Journal of Structural and Construction Engineering, 12(11), 2026, pp. 227-250

Journal of Structural and Construction Engineering

www.jsce.ir

Optimizing Reinforced Concrete Cantilever Retaining Walls Using Artificial Bee Colony Algorithm

Mehdi Shalchi Tousi[1*], Samane Laali[2], Payam Tarighi[1]

1. Assistant Professor, Faculty of Technology and Engineering, Ahlul Bayt International University, Tehran, Iran.
2. M.Sc., Department of Civil Engineering, University of Science and Culture, Tehran, Iran

ABSTRACT

The Artificial Bee Colony (ABC) Algorithm is a sophisticated optimization technique that is inspired by the intelligent behaviors of honey bee swarms. These behaviors, such as foraging and communication within complex social structures, serve as the foundation for the algorithm's effectiveness. In this paper, the ABC algorithm is utilized to optimize the design of reinforced cantilever concrete retaining walls, with the goal of minimizing both cost and weight. The results are compared to existing literature, demonstrating the success of the ABC algorithm in achieving the objectives. Furthermore, a comparison is conducted between the optimized design and a conventional manual design, revealing a significant reduction in cost and weight through optimization. Additionally, two types of reinforced concrete cantilever retaining walls-a T-shape wall with variable stem thickness and a standard T-shape wall-are presented and compared, considering their differing variables and constraints. These comparisons are made for two objective functions: the cost and weight of the wall. To further investigate the impact of initial parameters, such as the unit weight of soil and stem height, a sensitivity analysis is conducted. The robustness of the ABC algorithm in optimizing the cost and weight of reinforced concrete cantilever retaining walls is demonstrated by the results.

ARTICLE INFO

Receive Date: 07 January 2025
Revise Date: 14 April 2025
Accept Date: 06 May 2025

Keywords:
Retaining wall,
Bee Colony Algorithm,
Optimization,
Wall cost and weight,
T-shape walls

doi: 10.22065/jsce.2025.493295.3604

*Corresponding author: Mehdi Shalchi Tousi
Email address: mehdishalchi@abu.ac.ir

1- Introduction

The engineer's main goal is to reduce costs while ensuring safety. To achieve this, engineers have been using optimization methods to decrease the cost and weight of structures for a long time. This is especially important for large and essential structures in civil engineering, such as reinforced concrete cantilever retaining walls. Many researchers have studied the optimization of these walls using various methods. For example, Saribas and Erbatur [1] used nonlinear programming to optimize the wall for two objectives: cost and weight. They developed a computer program called RETOPT [1] for this purpose. They also conducted sensitivity analysis on initial parameters such as stem height, surcharge load, and soil unit weight. The results showed that the objective functions increased with an increase in stem height and surcharge, while they decreased with an increase in soil unit weight. Ceranic and Freyer [2] also optimized the cost of the wall using a simulated annealing algorithm. Other researchers have used different approaches such as target reliability [3], ant colony algorithm [5], foraging bacterial algorithm [6], genetic, simulated annealing, and particle swarm optimization algorithms [7], charged system search algorithm [8], intelligent water drops algorithm (IWDA) [9], Cuckoo Optimization Algorithm (COA) [10], and Gases Brownian Motion Algorithm (GBMOA) [11].

In the study by Kalmsi et al. [12], the Gray Wolf Optimization method is used to discuss the optimization of a concrete retaining wall. The purpose of their article is to design a low-weight cantilever reinforced concrete retaining wall with a shear key, utilizing the GWO algorithm. In the study by Romani et al. [13], a new discretization technique is developed based on reinforcement learning and transfer functions. The objective functions in this study are cost and CO_2 emissions. A comprehensive comparison is made with various metaheuristics and brute force methods, and the results consistently show that the use of S-shaped transfer functions leads to more robust outcomes.

In the study by Keivanian et al. [14], a fuzzy adaptive algorithm was used to identify sustainable, economical, and earthquake-resistant concrete cantilever retaining walls. The study considered 26 limitations for structural strengths and geotechnical stability, as well as 12 geometric variables, in order to handle the weights and forces resulting from earth pressures on the walls. The results demonstrate that the proposed algorithm can successfully achieve low-cost, low-weight, and low-CO_2 emission designs for reinforced concrete walls under 9 different seismic conditions, outperforming other design optimizers.

In the study by Tayfur [15], the use of the Teaching-Learning Based Optimization (TLBO) algorithm and an improved version (I-TLBO) with agents was explored for optimizing cantilever retaining walls. The design process focused on minimizing both the weight and cost of the wall as objectives.

In the study by Shenouda and Ali [16], a new model was implemented in MATLAB to optimize the design of cantilever retaining wall elements. This design model was coupled with the shuffled complex evolution algorithm, developed at the University of Arizona (SCE-UA). The results demonstrated that the use of the SCE-UA method yielded superior results compared to other algorithms.

In the study by Boukhatem et al. [17], a genetic algorithm was utilized to optimize retaining walls. The design variables, including geometric parameters such as the thickness of the veil and sole, as well as the lengths of the foot and heel, were incorporated into the problem formulation. The findings suggest that this algorithm is a dependable tool for addressing optimization challenges, particularly in the context of retaining wall design.

Recent studies on optimizing reinforced concrete cantilever retaining walls have leveraged artificial intelligence-based algorithms to enhance cost efficiency and structural performance.

Khajehzadeh et al. [18] proposed a Modified Particle Swarm Optimization (MPSO) algorithm for the cost-effective design of spread footings and retaining walls, integrating geotechnical and structural constraints. Their method outperformed standard PSO and PSOPC in convergence speed and accuracy, achieving up to 6% cost reduction in optimized designs. The study also highlighted the sensitivity of design outcomes to soil properties like friction angle and Young's modulus, demonstrating MPSO's robustness in handling complex engineering constraints. Camp and Akin [19] applied the Big Bang–Big Crunch (BB-BC) optimization to design cost-effective cantilever retaining walls, demonstrating its efficiency over traditional methods while meeting ACI 318-05 constraints. Their study highlighted key sensitivities in design parameters and the benefits of base shear keys. Kashani et al. [20] compared four multi-objective optimization techniques, including NSGA-II and MOPSO, to balance cost, weight, and safety factors. Gandomi et al. [21] investigated evolutionary algorithms like Differential Evolution (DE) and Biogeography-Based Optimization (BBO) for nonlinear constrained designs, showing BBO's superior performance. Aydogdu [22] introduced an enhanced BBO method with Levy flights for seismic load considerations, demonstrating its effectiveness in reducing construction costs while maintaining regulatory compliance. Molina-Moreno et al. [23] optimized buttressed earth-retaining walls using a hybrid harmony search algorithm, highlighting the impact of soil properties on cost-efficient structural design. Temur and Bekdaş [24] applied Teaching-Learning-Based Optimization (TLBO) to design cost-effective cantilever retaining walls, incorporating 29 geotechnical and structural constraints per ACI 318-05. Their method outperformed PSO, BB-BC, and Harmony Search in robustness and convergence speed, achieving up to 10% cost reduction. The study highlighted the nonlinear impact of backfill slope angles (>20°) and linear relationships between costs and surcharge loads or soil friction angles.

In this paper, the optimization of a concrete cantilever retaining wall for cost and weight is achieved using the Artificial Bee Colony (ABC) algorithm, as presented by Karaboga and Basturk [8] and Karaboga and Ozturk [18]. This algorithm is a swarm intelligence and population-based stochastic optimization algorithm. As mentioned earlier, the ABC algorithm is inspired by the intelligent foraging behavior of honey bee swarms, which are divided into three groups: worker bees, onlookers, and scouts. To evaluate the effectiveness of the ABC algorithm, it is compared to other literature, such as the study by Saribas and Erbatur [1]. Additionally, the optimized retaining wall is compared to a manually designed one to demonstrate the importance and impact of optimization. The paper also includes a section on parametric studies, where the optimization of two retaining walls is presented and compared. Furthermore, a sensitivity analysis is conducted to assess the influence of initial parameters on objective functions, including the stem height and unit weight of the soil.

