

Investigating the Ability of Grey Wolf Optimizer to Solve Pressure Vessel Optimization Design Problem

Ismail Nasrat¹, Dawood Rahmani²

¹Assistant Professor, Department of Mathematics & Physics, Faculty of Engineering, Balkh University, Mazar-e-Sharif, Balkh, AFGHANISTAN.

²Assistant Professor, Department of Mathematics, Faculty of Education, Badakhshan University, Badakhshan, AFGHANISTAN.

¹Corresponding Author: ismaielnasrat@gmail.com

ORCID

<https://orcid.org/0009-0006-5294-2915>



www.jrasb.com || Vol. 4 No. 2 (2025): April Issue

Received: 06-05-2025

Revised: 08-05-2025

Accepted: 10-05-2025

ABSTRACT

This study investigates the effectiveness of the Grey Wolf Optimizer (GWO), a nature-inspired metaheuristic algorithm, in solving the complex and constrained pressure vessel design optimization problem. The primary objective is to minimize the cost and weight of a pressure vessel while satisfying multiple nonlinear constraints related to material thickness, radius, and length. The GWO algorithm is benchmarked against several other well-known metaheuristic algorithms, including DE, GA, PSO, and CMA-ES, under identical experimental settings. To handle constraints, the death penalty method is applied. Results demonstrate that GWO performs competitively, ranking second overall in solution quality and convergence speed, surpassed only by Differential Evolution (DE). The findings highlight GWO's potential as a robust and efficient optimization tool for solving real-world engineering design problems with complex search landscapes.

Keywords- Grey Wolf Optimizer, Metaheuristic Algorithm, Pressure Vessel Design, Constrained Optimization, Engineering Design.

I. INTRODUCTION

Metaheuristic optimization algorithms are advanced problem-solving methods used to find near-optimal solutions for complex problems where traditional techniques are slow or ineffective. While they don't guarantee perfect results, they deliver good solutions efficiently, making them ideal for large, constrained search spaces. Popular methods include Genetic Algorithms, Simulated Annealing, Particle Swarm Optimization, and Tabu Search. These approaches are especially valuable in engineering design tasks—like structural or electrical system optimization—where traditional methods struggle due to complexity,

multiple objectives, and nonlinearity. As such, metaheuristics have become essential tools in modern engineering.

Metametaheuristic optimization algorithms can be broadly categorized into five classes. Evolutionary techniques, inspired by biological evolution, use operations like crossover and mutation to evolve a population of solutions—key examples include Evolutionary Strategy (ES) [3], Genetic Algorithm (GA) [4], Differential Evolution (DE) [5], Biogeography-Based Optimization (BBO) [6], Genetic Programming (GP) [7], and CMA-ES [8]. Swarm intelligence methods mimic collective behaviors of animals or insects, where individual agents cooperate to find solutions; notable

examples are PSO [9], ACO [10], ABC [11,12], SMO [13], GWO [14], FA [15], CS [16,17], and COA [18]. Physics-based algorithms simulate physical processes, such as GSA [19], HSA [20], SA [21], BBBC [22], CSS [23], and CFO [24]. Human-based techniques are inspired by learning and behavior, including TLBO [25], HS [26], GSO [27], and TS [28]. Finally, mathematics-based approaches apply mathematical functions to guide the search process, including SCA [29], SSO [30], AOA [31], SFS [32], and TSA [33].

In this paper, we investigate the effectiveness of the Grey Wolf Optimizer (GWO) in solving the pressure vessel engineering design problem. The study focuses on evaluating GWO's ability to handle the problem's nonlinear, constrained, and multi-variable nature, aiming to find cost-effective and feasible design solutions.

The remainder of this paper is organized as follows: Section 2 provides an overview of the Grey Wolf Optimizer (GWO) algorithm. In Section 3, the pressure vessel design problem is introduced in detail, including its objective function and associated constraints. Section 4 discusses the experimental setup, results, and comparative performance analysis. Finally, Section 5 offers concluding remarks and summarizes the key findings of the study.

