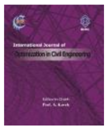
INTERNATIONAL JOURNAL OF OPTIMIZATION IN CIVIL ENGINEERING
Int. J. Optim. Civil Eng., 2024; 14(4):539-551



A COMPARATIVE STUDY OF SINGLE-OBJECTIVE AGAINST MULTI-OBJECTIVE OPTIMIZATION IN STRUCTURAL

S. TalatAhari*, †
School of Computing, Macquarie University, Sydney, Australia

ABSTRACT

Structural optimization plays a critical role in improving the efficiency, cost-effectiveness, and sustainability of engineering designs. This paper presents a comparative study of single-objective and multi-objective optimization in the structural design process. Single-objective problems focus on optimizing just one objective, such as minimizing weight or cost, while other important aspects are treated as constraints like deflections and strength requirements. Multi-objective optimization addresses multiple conflicting objectives, such as balancing cost, with displacement treated as a secondary objective and strength requirements defined as constraints within the given limits. Both optimization approaches are carried out using Chaos Game Optimization (CGO). While single-objective optimization produces a definitive optimal solution that can be used directly in the final design, multi-objective optimization results in a set of trade-off solutions (Pareto front), requiring a decision-making process based on design codes and practical criteria to select the most appropriate design. Through a real-world case study, this research will assess the performance of both optimization strategies, providing insights into their suitability for modern structural engineering challenges.

Keywords: Single-objective Optimization; Multi-objective Optimization; Structural Optimization; Chaos Game Optimization.

Received: 9 September 2024; Accepted: 13 October 2024

[Downloaded from education.iust.ac.ir on 2025-03-10]

[DOI: 10.22068/ijoce.2024.14.4.606]

1. INTRODUCTION

In the realm of structural engineering, optimization plays a crucial role in enhancing the performance, safety, and cost-effectiveness of designs. As structures become increasingly complex, engineers are challenged to balance various design objectives while adhering to stringent safety standards and regulations. Optimization methods serve as powerful tools that

*Corresponding author: School of Computing, Macquarie University, Sydney, Australia
†E-mail address: Siamak.TalatAhari@mq.edu.au (S. TalatAhari)

[Downloaded from education.iust.ac.ir on 2025-03-10]

[DOI: 10.22068/ijoce.2024.14.4.606]

enable engineers to achieve the best possible design solutions, ensuring that structures meet both functional requirements and regulatory constraints [1,2]. This paper focuses on the comparative analysis of single-objective and multi-objective optimization techniques, exploring their implications for structural design.

Single-objective optimization is a traditional approach that concentrates on optimizing a single parameter, such as minimizing weight, cost, or material usage. While this method simplifies the decision-making process, it can inadvertently lead to suboptimal designs, as other important performance metrics—such as deflection and strength requirements —are treated as constraints rather than objectives, [3]. These constraints are essential for ensuring that the final design adheres to safety and performance standards. Here, effectively managing constraints is critical yet challenging, as they often involve complex interactions between various factors. For instance, meeting the weight reduction objective may lead to designs that push the limits of allowable deflection or material strength requirements, which could compromise structural integrity.

Engineers must carefully balance these constraints to ensure that the optimized design is not only efficient but also safe and functional, [4]. If constraints are not handled properly, it can result in designs that, while optimal for one objective, fail to satisfy critical safety requirements, potentially leading to structural failures or performance deficiencies. Also, the technique selected for this aim has some effects on the performance of the optimization algorithms as well. For instance, when using the penalty method to handle constraints, the choice of penalty parameters can greatly influence the optimization process. These parameters dictate how severely violations of constraints are penalized, impacting the algorithm's ability to navigate the design space effectively, [5]. Given these complexities, understanding and effectively managing constraints is crucial in the optimization process, [6-9]. Failure to address these challenges may result in suboptimal designs that do not perform as intended. This highlights the necessity for more comprehensive approaches, such as multi-objective optimization, which can better account for the interplay between objectives and constraints, ultimately leading to safer and more effective structural solutions, [10].

Defining the problem as a multi-objective optimization empowers engineers to simultaneously tackle multiple conflicting objectives, offering a more comprehensive approach. This facilitates the consideration of trade-offs among various performance aspects—such as cost, displacement, or strength requirements —thereby providing a holistic perspective on the design challenges at hand, [11-13]. By employing techniques like Pareto optimization, engineers can delineate a set of optimal solutions, known as the Pareto front, which illustrates the best possible compromises among competing objectives, [14]. However, navigating this complexity presents challenges in decision-making, as choosing the most suitable design from the Pareto front necessitates careful evaluation of design codes and project requirements, [15]. Furthermore, the performance of multi-objective optimization can be influenced by the specific techniques and parameters employed and the strategies used to handle constraints [16].

