International Journal of Information Technology and Computer Engineering
Vol: 06, No. 01, Jan-June 2026
ISSN: 2455-5290, DOI: 10.55529/ijitc.61.78.93

Research Paper

Constrained pressure-vessel design optimization using the scuba diver optimization algorithm: parameter sensitivity and statistical validation

Saman M. Almufti[1*], Amira Bibo Sallow[2]
[1*,2]Department of Information Technology, Technical College of Duhok, Duhok Polytechnic University, Duhok, Iraq.
[1]Department of Information Technology, Technical College of Informatics-Akre, Akre University for Applied Sciences, Duhok, Iraq.

Article Info

Article History:
Received: 28 December 2025
Revised: 07 March 2026
Accepted: 14 March 2026
Published: 30 April 2026

Keywords:
Constrained Engineering Optimization
Mixed-Discrete Optimization
Pressure-Vessel Design
Scuba Diver Optimization Algorithm
Parameter Sensitivity
Nonparametric Statistics

ABSTRACT

The pressure-vessel benchmark is a mixed-discrete nonlinear engineering problem in which shell and head thicknesses follow manufacturing increments while radius and cylindrical length remain continuous. This study applies the Scuba Diver Optimization Algorithm (SDOA) to the standard formulation and examines ten parameter configurations through 30 independent runs each. An oxygen-regulated state model assigns global exploration, moderate exploration, exploitation, fine tuning, or reset, while a repair procedure enforces the 0.0625-in thickness increment, variable bounds, minimum wall thicknesses, and required volume. All 300 runs ended with feasible designs. S7 achieved the best mean rank (4.000), the smallest standard deviation (0.055056), a mean cost of 6059.766062, and a 73.3% success rate within 0.001% of the proven optimum. S9 produced the closest individual design, with cost 6059.714482843 and a relative gap of 0.000002439%. Friedman and Holm-adjusted Wilcoxon analyses confirmed significant configuration effects. The results show that parameter selection mainly affects repeated-run consistency and search-stage allocation rather than attainable best-case accuracy.

Corresponding Author:
Saman M. Almufti
Department of Information Technology, Technical College of Duhok, Duhok Polytechnic University, Duhok, Iraq.
Email: Saman.Almofty@gmail.com

1. INTRODUCTION

Pressure-vessel design requires simultaneous decisions about material thickness, geometry, fabrication, capacity, and safety. The classical benchmark minimizes the manufacturing cost of a cylindrical shell with hemispherical heads. Its design vector contains shell thickness, head thickness, inner radius, and cylindrical length. The two thicknesses are restricted to commercial plate increments, whereas the geometric dimensions are continuous. This mixed-discrete structure, combined with nonlinear stress and volume constraints, makes the problem a standard test for engineering optimization [1].

The standard formulation has a proven global minimum of 6059.714335048436 at a design located close to active shell-thickness and volume constraints [2]. This reference is important because the literature includes results obtained under altered bounds, continuous thickness assumptions, or incomplete feasibility enforcement. A numerical result below the proven mixed-discrete optimum is not evidence of improvement unless the same manufacturing increments, length bound, and original inequalities are satisfied. Therefore, the present study states the variable domains, repair rules, feasibility tolerance, and reference optimum explicitly.

Constraint handling is central to this benchmark. Penalty methods convert violations to a scalar addition, but their behaviour depends on coefficient scaling and on the different numerical magnitudes of the constraints [3]. Feasibility rules avoid part of this sensitivity by preferring feasible candidates, comparing objective values among feasible candidates, and comparing violation magnitudes among infeasible candidates [4]. Reviews of constrained evolutionary optimization also show that repair, ranking, and penalty design can affect results as strongly as the movement operator itself [5]. The SDOA implementation therefore uses problem-structured repair followed by verification with the original inequalities. Metaheuristics are attractive for such problems because they do not require derivatives and can combine global movement, information sharing, and local refinement [6]. Their stochastic nature, however, prevents reliable conclusions from a single best run. The no-free-lunch principle further implies that no optimizer is uniformly superior over all problem classes [7]. Consequently, this paper evaluates ten SDOA parameter settings through 30 independent runs each and reports distributions, convergence, feasibility, runtime, run-wise ranks, and nonparametric tests.

The study has four objectives. First, it gives a reproducible pressure-vessel formulation of SDOA, including integer-thickness repair and feasibility-first acceptance. Second, it measures the sensitivity of SDOA to population size, iteration budget, current strength, mutation rate, oxygen decay, and elite size. Third, it links final performance to the internal use of global exploration, moderate exploration, exploitation, fine tuning, and reset. Fourth, it selects a configuration using repeated-run evidence rather than isolated best-case accuracy. Sections 2 and 3 present related work and methodology separately, as required by the journal format. Section 4 reports the results and discussion, and Section 5 concludes the paper.

2. RELATED WORK

2.1 Metaheuristics for Constrained Mechanical Design

Differential evolution (DE) forms trial vectors from scaled population differences and has become a standard continuous optimizer [8]. Particle swarm optimization (PSO) updates particles from personal and social information [9]. Teaching-learning-based optimization (TLBO) uses teacher and learner phases and has been applied directly to constrained mechanical design [10]. These methods represent different information-transfer mechanisms, but each depends on boundary treatment and feasibility handling when applied to the pressure-vessel benchmark.

Later population-based algorithms introduced state-dependent movement. Grey Wolf Optimizer (GWO) uses a leadership hierarchy and encircling model [11]. The Whale Optimization Algorithm (WOA) alternates shrinking encirclement, spiral movement, and exploration [12]. The Sine Cosine Algorithm (SCA) controls movement around the current best through trigonometric functions [13]. The Salp Swarm Algorithm (SSA) combines leader movement with a follower chain [14], while Harris Hawks Optimization

(HHO) switches among exploration and exploitation strategies according to an energy model [15]. These algorithms show that changing operator behaviour over time or state can be useful, but published best values remain meaningful only when the same mixed-discrete formulation is enforced.

2.2 Methodological Requirements for Metaheuristic Assessment

Recent methodological discussions emphasize transparent algorithms, controlled computational budgets, repeated runs, and disclosure of parameter settings. Nature-inspired optimization remains useful, but new metaphors do not replace the need for identifiable operators and reproducible experiments [16]. Nonparametric statistical tests are recommended because stochastic optimizer outputs are often non-normal, heteroscedastic, or affected by ties [17]. Multiple-comparison procedures are also necessary when many configurations are examined simultaneously [18].

General reviews describe metaheuristics as combinations of diversification, intensification, memory, and adaptation rather than as isolated biological analogies [19]. Taxonomies of bio-inspired methods similarly classify algorithms by their search mechanisms and population structure [20]. Previous constrained-design studies have shown that optimizer performance depends on the interaction between movement rules and feasibility treatment [21]. Historical surveys also indicate that many modern methods reuse established mechanisms such as random perturbation, elitism, local search, and information sharing under new state models [22]. SDOA is evaluated here at this operational level: its claimed contribution is the coordination of known search functions by oxygen-regulated stage allocation, not the metaphor alone.

2.3 Research Gap

The pressure-vessel literature contains many best-run comparisons, but fewer studies examine how parameter choices alter reliability and stage usage under one fixed implementation. A configuration may reach the optimum once while producing a weak mean, wide dispersion, or frequent near-feasible outcomes. The present work addresses this gap by comparing ten SDOA configurations with matched seed identifiers, complete feasibility verification, convergence records, and paired statistical analysis. The experiment also records the percentage of diver-iterations assigned to each stage, enabling the numerical results to be interpreted through the actual internal search schedule.