2- Introduction of Artificial Bee Colony (ABC) Algorithm

In the ABC algorithm [19], the spatial configuration of a food source signifies a potential resolution to the optimization dilemma, while the quantity of nectar associated with a food source is indicative of the quality (fitness) of the corresponding resolution. The count of worker bees, alongside that of onlooker bees, is equivalent to the number of solutions present within the population. In the initial phase, the ABC algorithm generates a randomly allocated initial population P (C = 0) comprising SN solutions (food source positions), where SN represents the magnitude of worker bees or onlooker bees. Each solution x_i (i = 1, 2, ..., SN) is characterized as a D-dimensional vector, with D denoting the quantity of optimization parameters. Subsequent to initialization, the population of positions (solutions) undergoes iterative cycles, C = 1, 2, ..., MCN, of the search activities conducted by worker bees, onlooker bees, and scout bees. A worker bee modifies the position (solution) retained in her memory based on local information (visual information) and evaluates the nectar amount (fitness value) of the newly derived source (new solution). Should the nectar amount of the new source exceed that of the preceding one, the bee retains the new position in memory while discarding the old one; conversely, she maintains the previous position in her recollection. Upon completion of the search process by all worker bees, they disseminate the nectar information regarding the food sources along with their positional data to the onlooker bees. An onlooker bee assesses the nectar information obtained from all worker bees and selects a food source with a probability that correlates with its nectar amount. Similar to the worker bee, she modifies the position in her memory and evaluates the nectar amount of the prospective source. If the nectar of the new source surpasses that of the previous one, the bee retains the new position in memory and discards the old one. The principal steps of the algorithm are delineated as follows:

1: Initialize Population
2: **repeat**
3: Allocate the worker bees to their respective food sources
4: Allocate the onlooker bees to the food sources contingent upon their nectar amounts
5: Dispatch the scouts to the exploratory area for the identification of new food sources
6: Retain the optimal food source identified to date
7: **until** criteria are satisfied

In the Artificial Bee Colony (ABC) algorithm, each iteration of the search process is delineated into three distinct phases: the allocation of worker bees to their respective food sources followed by the assessment of their nectar quantities; subsequent to the dissemination of nectar information regarding the food sources, the selection of food source regions by the onlookers occurs, accompanied by the evaluation of the nectar amounts of these sources; finally, the identification of scout bees takes place, which are dispatched randomly to potential new food sources. During the initialization phase, a subset of food sources is randomly chosen by the bees, and their corresponding nectar amounts are ascertained. In the initial phase of the iteration, these bees return to the hive and convey the nectar information of the sources to the bees situated in the dance area. A bee that is stationed in the dance area and is poised to make a decision regarding the selection of a food source is designated as an onlooker, whereas the bee that visits the food source it had previously encountered is referred to as an worker bee. Following the exchange of information with onlookers, each worker bee proceeds to the food source area it had visited in the prior cycle, as this particular food source is retained in its memory, and subsequently selects a new food source utilizing visual information from the vicinity of the one stored in its memory, along with evaluating its nectar quantity. In the second phase, an onlooker selects a food source area based on the nectar information disseminated by the worker bees within the dance area. As the nectar quantity of a food source escalates, the likelihood of its selection correspondingly increases. Upon reaching the designated area, the onlooker selects a new food source in proximity to the one remembered, utilizing visual information analogous to that employed by the worker bees. The identification of the new food source is executed by the bees through a comparative analysis of food source locations based on visual cues. In the third phase of the iteration, when the nectar of a food source is deemed abandoned by the bees, a scout bee randomly identifies a new food source to supplant the one that has been abandoned. In our proposed model, during each cycle, at most one scout bee is permitted to venture outside in search of a new food source, while the quantities of employed and onlooker bees are maintained as equal. These three phases are reiterated through a predetermined number of iterations referred to as the Maximum Cycle Number (MCN), or until a specified termination criterion is fulfilled. An artificial onlooker bee selects a food source predicated upon the probability value associated with that particular food source, denoted as p_i, which is calculated by the subsequent expression (1):

$$p_i = \frac{fit_i}{\sum_{n=1}^{SN} fit_n} \quad (1)$$

where fit_i is the fitness value of the solution i which is proportional to the nectar amount of the food source in the position i and SN is the number of food sources which is equal to the number of worker bees or onlooker bees.
In order to produce a candidate food position from the old one in memory, the ABC uses the following expression (2):

$$v_{ij} = x_{ij} + \varphi_{ij}(x_{ij} - x_{kj}) \quad (2)$$

Where $k \in \{1, 2, \dots, SN\}$ and $j \in \{1, 2, \dots, D\}$ are randomly chosen indexes. Although k is determined randomly, it has to be different from i. φ_{ij} is a random number between [-1, 1]. It controls the production of neighbor food sources around x_{ij} and represents the comparison of two food positions visually by a bee. As can be seen from (2), as the difference between the parameters of the x_{ij} and x_{kj} decreases, the perturbation on the position x_{ij} gets decreased, too. Thus, as the search approaches the optimum solution in the search space, the step length is adaptively reduced. If a parameter value generated by this operation surpasses its established threshold, the parameter can be adjusted to a permissible value. In this study, the value of the parameter that exceeds its threshold is constrained to its maximum allowable value.
The food source from which the nectar is forsaken by the bees is substituted with a novel food source by the scout bees. In the Artificial Bee Colony (ABC) algorithm, this is represented by the random generation of a position and substitution with the forsaken one. In ABC, if a position cannot be enhanced further after a specified number of iterations, that food source is deemed to be abandoned. The specified number of iterations is a critical control parameter within the ABC algorithm, referred to as the "limit" for abandonment. Assume that the abandoned source is x_i and $j \in \{1, 2, \dots, D\}$, then the scout discovers a new food source to be replaced with x_i. This operation can be defined as in (3)

$$x_i^j = x_{min}^j + rand[0,1](x_{max}^j - x_{min}^j) \quad (3)$$

After each candidate food source position v_{ij} is generated and subsequently assessed by the artificial bee, its efficacy is juxtaposed with that of its predecessor. Should the novel food source exhibit an equivalent or superior nectar quality in comparison to the prior source, it is substituted for the older one within the memory framework. Conversely, if it does not meet this criterion, the original source is preserved in memory. In essence, a greedy selection mechanism is employed as the evaluative process between the previous and candidate sources.
In totality, the Artificial Bee Colony (ABC) algorithm incorporates four distinct selection methodologies: (1) a global probabilistic selection method, wherein the probability value is derived from equation (1), utilized by the onlooker bees to identify promising regions, (2) a local probabilistic selection process conducted within a specific area by the worker bees and the onlookers, reliant on visual cues such as color, shape, and scent of the flowers (sources), whereby bees are unable to ascertain the type of nectar source until they reach the appropriate

location and differentiate among the sources present based on olfactory signals, as articulated by equation (2), (3) a local selection referred to as the greedy selection process, executed by both onlooker and worker bees, whereby if the nectar quantity of the candidate source surpasses that of the current source, the bee discards the current source and commits the candidate source to memory as produced by equation (2); otherwise, the existing food source is retained in memory, and (4) a random selection process executed by scouts, as delineated in equation (3). It is evident from the aforementioned discourse that there exist three control parameters inherent to the fundamental ABC: the quantity of food sources, which corresponds to the number of employed or onlooker bees (SN), the limit value, and the maximum cycle number (MCN).

In the context of honeybee behavior, the recruitment rate serves as an indicator of the rapidity with which the bee colony locates and utilizes a newly identified food source. Analogously, artificial recruitment may be construed as a metric for the expediency with which viable solutions or high-quality solutions to complex optimization challenges can be ascertained. The survival and advancement of the bee colony are contingent upon the swift identification and effective exploitation of optimal food resources. Correspondingly, the successful resolution of intricate engineering dilemmas is associated with the relatively prompt discovery of effective solutions, particularly for problems necessitating real-time resolution. Within a comprehensive search framework, the processes of exploration and exploitation must be executed concurrently. In the ABC algorithm, while onlookers and worker bees engage in the exploitation process within the search domain, the scouts oversee the exploration process. A detailed pseudo-code of the ABC algorithm is provided below:

1: Initialize the population of solutions x_i, i =1, ..., SN
2: Evaluate the population
3: cycle = 1
4: **repeat**
5: Produce new solutions v_i for the worker bees by using (2) and evaluate them
6: Apply the greedy selection process for the worker bees
7: Calculate the probability values Pi for the solutions x_i by (1)
8: Produce the new solutions v_i for the onlookers from the solutions x_i selected depending on P_i and evaluate them
9: Apply the greedy selection process for the onlookers
10: Determine the abandoned solution for the scout, if exists, and replace it with a new randomly produced solution x_i
by (3)
11: Memorize the best solution achieved so far
12: cycle = cycle + 1
13: **until** cycle = MCN

The ABC algorithm flowchart is shown in Fig. 1.