Although structural optimization has been extensively explored in both single-objective and multi-objective formats by numerous researchers, a comprehensive analysis of the advantages and disadvantages of these approaches within the field of structural optimization has not been thoroughly examined. Therefore, this paper aims to fill this gap by providing a detailed study. To ensure a fair comparison, we employ a single method for both single-objective and multi-objective optimization problems. This research utilizes Chaos Game Optimization (CGO) [17,18] as the foundational technique for both approaches. By investigating a real-world case study, we aim to evaluate the effectiveness and applicability of each optimization strategy in structural design. The findings will illuminate the strengths and weaknesses inherent in both approaches, ultimately offering valuable insights to help engineers make informed decisions that balance performance, cost, and safety in contemporary structural projects.

2. PROBLEM STATEMENT

From a mathematic point of view, a single-objective optimization problem refers to the process of optimizing a single performance criterion while considering various constraints, [19]. In this problem, the objective function $f(x)$ is minimized (or maximized) subject to a set of constraints $g_i(x)$ and $h_j(x)$. The mathematical formulation can be expressed as follows:

$$\begin{aligned} &\min f(x) \\ &subject\ to: \\ &g_i(x) \leq 0.\ \ i = 1.2.\ldots.m \\ &h_j(x) = 0.\ \ j = 1.2.\ldots.p \end{aligned} \tag{1}$$

in which, x is the vector of decision variables; m is the number of inequality constraints and p is the number of equality constraints.

Multi-Objective Optimization problem involves optimizing two or more conflicting objectives simultaneously. In this approach, the objective functions $f_k(x)$ for $k = 1,2,\ldots,n$ are minimized (or maximized) while considering some equality or inequality constraints, [20]. The mathematical formulation can be expressed as follows:

$$\begin{aligned} &\min f_1(x).f_2(x).\ldots.f_n(x) \\ &subject\ to: \\ &g_i(x) \leq 0.\ \ i = 1.2.\ldots.m \\ &h_j(x) = 0.\ \ j = 1.2.\ldots.p \end{aligned} \tag{2}$$

where, $f_k(x)$ is the kth objective function.

In structural optimization problems, the primary aim is often to minimize weight [21-25]

[Downloaded from education.iust.ac.ir on 2025-03-10]

[DOI: 10.22068/ijoce.2024.14.4.606]

which can be defined mathematically as:

$$\min W = \sum_{i=1}^{N_d} \rho_i, A_i \sum_{j=1}^{N_t} L_j \tag{3}$$

where, ρ_i and A_i are the unit weight and length of the design section determined for member group i, respectively; N_t is the total number of all structural members in group i and L_j is the length of the *jth* member belonging to the *ith* group.

Additionally, important performance aspects such as displacement (or drift for buildings) need to be considered, which can be defined as a constraint for a single-objective problem:

- ***Displacement constraint***

$$C_D^t = \Delta_{MaxJ} - \Delta_{Max}^a \leq 0 \tag{4}$$

$$C_F^d = \left[\delta_J\right]_F - [\delta^a]_F \leq 0 \tag{5}$$

A comparison between the maximum lateral displacement of the considered structure in the Dth direction for $D = 1.2. \dots . ND$ under Jth design load combination, (Δ_{MaxJ}) regarding the maximum allowable lateral displacement (Δ_{Max}^a) is provided by Eq. (4). The Eq. (5) compares the inter-story drift of the Fth story for $F = 1.2. \dots . NF$ (NF is the total number of stories) under the Jth design load combination $\left[\delta_J\right]_F$ against the related permitted value $[\delta^a]_F$, [26].

- ***Strength requirements***

Based on the AISC-LRFD [27] code for steel structure design, the following constraints must be fulfilled for the design sections' strength requirements:

$$C_{IEL}^i = \left[\frac{P_{uJ}}{\varphi P_n}\right]_{IEL} + \frac{8}{9}\left(\frac{M_{uxJ}}{\varphi_b M_{nx}} + \frac{M_{uyJ}}{\varphi_b M_{ny}}\right)_{IEL} - 1 \leq 0 \quad for \quad \left[\frac{P_{uJ}}{\varphi P_n}\right]_{IEL} \geq 0{,}2 \tag{6}$$

