charged System Search (CSS) algorithm was developed by Kaveh and Talatahari [25, 26] as an efficient population-based meta-heuristic using some principles from physics and mechanics. CSS employs the Newtonian laws of mechanics and the Coulomb laws from electrostatics. In this algorithm each charged particle is an agent with a predetermined radius. The magnitudes of the charge of the particles specify their quality. Each particle creates an electric field, exerting a force on other electrically charged objects. Therefore, charged particles can affect each other based on their fitness values and their separation distance. The quantity of the resultant force is determined by using the electrostatics laws, and the quality of the movement is determined using Newtonian mechanics laws. In each iteration, transitions of particles can be induced by electric fields leading to particle-particle electrostatic interactions with the aim of attracting or repelling the particles toward the optimum position.

2.1.7 Ray Optimization algorithm

Kaveh and Khayatizad [27] developed the Ray Optimization (RO) algorithm as a novel population-based meta-heuristic conceptualized on Snell's light refraction law when light travels from a lighter medium to a darker medium. Based on Snell's light refraction law, when light moves from a medium to another, it refracts. Its direction changes in a way that gets closer to the normal vector when it passes from a lighter medium to a darker one. This physical behavior is the essence of the RO. The agents of RO are considered as beginning points of rays of light updated in the search space or traveled from a medium to another one based on Snell's light refraction law. Each ray of light is a vector so that its beginning point is the previous position of the agent in the search space, its direction and length is the searching step size in the current iteration, and its end point is the current position of the agent achieved by adding the step size to the beginning point. The new position of agents is updated to explore the search space and converge to the global or near-global optimum. In fact, RO aims to improve

the quality of the solutions by refracting the rays toward the promising points obtained based on the best-known solution by each agent and all of them.

2.1.8 Tug of War Optimization algorithm

Inspired by the game tug of war, Kaveh and Zolghadr [28] developed a population-based meta-heuristic algorithm denoted as Tug of War Optimization (TWO) algorithm. TWO considers each candidate solution as a team participating in a series of rope pulling competitions. The teams exert pulling forces on each other based on the quality of the solutions they represent. The competing teams move to their new positions according to Newtonian laws of mechanics. TWO starts from a set of randomly generated initial candidate solutions. Each candidate solution is considered as a team, and all population form a league. The weight of teams is determined based on the quality of the corresponding solutions, and the amount of pulling force that a team can exert on the rope is assumed to be proportional to its weight. Naturally, the opposing team will have to maintain at least the same amount of force in order to sustain its grip on the rope. The lighter team accelerates toward the heavier team, and this forms the convergence operator of the TWO. In each iteration of the algorithm, the league is updated by a series of team-team rope pulling competitions with the aim of attracting teams toward the optimum position.

2.1.9 Water Evaporation Optimization algorithm

Inspired by evaporation of a tiny amount of water molecules on the solid surface with different wettability, Kaveh and Bakhshpoori [29] developed a novel meta-heuristic called Water Evaporation Optimization (WEO). WEO considers water molecules as algorithm individuals. Solid surface or substrate with variable wettability is reflected as the search space. Decreasing the surface wettability (substrate changed from hydrophilicity to hydrophobicity) reforms the water aggregation from a monolayer to a sessile droplet. Such a behavior is consistent with how the layout of individuals changes to each other as the algorithm progresses. Decreasing wettability of the surface can represent the decrease of objective function for a minimizing optimization problem. Evaporation flux rate of the water molecules is considered as the most appropriate measure for updating the individuals which its pattern of change is in good agreement with the local and global search ability of the algorithm and can help WEO to have significantly well-converged behavior and simple algorithmic structure.

2.1.10 Vibrating Particles System algorithm

Vibrating Particles System (VPS) algorithm is a meta-heuristic search algorithm suggested by Kaveh and Ilchi Ghazaan [30]. This algorithm is motivated by the free vibration of systems with single degree of freedom having a viscous damper. Similar to other population-based meta-heuristics, VPS starts with a random set of initial solutions and considers them as the free vibrated single degree of freedom systems with viscous damper. For under-damped conditions, each free vibrated system or vibrating particle oscillates and returns to its equilibrium state. As the optimization process proceeds, using the combination of randomness and exploitation of the obtained results, VPS iteratively improves the quality of the particles by oscillating them toward the equilibrium position. The equilibrium position of each particle is considered as three parts, the best position achieved so far across the whole population (HP), a good particle (GP), and a bad particle (BP). In this way the main features of the VPS consists of three essential concepts, self-adaptation (particle moves toward HB), cooperation (the GP and BP, that are selected from particles themselves, can influence the new position of the particles), and competition (the influence of GP being higher than that of BP).