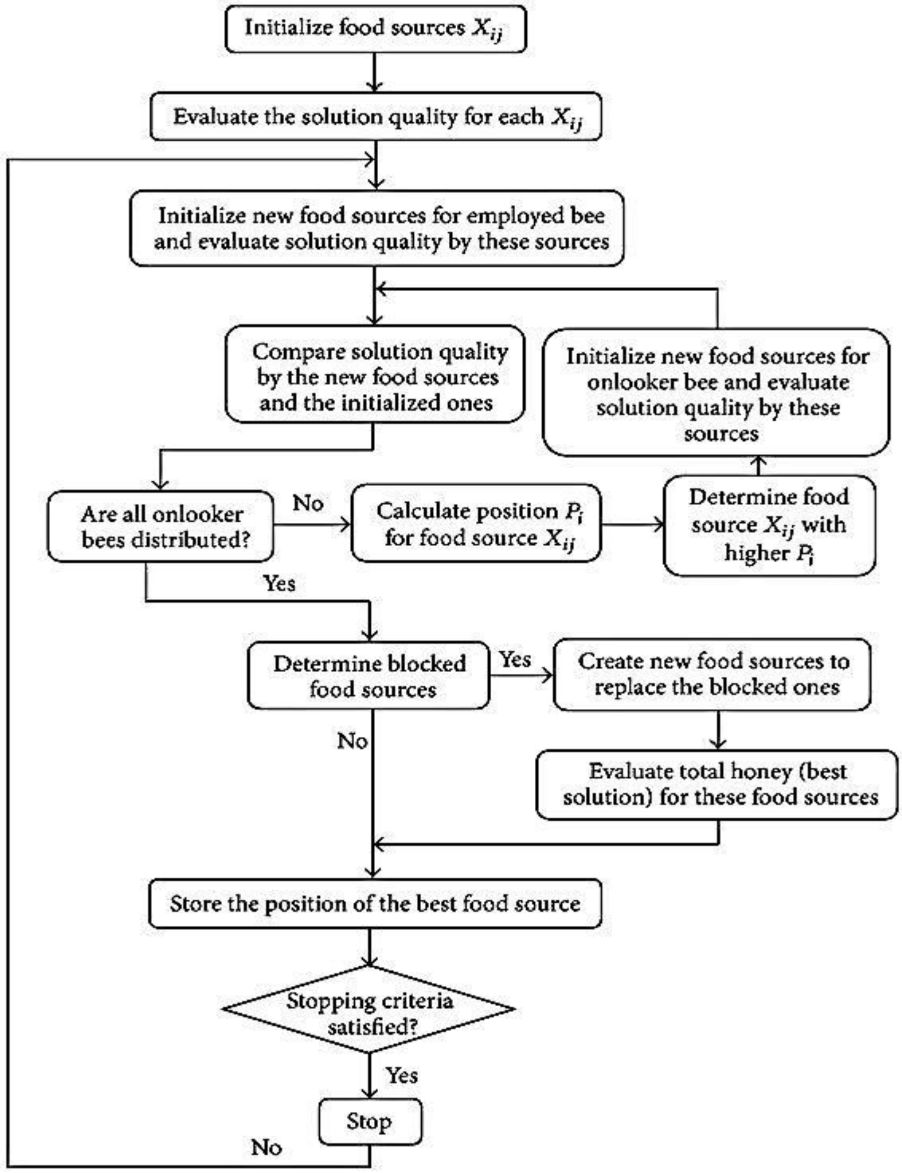


Fig. 1 The ABC algorithm flowchart [20]

The Artificial Bee Colony (ABC) algorithm is indeed initially designed for unconstrained optimization problems. However, it can be effectively adapted to handle constrained optimization problems, such as the design of reinforced concrete cantilever retaining walls, through the incorporation of constraint-handling techniques.

In this study, the constraints are primarily related to geotechnical and structural requirements, such as overturning stability, sliding stability, bearing capacity, and steel reinforcement limits, among others. To address these constraints within the ABC algorithm, a penalty function approach is employed. This method transforms the constrained optimization problem into an unconstrained one by adding a penalty term to the objective function. The penalty term increases the objective function value when constraints are violated, thereby discouraging the algorithm from exploring infeasible solutions. The penalty function can be expressed as by Eq. (4):

$$F(x) = f(x) + P(x) \tag{4}$$

where:

- $F(x)$ is the penalized objective function,
- $f(x)$ is the original objective function (either cost or weight),
- $P(x)$ is the penalty term, which is a function of the constraint violations.

The penalty term $P(x)$ is typically defined by Eq. (5):

$$P(x) = \sum_{i=1}^{m} \lambda_i . \max\,(0, g_i(x))^2 \tag{5}$$

where:

- λ_i is a penalty coefficient for the i-th constraint,
- $g_i(x)$ represents the i-th constraint function,
- m is the total number of constraints.

By incorporating this penalty function, the ABC algorithm is guided to search for solutions that not only minimize the cost or weight but also satisfy all the necessary constraints. This approach ensures that the final optimized design is both feasible and efficient.
In the ABC algorithm, parameters such as the number of worker bees, onlooker bees, and scout bees are carefully tuned to create an appropriate balance between exploration and exploitation. In the first 100 iterations, scout bees play a significant role in exploring the search space, helping the algorithm identify promising regions. After this phase, worker and onlooker bees focus more on exploiting the identified regions. These settings allow the ABC algorithm to quickly move toward reducing the objective function in the early stages.

2-1- Impact of Swarm Size (Population) on the Optimization Process

The swarm size (population size) in the Artificial Bee Colony (ABC) algorithm plays a critical role in determining the balance between exploration and exploitation during the optimization process.

2-2- Key Effects of Swarm Size on Optimization Performance:

- **Exploration Capability** – A larger population enhances the algorithm's ability to explore a wider range of design solutions, reducing the risk of premature convergence to local optima.
- **Exploitation Refinement** – While exploration is crucial, a sufficiently large swarm also allows for finer exploitation around promising regions, improving solution accuracy.
- **Computational Efficiency** – The selected swarm size (1000) strikes a balance between solution quality and computational cost, as further increases did not yield significant improvements but required more runtime.
- **Consistency in Results** – Multiple runs confirmed that this swarm size consistently produced optimal or near-optimal designs for reinforced concrete cantilever retaining walls, as validated by comparisons with existing literature (Section 4).

2-3- Justification for Swarm Size Selection:

- **Problem Complexity**: The optimization involves multiple design variables (e.g., wall dimensions, steel reinforcement) and constraints (geotechnical and structural), necessitating a sufficiently large population to cover the search space effectively.

- **Empirical Testing**: Smaller swarm sizes (e.g., 500) led to higher variability in results, while larger sizes (e.g., 1500) increased runtime without substantial gains in optimization performance.
- **Algorithm Robustness**: The chosen swarm size ensured reliable convergence across different initial conditions, reinforcing the algorithm's applicability in practical engineering design.

This analysis has been incorporated into the revised manuscript (Section 2) to clarify the role of swarm size in the study.
The swarm size (population) in the ABC algorithm was set to 1000 to ensure a balance between exploration and exploitation. This value was determined through preliminary trials, where it demonstrated consistent convergence and robustness for the high-dimensional optimization problem at hand. Larger swarm sizes were tested but showed diminishing returns in solution quality while increasing computational time. The selected size effectively navigated the design space, as evidenced by the algorithm's performance in comparative studies (Section 4).

3- The Requirements and Specifications of Concrete Retaining Wall

The design variables, constraints and objective functions are necessary for each optimization problem. In order to introduce these specifications, a T-shape wall is considered that it is shown in Fig. 2.