$$C_{IEL}^i = \left[\frac{P_{uJ}}{2\varphi P_n}\right]_{IEL} + \left(\frac{M_{uxJ}}{\varphi_b M_{nx}} + \frac{M_{uyJ}}{\varphi_b M_{ny}}\right)_{IEL} - 1 \leq 0 \quad for \quad \left[\frac{P_{uJ}}{\varphi P_n}\right]_{IEL} < 0{,}2 \tag{7}$$

$$C_{IEL}^v = \frac{(V_{uJ})_{IEL}}{(\varphi_v V_n)_{IEL}} - 1 \leq 0 \tag{8}$$

where, IEL is the element number as $IEL = 1.2. \dots . NEL$ and NEL is the overall number of elements; J is the load combination number as $J = 1.2. \dots . N$ and N is the total number of all design load combinations; P_{uJ} is the required compressive or tensile (axial) strength, under Jth design load combination; M_{uxJ} and M_{uyJ} are the total flexural strengths required for bending of structural elements about x and y, under the Jth design load combination,

[Downloaded from education.iust.ac.ir on 2025-03-10]
[DOI: 10.22068/ijoce.2024.14.4.606]

respectively; where for strong and weak axes bending, the x and y subscripts utilized as the relating symbols, respectively. P_n, M_{nx} and M_{ny} are the nominal compressive or tensile (axial) and flexural (for bending of structural elements about x and y axes) strengths of the IELth member under consideration. φ is the axial strength's resistance factor formulated regarding to the yielding of the gross section (0.85 for compression and 0.9 for tension) and φ_b is the flexural resistance factor (0.9). V_{uJ} is the shear strength required under Jth design load combination and V_n is the nominal shear strength of the IELth considered member and φ_v is 0.9.

For multi-objective structural optimization problems, the objectives are often characterized by the need to balance several conflicting criteria simultaneously. These objectives typically include:

- Minimizing Weight: this objective is similar to the single objective problem and is defined by Eq. 3.
- Minimizing Lateral Displacement (Drift): Ensuring that the maximum lateral displacement under design loads is kept within allowable limits, which is crucial for the structural stability and comfort of occupants.

$$\min C_D^t = \Delta_{MaxJ} - \Delta_{Max}^a \, C_D^t \tag{9}$$

$$\min C_F^d = \left[\delta_J\right]_F - \left[\delta^a\right]_F \tag{10}$$

- Controlling strength requirements: In addition to these objectives, it is essential to ensure that the stresses within the structural elements remain within allowable limits to prevent failure and maintain structural integrity (Eqs. 6-8).

3. A REVIEW ON SINGLE- AND MULTI-OBJECTIVE CHAOS GAME OPTIMIZATION

Chaos Game Optimization (CGO) is inspired by the principles of chaos theory and fractal geometry [17,18]. This method uses chaotic game concepts to generate solutions for optimization problems, offering advantages in terms of exploration and exploitation of the solution space. Below, we explore the methods for single-objective and multi-objective optimization using CGO.

3.1 Single-Objective Chaos Game Optimization

The original CGO is developed for a single objective optimization problem. The method generally involves the following steps, [17,18]:

1. **Initialization**: A set of initial solutions (seeds) is generated randomly within the defined search space. These solutions represent candidate designs for the optimization problem.

[Downloaded from education.iust.ac.ir on 2025-03-10]

[DOI: 10.22068/ijoce.2024.14.4.606]

2. **Generate a Sierpinski triangle:** For each of the eligible seeds in the search space (X_i), a temporary triangle is drawn with three seeds as follows:

   - The position of the so far found Global Best (GB),
   - The position of the Mean Group (MG_i),
   - The position of the *ith* solution candidate (X_i) as the selected seed.

The GB refers to the so far found best solution candidate which has the highest eligibility levels and the MG_i refers to the mean values of some randomly selected eligible seeds with an equal probability of including the current considered initial eligible seed (X_i). The GB and MG_i alongside the selected eligible seed (X_i) are considered as three vertices of a Sierpinski triangle.

3. **Generating New Seeds**: The new seeds are generated based on the following four equations:

$$Seed_i^1 = X_i + \alpha_i \times (\beta_i \times GB - \gamma_i \times MG_i). \qquad i = 1.2. \dots .ns, \tag{11}$$

$$Seed_i^2 = GB + \alpha_i \times (\beta_i \times X_i - \gamma_i \times MG_i). \qquad i = 1.2. \dots .ns, \tag{12}$$

$$Seed_i^3 = MG_i + \alpha_i \times (\beta_i \times X_i - \gamma_i \times GB). \qquad i = 1.2. \dots .ns, \tag{13}$$

$$Seed_i^4 = X_i\left(x_i^k = x_i^k + R\right). \quad k = [1.2. \dots .d] \qquad i = 1.2. \dots .ns, \tag{14}$$

   where, X_i is the *ith* solution candidate, GB is the so far found global best, and MG_i is the mean value of some selected eligible seeds. α_i is the randomly generated factorial for modeling the movement limitations of the seeds while each of the β_i and γ_i represent a random integer of 0 or 1 for modeling the possibility of rolling a dice. k is a random integer in the interval of $[1, d]$ and R is a uniformly distributed random number in the interval of [0,1], [17].