2.1.11 Cyclical Parthenogenesis algorithm

Cyclical Parthenogenesis Algorithm (CPA) was developed by Kaveh and Zolghadr [31]. CPA is inspired by social behavior and reproduction of zoological species like aphids. In this algorithm each candidate solution is considered as an aphid and groups them into a number of colonies with identical number of aphids each inhabiting a host plant. Each colony tries to improve the quality of its aphids iteratively by reproduction mechanisms with and without mating and by a chance to get merit from other colonies using information exchange mechanism. The role (female or male) of each aphid in each colony can be decided depending on the quality. Each colony reproduces independently with the aim of improving the position of its aphids in the search space. On the other hand, to prevent the reproduction of colonies in an independent manner, benefiting the winged aphids, colonies can exchange a level of information between themselves. Colony improvements exchange between them and information are repeated in the cyclic body of the algorithm to satisfy the stopping criterions aiming to direct each colony toward a better position in the search space.

2.2 Optimization problem definition

The purpose of this research is to introduce a framework to optimum design of the reinforced concrete cantilever retaining walls shown in Fig. 1. The optimum design problem is performed under both static and dynamic loading conditions. The design under dynamic loading condition is performed for different values of horizontal and vertical acceleration coefficients as shown in Table 1. The specified parameters of the optimization problem, including soil parameters, design parameters, and retaining wall parameters, are listed in Table 2.

2.2.1 Design variables of the optimization problem

Continuous design variables of the cantilever retaining wall are shown in Fig. 2. The design variables are dimensions of the cantilever retaining wall. The design variables consist of the thickness of top stem (T_1), the thickness of key and stem (T_2), the toe width (T_3), the heel width (T_4), the height of top stem (T_5), the footing thickness (T_6), and the key depth (T_7). The upper and lower bounds of the design variables are presented in Table 3.

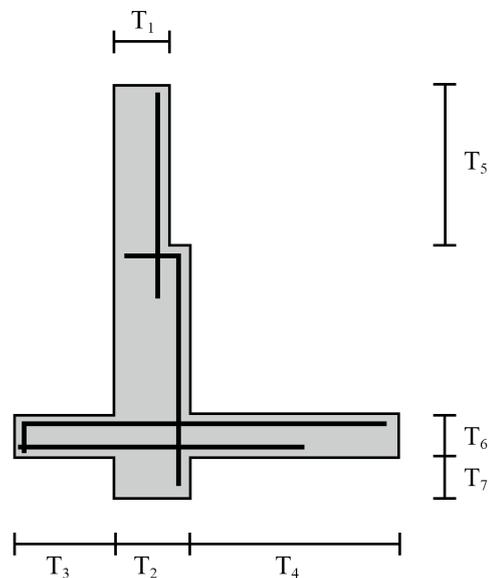


Fig. 1 Schematic view of the reinforced concrete cantilever retaining wall

Table 1 Horizontal and vertical acceleration coefficients

Case number	Horizontal acceleration coefficient (k_h)	Vertical acceleration coefficient (k_v)
Case 1	0	0
Case 2	0.15	0
Case 3	0.15	0.15
Case 4	0.15	0.075

Table 2 Specified parameters of the optimization problem

Parameter	Symbol	Unit	Value
Surcharge load	W_s	kN/m ²	10
Height of stem	H	m	6.1
Allowable soil pressure	q_a	kN/m ²	300
Concrete cover	d_c	cm	5
Unit weight of concrete	γ_c	kN/m ³	24
Unit weight of reinforcement steel	γ_s	kN/m ³	78
Unit weight of backfill soil	γ_b	kN/m ³	22
Internal friction angle of soil	θ	°	35
Base friction coefficient	μ	-	tan(θ)
Height of soil in front of wall	h_p	m	0
Yield strength of rebar	f_c	MPa	300
Yield strength of concrete	f_y	MPa	25
Design load factor	γ	-	1.6

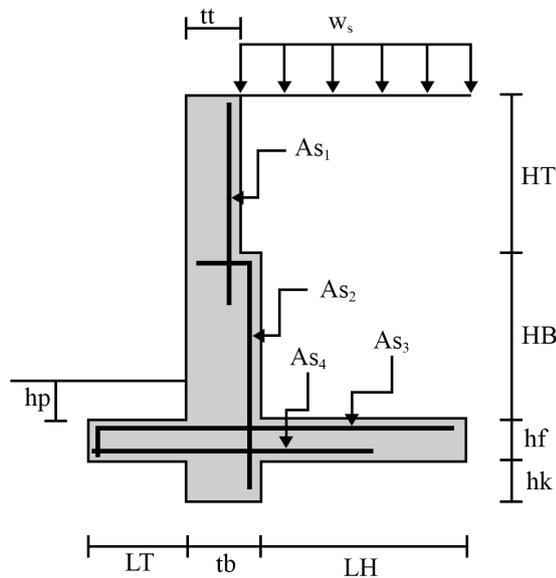


Fig. 2 Design variables of the reinforced concrete cantilever retaining wall

2.2.2 Objective function of the optimization problem

In this research, it is tried to achieve an optimum design for the cantilever retaining wall by minimizing its cost function. Therefore, the objective function can be expressed as equation Eq. (1). Therefore, to formulate the problem of minimizing the cost function of a cantilever retaining wall subject to stability and strength constraints, we have:

Minimize:

$$\bar{Q} = V_{Conc} + \frac{(C_3 + C_4)}{(C_1 + C_2)} \times W_{Steel} \tag{1}$$

Subject to:

$$FS_{(Overturning)} \geq 1.5 \tag{2}$$

$$FS_{(Sliding)} \geq 1.5 \tag{3}$$

$$FS_{(Bearing Capacity)} \geq 2 \tag{4}$$

$$M_u / (\phi_b M_n) \leq 1 \tag{5}$$

$$V_u / (\phi_v V_n) \leq 1 \tag{6}$$

In the above equations, V_{Conc} and W_{Steel} are volume of the concrete per unit of length and weight of the reinforcement steel per unit of length, respectively (m³/m and kg/m). \bar{Q} is the cost function of the cantilever retaining wall. In addition, C_1 and C_2 are cost of the concrete and steel per unit of length, respectively (\$/m). C_3 and C_4 are cost of the concreting and reinforcing per unit of length, respectively (\$/m). Furthermore, $FS_{(Overturning)}$ and $FS_{(Sliding)}$ are the factors of safety with respect to overturning about the toe and sliding, respectively. $FS_{(Bearing Capacity)}$ is the factor of safety against bearing capacity failure. M_u and V_u are maximum flexural moment and maximum shear force, respectively. In addition, ϕ_b and ϕ_v are strength reduction factors for flexure and shear, respectively. M_n and V_n are nominal flexural resistance and nominal shear resistance, respectively. The factors of safety against sliding and overturning under dynamic loading condition can be reduced to 75 % of these factors under static loading condition. Furthermore, the value of allowable soil pressure (q_a) used for static loading condition can increase by 33 % for dynamic loading condition [32]. Equations (4) and (5) must be checked for all critical sections of the cantilever retaining wall. The value of $\frac{(C_3 + C_4)}{(C_1 + C_2)}$ parameter depends on different things (e.g. country, time, etc.); but experience has shown that its value ranges from 0.035 to 0.045. In this paper, similar to Kaveh and Jafarpour Laien [10], the value of this parameter is assumed to be 0.04.

Table 3 Upper and lower bounds of the design variables

Design variable	Thickness of top stem	Thickness of key and stem	Toe width	Heel width	Height of top stem	Footing thickness	Key depth
	(T ₁)	(T ₂)	(T ₃)	(T ₄)	(T ₅)	(T ₆)	(T ₇)
Lower bound (m)	0.3	0.3	0.45	1.8	1.5	0.3	0.2
Upper bound (m)	0.6	0.6	1.2	3	6.1	0.9	0.9

Table 5 Optimum designs at different stages of optimization (Coulomb method)

No. of analyses	Cost function										
	ABC	BB-BC	CPA	CS	CSS	ICA	RO	TLBO	TWO	VPS	WEO
1000	7.136	6.984	7.012	7.303	7.105	7.465	7.024	7.055	7.147	6.989	7.247
2000	6.967	6.957	6.972	7.004	7.003	7.272	7.000	6.946	7.065	6.942	7.072
3000	6.958	6.950	6.953	6.955	6.983	6.980	6.968	6.943	6.994	6.941	7.012
4000	6.954	6.947	6.947	6.952	6.983	6.966	6.968	6.942	6.962	6.941	6.966
5000	6.946	6.946	6.946	6.951	6.970	6.961	6.968	6.942	6.957	6.941	6.963

Table 6 Optimum designs utilizing Rankine method

Design variable	Algorithms										
	ABC	BB-BC	CPA	CS	CSS	ICA	RO	TLBO	TWO	VPS	WEO
T ₁ (cm)	30.00	30.02	30.00	30.00	30.00	30.00	30.00	30.00	30.02	30.00	30.00
T ₂ (cm)	49.54	52.35	52.17	51.86	50.39	52.29	51.15	51.63	51.81	50.77	51.18
T ₃ (cm)	117.84	117.20	117.33	117.45	117.79	117.39	117.19	117.19	117.31	117.33	117.52
T ₄ (cm)	241.74	241.21	240.79	240.90	241.78	240.52	242.29	241.65	241.30	242.00	241.90
T ₅ (cm)	313.66	313.81	313.65	314.28	313.65	313.64	308.74	313.64	315.06	313.65	312.79
T ₆ (cm)	48.39	48.22	48.25	48.35	48.33	48.27	48.29	48.21	48.26	48.25	48.25
T ₇ (cm)	20.00	20.27	20.00	20.00	20.00	20.00	20.00	20.00	20.00	20.00	20.00
A _{s1} (cm ² /m)	11.20	11.21	11.20	11.26	11.20	11.20	11.20	11.20	11.33	11.20	11.20
A _{s2} (cm ² /m)	40.00	37.26	37.43	37.71	39.13	37.32	38.38	37.93	37.76	38.75	38.35
A _{s3} (cm ² /m)	19.78	19.70	19.72	19.77	19.75	19.73	19.74	19.70	19.72	19.72	19.72
A _{s4} (cm ² /m)	19.78	19.70	19.72	19.77	19.75	19.73	19.86	19.70	19.72	19.72	19.76
Best cost	7.376	7.375	7.373	7.379	7.374	7.374	7.390	7.370	7.375	7.370	7.375
Average cost	7.850	7.430	7.583	7.688	7.476	7.684	7.691	7.448	7.709	7.553	7.589
Std. Dev.	1.412	0.185	0.928	0.942	0.572	1.379	1.324	0.288	1.330	1.095	0.749
No. of analyses	5000	5000	5000	5000	5000	5000	5000	5000	5000	5000	5000