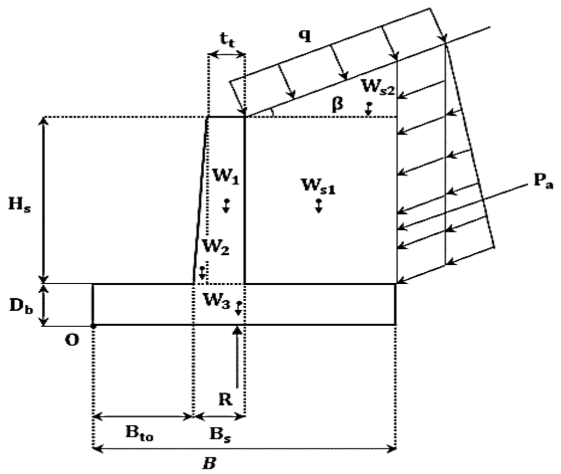


Fig. 2 The considered T-shape wall

3-1- Design Variables

All parameters related to wall design are considered as design variables, including wall dimensions and required steel values for the toe, heel, and stem. These variables are listed in Table 1. It should be noted that the compressive steel areas are also included as design variables. In some cases, the design process may require additional compressive steel if the tensile steel is not sufficient to handle the applied moment. The units for steel area are cm^2 per unit length of wall. The number of required bars is determined using MATLAB software, following the criteria outlined in the American Concrete Institute Code (ACI 318-08) [21]. The design variables have both lower and upper bounds, as specified by Saribas and Erbatur [1] and listed in Table 2. It is important to note that the maximum and minimum values for steel areas are constrained by the ACI code [21]. The values listed in Table 2 are initial values and will be used to start the optimization process.

Table 1 The design variables of concrete retaining wall

type	Name	Unit	Symbol
Wall dimension variables	Total base width	m	B
	Toe width		B_{to}
	Stem thickness at bottom		B_s
	Thickness of base		D_b
	Stem thickness at top		t_t
Steel area variables	Stem tensile steel area	$\left(cm^2/m\right)$	AstS
	Toe tensile steel area		AstT
	Heel tensile steel area		AstH
	Stem compressive steel area		AscS
	Toe compressive steel area		AscT
	Heel compressive steel area		AscH
Software output	Number of stem tensile steel	–	n_1
	Number of toe tensile steel		n_2
	Number of heel tensile steel		n_3
	Number of stem compressive steel		n_4
	Number of toe compressive steel		n_5
	Number of heel compressive steel		n_6

Table 2 The upper and lower bands of variables

Variables name	Unit	Lower bound	Upper bound
Total base width	m	$(24 \times H_s)/55$	$(7 \times H_s)/9$
Toe width	m	$(8 \times H_s)/55$	$(7 \times H_s)/27$
Stem thickness at bottom	m	0.2	$H_s/9$
Thickness of base	m	$H_s/11$	$H_s/9$
Stem thickness at top	m	0.2	0.3
Steel area	cm^2/m	0	80

3-2- Design Constraints

It is obvious that the optimization problem is performed by controlling some constraints. In this study, the constraints include the geotechnical and structural constraints that are defined by $g_i(x)$. It should be noted that the numbers of constraint are m. The considered constraints in this paper are shown in Table 3. All constraints are controlled based on ACI 318 [21], Bowles [22] and other references books. Furthermore, the bearing capacity and lateral earth pressure are calculated based on Hansen and Rankine methods, respectively.

$$g_i(x) \leq 0 \ , \qquad i = 1,2, \ldots, m \tag{6}$$

Table 3 The design constraints

Names of constraints	Unit	Names of constraints	Unit
Overturning stability	kN. m	Yielding of tensile steel	–
Sliding stability	kN	Yielding of compressive steel	–
No tension condition in foundation	m	Minimum footing depth	m
Bearing capacity	kPa	Stem slope control	–
Shear control	kN	Minimum distance of tensile steel	m
Moment control	kN. m	Minimum distance of compressive steel	m
Minimum of tensile steel	–	Maximum distance of tensile steel	m
Maximum of tensile steel	–	Maximum distance of compressive steel	m

3-2-1- Geotechnical Constraints:

- **Overturning Stability:** This constraint ensures that the resisting moment due to the weight of the wall and the backfill soil is greater than the overturning moment caused by lateral soil pressure. A safety factor of 1.5 is typically used (Eq. 7).

$$\text{FS}_{\text{overturning}} = \frac{\sum M_{resisting}}{\sum M_{overturning}} \geq 1.5 \quad (7)$$

- **Sliding Stability:** This constraint ensures that the resisting force due to friction between the base of the wall and the underlying soil is greater than the sliding force caused by lateral soil pressure. A safety factor of 1.5 is typically used (Eq. 8).

$$\text{FS}_{\text{sliding}} = \frac{\sum F_{resisting}}{\sum F_{sliding}} \geq 1.5 \quad (8)$$

- **No Tension Condition in Foundation:** This constraint ensures that no tensile stress is generated in the foundation. This is verified by checking the stress distribution in the foundation to ensure that all stresses are compressive (Eq. 9).

$$\sigma_{min} \geq 1 \quad (9)$$

- **Bearing Capacity:** This constraint ensures that the stress applied to the soil beneath the foundation does not exceed the allowable bearing capacity of the soil. A safety factor of 3 is typically used (Eq. 10).

$$FS_{bearing} = \frac{q_{ult}}{q_{applied}} \geq 3 \quad (10)$$

3-2-2- Structural Constraints:

- **Shear Control:** This constraint ensures that the shear resistance of the concrete and steel is greater than the applied shear force on the wall (Eq. 11). This control is performed for different sections of the wall (stem, toe, and heel).

$$V_{applied} \leq \varphi V_{capacity} \quad (11)$$

where ϕ is the strength reduction factor.

- **Moment Control:** This constraint ensures that the flexural resistance of the concrete and steel is greater than the applied bending moment on the wall (Eq. 12). This control is also performed for different sections of the wall.

$$M_{applied} \leq \varphi M_{capacity} \quad (12)$$

- **Yielding of Tensile and Compressive Steel:** This constraint ensures that the stress in the tensile and compressive steel does not exceed the yield strength of the steel (Eq. 13).

$$\sigma_{steel} \leq f_y \quad (13)$$

- **Minimum and Maximum Steel Percentage:** This constraint ensures that the percentage of steel used in the wall is within the allowable limits specified by the design code (Eq. 14).

$$\rho_{min} \leq \rho \leq \rho_{max} \quad (14)$$

- **Minimum and Maximum Distance of Steel Bars:** This constraint ensures that the spacing of the steel bars in the wall is within the allowable limits specified by the design code (Eq. 15).

$$s_{min} \le s \le s_{max} \quad (15)$$

3-2-3- Geometric Constraints:

- **Stem Slope Control:** This constraint ensures that the slope of the stem is within the allowable range to ensure structural and construction feasibility.
- **Minimum Footing Depth:** This constraint ensures that the depth of the footing is greater than the minimum allowable depth specified by the design code.

3-2-4- Economic Constraints:

- **Material Cost:** This constraint ensures that the total cost of materials used in the wall (concrete and steel) does not exceed the maximum allowable cost specified by the designer.

3-2-5- Weight Constraints:

- **Wall Weight:** This constraint ensures that the total weight of the wall does not exceed the maximum allowable weight specified by the designer.

3-3- Objective Functions

To optimize the concrete retaining wall, two objective functions presented by Saribas and Erbatur [1] are considered. Once the constraints have been controlled, the wall will be optimized from two perspectives: cost and weight. These objective functions are defined as Eq. (7) and (8):

$$f(C) = C_s W_s + C_c V_c \quad (7)$$

$$f(W) = W_s + 100 V_c \gamma_c \quad (8)$$

The units of objective functions parameters are presented in Table 4. It is important to note that the required development length of bars (l_{dh}, l_{dc}) are needed for calculation of W_s. The values of these parameters are obtained based on ACI 318 [21].

Table 4 The unit of objective functions parameters

Parameter	Name	Unit
f(C)	The cost objective function	$ per unit length of the wall

$f(W)$	The weight objective function	kN in per unit length of the wall
C_s	The cost of steel unit	$\$/kg$
C_c	The cost of concrete unit (forming, concretion, vibration and etc.)	$\$/m^3$
W_s	The steel weight in the wall length unit	kN
V_c	The concrete volume in the wall length unit	m^3
γ_c	The weight of concrete unit	kN/m^3

4- The ABC Algorithm Verification

4-1- The Comparison with Saribas and Erbatur Research (1996)

This verification is being conducted to demonstrate the effectiveness of the ABC algorithm in optimization. As previously mentioned, Saribas and Erbatur utilized a differential approach and a specialized program, RETOPT [1], to optimize the retaining wall. This method has been proven to have a high level of accuracy compared to other methods. In other words, other metaheuristic methods strive to achieve results that are similar to those obtained through the differential approach, thus showcasing their performance. To further illustrate this, two examples from Saribas and Erbatur's research [1] are presented and compared to the results obtained through the ABC algorithm. The T-shape wall used in the study is depicted in Figure 2. Additionally, Table 5 displays the initial and fixed parameters for the two examples based on Saribas and Erbatur's research [1], while Table 6 shows the number of variables (7) and constraints (10). Although the safety factors for design are provided in Table 5, a more detailed discussion of the soil mechanics models used in stability analysis is warranted. The stability of the retaining wall is analyzed based on conventional geotechnical principles, including Rankine's and Coulomb's earth pressure theories for lateral earth pressures, as well as bearing capacity analysis using Hansen's method. The wall stability is evaluated by considering overturning, sliding, and bearing capacity failures, ensuring that deformations remain within acceptable limits. Additionally, the interaction between the retaining wall and the backfill is assessed under various loading conditions, including surcharge and seismic effects where applicable. These considerations align with established design standards such as ACI 318 and recommendations from previous studies on optimized retaining wall design.