II. GREY WOLF OPTIMIZER (SCA)

The Grey Wolf Optimizer (GWO) is a nature-inspired metaheuristic algorithm that mimics the hunting behavior of grey wolves in a pack. It was proposed by Seyedali Mirjalili in 2013 [14] and has gained popularity due to its simplicity and effectiveness in solving optimization problems. In GWO, a population of candidate solutions, represented as grey wolves, is iteratively updated to search for the optimal solution. The algorithm imitates the social hierarchy and hunting behavior of wolves to balance exploration and exploitation during the search process. The GWO algorithm consists of four main steps: initialization, updating the position of alpha, beta, and delta wolves, updating the position of other wolves, and boundary handling. The mathematical model of GWO consists of three main steps: encircling prey, hunting, and attacking (updating positions). Here's a summary of the mathematical formulation:

Encircling: Wolves estimate the position of prey and move around it. The encircling behavior is mathematically modeled as:

$$\vec{D} = |\vec{C} \cdot \vec{X}_p(t) - \vec{X}(t)| \quad (1)$$

$$\vec{X}(t+1) = \vec{X}_p(t) - \vec{A} \cdot \vec{D} \quad (2)$$

Where $\vec{X}_p(t)$ is the position vector of the prey (best solution found so far), $\vec{X}(t)$ is the position vector of a grey wolf, $\vec{A}$ and $\vec{C}$ are coefficient vectors and finally $\vec{D}$ is the distance between the wolf and the prey. The vectors $\vec{A}$ and $\vec{C}$ are defined as follow:

$$\vec{A} = 2\vec{a} \vec{r}_1 - \vec{a}, \vec{C} = 2\vec{r}_2 \quad (3)$$

Where $\vec{a}$ linearly decreases from 2 to 0 over the course of iterations, $\vec{r}_1$ and $\vec{r}_2$ are random vectors in [0,1].

Hunting: The GWO algorithm assumes that the **alpha** (α), **beta** (β), and **delta** (δ) wolves guide the rest of the wolves toward the prey. The positions are updated based on these three leaders:

$$\vec{D}_\alpha = |\vec{C} \cdot \vec{X}_\alpha - \vec{X}|$$

$$\vec{D}_\beta = |\vec{C} \cdot \vec{X}_\beta - \vec{X}|$$

$$\vec{D}_\delta = |\vec{C} \cdot \vec{X}_\delta - \vec{X}|$$

$$\vec{X}_1 = \vec{X}_\alpha - \vec{A}_\alpha \cdot \vec{D}_\alpha$$

$$\vec{X}_2 = \vec{X}_\beta - \vec{A}_\beta \cdot \vec{D}_\beta$$

$$\vec{X}_3 = \vec{X}_\delta - \vec{A}_\delta \cdot \vec{D}_\delta$$

$$\vec{X}(t+1) = \frac{\vec{X}_1 + \vec{X}_2 + \vec{X}_3}{3} \quad (4)$$

Attacking Prey: As $\vec{a} \rightarrow 0$, the wolves exploit the best solutions by reducing their movement range, allowing convergence toward the prey. This mathematical model enables the GWO algorithm to balance **exploration** and **exploitation**, efficiently searching the solution space.

The pseudo code of GWO is given as follows:

Algorithm: Pseudo code of GWO

Introduce the variable bounds and required parameters.

Generate the initial populations.

Calculate the fitness

While t <= T

Find the three best solution

For i=1:Population size

Update the parameters r_1 and r_2

Update the position X_i using (4)

End if

Check the bounds

Calculate the fitness value

Apply the greedy selection

End for

End while

Report the best solution and fitness

III. PRESSURE VESEL DESIGN PROBLEM

Pressure vessels are critical structures used to store or transport high-pressure fluids and gases. Their design must ensure safety, prevent leaks or failures, and enhance operational efficiency by reducing energy loss, maintenance, and downtime. Effective design also cuts material and manufacturing costs while ensuring durability and performance. The pressure vessel design optimization problem aims to minimize weight and cost while meeting safety and performance standards. Key factors include:

- **Material selection:** Strong, corrosion-resistant, cost-effective, and lightweight.
- **Shape optimization:** Efficient stress distribution with minimal material use.
- **Thickness optimization:** Safe wall thickness with reduced weight and cost.
- **Performance criteria:** Compliance with stress limits, fatigue life, and pressure containment.
- **Cost efficiency:** Balancing material, manufacturing, and operational costs.