4. **Evaluation**: Each new solution is checked for boundary condition and then it is evaluated based on the defined objective function and selected the constraint handling method.
5. **Update Mechanism**: The best-obtained solution is replaced with the worst one.
6. **Convergence**: The algorithm iteratively generates/updates solutions (steps 2-5) until convergence criteria are met (a predefined number of iterations). The best solution obtained at the end of this process is considered the optimal solution.

3.2 Multi-Objective Chaos Game Optimization

Multi-objective Chaos Game Optimization (CGO) extends the single-objective approach by handling multiple conflicting objectives simultaneously, [28,29]. While many of the steps—such as initialization, Sierpinski triangle generation, and seed generation equations—mirror those of the single-objective version, there are critical differences due to the nature of

[Downloaded from education.iust.ac.ir on 2025-03-10]

[DOI: 10.22068/ijoce.2024.14.4.606]

multi-objective optimization.

1. **Initialization**: As in the single-objective CGO, a set of initial seeds is generated randomly. However, in multi-objective problems, each seed will be evaluated based on multiple objectives rather than a single one.
2. **Sierpinski Triangle Construction and Seed Generation**: Similar to the single-objective approach, a Sierpinski triangle is created for each seed using the positions of the Global Best, the Mean Group, and a randomly selected candidate seed. The equations used to generate new seeds also remain the same.
3. **Evaluation and Pareto Dominance**: Unlike the single-objective version, where solutions are ranked based on one objective, the multi-objective CGO evaluates each solution against multiple objective functions. The concept of Pareto dominance is employed here, wherein a solution is considered Pareto optimal if no other solution can improve one objective without worsening another.
4. **Update Mechanism and Maintaining Pareto Front**: Instead of focusing on a single best solution, multi-objective CGO maintains a Pareto front—a set of non-dominated solutions. The Pareto front represents the best trade-offs between competing objectives. Solutions that do not belong to this front are updated or replaced, ensuring that only optimal trade-offs are retained.
5. **Convergence**: The algorithm continues iterating until the Pareto front stabilizes, either by reaching a maximum number of iterations or when the solution set shows little to no improvement in terms of Pareto optimality.

4. NUMERICAL STUDY

In this section, we evaluate the performance of single-objective and multi-objective Chaos Game Optimization (CGO) using a real-world benchmark structure widely recognized in structural optimization. The benchmark example is a 10-story building with 1024 structural elements, [5]. For this structure, the total number of design variables is 32. These design variables are grouped, meaning that a set of structural members, based on their geometry and function, share the same design section. This grouping significantly reduces the number of independent variables to optimize, while maintaining structural integrity and performance.

To ensure a fair comparison between single-objective and multi-objective optimization approaches, we have adopted a common penalty function for handling constraints, ensuring that both methods similarly penalize constraint violations. Additionally, the number of function evaluations is set to 15,000 for both methods to ensure consistency in computational effort. The other required parameters for both methods are also kept constant. By maintaining these consistent parameters across both methods, this study aims to provide a balanced comparison of single-objective and multi-objective optimization for real-world structural design problems.

It should be noted that the final results obtained by both methods should be double-checked to ensure that all constraints are satisfied. In the case of the single-objective algorithm, if the obtained optimum solution is feasible—i.e., all constraints are met—then the solution is considered the final design and can be implemented immediately without further adjustments.

[Downloaded from education.iust.ac.ir on 2025-03-10]

[DOI: 10.22068/ijoce.2024.14.4.606]

However, for the multi-objective method, an additional step is required: decision-making. In multi-objective optimization, the result is not a single solution but a set of Pareto-optimal solutions, each representing a different trade-off between competing objectives, such as weight and deflection. Selecting the most appropriate solution from this set involves carefully evaluating the trade-offs, often regarding design codes and project requirements. This decision-making process is crucial for ensuring that the chosen design balances performance, safety, and cost effectively, while still adhering to the required structural standards.