Table 7 Optimum designs at different stages of optimization (Rankine method)

No. of analyses	Cost function										
	ABC	BB-BC	CPA	CS	CSS	ICA	RO	TLBO	TWO	VPS	WEO
1000	7.976	7.399	7.567	7.761	7.424	7.519	7.481	7.405	7.576	7.398	7.539
2000	7.618	7.383	7.376	7.556	7.380	7.382	7.455	7.375	7.449	7.374	7.403
3000	7.420	7.380	7.373	7.452	7.377	7.375	7.424	7.371	7.406	7.371	7.394
4000	7.380	7.375	7.373	7.384	7.374	7.374	7.400	7.370	7.383	7.370	7.390
5000	7.376	7.375	7.373	7.379	7.374	7.374	7.390	7.370	7.375	7.370	7.375

Table 8 Optimum designs utilizing Mononobe-Okabe method (Case 1)

Design variable	Algorithms										
	ABC	BB-BC	CPA	CS	CSS	ICA	RO	TLBO	TWO	VPS	WEO
T ₁ (cm)	30.00	30.00	30.00	30.01	30.09	30.00	30.00	30.00	30.00	30.00	30.00
T ₂ (cm)	49.79	52.30	50.31	49.16	48.52	48.57	51.49	50.20	49.92	50.61	51.31
T ₃ (cm)	69.78	69.22	69.59	69.84	70.11	70.08	69.59	69.61	69.66	69.54	69.42
T ₄ (cm)	225.04	224.29	225.00	225.57	225.38	225.37	224.42	225.06	225.18	224.87	224.56
T ₅ (cm)	337.38	336.34	337.38	338.98	339.47	337.37	335.60	337.38	337.91	337.40	337.48
T ₆ (cm)	39.08	38.91	39.01	39.12	39.22	39.20	38.98	39.02	39.04	38.99	38.94
T ₇ (cm)	20.00	20.00	20.00	20.00	20.15	20.00	20.00	20.00	20.01	20.04	20.04

Design variable	Algorithms										
	ABC	BB-BC	CPA	CS	CSS	ICA	RO	TLBO	TWO	VPS	WEO
A _{s1} (cm ² /m)	11.20	11.20	11.20	11.35	11.35	11.20	11.20	11.20	11.25	11.20	11.21
A _{s2} (cm ² /m)	33.54	31.52	33.10	34.09	34.67	34.63	32.15	33.20	33.43	32.86	32.29
A _{s3} (cm ² /m)	15.44	15.40	15.40	15.46	15.50	15.50	15.39	15.41	15.42	15.40	15.37
A _{s4} (cm ² /m)	15.44	15.36	15.40	15.46	15.50	15.50	15.39	15.41	15.42	15.39	15.37
Best cost	6.222	6.229	6.219	6.225	6.232	6.226	6.228	6.220	6.220	6.220	6.220
Average cost	6.335	6.286	6.317	6.330	6.368	6.380	6.322	6.278	6.341	6.334	6.433
Std. Dev.	0.373	0.207	0.267	0.258	0.304	0.344	0.270	0.177	0.304	0.281	0.530
No. of analyses	5000	5000	5000	5000	5000	5000	5000	5000	5000	5000	5000

Table 9 Optimum designs at different stages of optimization (Mononobe-Okabe method, Case 1)

No. of analyses	Cost function										
	ABC	BB-BC	CPA	CS	CSS	ICA	RO	TLBO	TWO	VPS	WEO
1000	6.481	6.249	6.301	6.393	6.394	6.463	6.310	6.258	6.365	6.289	6.410
2000	6.240	6.240	6.233	6.253	6.235	6.250	6.269	6.231	6.242	6.239	6.247
3000	6.223	6.231	6.220	6.237	6.233	6.230	6.254	6.222	6.224	6.229	6.227
4000	6.222	6.230	6.219	6.230	6.233	6.226	6.228	6.220	6.221	6.224	6.221
5000	6.222	6.229	6.219	6.225	6.232	6.226	6.228	6.220	6.220	6.220	6.220

Table 10 Optimum designs utilizing Mononobe-Okabe method (Case 2)