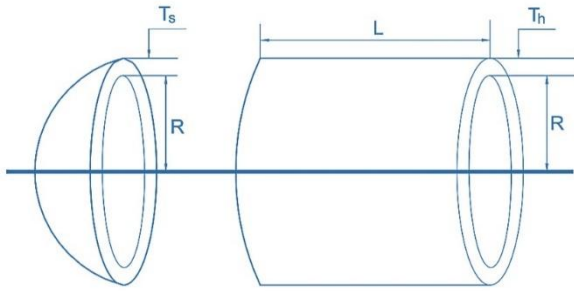


Figure 1. The schematic view of presser vessel [37].

Figure 1 shows a schematic representation of the pressure vessel design problem, which aims to minimize welding, material, and forming costs [34]. The problem involves four decision variables: head thickness (T_h), shell thickness (T_s), length of the cylindrical section excluding the head (L), and inner radius (R). It also includes four constraints, with a full mathematical formulation available in [35]. Optimal solutions obtained using various optimization algorithms are listed in Tables 3 and 4, while the convergence of the objective function, $f(\vec{x})$, is illustrated in Figures 6 to 8. The mathematical model is defined as follows:

Let $[x_1, x_2, x_3, x_4] = [T_s, T_h, R, L]$

$$f(\vec{x}) = 0.6224x_1x_3x_4 + 1.7781x_2x_3^2 + 3.1661x_1^2x_4 + 19.84x_1^2x_3$$

Subject to

$$g_1(\vec{x}) = -x_1 + 0.0193x_3 \leq 0,$$

$$g_2(\vec{x}) = -x_2 + 0.00954x_3 \leq 0,$$

$$g_3(\vec{x}) = -\pi x_3^2x_4 - \frac{4}{3}\pi x_3^3 + 1296000 \leq 0,$$

$$g_4(\vec{x}) = x_4 - 240 \leq 0,$$

Variable range $0 \leq x_1, x_2 \leq 99, 10 \leq x_3, x_4 \leq 200$.

To address this optimization problem, engineers use advanced computational methods like finite element analysis (FEA) and optimization algorithms to evaluate and refine design options based on specific objectives and constraints. It's also essential to account for real-world factors such as manufacturing limitations, inspection needs, and regulatory standards to ensure the final design is both practical and safe.

IV. EXPERIMENTAL RESULTS

This section evaluates the effectiveness and applicability of the Grey Wolf Optimizer (GWO) in

addressing the pressure vessel design problem, a complex constrained engineering task that poses significant challenges for optimization techniques. To manage the design constraints, an appropriate constraint-handling mechanism is essential within the optimization process. Although various methods exist in the literature, this study adopts the simplest approach known as the death penalty method to perform the experiments [36]. Detailed parameter settings and experimental background are the same as Sabery et al. [40].

Parameter Settings: In the experiments conducted in this section, the maximum number of iterations was set to 500, and the population size was 30. It is important to note that the population size in CMA-ES is defined as its standard formula, which results in a value of 50, where D is the problem's dimension, set to 4. To ensure reliability, each algorithm was executed 10 times to identify the optimal solution. The experiments were carried out using MATLAB, with the experimental setup and algorithm control parameters identical to those used by Sabery et al. [40].

V. RESULTS AND DISCUSSIONS

This section presents the experimental results obtained using the Grey Wolf Optimizer (GWO) alongside various other metaheuristic algorithms. To assess the effectiveness of GWO, its performance was compared with several well-established optimization techniques, including ABC, ACO, BBO, CMA-ES, CS, DE, GA, GSA, HSA, PSO, SSO, TLBO, and TSA. A detailed comparison between SCA and these algorithms is summarized in Table 1. As shown in Table 1, Differential Evolution (DE) achieved the highest overall performance, with GWO ranking as the second-best performing algorithm.