Table 5 Initial and fixed parameters in Saribas and Erbatur research [1]

Parameter	Symbol	Example 1	Example 2
Height of stem (m)	H_s	3	4.5
Stem thickness at the top (m)	t_t	0.2	0.25

Yield strength of reinforcing steel (MPa)	F_y	400	400
Compressive strength of concrete (MPa)	f'_c	21	21
Concrete cover (cm)	d_{co}	7	7
Maximum steel percentage	ρ_{max}	0.016	0.016
Minimum steel percentage	ρ_{min}	0.00333	0.00333
Shrinkage and temporary reinforcement percent	ρ_{st}	0.002	0.002
Diameter of bar (cm)	ϕ_{bar}	1.2	1.4
Surcharge load (kPa)	q	20	30
Backfill slope (Degree)	β	10	15
Internal friction angle of retained soil (Degree)	ϕ	36	36
Internal friction angle of base soil (Degree)	ϕ'	0	34
Unit weight of retained soil (kN/m^3)	γ_s	17.5	17.5
Unit weight of base soil (kN/m^3)	γ'_s	18.5	18.5
Unit weight of concrete (kN/m^3)	γ_c	23.5	23.5
Cohesion of base soil (kPa)	c	125	100
Depth of soil in front of wall (m)	D_f	0.5	0.75
Cost of steel ($/kg)	C_s	0.4	0.4
Cost of concrete ($\$/m^3$)	C_c	40	40
Factor of safety against sliding	SF_s	1.5	1.5
Factor of safety for overturning stability	SF_o	1.5	1.5
Factor of safety for bearing capacity	SF_b	3	3

Table 6 List of the variables and constraints for the first case of verification

Variables	Constraints
Total base width	Shear at bottom of stem
Toe width	Moment at bottom of stem
Stem thickness at the bottom	Overturning stability
Thickness of base	Sliding stability
Area of stem tensile steel	No tension condition in foundation
Area of toe tensile steel	Bearing capacity
Area of heel tensile steel	Toe shear
	Toe moment
	Heel shear
	Heel moment

The results obtained from the experiment are displayed in Table 7. As can be seen, there is a slight difference between the cost and weight objective functions of the ABC algorithm and RETOPT [1]. This indicates that the ABC algorithm is effective in optimizing the retaining wall from both a cost and weight perspective. While the method used by Saribas and Erbatur [1] is highly accurate, it is time-consuming and not suitable for larger problems. In contrast, the ABC algorithm runs quickly and reaches the optimal solution. The values of the variables at the optimal points for the cost and weight objective functions in Examples 1 and 2 are presented in Tables 8 and 9, respectively. Additionally, the values of the constraints at the optimal points for Examples 1 and 2 are shown in Tables 10 and 11, respectively. These tables demonstrate that there is a small difference in the values of the variables and constraints, further highlighting the effectiveness of the ABC algorithm in optimization.

Table 7 The cost and weight objective functions in first verification

Method	**Example 1**		**Example 2**	
	Cost ($/m)	Weight (kN/m)	Cost ($/m)	Weight (kN/m)
RETOPT [1]	82.474	24.49	189.546	51.81

ABC algorithm	82.53	24.51	189.758	51.83
ACO algorithm [5]	-	-	201.185	54.41
BFOA algorithm [6]	-	-	190.574	52.39
The differences between RETOPT [1] and other methods				
RETOPT [1] and ABC algorithm	%0.067	%0.036	%0.11	%0.087
RETOPT [1] and ACO algorithm [5]	-	-	%6.14	%4.91
RETOPT [1] and BFOA algorithm [6]	-	-	%0.542	%1.18

Table 8 The variables values of cost and weight optimum points for Example 1

Design variables		Optimum values for minimum Cost		Optimum values for minimum Weight	
		RETOPT [1]	ABC algorithm	RETOPT [1]	ABC algorithm
X_1	Total base width (m)	1.578	1.580	1.574	1.575
X_2	Toe width (m)	0.436	0.479	0.441	0.472
X_3	Stem thickness at the bottom (m)	0.258	0.259	0.200	0.200
X_4	Thickness of base (m)	0.273	0.273	0.273	0.273
X_5	Area of stem tensile steel (cm^2/m)	12.574	12.44	21.072	21.035
X_6	Area of toe tensile steel (cm^2/m)	6.551	6.551	6.551	6.551
X_7	Area of heel tensile steel (cm^2/m)	6.551	6.551	6.681	6.551

Table 9 The variables values of cost and weight optimum points for Example 2

Design variables		Optimum values for minimum Cost		Optimum values for minimum Weight	
		RETOPT [1]	ABC algorithm	RETOPT [1]	ABC algorithm
X_1	Total base width (m)	2.254	2.259	2.238	2.242
X_2	Toe width (m)	0.655	0.678	0.655	0.686
X_3	Stem thickness at the bottom (m)	0.417	0.432	0.300	0.300
X_4	Thickness of base (m)	0.409	0.409	0.409	0.409
X_5	Area of stem tensile steel (cm^2/m)	23.475	22.295	41.626	41.547
X_6	Area of toe tensile steel (cm^2/m)	11.059	11.059	11.059	11.059
X_7	Area of heel tensile steel (cm^2/m)	11.059	11.059	11.059	11.059

Moreover, Ghazavi and Bazzazian [5] and Ghazavi and Salavati [6] optimized the Example 2 of this research by ACO algorithm and BFOA algorithm, respectively and then reached to acceptable results. However, these results that are shown in Table 7 are more than ABC algorithm results. In other words, these results show that the ABC algorithm has better performance in comparison with ACO and BFOA algorithms.

Table 10 The constraints values of cost and weight optimum points for Example 1

Constraints	Minimum cost value		Minimum weight value	
	RETOPT [1]	ABC algorithm	RETOPT [1]	ABC algorithm

g_1	Shear at bottom of stem	55.893-	-56.973	-17.985	-18.0714
g_2	Moment at bottom of stem	0	0	0	0
g_3	Overturning stability	73.333-	-71.878	-74.001	-72.823
g_4	Sliding stability	-69.823	-70.302	-69.212	-69.4681
g_5	No tension condition in foundation	0	0	0	0
g_6	Bearing capacity	-104.663	-108.778	-102.204	-105.253
g_7	Toe shear	-51.98	-48.744	-49.76	-47.2439
g_8	Toe moment	-27.165	-24.497	-26.455	-24.415
g_9	Heel shear	-60.808	-60.025	-61.734	-60.336
g_{10}	Heel moment	-2.102	-4.201	0	-0.605

Table 11 The constraints values of cost and weight optimum points for Example 2

Constraints		Minimum cost value		Minimum weight value	
		RETOPT [1]	ABC algorithm	RETOPT [1]	ABC algorithm
g_1	Shear at bottom of stem	75.661-	-85.592	0	-0.162
g_2	Moment at bottom of stem	0	-0.002	0	-0.002
g_3	Overturning stability	228.051-	-226.642	-229.756	-227.82
g_4	Sliding stability	-142.246	-141.057	-143.491	-141.884
g_5	No tension condition in foundation	0	0	0	0
g_6	Bearing capacity	-1762.754	-1765.99	-1756.297	-1759.62
g_7	Toe shear	-42.701	-41.269	-26.989	-33.745
g_8	Toe moment	-67.2	-64.531	-68.555	-60.926
g_9	Heel shear	-76.315	-76.445	-78.083	-77.0957
g_{10}	Heel moment	-13.834	-18.774	1.542	-5.983

4-2- The Comparison with Manual Design

In order to show the importance of optimization, the optimized wall is compared with conventional design of engineering. For this section, the example is selected from Bowles [22]. In this example, the wall is the normal T-shape wall that is shown in Fig. 3. The initial parameters are shown in Table 12. The variables and constraints are equal to Table 1 and 3, respectively.