Design variable	Algorithms										
	ABC	BB-BC	CPA	CS	CSS	ICA	RO	TLBO	TWO	VPS	WEO
T ₁ (cm)	30.00	30.00	30.00	30.01	30.00	30.24	30.01	30.02	30.02	30.00	30.00
T ₂ (cm)	58.15	59.49	58.79	59.07	59.39	56.42	60.000	59.10	57.85	58.62	60.00
T ₃ (cm)	103.63	103.52	103.44	103.42	102.93	104.53	102.81	103.38	103.66	103.49	103.19
T ₄ (cm)	290.47	289.95	290.32	290.21	291.25	291.37	290.91	290.16	290.71	290.40	289.78
T ₅ (cm)	279.10	278.43	279.12	280.05	279.12	283.45	287.83	279.20	280.89	279.23	279.68
T ₆ (cm)	54.60	54.52	54.52	54.53	54.33	54.82	54.25	54.50	54.61	54.54	54.43
T ₇ (cm)	20.00	20.05	20.00	20.00	21.22	20.03	20.20	20.00	20.00	20.01	20.03
A _{s1} (cm ² /m)	11.20	11.20	11.20	11.31	11.20	11.58	12.22	11.21	11.40	11.21	11.27
A _{s2} (cm ² /m)	48.18	46.80	47.51	47.23	46.91	50.10	46.30	47.19	48.50	47.69	46.30
A _{s3} (cm ² /m)	22.68	22.64	22.64	22.65	23.19	22.78	23.19	22.63	22.69	22.65	22.62
A _{s4} (cm ² /m)	22.68	22.64	22.64	22.65	22.56	22.78	22.52	22.63	22.69	22.65	22.60
Best cost	8.679	8.683	8.676	8.679	8.697	8.704	8.696	8.677	8.682	8.677	8.677
Average cost	8.893	8.756	8.810	8.993	8.819	9.049	8.804	8.961	9.009	8.803	8.782
Std. Dev.	0.851	0.216	0.805	1.047	0.660	0.995	0.362	1.160	1.171	0.374	0.464
No. of analyses	5000	5000	5000	5000	5000	5000	5000	5000	5000	5000	5000

Table 11 Optimum designs at different stages of optimization (Mononobe-Okabe method, Case 2)

No. of analyses	Cost function										
	ABC	BB-BC	CPA	CS	CSS	ICA	RO	TLBO	TWO	VPS	WEO
1000	8.713	8.721	8.731	9.070	8.768	9.018	8.731	8.980	9.140	8.754	8.713
2000	8.681	8.688	8.707	8.777	8.730	8.922	8.705	8.691	8.769	8.698	8.704
3000	8.680	8.685	8.680	8.700	8.710	8.748	8.705	8.679	8.700	8.678	8.680
4000	8.680	8.683	8.677	8.685	8.697	8.718	8.702	8.677	8.690	8.677	8.678
5000	8.679	8.683	8.676	8.679	8.697	8.704	8.696	8.677	8.682	8.677	8.677

Table 15 Optimum designs at different stages of optimization (Mononobe-Okabe method, Case 4)

No. of analyses	Cost function											
	ABC	BB-BC	CPA	CS	CSS	ICA	RO	TLBO	TWO	VPS	WEO	
1000	8.652	8.277	8.280	8.312	8.283	8.462	8.394	8.297	8.601	8.355	8.789	
2000	8.299	8.277	8.251	8.263	8.283	8.297	8.293	8.245	8.331	8.252	8.449	
3000	8.250	8.254	8.249	8.251	8.282	8.268	8.253	8.241	8.260	8.245	8.322	
4000	8.243	8.249	8.247	8.246	8.255	8.258	8.253	8.241	8.255	8.243	8.261	
5000	8.241	8.247	8.246	8.245	8.254	8.253	8.253	8.241	8.249	8.242	8.260	