4.1 10-story Building

The ten-story frame consists of 1026 members [5]: 580 beams, 96 bracing elements, and 350 columns. Stability is achieved through inverted X-bracing and moment-resistant connections as shown in Fig. 1. For practical manufacturing, the members are grouped into 32 design variables based on elevation and plan. Columns are divided into five groups, beams into outer and inner groups, and bracings into one group. In total, there are 20 column groups, 8 beam groups, and 4 bracing groups. The ten design load combinations are considered including live loads on the floors and roof, along with uniformly distributed dead loads and the structure’s weight. Earthquake loads are also added, with base shear proportional to the structure's dead load. The beams are braced along their lengths by the floor system, while columns and bracings are unbraced. The effective length factor is set to 1 for beams and bracings, with specific consideration for buckling in the main direction for columns.

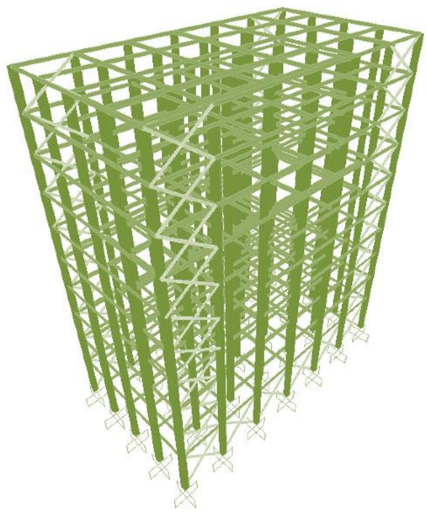


Figure 1: Schematic view of the 10-story building, [30]

4.2. Results and Discussion

Fig. 2 illustrates the convergence history of the single-objective CGO algorithm applied to the numerical example. The best result weighs 592 tons, with a maximum stress ratio of 0.9851 and a maximum drift of 0.9915, indicating that the final design is feasible. The

[Downloaded from education.iust.ac.ir on 2025-03-10]

[DOI: 10.22068/ijoce.2024.14.4.606]

algorithm required approximately 7500 evaluations to produce a design weighing less than 600 tons. The mean value across different runs was 634 tons, which is about 7% higher than the best result. The convergence pattern shows that after a certain number of evaluations, the algorithm consistently improved the design, indicating a balanced exploration of the design space and exploitation of promising solutions. Although convergence slowed down after reaching 600 tons, the algorithm demonstrated efficiency in refining the design further. Additionally, the low variance between the best and average results across runs suggests that the algorithm is robust, reliably producing near-optimal solutions with little deviation.

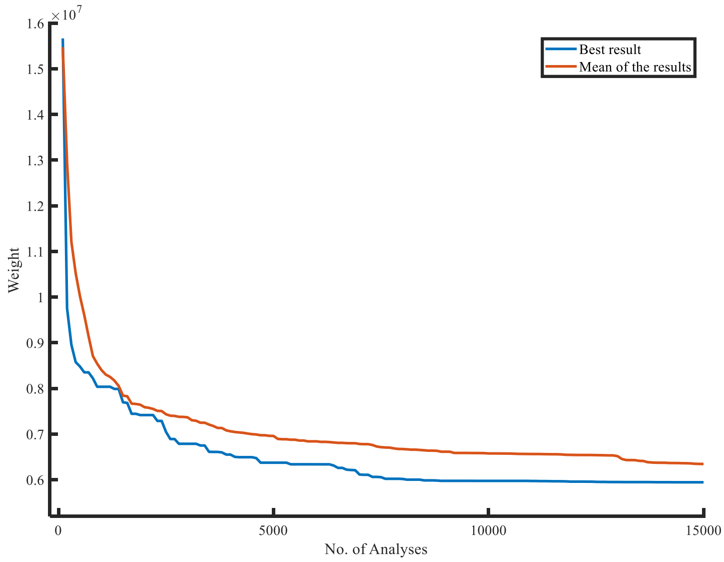


Figure 2: Convergence history obtained by the single-objective CGO

Fig. 3 presents the Pareto front generated by the multi-objective CGO for this example. As illustrated in Fig. 3(a), the structure's weight ranges from 530 tons to approximately 1500 tons, while the maximum drift varies between 0.5 and 1.44. Designs with a maximum drift exceeding 1 are not feasible from an engineering perspective, while designs with weights greater than 700 tons are inefficient. Therefore, as depicted in Fig. 3(b), the feasible/efficient solutions are concentrated around the region where the maximum drift is close to 1.