Table 12 The initial parameters of Bowles example [22]

Parameter	Symbol	Value
Height of stem (m)	H_s	2.44

Concrete cover (cm)	d_{co}	5
Shrinkage and temporary reinforcement percent	ρ_{st}	0.0018
Diameter of bars (cm)	ϕ_{bar}	2
Surcharge load (kPa)	q	12
Backfill slope (degree)	β	0
Internal friction angle of retained soil (degree)	ϕ	36
Internal friction angle of base soil(degree)	$\acute{\phi}$	0
Unit weight of retained soil (kN/m^3)	$\acute{\gamma}_s$	18.86
Unit weight of base concrete (kN/m^3)	γ_c	23.6
Unit weight of soil (kN/m^3)	γ_s	17.3
Cohesion of base soil (kPa)	c	120
Depth of soil in front of wall (m)	D_f	1.22
Factor of safety for bearing capacity	SF_b	3
Factor of safety against sliding	SF_s	1.5
Factor of safety against overturning	SF_o	1.5

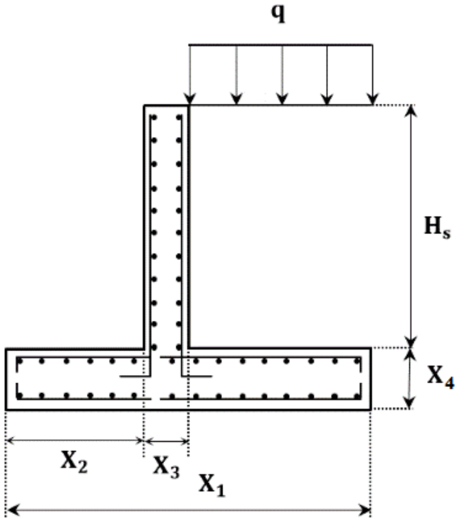


Fig 3 The geometric model of retaining wall in comparison with manual design

The results of the objective functions are presented in Table 13. It is evident that the optimization has a significant impact on both the wall cost and weight, resulting in a decrease of 46.57% and 45.89%, respectively. These reductions are considered generous in the field of civil engineering and hold great importance. Furthermore, the verification process took approximately 8 minutes, which is longer than the previous verification conducted by Saribas and Erbatur [1]. This can be attributed to the increase in the number of constraints, which has led to a longer run time.

The optimal values for cost and weight variables are presented in Table 14. These results demonstrate that the ABC algorithm is the best choice for optimization. Additionally, the convergence graphs for the cost and weight objective functions are shown in Figures 4 and 5, respectively. As shown, the ABC algorithm is able to quickly find the optimal point with a low number of iterations, indicating its rapid convergence. The initial decline in the ABC algorithm curve highlights its effectiveness in exploration.

Table 13 The optimum objective functions with ABC algorithm and manual design of Bowles example [22]

Method	Objective function	Value	Run time (s)
Bowles (manual design) [22]	Cost ($/m)	86.7692	High – manual design

	Weight (kN/m)	34.77	High – manual design
ABC algorithm	Cost ($/m)	46.3564	497.8609
	Weight (kN/m)	18.80	504.34

Table 14 The optimum variables with ABC algorithm in second verification-comparison with manual design

	Design parameters and output software	Optimum value	
		Cost	Weight
X_1	Total base width (m)	1.3934	1.3933
X_2	Toe width (m)	0.4812	0.4768
X_3	Stem thickness at the bottom (m)	0.200	0.200
X_4	Thickness of base (m)	0.2218	0.2218
X_5	Stem tensile steel area (cm^2/m)	6.2835	6.2831
X_6	Toe tensile steel area (cm^2/m)	6.2832	6.2833
X_7	Heel tensile steel area (cm^2/m)	6.2833	6.2832
X_8	Stem compressive steel area (cm^2/m)	0	0
X_9	Toe compressive steel area (cm^2/m)	0	0
X_{10}	Heel compressive steel area (cm^2/m)	0	0
X_{11}	Number of stem tensile steel	3	3
X_{12}	Number of toe tensile steel	3	3
X_{13}	Number of heel tensile steel	3	3
X_{14}	Number of stem compressive steel	0	0
X_{15}	Number of toe compressive steel	0	0
X_{16}	Number of heel compressive steel	0	0

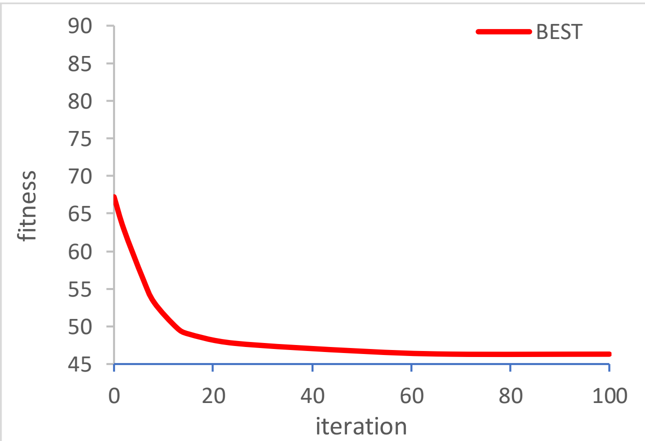


Fig 4 Convergence graph of the cost objective function

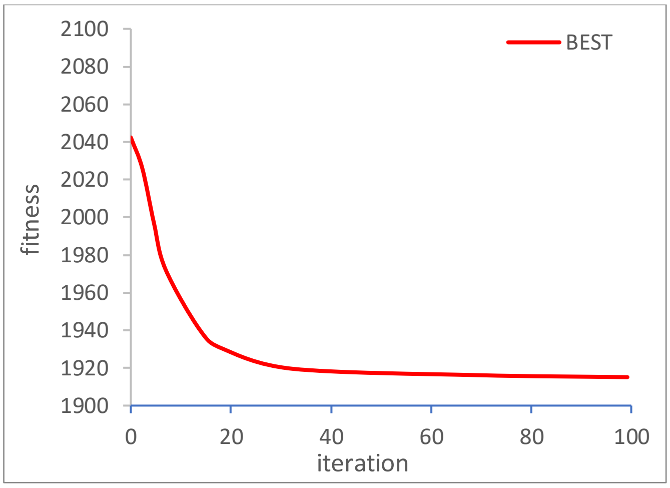


Fig 5 Convergence graph of the weight objective function

In Figures 4 and 5, the changes in the objective function (cost and weight) during the first 100 iterations are clearly shown. As can be observed, in the early iterations, the ABC algorithm quickly moves toward reducing the objective function. This reduction is particularly noticeable in the first 50 iterations, after which the changes in the objective function gradually decrease. This behavior demonstrates the algorithm's ability to find optimal regions in the early stages of the search.

5- Comparative Analysis of Wall Geometries

In this section, two types of reinforced concrete cantilever retaining walls are considered for comparison: Type 1, which is a T-shape wall with variable thickness in the stem, and Type 2, which is a normal T-shape wall with uniform thickness. These wall types are illustrated in Figure 6. The comparison aims to evaluate the impact of geometric variations on the cost and weight objective functions. The required and fixed parameters are considered based on Table 5 - Example 2. In other words, this study aims to demonstrate the impact of wall geometric figure on cost and weight objective functions. The variables and constraints are similar to those in Table 1 and 3, respectively. The results for a normal T-shape and a T-shape with varying stem thickness are presented in Table 15. As shown in this table, the wall geometric figure has a significant effect on reducing wall cost and weight. The normal T-shape wall results in an increase of 7.897% and 8.922% in the cost and weight objective functions, respectively. It is evident that the T-shape wall with varying stem thickness is superior to the normal T-shape wall in terms of both objective functions.