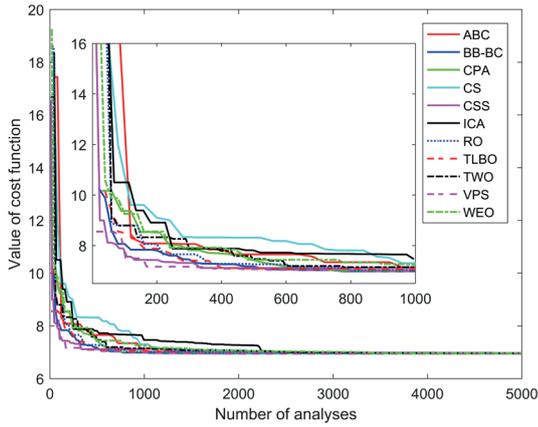


Fig. 3 Convergence histories utilizing Coulomb method

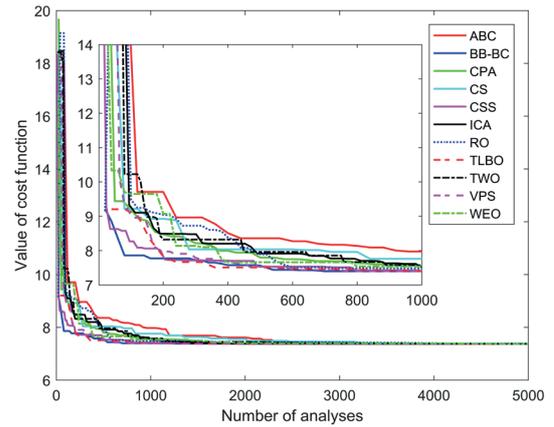


Fig. 4 Convergence histories utilizing Rankine method

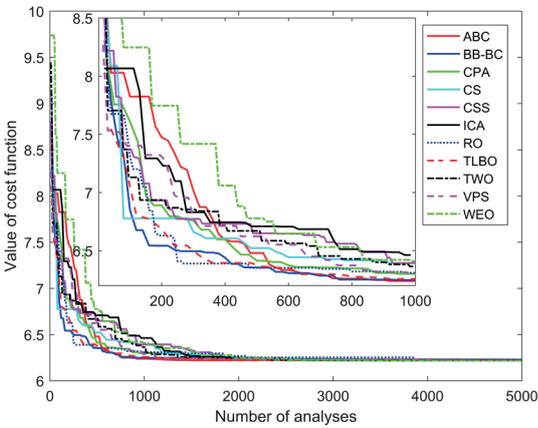


Fig. 5 Convergence histories utilizing Mononobe-Okabe method (Case 1)

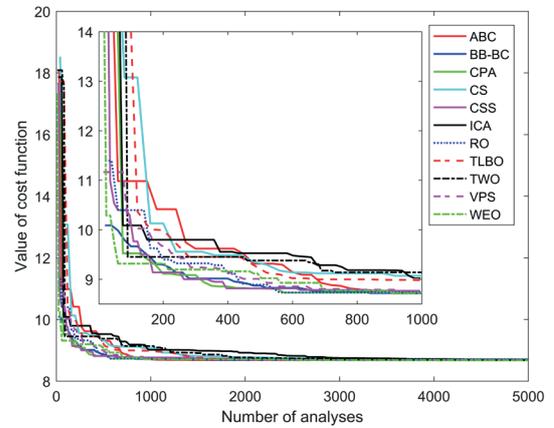


Fig. 6 Convergence histories utilizing Mononobe-Okabe method (Case 2)

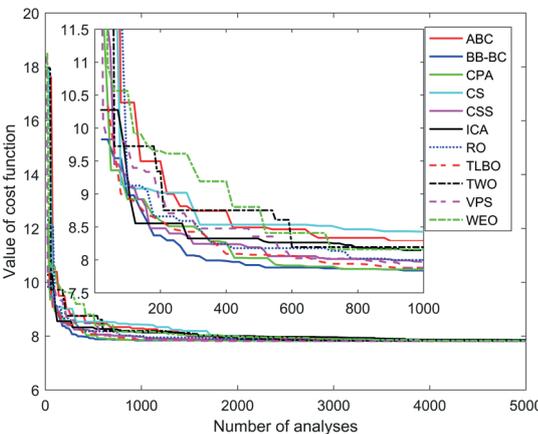


Fig. 7 Convergence histories utilizing Mononobe-Okabe method (Case 3)

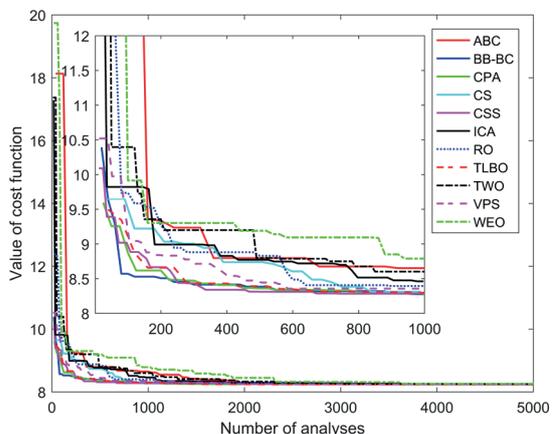


Fig. 8 Convergence histories utilizing Mononobe-Okabe method (Case 4)

4 Conclusions

In this paper, eleven population-based meta-heuristic algorithms are employed for optimum design of reinforced concrete cantilever retaining walls under static and dynamic loading conditions. The algorithms consist of Artificial Bee Colony, Big Bang-Big Crunch, Teaching-Learning-Based Optimization, Imperialist Competitive Algorithm, Cuckoo Search, Charged System Search, Ray Optimization, Tug of War Optimization, Water Evaporation Optimization, Vibrating Particles System, and Cyclical Parthenogenesis Algorithm. The objective of optimization is to minimize the cost of the cantilever retaining wall subject to stability and strength constraints. The design is based on ACI 318-05. Coulomb and Rankine methods are used to determine lateral earth pressures under static loading condition, and the Mononobe-Okabe method is employed for dynamic loading condition. The results confirm that the cantilever retaining

walls designed based on the Coulomb method is lighter compared to those designed based on the Rankine method. The results of Mononobe-Okabe method indicate that the horizontal acceleration coefficient has a direct effect on the cost of the cantilever retaining wall, while the vertical acceleration coefficient has a reverse effect on it. In general, the optimization results confirm efficiency and accuracy of the utilized algorithms. All the utilized algorithms can converge to high-quality optimum designs rapidly. The results indicate superiority of the BB-BC, TLBO, CPA, VPS, and CSS algorithms in both aspects of convergence rate and accuracy compared to other employed algorithms. The convergence histories of the mentioned algorithms indicate that they have a close performance.