A further gap concerns comparability across pressure-vessel studies. Some articles report continuous thicknesses even though the standard benchmark requires multiples of 0.0625 in; others use a 240-in length bound, omit the head-thickness restriction, or accept small negative volume margins as feasible. These differences can produce costs below the proven optimum without solving the same problem. The present analysis avoids that ambiguity by retaining the standard 200-in upper bound, repairing thicknesses to the manufacturing grid, and evaluating every final design with the original four inequalities. The reported comparison is therefore formulation specific rather than a general claim against all variants of the benchmark.

The related literature also indicates that constraint handling and local refinement are often responsible for much of the performance attributed to a named optimizer. Differential evolution, swarm methods, and teaching-learning approaches can all perform well when their candidate solutions are repaired or ranked consistently. For this reason, the present methodology separates three elements: the SDOA movement rules, the oxygen-based allocation of those rules, and the pressure-vessel repair procedure. This separation permits the observed accuracy to be interpreted conservatively. The results demonstrate the behaviour of the complete implementation; they do not imply that oxygen regulation alone accounts for every improvement.

3. METHODOLOGY

3.1 Pressure-Vessel Mathematical Model

The general constrained minimization problem and the specific pressure-vessel objective are written as follows. The design vector is x = [T_s, T_h, R, L], where T_s and T_h are the shell and head thicknesses, R is the inner radius, and L is the cylindrical length.

$$\min_{\mathbf{x}} f(\mathbf{x}) \quad \text{subject to} \quad g_j(\mathbf{x}) \le 0,\ j = 1, \dots, m, \quad \mathbf{x}^L \le \mathbf{x} \le \mathbf{x}^U. \tag{1}$$

$$f(\mathbf{x}) = 0.6224 T_s R L + 1.7781 T_h R^2 + 3.1661 T_s^2 L + 19.84 T_s^2 R. \tag{2}$$

$$g_1(\mathbf{x}) = -T_s + 0.0193 R \le 0. \tag{3}$$

$$g_2(\mathbf{x}) = -T_h + 0.00954 R \le 0. \tag{4}$$

$$g_3(\mathbf{x}) = -\pi R^2 L - \frac{4\pi}{3} R^3 + 1{,}296{,}000 \le 0. \tag{5}$$

$$g_4(\mathbf{x}) = L - 240 \le 0. \tag{6}$$

$$0.0625 \le T_s, T_h \le 6.1875, \qquad 10 \le R, L \le 200. \tag{7}$$

$$T_s = 0.0625 k_s, \qquad T_h = 0.0625 k_h, \qquad k_s, k_h = 1, 2, \dots. \tag{8}$$

The objective combines shell material, head material, forming, and welding terms. Constraints (3) and (4) impose minimum wall thicknesses, (5) enforces the required internal volume, and (6) retains the conventional maximum length. The discrete rule in (8) forces both thickness variables to multiples of 0.0625 in. Table 1 summarizes the domains, and Figure 1 shows the corresponding pressure-vessel geometry and design variables.

Table 1. Pressure-Vessel Design Variables and Domains

Variable	Meaning	Type	Lower	Upper	Rule
T_S	Shell Thickness (In)	Discrete	0.0625	6.1875	0.0625 Increment
T_H	Head Thickness (In)	Discrete	0.0625	6.1875	0.0625 Increment
R	Inner Radius (In)	Continuous	10	200	Bounded
L	Cylindrical Length (In)	Continuous	10	200	Bounded

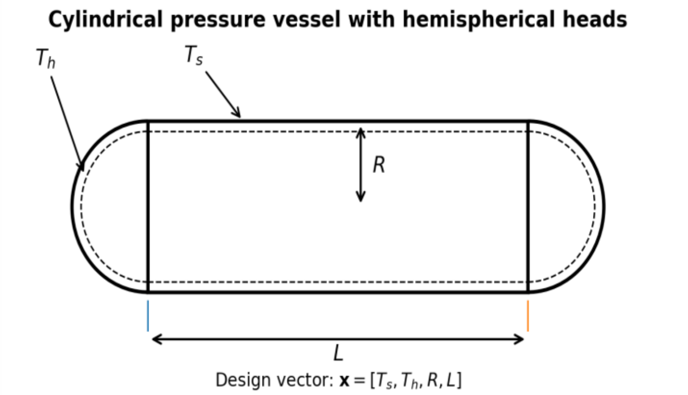

Figure 1. Pressure-Vessel Geometry and Design Variables

3.2 SDOA Search Concept and Population Representation

Each diver represents one candidate design vector. The algorithm stores the position, objective value, normalized violation, oxygen level, and current depth stage of every diver. The best feasible or least-penalized diver is maintained as the global guide. The population is initialized through a stratified distribution in each variable, after which one third of the candidates are replaced by uniformly random designs. For variable j, the stratified initialization is

$$x_{i,j}^{(0)} = L_j + \left(i - 1 + r_{i,j}\right) \frac{U_j - L_j}{N}, \quad i = 1, \dots, N, \tag{9}$$

Where$r_{i,j} \sim U(0,1)$, N is the number of divers, and L_j and U_j are the variable bounds. The initial candidates are repaired before evaluation.

3.3 Adaptive Current, Mutation, Oxygen, and Depth States

The current strength decreases quadratically:

$$c_t = c_0\left(1 - \frac{t}{T}\right)^2, \tag{10}$$

Where c_0 is the initial current strength, t is the current iteration, and T is the maximum iteration count. The mutation probability decreases linearly:

$$\mu_t = \mu_0\left(1 - \frac{t}{T}\right). \tag{11}$$

For non-elite divers, oxygen is updated before stage assignment as

$$O_i^{(t)} = O_i^{(t-1)}\exp\left(-\alpha\frac{t}{T}\right), \tag{12}$$

Where α is the oxygen-decay coefficient. Accepted candidates receive an oxygen increment of 5, capped at the initial level. A reset restores the oxygen level to its initial value. Elite positions and their stored oxygen values are preserved at the end of each iteration.

The depth stage is selected by

$$D_i^{(t)} = \begin{cases} 1, & O_i^{(t)} > 75, \\ 2, & 55 < O_i^{(t)} \leq 75, \\ 3, & 35 < O_i^{(t)} \leq 55, \\ 4, & 15 < O_i^{(t)} \leq 35, \\ 5, & O_i^{(t)} \leq 15. \end{cases} \tag{13}$$

The thresholds connect the physiological state to five movement modes. Oxygen recovery and reset allow a diver to return to an earlier stage, so stage usage is not constrained to a one-way progression.

3.4 Stage-Specific Search Operators

3.4.1 D1 Global Exploration

D1 applies a Cauchy-like heavy-tailed step:

$$\mathrm{x}_i' = \mathrm{x}_i + c_t(\mathrm{U} - \mathrm{L}) \odot \tan[\pi(\mathrm{r} - 0.5)]\left(1 - \frac{t}{T}\right), \tag{14}$$

Where **r**a vector of independent uniform is random values and $\odot$ denotes component-wise multiplication. The heavy tail permits occasional long movements, while the time factor contracts the step toward the end of the run.