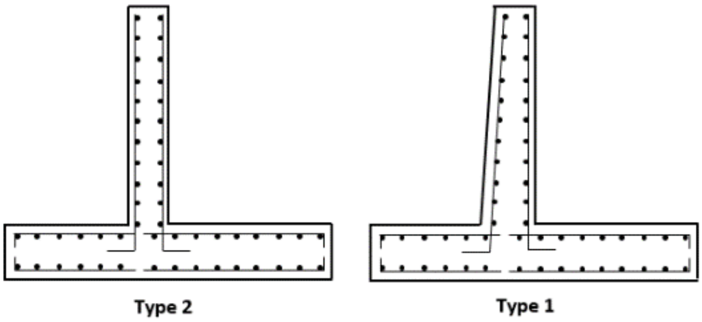


Fig 6 The two wall types in parametric studies section

Table 15 The obtained results of cost and weight objective function for parametric studies

Objective function	Type 1	Type 2	The difference of type 1 and 2
Cost ($/m)	149.3302	161.1238	%7.897
Weight (kN/m)	49.40	53.81	%8.922

6- Sensitivity Analysis

The initial parameters in the optimization process are typically held constant; however, these parameters can significantly influence both the cost and weight of the retaining wall. To better understand the impact of these parameters, a more detailed sensitivity analysis is conducted, focusing on two key parameters: the unit weight of the soil and the stem height. Additionally, the internal friction angle of the soil, which is linearly related to the unit weight of the soil, is also examined. The values considered for these parameters are presented in Table 16.

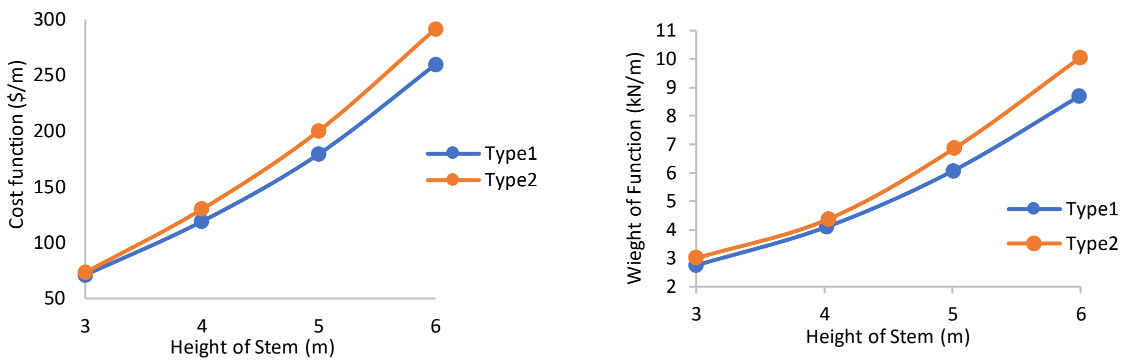
Table 16 The values of parameters in sensitivity analysis section

Parameter	symbol	value	comments
Stem height (m)	H_s	3, 4.5, 6	-
Unit weight of retained soil (kN/m³)	γ_s	15-16-17-18	These parameters are changed together with assuming linear relationship
Internal friction angle of soil (degree)	ϕ	30°, 33.33°, 36.66°, 40°	

6-1- The effect of stem height on objective functions

The stem height is a critical parameter in the design of retaining walls, as it directly influences the structural stability and material requirements. Figure 7 illustrates the impact of varying stem height on both the cost and weight objective functions for the two types of walls considered in this study. As expected, both objective functions increase significantly with an increase in stem height. For instance, when the stem height is doubled from 3 meters to 6 meters, the wall cost increases by more than 250%, while the weight increases by approximately 200% to 250%. This nonlinear relationship highlights the importance of optimizing stem height to achieve cost-effective and lightweight designs.

Furthermore, the results indicate that the normal T-shape wall experiences a more pronounced increase in both cost and weight compared to the T-shape wall with variable stem thickness. This suggests that the geometric configuration of the wall plays a significant role in determining its overall performance, particularly under varying stem heights.



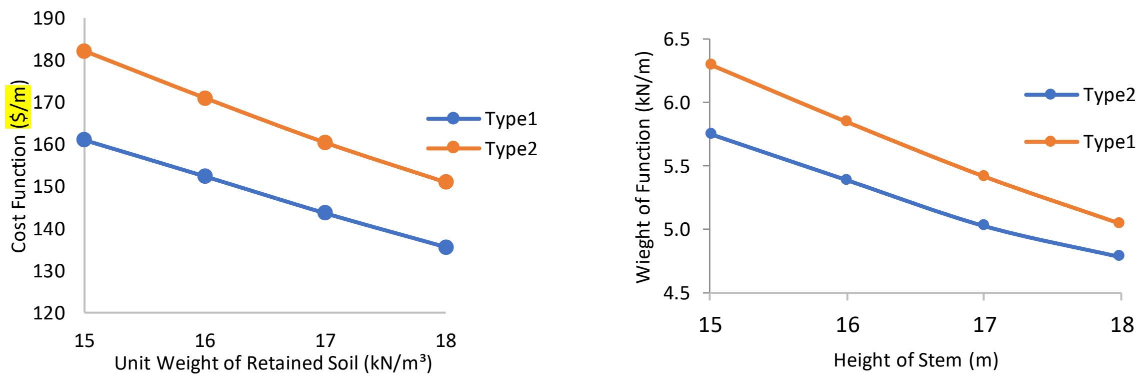
(a) (b)

Fig. 7 The effect of stem height on objective functions for two wall types (a) Cost objective function (in $/m) as stem height increases (b) Weight objective function (in kN/m) as stem height increases.

6-2- The Effect of Soil Unit Weight and Internal Friction Angle on Objective Functions

The unit weight of the soil and the internal friction angle are key factors that influence the lateral earth pressure acting on the retaining wall. Figure 8 presents the impact of these parameters on the cost and weight objective functions for both wall types. As shown, both objective functions decrease as the unit weight of the soil increases. This is because a higher unit weight of soil generally leads to a higher internal friction angle, which in turn reduces the lateral earth pressure and, consequently, the required wall dimensions and reinforcement.

The reduction in cost and weight is not uniform across all values of soil unit weight. For example, a decrease in soil unit weight from 18 kN/m³ to 15 kN/m³ results in a 15% to 20% increase in both cost and weight for both wall types. This indicates that the normal T-shape wall is more sensitive to changes in soil properties, with its cost and weight increasing by approximately 2% to 3% more than the T-shape wall with variable thickness.



(a) (b)

Fig. 8 The effect of soil unit weight and internal friction angle on objective functions for two wall types (a) Cost objective function (in $/m) as soil unit weight and internal friction angle vary (b) Weight objective function (in kN/m) as soil unit weight and internal friction angle vary.

6-3- Advanced Sensitivity Analysis Using Regression Models

To further enhance the sensitivity analysis, a regression-based approach is employed to quantify the relationship between the key parameters (stem height, soil unit weight, and internal friction angle) and the objective functions. The regression models are developed using the data obtained from the optimization process, and the results are presented in Table 17.

Table 17 Regression coefficients for cost and weight objective functions

Parameter	Cost Objective Function (Coefficient)	Weight Objective Function (Coefficient)
Stem Height (m)	0.45	0.38
Unit Weight of Soil (kN/m³)	-0.12	-0.10
Internal Friction Angle (degree)	-0.08	-0.07

The regression coefficients indicate that stem height has the most significant impact on both cost and weight, followed by the unit weight of the soil and the internal friction angle. This quantitative analysis provides a more robust understanding of the relative importance of each parameter and can guide engineers in making informed design decisions.

6-4- Discussion on Sensitivity Analysis Results

The sensitivity analysis reveals that both stem height and soil properties play a crucial role in determining the cost and weight of retaining walls. The results underscore the importance of considering these parameters during the design phase, particularly when optimizing for cost and weight. The regression models developed in this study provide a valuable tool for engineers to predict the impact of varying design parameters on the overall performance of retaining walls.

Furthermore, the analysis highlights the advantages of using a T-shape wall with variable stem thickness, as it demonstrates better performance under varying soil conditions and stem heights compared to the normal T-shape wall. This finding aligns with the broader goal of achieving cost-effective and lightweight designs in civil engineering projects.

7- Conclusion

1-The optimization process significantly impacts the cost and weight of walls. This indicates that through careful design and material selection, structures can become more cost-effective and lighter, which is essential in construction projects.

2-The present study introduces two types of walls to examine the effects of geometric shapes. It was found that the normal T-shaped wall has higher values for both cost and weight objective functions. This suggests that certain designs may lead to increased costs and material usage, highlighting the need for efficient design choices.

3-A key finding is that the T-shaped wall with variable thickness achieves the objective functions of minimum cost and weight. This emphasizes the importance of design flexibility and demonstrates that variations in wall thickness can optimize performance and reduce costs.