Compliance with ethical standards

Conflict of interest: No potential conflict of interest was reported by the authors.

References

- [1] Yepes, V., Alcalá, J., Perea, C., Gonzalez-Vidoso, F. "A parametric study of optimum earth-retaining walls by simulated annealing", *Engineering Structures*, 30(3), pp. 821–830, 2018. <https://doi.org/10.1016/j.engstruct.2007.05.023>
- [2] Kaveh, A., Shakouri Mahmud Abadi, A. "Harmony search based algorithms for the optimum cost design of reinforced concrete cantilever retaining walls", *International Journal of Civil Engineering*, 9(1), pp. 1–8, 2011. [online] Available at: <http://ijce.iust.ac.ir/article-1-292-en.html> [Accessed: 14 December 2019]
- [3] Camp, C. V., Akin, A. "Design of Retaining Walls Using Big Bang-Big Crunch Optimization", *Journal of Structural Engineering*, 138(3), pp. 438–448, 2012. [https://doi.org/10.1061/\(ASCE\)ST.1943-541X.0000461](https://doi.org/10.1061/(ASCE)ST.1943-541X.0000461)
- [4] Khajezadeh, M., Taha, M. R., Eslami, M. "Efficient gravitational search algorithm for optimum design of retaining walls", *Structural Engineering Mechanics*, 45(1), pp. 111–127, 2013. <https://doi.org/10.12989/sem.2013.45.1.111>
- [5] Kaveh, A., Behnam, A. F. "Charged System Search Algorithm for the Optimum Cost Design of Reinforced Concrete Cantilever Retaining Walls", *Arabian Journal of Science and Engineering*, 38(3), pp. 563–570, 2013. <https://doi.org/10.1007/s13369-012-0332-0>
- [6] Kaveh, A., Khayatazad, M. "Optimal design of cantilever retaining walls using ray optimization method", *Iranian Journal of Science and Technology-Transaction on Civil Engineering*, 38(C1), pp. 261–274, 2014.
- [7] Kaveh, A., Soleimani, N. "CBO and DPSO for optimum design of reinforced concrete cantilever retaining walls", *Asian Journal of Civil Engineering*, 16(6), pp. 751–774 2015. [online] Available at: <https://ajce.bhrc.ac.ir/Portals/25/PropertyAgent/2905/Files/5784/751.pdf> [Accessed: 14 December 2019]
- [8] Kaveh, A., Farhoudi, N. "Dolphin Echolocation Optimization for Design of Cantilever Retaining Walls", *Asian Journal of Civil Engineering*, 17(2), pp. 193–211, 2016. [online] Available at: <https://ajce.bhrc.ac.ir/Portals/25/PropertyAgent/2905/Files/6842/193.pdf> [Accessed: 14 December 2019]
- [9] Kaveh, A., Jafarpour Laien, D. "Optimal design of reinforced concrete cantilever retaining walls using CBO, ECBO and VPS algorithms", *Asian Journal of Civil Engineering*, 18(4), pp. 657–671, 2017. [online] Available at: <https://ajce.bhrc.ac.ir/Portals/25/PropertyAgent/2905/Files/8806/657.pdf> [Accessed: 14 December 2019]
- [10] Kaveh, A. "Applications of Metaheuristic Optimization Algorithms in Civil Engineering", Springer, Cham, Switzerland, 2017. <https://doi.org/10.1007/978-3-319-48012-1>
- [11] Kaveh, A. "Advances in Metaheuristic Algorithms for Optimal Design of Structures", Springer, Cham, Switzerland, 2017. <https://doi.org/10.1007/978-3-319-46173-1>
- [12] Kaveh, A., Sabeti, S. "Optimal Design of Jacket Supporting Structures for Offshore Wind Turbines Using CBO and ECBO Algorithms", *Periodica Polytechnica Civil Engineering*, 62(3), pp. 545–554, 2018. <https://doi.org/10.3311/PPci.11651>
- [13] Kaveh, A., Farhadmanesh, M. "Optimal Seismic Design of Steel Plate Shear Walls Using Metaheuristic Algorithms", *Periodica Polytechnica Civil Engineering*, 63(1), pp. 1–17, 2019. <https://doi.org/10.3311/PPci.12119>
- [14] Kaveh, A., Kabir, M. Z., Bohlool, M. "Optimal Design of Multi-Span Pitched Roof Frames with Tapered Members", *Periodica Polytechnica Civil Engineering*, 63(1), pp. 77–86, 2019. <https://doi.org/10.3311/PPci.13107>
- [15] Kaveh, A., Bakhshpoori, T. "Metaheuristics: Outlines, MATLAB Codes and Examples", Springer, Cham, Switzerland, 2019. <https://doi.org/10.1007/978-3-030-04067-3>