Figures 2 illustrate the convergence behavior of GWO in comparison with other algorithms, showing that GWO steadily converges toward a near-optimal solution. Initially, GWO emphasizes exploration and gradually shifts to exploitation, as reflected in the convergence trend depicted in the boxplot in Figure 3.

Table 1. The optimum table of pressure vessel problem.

Algo	Decision Variables				$f(\vec{x})$
	T_s	T_h	R	L	
ABC	53.414	43.323	59.826	118.526	6742.04
ACO	17.532	8.931	56.602	60.8887	6814.89
BBO	17.340	8.917	56.048	63.042	6840.89
CMA	15.080	8.370	47.118	136.588	6932.20
ES	12.961	7.190	42.392	173.519	6070.14
CS	12.961	7.190	42.392	173.519	6070.14
DE	13.024	6.968	42.098	176.636	6059.71
GA	16.196	8.072	52.351	86.1028	6531.35
GSA	16.148	12.139	50.149	98.2968	7671.27
GWO	13.076	6.937	42.4143	173.147	6064.99
HAS	19.312	9.356	56.705	58.280	7452.66
JA	14.204	7.432	45.235	150.894	6310.27

PSO	16.923	8.360	53.538	86.768	6735.20
SCA	13.872	7.415	44.246	156.280	6240.96
SSO	13.772	7.177	44.007	156.230	6159.71
TLBO	14.736	7.704	47.269	136.561	6321.79
TSA	17.970	9.015	57.438	62.2124	6987.91

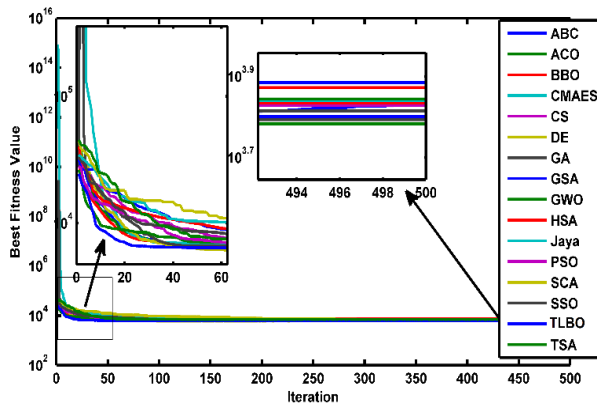


Figure 2. Convergence curve of algorithms

Table 2 presents a summary of the search agent success rates for GWO, SCA, Jaya, SSO, and DE across different population sizes. During each iteration of the algorithms, search agents attempt to update their positions. However, due to the greedy selection mechanisms employed by many algorithms, these updates often fail. To evaluate how dynamic each algorithm is, it is essential to measure how frequently search agents are successfully updated throughout the optimization process.

To quantify this, a trial counter was assigned to each search agent. When an agent is successfully updated, its counter is incremented by one; otherwise, it remains unchanged. If the total number of iterations is denoted by T , and a search agent is updated T_a times, the success rate is calculated as $(T_a/T) \times 100\%$, following the method defined by Sabery et al. [40].

In Figure 3, the horizontal axis represents population size, while the vertical axis shows the success rate of each algorithm. According to Table 2, GWO's search agent success rate does not exceed 4%, whereas DE achieves an average success rate of around 16%. Figure 3 illustrates a comparative view of the success rates for SCA, GWO, SSO, Jaya, and DE across various population sizes.

Table 2. Search agent success rate

Pop	Optimization Algorithms				
	SCA	GWO	Jaya	SSO	DE
$N = 5$	4.178	3.576	28.08	4.008	13.54
$N = 10$	3.914	3.496	12.77	4.164	25.85
$N = 20$	3.898	2.955	8.628	4.179	17.155
$N = 30$	3.784	3.046	4.892	4.064	16.18
$N = 40$	3.928	2.913	3.489	4.221	15.83
$N = 50$	3.802	2.8372	3.584	4.1388	15.57

$N = 60$	3.764	2.798	2.972	4.1503	15.43
$N = 70$	3.806	2.644	1.894	4.36	15.46
$N = 80$	3.762	2.715	2.001	4.069	15.21
$N = 90$	3.802	2.613	2.057	4.191	15.30
$N = 100$	3.7834	2.6044	1.9772	4.1136	15.4038