4-This research also investigates how stem height and unit soil weight affect the objective functions. It was observed that as stem height increases, the cost and weight objectives also rise, while an increase in unit soil weight leads to a decrease in these objectives. Understanding these relationships is crucial for engineers when designing walls that are subjected to various soil conditions.

5- Compared to other optimization algorithms such as Genetic Algorithm (GA) and Ant Colony Optimization (ACO), the ABC algorithm quickly moves toward reducing the objective function in the first 100 iterations. This behavior demonstrates the high capability of the ABC algorithm in exploring the search space and finding optimal regions in the early stages. While some algorithms may perform slower in the early stages, ABC, due to its population-based structure and the intelligent behavior of bees, rapidly progresses toward reducing the objective function.

References

[1] Saribas, A., Erbatur, F. (1996). “Optimization and sensitivity of retaining structures”, Journal of Geotechnical Engineering, 122(8), 649-656. https://doi.org/10.1061/(ASCE)0733-9410(1996)122:8(649)

[2] Ceranic, B., Fryer, B. and Baines, R.W. (2001). “An application of simulated annealing to the optimum design of reinforced concrete retaining structures”. Computers and Structures, 79, 1569-1581. https://doi.org/10.1016/S0045-7949(01)00037-2

[3] Sivakumar, G. L. and Munwar, B. (2008). “Optimum design of cantilever retaining walls using target reliability approach”. International Journal of Geomechanics, 8(4), 240-252. https://doi.org/10.1061/(ASCE)1532-3641(2008)8:4(240)

[4] Yepes, V., Alcala, J., Perea, C., Gonzalez-Vidosa, F. (2008). “A parametric study of optimum earth retaining walls by simulated annealing”. Engineering Structures, 30(3), 821-830. https://doi.org/10.1016/j.engstruct.2007.05.023

[5] Ghazavi, M., Bazzazian Bonab, S. (2011). “Learning from ant society in optimizing concrete retaining walls”. Journal of Technology & Education, 5(3), 234-241. https://doi.org/10.22061/tej.2011.285

[6] Ghazavi, M. and Salavati, V. (2011). “Sensitivity analysis and design of reinforced concrete cantilever retaining walls using bacterial foraging optimization algorithm”. Bundesanstalt für Wasserbau ISBN 978-3-939230-01-4.

[7] Kaveh, A. and Behnam, A. F. (2013). “Charged system search algorithm for the optimum cost design of reinforced concrete cantilever retaining wall”. Arabian Journal of Science and Engineering, 38, 563–570. https://doi.org/10.1007/s13369-012-0332-0

[8] Karaboga, D. and Basturk, B. (2008). “On the performance of artificial bee colony (ABC) algorithm”. Applied Soft computing, 8, 687-697. https://doi.org/10.1016/j.asoc.2007.05.007

[9] Shalchi Tousi, M., Laali, S., Ghazavi, M. (2022), “Optimization of reinforced concrete cantilever retaining walls by the use of intelligent water drops algorithm (IWDA)”, International Journal of Advanced Structural Engineering, 12(3), 619–624. https://doi.org/10.1007/ijase.2023.1971683.1059

[10] Shalchi Tousi, M., Laali, S. (2022). “Optimum Design of Reinforced Concrete Cantilever Retaining Walls by Cuckoo Optimization Algorithm (COA)”, International Journal of Advanced Structural Engineering, 12(4), 645–662. https://doi.org/10.1007/ijase.2023.1971723.1060

[11] Shalchi Tousi , M., Ghazavi, M., Laali, S. (2021). “Optimizing Reinforced Concrete Cantilever Retaining Walls Using Gases Brownian Motion Algorithm (GBMOA)”, Journal of Soft Computing in Civil Engineering, 5(1), 1-18. https://doi.org/10.22115/scce.2021.248638.1256

[12] Kalmeci, E.N., Ikizler, S.B., Dede, T., Angin, Z. (2020), “Design of reinforced concrete cantilever retaining wall using Grey wolf optimization algorithm”, Structures, 23, 245-253. https://doi.org/10.1016/j.istruc.2019.09.013

[13] Romani, J., Ossandón, D., Sepúlveda, R., Astudillo, N., Yepes, V. (2023), “Optimizing Retaining Walls through Reinforcement Learning Approaches and Metaheuristic Techniques”, Mathematics, 11, 2104. https://doi.org/10.3390/math11092104

[14] Keivanian, F., Chiong, R., Kashani, A. and Gandomi, A. (2023), “A fuzzy adaptive metaheuristic algorithm for identifying sustainable, economical, and earthquake-resistant reinforced concrete cantilever retaining walls”, Journal of Computational Science, 70, 101978. https://doi.org/10.1016/j.jocs.2023.101978

[15] Tayfur, B., KAMILOĞLU, H. (2024), “Optimization of Cantilever Retaining Wall Design Using Improved Teaching-Learning-Based Optimization Algorithms”, Journal of Experimental and Computational Engineering, 3(2), 134-150. https://doi.org/10.62520/fujece.1430236

[16] Shenouda, M. and Ali, M. (2024), “Simulation-Optimization Model for the Structural Design of Cantilever Retaining walls”, Asian Journal of Civil Engineering, 17. https://doi.org/10.1007/s42107-024-01028-6

[17] Boukhatem, G., Moufida, M., Kamel, G., Bekkouche Souhila, R. (2023), “Mono-objective Optimization of Retaining Wall Using Genetic Algorithm”, Journal of Civil Eng., 18(1). https://doi.org/10.2478/sspjce-2023-0012

[18] Khajehzadeh, M., Taha, M.R., El-Shafie, A. et al. (2011). "Modified particle swarm optimization for optimum design of spread footing and retaining wall", J. Zhejiang Univ. Sci., 12, 415–427. https://doi.org/10.1631/jzus.A1000252

[19] Camp, Ch., Akin, A. (2012). "Design of Retaining Walls Using Big Bang–Big Crunch Optimization", Journal of Structural Engineering, 138(3), https://doi.org/10.1061/(ASCE)ST.1943-541X.0000461

[20] Kashani, A.R., Gandomi, A.H., Azizi, K. et al. (2022), "Multi-objective optimization of reinforced concrete cantilever retaining wall: a comparative study", Struct Multidisc Optim, 65(262). https://doi.org/10.1007/s00158-022-03318-6
[21] Gandomi, A.H., Kashani, A.R., Roke, D.A. et al. (2017). "Optimization of retaining wall design using evolutionary algorithms", Struct Multidisc Optim, 55, 809–825. https://doi.org/10.1007/s00158-016-1521-3
[22] Ibrahim Aydogdu (2016), "Cost optimization of reinforced concrete cantilever retaining walls under seismic loading using a biogeography-based optimization algorithm with Levy flights", Engineering Optimization. http://dx.doi.org/10.1080/0305215X.2016.1191837
[23] Molina-Moreno, F., Garsia-Segura, T., Marti, J., Yepes, V. (2017), " Optimization of buttressed earth-retaining walls using hybrid harmony search algorithms", Engineering Structures, 134, 205-216. http://dx.doi.org/10.1016/j.engstruct.2016.12.042
[24] Temur, Rasim, and Gebrail Bekdaş. (2016), "Teaching Learning-Based Optimization for Design of Cantilever Retaining Walls", Structural Engineering and Mechanics, 57(4), 763-783. https://doi.org/10.12989/sem.2016.57.4.763.
[25] Karaboga, D. and Ozturk, C. (2011). "A novel clustering approach: artificial bee colony (ABC) algorithm". Applied Soft computing, 11(1), 652-657. https://doi.org/10.1016/j.asoc.2009.12.025
[26] Karaboga, D., (2005). "An idea based on honey bee swarm for numerical optimization", Tech. rep., technical report-tr06, Erciyes University, Engineering Faculty, Computer Engineering Department.
[27] Shah, Habib & Tairan, Nasser & Mashwani, Wali & Alsewari, AbdulRahman & Jan, Muhammad Asif & Badshah, Gran. (2017). "Hybrid Global Crossover Bees Algorithm for Solving Boolean Function Classification Task", Conference Paper in Lecture Notes in Computer Science.
[28] ACI 318-08, (2008). "Building code requirements for structural concrete and commentary". American Concrete Institute International.
[29] Bowles, J., (1992). "Foundation analysis and design". 6th edition, McGraw-Hill, New York.