The designs within the red circle represent feasible options with a maximum drift value of less than 1, making them viable for immediate use. One significant advantage of the multi-objective method, compared to the single-objective approach, is that it offers multiple solutions, allowing the designer to compare them and choose the most appropriate one based on other considerations (such as practicality or market demands) not accounted for in the optimization process. Furthermore, some valuable solutions lie within the green circle, which represents designs with slightly better weight but minor drift violations. In certain scenarios, these designs can be considered acceptable or can be adjusted to meet the constraints. These solutions provide a broader perspective when compared to fully feasible ones. Another critical set of solutions is highlighted in the orange region of Fig. 3(b). For projects where safety is

[Downloaded from education.iust.ac.ir on 2025-03-10]

[DOI: 10.22068/ijoce.2024.14.4.606]

paramount, and a stricter design is preferred, these solutions offer more conservative options by avoiding designs that push the drift limit. This region provides safer, more reliable solutions.

a)

b)

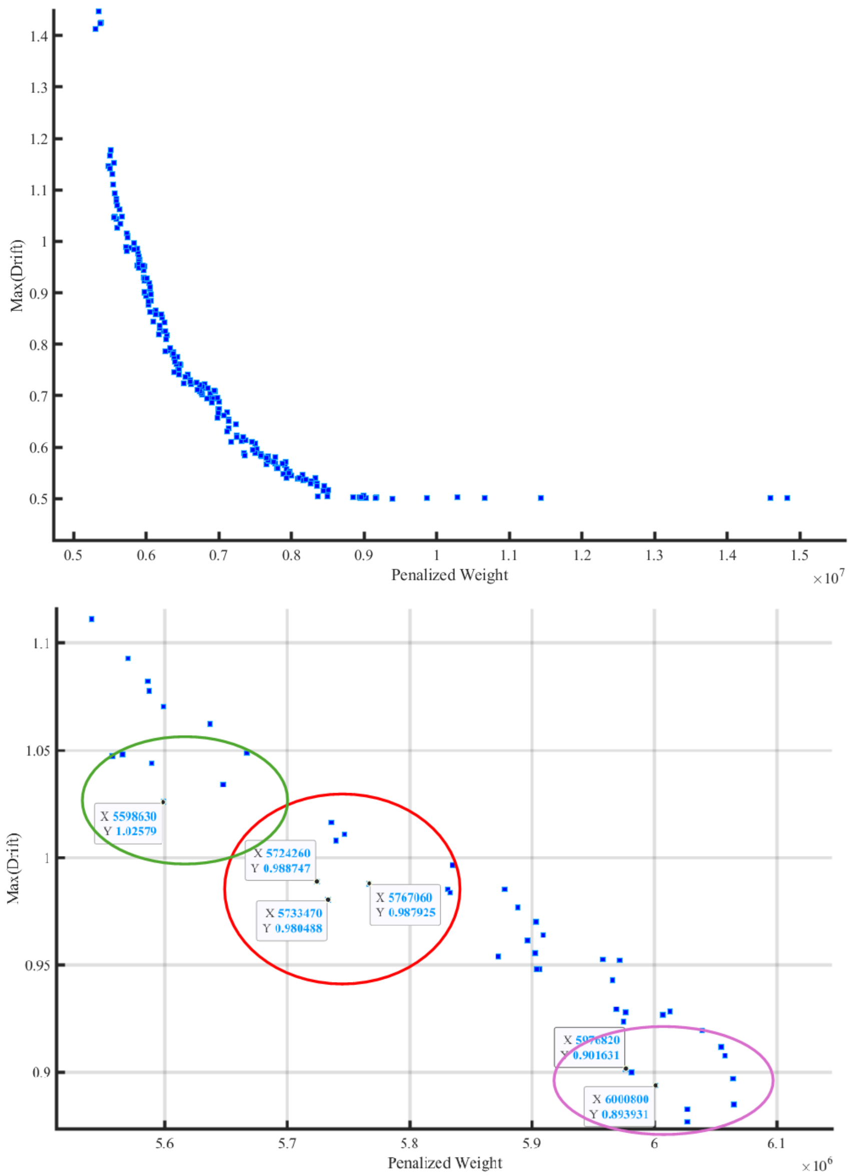


Figure 3: Pareto front obtained by the multi-objective CGO

Table 1 presents the best 30 results (based on weight) obtained by the multi-objective algorithm. The optimal feasible result weighs 572 tons, which is almost 20 tons lighter than the result obtained by the single-objective method. Therefore, the multi-objective approach

[Downloaded from education.iust.ac.ir on 2025-03-10]

[DOI: 10.22068/ijoce.2024.14.4.606]

not only offers flexibility in decision-making but also delivers more efficient designs in terms of weight optimization.