3.4.2 D2 Moderate Exploration

With probability 0.7, D2 forms a convex crossover with the global best:

$$\mathrm{x}_i' = \eta \mathrm{x}_i + (1 - \eta)\mathrm{x}_{best}, \quad \eta \sim U(0,1). \tag{15}$$

Otherwise, it applies a bounded random walk:

$$\mathrm{x}_i' = \mathrm{x}_i + c_t(\mathrm{U} - \mathrm{L}) \odot (2\mathrm{r} - 1). \tag{16}$$

The two alternatives combine guided relocation with moderate unguided movement.

3.4.3 B1 Exploitation

B1 applies coordinate-wise local search. For each dimension, positive and negative trial movements are evaluated. The thickness step is exactly 0.0625 for T_s andT_h, while the continuous-variable step decreases from 2% of the corresponding range to a minimum fraction near the final iteration. The best local candidate is retained. With probability 0.6, the locally improved design is then crossed with a randomly selected elite diver. This stage therefore uses deterministic neighborhood checking followed by elite information exchange.

3.4.4 D3 Fine-Tuning

When a uniform random value is below μ_t, D3 applies non-uniform mutation. For variable j,

$$\Delta_{j,t} = 1 - r^{(1-t/T)^b}, \qquad x_{i,j}{}' = \begin{cases} x_{i,j} + \Delta_{j,t}(U_j - x_{i,j}), & r_1 < 0.5, \\ x_{i,j} - \Delta_{j,t}(x_{i,j} - L_j), & r_1 \geq 0.5, \end{cases} \tag{17}$$

With $b = 5$. The mutation magnitude approaches zero as t approaches T.

3.4.5 Reset Diversification

At stage 5, a diver is reinitialized with probability 0.4 or whenever its stored violation is positive. The repaired new design replaces the current candidate only if it improves the feasibility-first penalized fitness. Reset therefore introduces diversity without automatically discarding a better incumbent.

3.5 Global-Best Communication and Elitism

After the stage operator and greedy acceptance, a non-best diver can sample a neighborhood of the global best. The communication candidate is

$$\mathrm{x}_{comm} = \mathrm{x}_{best} + 0.1(\mathrm{U} - \mathrm{L}) \odot (\mathrm{r} - 0.5). \tag{18}$$

The communication probability is $0.6 O_i / O_0$, so divers with higher oxygen communicate more frequently. The candidate is repaired and accepted only if its penalized fitness is better. Before the divers are updated, the best K candidates are stored. Their positions and oxygen levels are restored after the population loop, which provides explicit elitism.

3.6 Pressure-Vessel Repair and Feasibility Ranking

The repair procedure first clips each variable to its bounds and rounds the two thickness variables to the nearest 0.0625-in increment. If the current radius does not provide the required volume, the minimum required cylindrical length is

$$L_{req} = \frac{1{,}296{,}000 - \frac{4\pi}{3} R^3}{\pi R^2}. \tag{19}$$

When $L_{req} \leq 200$, L is increased to at least this value. If $L_{req} > 200$, a bisection search increases R until the required volume is attained at $L = 200$. The minimum discrete thicknesses are then enforced:

$$T_s \leftarrow \max\left(T_s, 0.0625\mathrm{ceil}\left(\frac{0.0193R}{0.0625}\right)\right), \quad T_h \leftarrow \max\left(T_h, 0.0625\mathrm{ceil}\left(\frac{0.00954R}{0.0625}\right)\right). \tag{20}$$

Positive constraint violations are normalized to prevent the volume constraint from dominating solely because of its scale:

$$CV(\mathbf{x}) = \frac{\max(g_1,0)}{6.1875} + \frac{\max(g_2,0)}{6.1875} + \frac{\max(g_3,0)}{1{,}296{,}000} + \frac{\max(g_4,0)}{240}, \tag{21}$$

The scalar ranking value is

$$F(\mathbf{x}) = \begin{cases} f(\mathbf{x}), & CV(\mathbf{x}) \leq 10^{-12}, \\ 10^{10} + 10^6 CV(\mathbf{x}) + f(\mathbf{x}), & \text{otherwise}. \end{cases} \tag{22}$$

The large offset ensures that every feasible design is preferred to every infeasible design within the stated bounds. Final feasibility is verified using the original constraints, not the repaired state alone.

3.7 Algorithm Sequence and Computational Complexity

Figure 2 summarizes the complete procedure. The nominal population-iteration work is $N \times T$, but the actual objective-evaluation count is larger because B1 local search evaluates up to two candidates per dimension and the communication step can introduce another evaluation. With a fixed four- dimensional pressure-vessel model, the dominant computational order remains approximately $O(NTD)$, where $D = 4$. The repair bisection uses at most 60 inexpensive volume calculations when the current radius is insufficient at the maximum length.