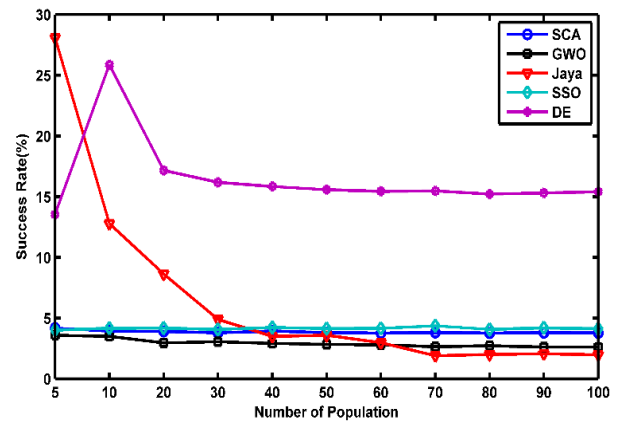


Figure 3. GWO search agent success rate

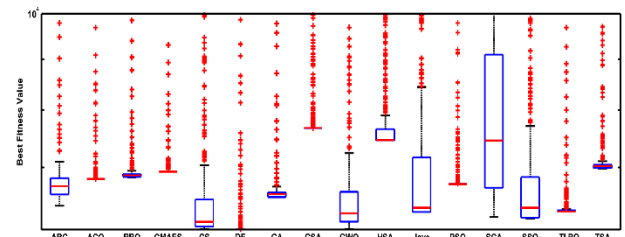


Figure 4. Boxplot of algorithms performance

VI. CONCLUSION

This study thoroughly investigated the application of the Grey Wolf Optimizer (GWO) to solve the pressure vessel design problem a classic example of a complex, constrained, nonlinear engineering optimization task. The results demonstrate that GWO is not only capable of effectively navigating the multidimensional and constrained design space but also consistently converges toward high-quality solutions. By mimicking the leadership hierarchy and cooperative hunting strategies of grey wolves, the algorithm balances exploration and exploitation, enabling efficient optimization. Through comparative analysis with other well-known metaheuristic algorithms such as DE, PSO, GA, CMA-ES, and others, GWO achieved the second-best performance, following Differential Evolution (DE), highlighting its competitive robustness and reliability. Additionally, the convergence plots confirm the algorithm's stability and efficiency in reaching near-optimal results over multiple runs.

In conclusion, the Grey Wolf Optimizer proves to be a promising and efficient tool for solving real-world engineering optimization problems like pressure vessel design. Its simplicity, adaptability, and strong

performance in constrained environments make it a valuable addition to the suite of nature-inspired optimization methods. Future work may involve hybridizing GWO with other algorithms or enhancing its constraint-handling mechanisms to further improve solution quality and computational efficiency.