Table 1: The best 30 results (based on weight) obtained by the multi-objective algorithm

No.	Max Ratio	Max Drift	Weight	No.	Max Ratio	Max Drift	Weight
1	1.41	1.00	527.5	16	1.08	0.93	558.6
2	1.44	0.96	534.2	17	1.07	0.95	558.7
3	1.42	0.97	537.2	18	1.04	0.91	558.9
4	1.42	0.98	537.7	19	1.02	0.89	559.9
5	1.14	0.97	548.0	20	1.07	0.93	559.9
6	1.16	0.98	549.6	21	1.06	0.92	563.7
7	1.14	0.93	550.0	22	1.03	0.90	564.7
8	1.17	0.99	551.7	23	1.04	0.90	566.7
9	1.13	0.94	552.9	24	**0.98**	**0.90**	**572.4**
10	1.11	0.94	554.1	25	**0.98**	**0.89**	**573.3**
11	1.15	0.95	555.2	26	1.01	0.93	573.5
12	1.15	0.95	555.4	27	**1**	**0.91**	**574.0**
13	1.04	0.99	555.7	28	1.01	0.91	5747161
14	1.04	0.99	556.5	29	**0.98**	**0.89**	**576.7**
15	1.09	0.96	557.0	30	**0.98**	**0.87**	**583.1**

It is important to note that the computational cost is higher for the multi-objective algorithm. This is due to the need to archive many points on the Pareto front, along with the additional computations required for operations like dominance sorting. Despite the higher cost, the multi-objective approach provides greater flexibility and appears to be a valuable trade-off. When comparing this with the single-objective result, the single-objective CGO focuses solely on minimizing weight, which limits the design to a single feasible solution. While the single-objective CGO ensures a highly optimized result for the given objective, it lacks the flexibility and variety of solutions that a multi-objective method offers, where designers can consider trade-offs between weight and other factors. In practical terms, the multi-objective approach delivers a range of solutions, providing the designer with the flexibility to evaluate multiple factors beyond the optimization criteria. This includes safety, practical constraints, and other real-world considerations, ensuring a well-rounded and informed decision-making process.

5. CONCLUSIONS

This study presents a comprehensive comparison between single-objective and multi-objective optimization approaches for structural design using the Chaos Game Optimization (CGO) algorithm. Through the analysis of a real-size benchmark building, the results

[Downloaded from education.iust.ac.ir on 2025-03-10]

[DOI: 10.22068/ijoce.2024.14.4.606]

demonstrate the distinct advantages and trade-offs of each method. The single-objective CGO proved effective in minimizing weight, providing a singular optimal solution that satisfies all constraints. However, it lacks the flexibility to explore various trade-offs between conflicting objectives, which are often crucial in real-world engineering scenarios. In contrast, the multi-objective CGO not only optimizes weight but also incorporates other essential factors such as structural stability and drift. This method delivers a set of Pareto-optimal solutions, allowing designers to balance competing objectives and select designs that best meet practical requirements, safety considerations, and market constraints. Despite the higher computational cost associated with multi-objective optimization, its ability to provide a range of feasible solutions offers significant value, enabling more informed and flexible decision-making in structural design. Ultimately, this research highlights the importance of using multi-objective optimization in engineering, where trade-offs between various factors need to be considered. To sum up, while the multi-objective approach provides a broader perspective, enhanced efficiency, and more practical solutions—making it a valuable tool for modern structural optimization tasks—there are significant challenges associated with controlling constraints and tuning parameters. Additionally, defining objectives in a manner that the optimization algorithm can effectively manage, while also ensuring that the solutions produced are usable and feasible, presents a key issue when employing multi-objective algorithms.

REFERENCES

1. Kaveh A. Advances in metaheuristic algorithms for optimal design of structures. Springer; 2014.
2. Yang XS, Talatahari S, Alavi AH. Metaheuristic applications in structures and infrastructures. Newnes; 2013.
3. Kirsch U. Structural optimization: fundamentals and applications. Springer Science & Business Media; 2012.
4. Zakian P, Kaveh A. Seismic design optimization of engineering structures: a comprehensive review. *Acta Mech*. 2023;**234**(4):1305-30.
5. Sadrekarimi N, Talatahari S, Azar BF, Gandomi AH. A surrogate merit function developed for structural weight optimization problems. *Soft Comput*. 2023;**27**:1533-63.
6. Kaveh A, Izadifard RA, Mottaghi L. Optimal design of planar RC frames considering CO2 emissions using ECBO, EVPS and PSO metaheuristic algorithms, *J Build Eng.* 2020;**28**, 101014.
7. Kaveh A, Khanzadi M, Alipour M. Fuzzy resource constraint project scheduling problem using CBO and CSS algorithms. *Int J Civil Eng.* 2016;**14**(5), 325–337.
8. Kaveh A, Rajabi F. Fuzzy-multi-mode Resource-constrained Discrete Time-cost-resource Optimization in Project Scheduling Using ENSCBO. *Periodica Polytechnica Civil Eng.* 2022;**66**(1), 50–62.
9. Kaveh, A, Vazirinia, Y. Chaotic vibrating particles system for resource-constrained project scheduling problem. *Scientia Iranica*, 2021;**27**(4A), 1826–1842.
10. Georgioudakis M, Plevris V. A comparative study of differential evolution variants in constrained structural optimization. *Front Built Environ*. 2020;**6**:102.