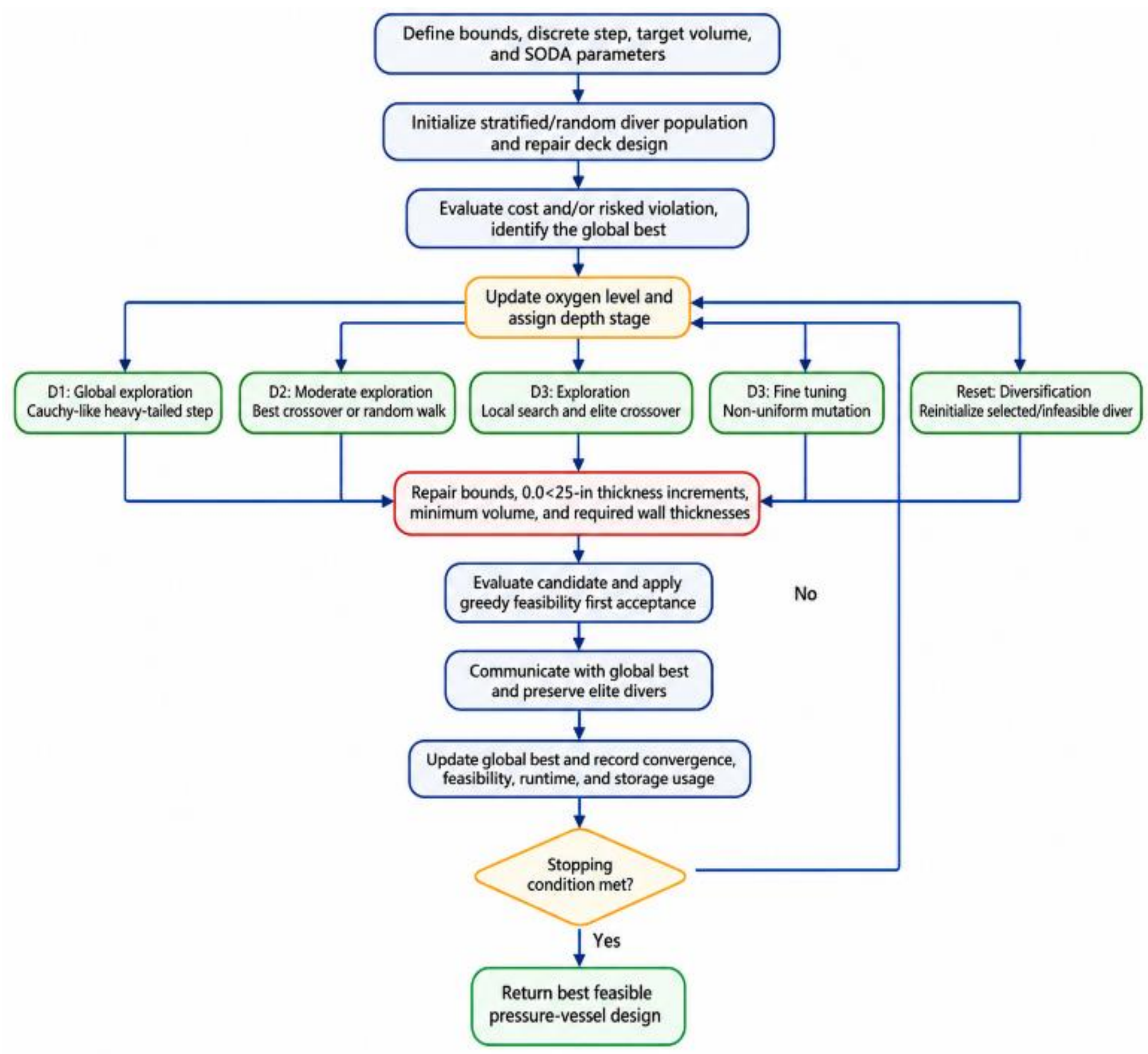

Figure 2. Computational Sequence of SDOA for Mixed-Discrete Pressure-Vessel Design

- Define the pressure-vessel objective, constraints, bounds, manufacturing increment, and SDOA parameters.
- Initialize, repair, and evaluate the diver population; then identify the initial global best and elite set.
- At each iteration, update the adaptive current strength, mutation probability, and oxygen levels.
- Assign every non-elite diver to D1 global exploration, D2 moderate exploration, B1 exploitation, D3 fine-tuning, or reset diversification.
- Apply the selected stage operator, repair the candidate, and evaluate its objective and normalized constraint violation.
- Apply feasibility-first greedy acceptance, oxygen recovery for accepted candidates, and probabilistic communication with the global best.
- Restore elite positions and oxygen states, update the global best, and record convergence, feasibility, runtime, and stage usage.
- Repeat until the maximum iteration count is reached and return the best feasible pressure-vessel design.

3.8 Experimental Design

Ten parameter configurations were selected to vary population size, iteration budget, current strength, mutation rate, oxygen decay, and elite size. Each setting was executed 30 times, producing 300 final observations. Seed identifiers 20260701-20260730 were reused across configurations as reproducible paired blocks. The experiments used an equation-equivalent compiled Python/Numba implementation because MATLAB and GNU Octave were unavailable in the execution environment. Compilation overhead was excluded from timing; runtime is therefore interpreted only as a relative measure within this experiment. Table 2 lists the ten parameter configurations examined in this study.

Table 2. SDOA Parameter Configurations

Set	Divers	Iter.	Current	Mutation	O2 Decay	Elite
S1	50	300	0.20	0.50	0.30	8
S2	60	400	0.15	0.40	0.25	12
S3	80	400	0.30	0.70	0.45	16
S4	100	400	0.30	0.80	0.50	20
S5	100	500	0.50	0.90	0.35	15
S6	100	500	0.18	0.50	0.70	25
S7	150	400	0.30	0.80	0.50	30
S8	80	600	0.30	0.80	0.50	16
S9	120	500	0.60	1.00	0.80	20
S10	150	200	0.50	0.40	0.20	20

Performance indicators were best, mean, median, standard deviation, interquartile range, worst value, relative gap from the proven optimum, success rate at three thresholds, convergence, feasibility, mean runtime, stage usage, and run-wise rank. The Friedman test assessed the omnibus null hypothesis of equal configuration performance [23]. Paired Wilcoxon signed-rank tests compared the selected setting with each alternative [24] and adjusted probabilities controlled multiplicity. Parameter-sensitivity analysis follows the same principle used in earlier constrained-design studies: conclusions should not depend on an unexamined single configuration [25].

4. RESULTS AND DISCUSSION

4.1 Descriptive Performance

All 300 runs ended with feasible designs. Every configuration produced a best value within 0.000114% of the proven optimum, indicating that repair consistently directed the population toward the narrow feasible region. The principal differences occurred in mean cost, dispersion, and the frequency of high-precision outcomes. Table 3 orders the settings by mean rank.

Table 3. Final-Cost Statistics Over 30 Runs

Set	Best	Mean	Median	SD	Mean Rank	Rank
S7	6059.715226	6059.766062	6059.737500	0.055056	4.000	1
S8	6059.714922	6059.794485	6059.745750	0.144997	4.033	2
S9	6059.714483	6059.777551	6059.747478	0.085912	4.333	3
S5	6059.715210	6059.788058	6059.751045	0.133933	4.567	4
S2	6059.721210	6059.883818	6059.772992	0.227922	5.933	5
S10	6059.714621	6059.998294	6059.821093	0.450105	6.200	6
S6	6059.714922	6060.007037	6059.793927	0.369014	6.267	7
S4	6059.715293	6059.873046	6059.798165	0.217239	6.367	8
S1	6059.715051	6059.952426	6059.804950	0.316203	6.600	9
S3	6059.715043	6060.036843	6059.802556	0.653482	6.700	10

S7 obtained the best mean rank (4.000), a mean cost of 6059.766062, and the smallest standard deviation (0.055056). S8 ranked second and had a competitive median, but one weaker run increased its standard deviation. S9 produced the lowest individual cost and remained strong by mean rank. S3 had the largest dispersion and the weakest rank despite a highly accurate best run. This contrast confirms that a single minimum is insufficient for selecting a dependable configuration. Figure 3 shows the mean convergence of the ten configurations, and Figure 4 shows the distribution of final cost error over the 30 runs.