REFERENCES

- [1] Bansal, Jagdish Chand, Pramod Kumar Singh, and Nikhil R. Pal, eds. *Evolutionary and swarm intelligence algorithms*. Vol. 779. Cham: Springer, 2019.
- [2] Yang, Xin-She. "Nature-inspired optimization algorithms: Challenges and open problems." *Journal of Computational Science* 46 (2020): 101104. <https://doi.org/10.1016/j.jocs.2020.101104>.
- [3] Rechenberg, Ingo. "Evolutionsstrategie." *Optimierung technischer Systeme nach Prinzipien der biologischen Evolution* (1973).
- [4] Holland, John H. *Adaptation in natural and artificial systems: an introductory analysis with applications to biology, control, and artificial intelligence*. MIT press, 1992.
- [5] Storn, Rainer, and Kenneth Price. "Differential evolution—a simple and efficient metaheuristic for global optimization over continuous spaces." *Journal of global optimization* 11 (1997): 341-359. <https://doi.org/10.1023/A:1008202821328>
- [6] Simon, Dan. "Biogeography-based optimization." *IEEE transactions on evolutionary computation* 12, no. 6 (2008): 702-713. DOI: 10.1109/TEVC.2008.919004
- [7] Koza, John R., and James P. Rice. "Automatic programming of robots using genetic programming." In *AAAI*, vol. 92, pp. 194-207. 1992.
- [8] Hansen, Nikolaus, Andreas Ostermeier, and Andreas Gawelczyk. "On the Adaptation of Arbitrary Normal Mutation Distributions in Evolution Strategies: The Generating Set Adaptation." In *ICGA*, pp. 57-64. 1995.
- [9] Kennedy, James, and Russell Eberhart. "Particle swarm optimization." In *Proceedings of ICNN'95-international conference on neural networks*, vol. 4, pp. 1942-1948. IEEE, 1995. DOI: 10.1109/ICNN.1995.488968
- [10] Dorigo, M., and T. Stützle. *"Ant Colony Optimization, Bradford Publisher."* (2004).
- [11] Karaboga, Dervis, and Bahriye Basturk. "A powerful and efficient algorithm for numerical function optimization: artificial bee colony (ABC) algorithm." *Journal of global optimization* 39 (2007): 459-471. <https://doi.org/10.1007/s10898-007-9149-x>
- [12] Karaboga, Dervis. *An idea based on honey bee swarm for numerical optimization*. Vol. 200. Technical report-tr06, Erciyes university, engineering faculty, computer engineering department, 2005.
- [13] Bansal, Jagdish Chand, Harish Sharma, Shimpi Singh Jadon, and Maurice Clerc. "Spider monkey optimization algorithm for numerical optimization." *Memetic computing* 6 (2014): 31-47. <https://doi.org/10.1007/s12293-013-0128-0>
- [14] Mirjalili, Seyedali, Seyed Mohammad Mirjalili, and Andrew Lewis. "Grey wolf optimizer." *Advances in engineering software* 69 (2014): 46-61. <https://doi.org/10.1016/j.advengsoft.2013.12.007>
- [15] Yang, Xin-She. "Firefly algorithm, stochastic test functions and design optimisation." *International journal of bio-inspired computation* 2, no. 2 (2010): 78-84. <https://doi.org/10.1504/IJBIC.2010.032124>
- [16] Yang, Xin-She, and Suash Deb. "Cuckoo search via Lévy flights." In *2009 World congress on nature & biologically inspired computing (NaBIC)*, pp. 210-214. Ieee, 2009. DOI: 10.1109/NABIC.2009.5393690
- [17] Yang, Xin-She, and Suash Deb. "Engineering optimisation by cuckoo search." *International Journal of Mathematical Modelling and Numerical Optimisation* 1, no. 4 (2010): 330-343. <https://doi.org/10.1504/IJMMNO.2010.03543>
- [18] Rajabioun, Ramin. "Cuckoo optimization algorithm." *Applied soft computing* 11, no. 8 (2011): 5508-5518. <https://doi.org/10.1016/j.asoc.2011.05.008>
- [19] Rashedi, Esmat, Hossein Nezamabadi-Pour, and Saeid Saryazdi. "GSA: a gravitational search algorithm." *Information sciences* 179, no. 13 (2009): 2232-2248. <https://doi.org/10.1016/j.ins.2009.03.004>
- [20] Yang, Xin-She. "Harmony search as a metametaheuristic algorithm." *Music-inspired harmony search algorithm: theory and applications* (2009): 1-14. https://doi.org/10.1007/978-3-642-00185-7_1
- [21] Kirkpatrick, Scott, C. Daniel Gelatt Jr, and Mario P. Vecchi. "Optimization by simulated annealing." *science* 220, no. 4598 (1983): 671-680. DOI: 10.1126/science.220.4598.671
- [22] Erol, Osman K., and Ibrahim Eksin. "A new optimization method: big bang–big crunch." *Advances in engineering software* 37, no. 2 (2006): 106-111. <https://doi.org/10.1016/j.advengsoft.2005.04.005>