[Downloaded from education.iust.ac.ir on 2025-03-10]

[DOI: 10.22068/ijoce.2024.14.4.606]

11. Kaveh, A, Fahimi-Farzam, M, Kalateh-Ahani, M. Performance-based multi-objective optimal design of steel frame structures: Nonlinear dynamic procedure. *Scientia Iranica*. 20215;**22**(2), 373–387.
12. Kaveh, A. Laknejadi, K, Alinejad, B. Performance-based multi-objective optimization of large steel structures. *Acta Mech.* 2012;**223**(2), 355–369.
13. Kaveh A, Kalateh-Ahani M, Fahimi-Farzam M. Constructability optimal design of reinforced concrete retaining walls using a multi-objective genetic algorithm. *Struct Eng Mech.* 2013;**47**(2), 227-245.
14. Wu J, Sigmund O, Groen JP. Topology optimization of multi-scale structures: a review. *Struct Multidiscip Optim*. 2021;**63**:1455-80.
15. Zavala GR, Nebro AJ, Luna F, Coello CAC. A survey of multi-objective metaheuristics applied to structural optimization. *Struct Multidiscip Optim*. 2014;**49**:537-58.
16. Deb K, Sindhya K, Hakanen J. Multi-objective optimization. In: Decision sciences. CRC Press; 2016:161-200.
17. Talatahari S, Azizi M. Optimization of constrained mathematical and engineering design problems using chaos game optimization. *Comput Ind Eng*. 2020;**145**:106560.
18. Talatahari S, Azizi M. Chaos game optimization: a novel metaheuristic algorithm. *Artif Intell Rev*. 2021;**54**(2):917-1004.
19. Diwekar UM. Introduction to applied optimization. Vol 22. Springer Nature; 2020.
20. Marler RT, Arora JS. Survey of multi-objective optimization methods for engineering. *Struct Multidiscip Optim*. 2004;**26**:369-95.
21. Haftka RT, Gürdal Z. Elements of structural optimization. Sci Business Media; 2012,**11**.
22. Kaveh A, Dadras A., Malek, NG. Robust Design Optimization of Multilayered Sandwich Panel under Uncertain Bounded Buckling Loads, *Structural and Multidisciplinary Optimization*, 2018.
23. Kaveh A, Ilchi Ghazaan M. A new VPS-based algorithm for multi-objective optimization problems, *Eng Comput.* 2020;**36**:1029–1040
24. Kaveh A, Massoudi MS. Multi-objective optimization using Charged System Search, *Scientia Iranica*, 2014; **6**(21), 1845-1860.
25. Kaveh A, Talatahari S. Hybrid charged system search and particle swarm optimization for engineering design problems, *Eng Comput.* 2011;**28** (4), 423-440 (2011).
26. Talatahari S, Azizi M. Optimal design of real-size building structures using quantum-behaved developed swarm optimizer. *Struct Des Tall Spec Build*. 2020;**29**(11):e1747.
27. Committee A. Specification for structural steel buildings (ANSI/AISC 360-10). American Institute of Steel Construction; 2010.
28. Jafari A, Saraee M, Nouhi B, Shishehgarkhaneh MB, Talatahari S. Optimum design of dam structures using multi-objective chaos game optimization algorithm. In: Hybrid Metaheuristics in Structural Engineering: Including Machine Learning Applications. Springer; 2023:127-44.
29. Yacoubi S, Manita G, Chhabra A, Korbaa O, Mirjalili S. A multi-objective chaos game optimization algorithm based on decomposition and random learning mechanisms for numerical optimization. *Appl Soft Comput*. 2023;**144**:110525.
30. Kaveh A, Khodadadi N, Azar BF, Talatahari S. Optimal design of large-scale frames with an advanced charged system search algorithm using box-shaped sections. *Eng Comput*. 2021;**37**:2521-41.

[Downloaded from education.iust.ac.ir on 2025-03-10]

[DOI: 10.22068/ijoce.2024.14.4.606]