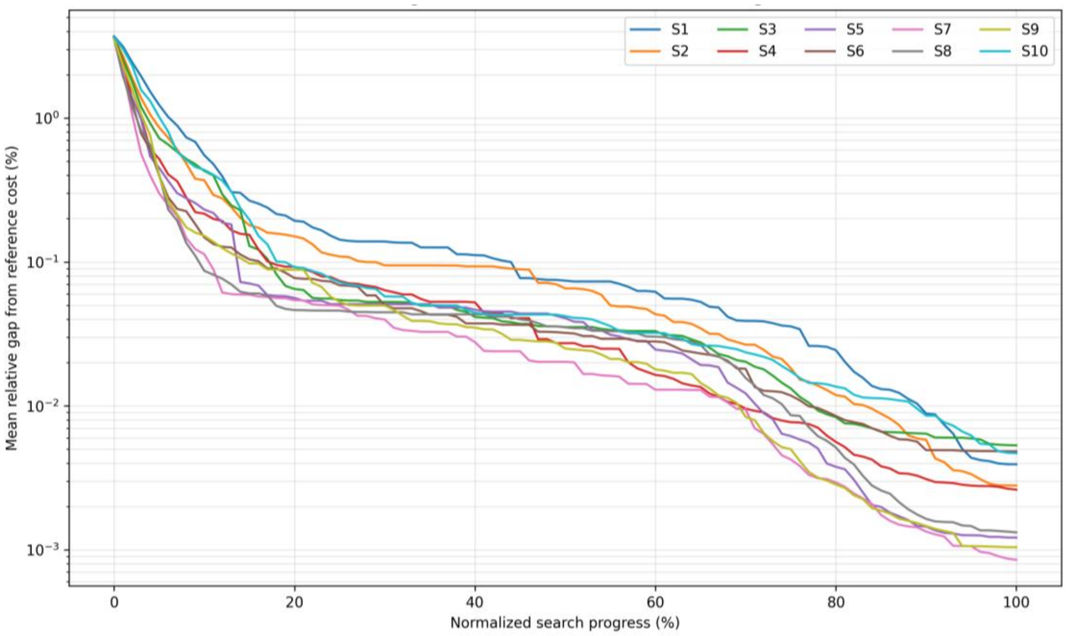

Figure 3. Mean Convergence of the Ten Sdoa Parameter Configurations

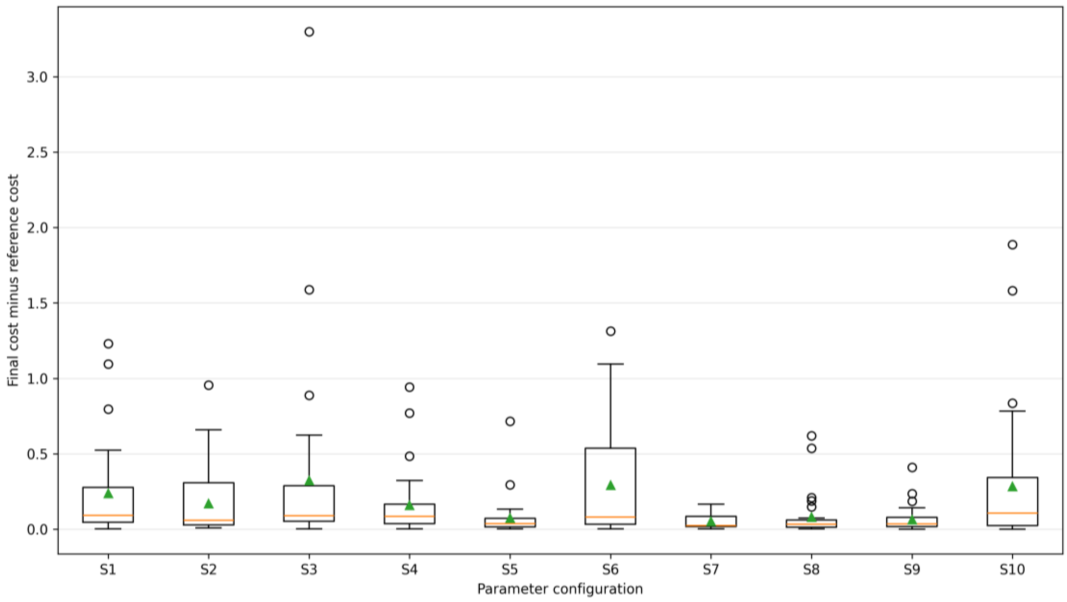

Figure 4. Distribution of Final Cost Error Over 30 Runs

4.2 Best Designs and Constraint Activity

The best designs from all settings selected T_s = 0.8125 and T_h = 0.4375, the same discrete thicknesses as the proven optimum. Remaining differences were concentrated in the final digits of R and L along the active volume boundary. S9 produced the closest design: T_s = 0.8125, T_h = 0.4375, R = 42.098445382, and L = 176.636603697. Its cost was 6059.714482843, only 0.000147794 above the proven optimum, corresponding to a relative gap of 0.000002439%. Table 4 reports the best solution obtained by each configuration.

Table 4. Best Solution Obtained by Each Configuration

Set	Cost	Gap (%)	T_S	T_H	R	L	Seed
S1	6059.715051267	0.000011819	0.8125	0.4375	42.098439715	176.636668717	20260706
S2	6059.721210162	0.000113456	0.8125	0.4375	42.098389144	176.637295378	20260703
S3	6059.715042995	0.000011683	0.8125	0.4375	42.098439783	176.636667875	20260725
S4	6059.715292858	0.000015806	0.8125	0.4375	42.098437731	176.636693299	20260720
S5	6059.715210054	0.000014440	0.8125	0.4375	42.098438411	176.636684873	20260703
S6	6059.714921869	0.000009684	0.8125	0.4375	42.098440777	176.636655551	20260722
S7	6059.715225823	0.000014700	0.8125	0.4375	42.098438282	176.636686478	20260702
S8	6059.714922364	0.000009692	0.8125	0.4375	42.098440773	176.636655601	20260703
S9	6059.714482843	0.000002439	0.8125	0.4375	42.098445382	176.636603697	20260708
S10	6059.714620630	0.000004713	0.8125	0.4375	42.098443251	176.636624900	20260710

Constraint-utilization analysis showed that the shell-thickness and volume constraints were effectively active at the best design, while the head-thickness and maximum-length conditions retained slack. This pattern agrees with the mathematical structure of the global optimum: cost decreases as material thickness and volume-related dimensions approach their minimum feasible values. The repair mechanism did not bypass the constraints; it reconstructed candidates near these boundaries and then evaluated the original inequalities. Figure 5 shows the constraint-utilization profile for the closest design.

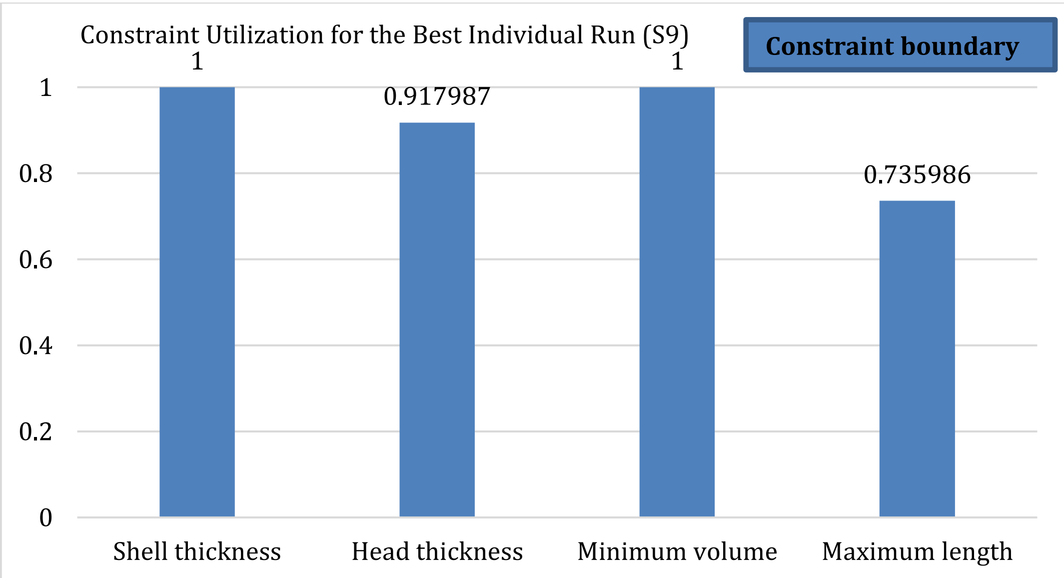

Figure 5. Constraint Utilization for the Closest SDOA Design

4.3 Success Rates, Statistical Tests, and Runtime

Success rates separate attainable accuracy from consistency. At the 0.01% and 0.001% thresholds, most settings succeeded frequently, but the strictest threshold exposed larger differences. S7 achieved a 73.3% success rate within 0.001% of the global optimum and combined this accuracy with the smallest dispersion. S8 and S9 were also competitive; weaker settings produced accurate individual solutions but failed more often to complete the final refinement. Figure 6 shows the success rates across the three accuracy thresholds.