- [16] ACI Committee "Building Code Requirements for Structural Concrete (ACI 318-05) and Commentary (ACI 318R-05)", American Concrete Institute, Farmington Hills, MI, USA, 2005.
- [17] Karaboga, D. "An idea based on honey bee swarm for numerical optimization", Department of Computer Engineering, Faculty of Engineering, Erciyes University, Erciyes, Turkey, TR06, October, 2005. [online] Available at: https://abc.erciyes.edu.tr/pub/tr06_2005.pdf [Accessed: 14 December 2019]
- [18] Erol, O. K., Eksin, I. "New optimization method: Big Bang-Big Crunch", *Advances in Engineering Software*, 37(2), pp. 106–111, 2006.
<https://doi.org/10.1016/j.advengsoft.2005.04.005>
- [19] Rao, R. V., Savsani, V. J., Vakharia, D. P. "Teaching-learning-based optimization: A novel method for constrained mechanical design optimization problems", *Computer-Aided Design*, 43(3), pp. 303–315, 2011.
<https://doi.org/10.1016/j.cad.2010.12.015>
- [20] Savsani, V. J., Tejani, G. G., Patel, V. K. "Truss topology optimization with static and dynamic constraints using modified sub-population teaching-learning-based optimization", *Engineering Optimization*, 48(11), pp. 1990–2006, 2016.
<https://doi.org/10.1080/0305215X.2016.1150468>
- [21] Tejani, G. G., Savsani, V. J., Patel, V. K., Bureerat, S. "Topology, shape, and size optimization of truss structures using modified teaching-learning based optimization", *Advances in Computational Design*, 2(4), pp. 313–331, 2017.
<https://doi.org/10.12989/acd.2017.2.4.313>
- [22] Tejani, G. G., Savsani, V. J., Patel, V. K. "Modified sub-population teaching-learning-based optimization for design of truss structures with natural frequency constraints", *Mechanics Based Design of Structures and Machines*, 44(4), pp. 495–513, 2016.
<https://doi.org/10.1080/15397734.2015.1124023>
- [23] Atashpaz-Gargari, E., Lucas, C. "Imperialist competitive algorithm: An algorithm for optimization inspired by imperialistic competition", In: 2007 IEEE Congress on Evolutionary Computation, Singapore, Singapore, 2007, pp. 25–28.
<https://doi.org/10.1109/CEC.2007.4425083>
- [24] Yang, X. S., Deb, S. "Engineering optimisation by cuckoo search", *International Journal of Mathematical Modelling and Numerical Optimisation (IJMMNO)*, 1(4), pp. 330–343, 2010.
<https://doi.org/10.1504/IJMMNO.2010.035430>
- [25] Kaveh, A., Talatahari, S. "A novel heuristic optimization: charged system search algorithm", *Acta Mechanica*, 213(3–4), pp. 267–289, 2010.
<https://doi.org/10.1007/s00707-009-0270-4>
- [26] Kaveh, A., Talatahari, S. "Optimal design of skeletal structures via the charged system search algorithm", *Structural Multidisciplinary Optimization*, 41(6), pp. 893–911, 2010.
<https://doi.org/10.1007/s00158-009-0462-5>
- [27] Kaveh, A., Khayatizad, M. "A new meta-heuristic method: Ray Optimization", *Computers & Structures*, 112–113, pp. 283–294, 2012.
<https://doi.org/10.1016/j.compstruc.2012.09.003>
- [28] Kaveh, A., Zolghadr, A. "A novel meta-heuristic algorithm: tug of war optimization", *International Journal of Optimization in Civil Engineering*, 6(4), pp. 469–492, 2016. [online] Available at: <http://ijocce.iust.ac.ir/article-1-265-en.html> [Accessed: 14 December 2019]
- [29] Kaveh, A., Bakhshpoori, T. "Water Evaporation Optimization: A novel physically inspired optimization algorithm", *Computers and Structures*, 167, pp. 69–85, 2016.
<https://doi.org/10.1016/j.compstruc.2016.01.008>
- [30] Kaveh, A., Ilchi Ghazaan, M. "A new meta-heuristic algorithm: vibrating particles system", *Scientia Iranica: Transaction on Civil Engineering*, 24(2), pp. 551–566, 2017.
<https://doi.org/10.24200/sci.2017.2417>
- [31] Kaveh, A., Zolghadr, A. "Cyclical parthenogenesis algorithm: A new meta-heuristic algorithm", *Asian Journal of Civil Engineering (BHRC)*, 18(5), pp. 673–701, 2017. [online] Available at: <https://ajce.bhrc.ac.ir/Portals/25/PropertyAgent/2905/Files/8863/AJCE%20673.pdf> [Accessed: 14 December 2019]
- [32] AASHTO "Standard Specifications for Highway Bridges", American Association of State Highway and Transportation Officials, Washington, DC., USA, 2002.