- [23] Kaveh, A., and Siamak Talatahari. "A novel metaheuristic optimization method: charged system search." *Acta mechanica* 213, no. 3-4 (2010): 267-289. DOI: <https://doi.org/10.1007/s00707-009-0270-4>
- [24] Formato, Richard. "Central force optimization: a new metametaheuristic with applications in applied electromagnetics." *Progress in electromagnetics research* 77 (2007): 425-491. doi:10.2528/PIER07082403
- [25] Rao, R. Venkata, Vimal J. Savsani, and D. P. Vakharia. "Teaching-learning-based optimization: an optimization method for continuous non-linear large scale problems." *Information sciences* 183, no. 1 (2012): 1-15. <https://doi.org/10.1016/j.ins.2011.08.006>
- [26] Zong Woo Geem, Joong Hoon Kim, Loganathan GV. A New Metaheuristic Optimization Algorithm: Harmony Search. *SIMULATION*. 2001;76(2):60-68. doi:10.1177/003754970107600201
- [27] S. He, Q. H. Wu and J. R. Saunders, "Group Search Optimizer: An Optimization Algorithm Inspired by Animal Searching Behavior," in *IEEE Transactions on Evolutionary Computation*, vol. 13, no. 5, pp. 973-990, Oct. 2009, doi: 10.1109/TEVC.2009.2011992.
- [28] Fred Glover, (1989) Tabu Search—Part I. *ORSA Journal on Computing* 1(3):190-206. <https://doi.org/10.1287/ijoc.1.3.190>
- [29] Mirjalili, Seyedali. "SCA: a sine cosine algorithm for solving optimization problems." *Knowledge-based systems* 96 (2016): 120-133. <https://doi.org/10.1016/j.knosys.2015.12.022>
- [30] Zhao, J., Tang, D., Liu, Z. et al. Spherical search optimizer: a simple yet efficient meta-metaheuristic approach. *Neural Comput & Applic* 32, 9777-9808 (2020). <https://doi.org/10.1007/s00521-019-04510-4>
- [31] Abualigah, Laith, Ali Diabat, Seyedali Mirjalili, Mohamed Abd Elaziz, and Amir H. Gandomi. "The arithmetic optimization algorithm." *Computer methods in applied mechanics and engineering* 376 (2021): 113609. <https://doi.org/10.1016/j.cma.2020.113609>
- [32] Salimi, Hamid. "Stochastic fractal search: a powerful metametaheuristic algorithm." *Knowledge-based systems* 75 (2015): 1-18. <https://doi.org/10.1016/j.knosys.2014.07.025>
- [33] Layeb, A. Tangent search algorithm for solving optimization problems. *Neural Comput & Applic* 34, 8853-8884 (2022). <https://doi.org/10.1007/s00521-022-06908-z>
- [34] Rao, Singiresu S. *Engineering optimization: theory and practice*. John Wiley & Sons, 2019.
- [35] Gandomi, A.H., Yang, X.S., Alavi, A.H. et al. Bat algorithm for constrained optimization tasks. *Neural Comput & Applic* 22, 1239-1255 (2013). <https://doi.org/10.1007/s00521-012-1028-9>
- [36] Corne, David, Marco Dorigo, Fred Glover, Dipankar Dasgupta, Pablo Moscato, Riccardo Poli, and Kenneth V. Price, eds. *New ideas in optimization*. McGraw-Hill Ltd., UK, 1999.
- [37] Varaee, Hesam, Naser Safaeian Hamzehkolaei, and Mahsa Safari. "A hybrid generalized reduced gradient-based particle swarm optimizer for constrained engineering optimization problems." *Journal of Soft Computing in Civil Engineering* 5, no. 2 (2021): 86-119.
- [38] URL: https://scholar.google.com/citations?view_op=view_citation&hl=en&user=TJHmrREAAAJ&citation_for_view=TJHmrREAAAJ:k_IJM867U9cC (Accessed, December 4, 2023).
- [39] URL: <https://forgedcomponents.com/what-are-some-of-the-uses-for-pressure-vessels/> (Accessed, December 4, 2023)
- [40] Sabery, G. A., Danishyar, G. H., & Osmani, M. A. (2024). Evaluation the performance of sin e cosine algorithm in solving pressure vessel engineering design problem. *J. Res. Appl. Sci. Biotechnol.*, 3(3), 38-46.