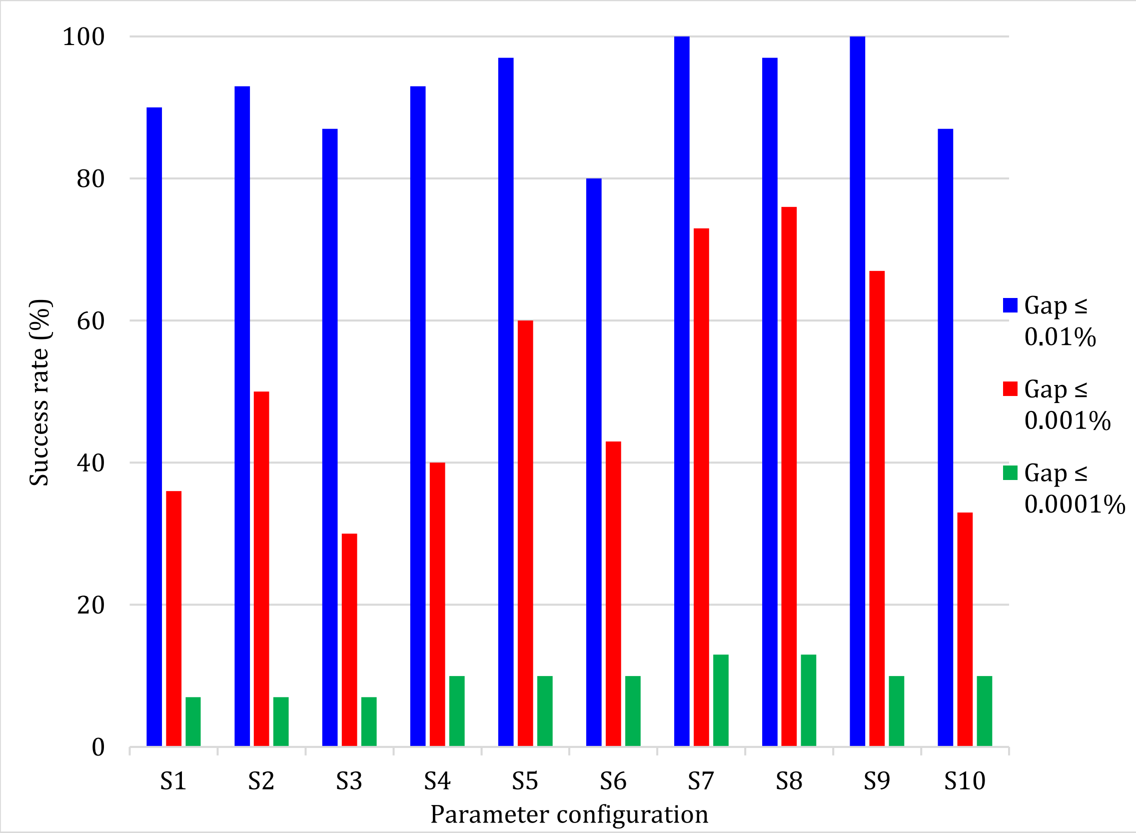

Figure 6. Success Rates under Increasingly Strict Accuracy Thresholds

The Friedman test rejected equal run-wise performance across the ten configurations (chi-square = 36.981818, df = 9, p = 2.65156 x 10^-5). Kendall's W indicated a non-negligible configuration effect. Holm-adjusted Wilcoxon comparisons showed that S7 significantly outperformed S1, S2, S3, S4, S6, and S10, while its differences from S5, S8, and S9 were not significant at 0.05. The correct conclusion is therefore that S7 is the best-ranked setting, while S5, S8, and S9 remain statistically competitive alternatives.

Runtime increased with the nominal population-iteration budget. S7 was not the fastest setting, but its higher cost was accompanied by lower dispersion and stronger success probability. S8 spent more iterations with a smaller population and achieved similar rank; S9 used more aggressive movement and obtained the closest individual solution. The computational trade-off therefore depends on whether the application prioritizes repeated-run reliability, the probability of an exceptionally accurate result, or reduced evaluation cost. Absolute runtime values should not be transferred to MATLAB because the reported measurements were produced by the compiled Python/Numba implementation.

4.4 Search-Stage Behaviour

Stage records explain why the parameter settings differed. Early iterations were dominated by D1 global exploration and D2 moderate exploration. As oxygen declined, B1 exploitation and D3 fine tuning became more frequent. Reset usage increased when oxygen was depleted, but accepted improvements temporarily raised oxygen and returned some divers to earlier movement modes. This feedback prevented the stage schedule from becoming a fixed monotonic sequence. Figure 7 shows the mean stage allocation for configuration S7.

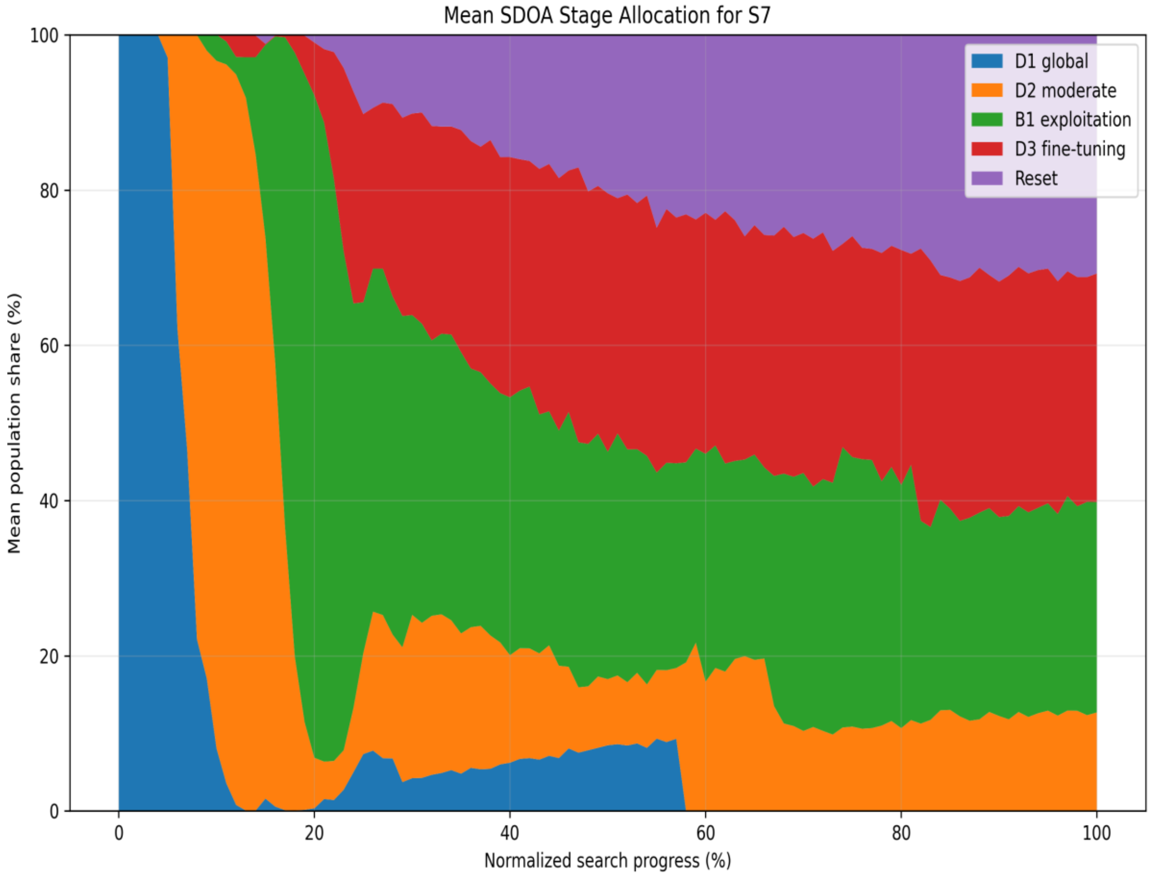

Figure 7. Mean Sdoa Stage Allocation for Configuration S7

S7 retained a substantial exploitation and fine-tuning share during the middle and late search while avoiding excessive reset activity. S9 reached the strongest single solution but its aggressive oxygen decay and movement parameters generated greater variation in stage usage. The results suggest that population size and iteration budget mainly determine how many opportunities the algorithm has to revisit the active constraint region, whereas oxygen decay determines how these opportunities are distributed among exploration, refinement, and reset.

4.5 Comparison with Established Search Behaviour

The observed convergence pattern is consistent with earlier mechanical-design applications of DE, PSO, TLBO, GWO, WOA, SCA, SSA, and HHO: large reductions occur early, while the final digits require sustained search near active constraints. SDOA differs in how this transition is scheduled. Rather than using one time-varying coefficient or a small set of deterministic phases, it assigns each diver independently according to oxygen. The population can therefore contain globally exploring, moderately exploring, exploiting, and fine-tuning candidates in the same iteration. Accepted improvements feed back into this allocation through oxygen recovery.

The best SDOA result does not improve the proven global minimum, nor should it. Its value is slightly higher because the final continuous coordinates differ in the last digits from the analytical solution. The relevant evidence is that the method repeatedly identifies the correct discrete thickness pair and approaches the active volume boundary with small error. The comparison with the known optimum therefore validates implementation accuracy, while the repeated-run statistics measure reliability. Claims of superiority based solely on values below 6059.714335 should be treated cautiously unless their formulations are demonstrably identical.

From an engineering perspective, the difference between the closest SDOA solution and the proven optimum is negligible relative to the scale of the cost model. From an algorithmic perspective, however, those final digits are informative because they distinguish configurations that repeatedly complete the last stage of refinement from those that only occasionally do so. S7 was selected because it combined small dispersion, strong rank, and high precision frequency, whereas S9 was retained as evidence that a more aggressive setting can produce the strongest individual run. This distinction between recommended configuration and closest isolated design is maintained throughout the interpretation.

4.6 Practical Interpretation and Limitations

The experiment supports three practical conclusions. First, problem-structured repair is essential for efficient mixed-discrete pressure-vessel search because it enforces the plate increment and reconstructs volume-feasible geometry before acceptance. Second, the attainable best value is relatively insensitive to parameter setting once the search budget is adequate; reliability is more sensitive. Third, stage-usage records are useful diagnostics because they reveal whether a weak configuration spends too much time in reset or too little time in late refinement.

The study also has limitations. It examines one four-variable benchmark, so the selected parameters should not be assumed optimal for other constrained problems. Repair uses analytical knowledge specific to the pressure vessel and therefore contributes to performance independently of SDOA. The tested settings do not constitute exhaustive parameter optimization. Runtime measurements are environment specific, and matched seed identifiers do not produce identical random-number streams when configurations consume different numbers of random values. These limitations do not invalidate the comparative result, but they restrict its scope to the stated implementation and protocol.

5. CONCLUSION

This paper presented a mixed-discrete pressure-vessel implementation of SDOA and evaluated ten parameter configurations through 300 independent runs. The algorithm combined oxygen-regulated stage selection with manufacturing-aware repair, feasibility-first ranking, global-best communication, and elitism. Every run ended with a feasible design. S7 achieved the best mean rank, the smallest standard deviation, a mean cost of 6059.766062, and a 73.3% success rate within 0.001% of the proven optimum. S9 produced the closest individual design at 6059.714482843, only 0.000147794 cost units above the global minimum.

The statistical analysis confirmed that parameter selection affects repeated-run performance. S7 significantly outperformed six configurations and remained statistically comparable with S5, S8, and S9. Stage records showed that stronger settings maintained a useful transition from exploration to exploitation and fine tuning without excessive reset. Future work should evaluate the same SDOA mechanisms on additional mixed-integer engineering problems, compare generic and problem-specific constraint handling under equal evaluation budgets, and study adaptive online control of oxygen thresholds and elite size. A further priority is direct MATLAB replication using reported evaluation counts, hardware specifications, and open run-level data so that algorithmic and implementation effects can be separated more precisely.

Acknowledgments

The authors acknowledge the academic and technical support provided by the Department of Information Technology, Technical College of Duhok, Duhok Polytechnic University, and the Department of Information Technology, Technical College of Informatics-Akre, Akre University for Applied Sciences.

Funding Information

This research received no external funding.

Author Contributions Statement

Name of Author	C	M	So	Va	Fo	I	R	D	O	E	Vi	Su	P	Fu
Saman M. Almufti	✓	✓	✓	✓		✓		✓	✓	✓	✓	✓	✓	✓
Amira Bibo Sallow		✓	✓	✓	✓	✓		✓	✓	✓		✓	✓	

C : **C**onceptualization
M : **M**ethodology
So : **So**ftware
Va : **Va**lidation
Fo : **Fo**rmal analysis
I : **I**nvestigation
R : **R**esources
D : **D**ata Curation
O : Writing - **O**riginal Draft
E : Writing - Review & **E**diting
Vi : **Vi**sualization
Su : **Su**pervision
P : **P**roject administration
Fu : **Fu**nding acquisition

Conflict of Interest Statement

The authors declare that they have no financial or non-financial conflicts of interest related to this work.

Informed Consent

Not applicable. The study used a mathematical engineering benchmark and did not involve human participants, identifiable personal information, or clinical data.

Ethical Approval

Not applicable. The research involved computational optimization only and did not include human participants, animals, biological materials, or interventions requiring institutional ethical approval.

Data Availability

The complete 300-run dataset, parameter configurations, descriptive statistics, convergence summaries, run-wise ranks, stage-usage records, constraint checks, and statistical-test outputs are available from the corresponding author on reasonable request. The numerical values required to interpret the principal findings are also reported in the article tables and figures.

REFERENCES

[1] B. K. Kannan and S. N. Kramer, 'An augmented Lagrange multiplier based method for mixed integer discrete continuous optimization and its applications to mechanical design', J. Mech. Des. N. Y., vol. 116, no. 2, pp. 405-411, June 1994. doi.org/10.1115/1.2919393

[2] X. S. Yang, C. Huyck, M. Karamanoglu, and N. Khan, 'True global optimality of the pressure vessel design problem: a benchmark for bio-inspired optimisation algorithms', Int. J. Bio-inspired Comput., vol. 5, no. 6, p. 329, 2013. doi.org/10.1504/IJBIC.2013.058910

[3] C. A. Coello Coello, 'Theoretical and numerical constraint-handling techniques used with evolutionary algorithms: a survey of the state of the art', Comput. Methods Appl. Mech. Eng., vol. 191, no. 11-12, pp. 1245-1287, Jan. 2002. doi.org/10.1016/S0045-7825(01)00323-1

[4] K. Deb, 'An efficient constraint handling method for genetic algorithms', Comput. Methods Appl. Mech. Eng., vol. 186, no. 2-4, pp. 311-338, June 2000. doi.org/10.1016/S0045-7825(99)00389-8

[5] E. Mezura-Montes and C. A. Coello Coello, 'Constraint-handling in nature-inspired numerical optimization: Past, present and future', Swarm Evol. Comput., vol. 1, no. 4, pp. 173-194, Dec. 2011. doi.org/10.1016/j.swevo.2011.10.001

[6] I. Boussaïd, J. Lepagnot, and P. Siarry, 'A survey on optimization metaheuristics', Inf. Sci. (Ny), vol. 237, pp. 82-117, July 2013. doi.org/10.1016/j.ins.2013.02.041

[7] D. H. Wolpert and W. G. Macready, 'No free lunch theorems for optimization', IEEE Trans. Evol. Comput, vol. 1, no. 1, pp. 67-82, Apr. 1997. doi.org/10.1109/4235.585893

[8] R. Storn and K. Price, J. Glob. Optim, vol. 11, no. 4, pp. 341-359, 1997. doi.org/10.1023/A:1008202821328

[9] J. Kennedy and R. Eberhart, 'Particle swarm optimization', in Proceedings of ICNN'95 - International Conference on Neural Networks, Perth, WA, Australia, 2002. doi.org/10.1109/ICNN.1995.488968

[10] R. V. Rao, V. J. Savsani, and D. P. Vakharia, 'Teaching-learning-based optimization: A novel method for constrained mechanical design optimization problems', Comput. Aided Des., vol. 43, no. 3, pp. 303-315, Mar. 2011. doi.org/10.1016/j.cad.2010.12.015

[11] S. Mirjalili, S. M. Mirjalili, and A. Lewis, 'Grey wolf optimizer', Adv. Eng. Softw., vol. 69, pp. 46-61, Mar. 2014. doi.org/10.1016/j.advengsoft.2013.12.007

[12] S. Mirjalili and A. Lewis, 'The whale optimization algorithm', Adv. Eng. Softw., vol. 95, pp. 51-67, May 2016. doi.org/10.1016/j.advengsoft.2016.01.008

[13] S. Mirjalili, 'SCA: A Sine Cosine Algorithm for solving optimization problems', Knowl. Based Syst., vol. 96, pp. 120-133, Mar. 2016. doi.org/10.1016/j.knosys.2015.12.022

[14] S. Mirjalili, A. H. Gandomi, S. Z. Mirjalili, S. Saremi, H. Faris, and S. M. Mirjalili, 'Salp Swarm Algorithm: A bio-inspired optimizer for engineering design problems', Adv. Eng. Softw., vol. 114, pp. 163-191, Dec. 2017. doi.org/10.1016/j.advengsoft.2017.07.002

[15] A. A. Heidari, S. Mirjalili, H. Faris, I. Aljarah, M. Mafarja, and H. Chen, 'Harris hawks optimization: Algorithm and applications', Future Gener. Comput. Syst., vol. 97, pp. 849-872, Aug. 2019. doi.org/10.1016/j.future.2019.02.028

[16] X.-S. Yang, 'Nature-inspired optimization algorithms: Challenges and open problems', J. Comput. Sci., vol. 46, no. 101104, p. 101104, Oct. 2020. doi.org/10.1016/j.jocs.2020.101104

[17] J. Derrac, S. García, D. Molina, and F. Herrera, 'A practical tutorial on the use of nonparametric statistical tests as a methodology for comparing evolutionary and swarm intelligence algorithms', Swarm Evol. Comput, vol. 1, no. 1, pp. 3-18, Mar. 2011. doi.org/10.1016/j.swevo.2011.02.002

[18] S. García, A. Fernández, J. Luengo, and F. Herrera, 'Advanced nonparametric tests for multiple comparisons in the design of experiments in computational intelligence and data mining: Experimental analysis of power', Inf. Sci. (Ny), vol. 180, no. 10, pp. 2044-2064, May 2010. doi.org/10.1016/j.ins.2009.12.010

[19] S. M. Almufti, A. Ahmad Shaban, Z. Arif Ali, R. Ismael Ali, J. A. Dela Fuente, and R. Rajab Asaad, 'Overview of metaheuristic algorithms', Polaris Global Journal of Scholarly Research and Trends, vol. 2, no. 2, pp. 10-32, Apr. 2023. doi.org/10.58429/pgjsrt.v2n2a144

[20] S. M. Almufti, R. Boya Marqas, and V. Ashqi Saeed, 'Taxonomy of bio-inspired optimization algorithms', J. Adv. Comput. Sci. Technol., vol. 8, no. 2, pp. 23-31, Aug. 2019. doi.org/10.14419/jacst.v8i2.29402

[21] S. M. Almufti, 'Vibrating Particles System Algorithm performance in solving Constrained Optimization Problem', Acad. J. Nawroz Univ., vol. 11, no. 3, pp. 231-242, Aug. 2022. doi.org/10.25007/ajnu.v11n3a1499

[22] S. M. Almufti, 'Historical survey on metaheuristics algorithms', Int. J. Sci. World, vol. 7, no. 1, p. 1, Nov. 2019. doi.org/10.14419/ijsw.v7i1.29497

[23] M. Friedman, 'The use of ranks to avoid the assumption of normality implicit in the analysis of variance', J. Am. Stat. Assoc., vol. 32, no. 200, pp. 675-701, Dec. 1937. doi.org/10.1080/01621459.1937.10503522

[24] F. Wilcoxon, 'Individual comparisons by ranking methods', Biom. Bull., vol. 1, no. 6, p. 80, Dec. 1945. doi.org/10.2307/3001968

[25] S. M. Almufti, 'Exploring the impact of Big Bang-Big Crunch algorithm parameters on Welded Beam Design problem resolution', Acad. J. Nawroz Univ., vol. 12, no. 4, pp. 1-16, Sept. 2023. doi.org/10.25007/ajnu.v12n4a1903

How to Cite Saman M. Almufti, Amira Bibo Sallow. (2026). Constrained Pressure-Vessel Design Optimization Using the Scuba Diver Optimization Algorithm: Parameter Sensitivity and Statistical Validation. International Journal of Information Technology and Computer Engineering, 6(1), 78-93. https://doi.org/10.55529/ijitc.61.78.93

BIOGRAPHIES OF AUTHORS

	**Saman M. Almufti**, is a researcher at both the Technical College of Informatics–Akre, Akre University for Applied Sciences, Duhok, Iraq, and the Department of Computer Science, College of Science, Knowledge University, Erbil. His research focuses on metaheuristic algorithms, swarm intelligence, and applied machine learning. He has published numerous papers in high-impact international journals. Email: Saman.Almofty@gmail.com
	**Amira Bibo Sallow**, is a lecturer at the Department of Information Technology, Technical College of Duhok, Duhok Polytechnic University, Iraq. Her research interests include information systems, data science, and software engineering. She is actively involved in academic and applied research projects related to digital transformation and computational intelligence. Email: amira.bibo@dpu.edu.krd