Chapter 5

Development and Comparative Assessment of a Hybrid Optimization Algorithm

5.1 Introduction

This chapter presents the design and development of a metaheuristic algorithm namely GWOSCAPSO hybridizing three popular optimization approaches namely the Grey Wolf Optimizer (GWO), Sine Cosine Algorithm (SCA) and Particle Swarm Optimization (PSO). The exploration and exploitation properties of GWO is enhanced with the application of the SCA and PSO algorithms. The performance of the hybrid algorithm is analyzed through the application to solve 23 benchmark functions. Furthermore, the performance of the hybrid GWOSCAPSO algorithm is compared with the results of widely used optimization algorithms already obtained in chapters 3 and chapter 4.

5.2 Design and development of a Hybrid Metaheuristic Algorithm (GWOSCAPSO)

The objective to locate an optimal solution within a complicated space remains a prevalent concern in several engineering optimization problems. Metaheuristic algorithms are a class of numerical methods that draw inspiration from natural phenomena [154] in the formulation of the optimization algorithm. This prevalent strategy of the hybridization of distinct algorithms integrates their respective strengths, to provide an improved algorithm [221]. Additionally, to achieve equilibrium between exploitation and exploration, it is essential to employ an effective optimization algorithm [138]. Exploitation refers to the algorithm's high efficacy in conducting localized search operations. The inclusion of exploration ability enhances the efficacy of an algorithm in identifying optimal initial placements, potentially in close proximity to the global minimum.

The GWO algorithm, which is based on hunting behavior and hierarchical structure of grey wolves in the natural world (Chapter 2, subsection 2.2.1.9) has shown a strong initial exploration capability [60], is selected as the base algorithm for hybridization. It is observed that the exploitation capability of PSO is very strong with inadequate exploration capability[88, 100, 229, 234]. The SCA algorithm [153] enables the exploitation and exploration phases of global optimization functions by using the Sine and Cosine functions.

To improve the performance of GWO, we use the SCA to update the position of the alpha wolves of the GWO algorithm while PSO is implemented to update the beta and delta wolves in order to improve the overall performance of the algorithms. The algorithm updates the equations for the alpha wolf taking into consideration the equations 2.23 and 2.24 using the SCA algorithm (from chapter 2, subsection 2.2.4). Additionally, the GWO algorithm performance is enhanced by implementing the exploitation ability of PSO by improving the update of beta and delta wolves of the GWO algorithm given by:

$$\begin{aligned} \overrightarrow{D_\beta} &= \overrightarrow{C_2}.\overrightarrow{X_\beta} - w \times \overrightarrow{X} \\ \overrightarrow{D_\delta} &= \overrightarrow{C_3}.\overrightarrow{X_\delta} - w \times \overrightarrow{X} \end{aligned} \tag{5.1}$$

$$\begin{aligned} \overrightarrow{X_2} &= \overrightarrow{X_\beta} - \overrightarrow{A_2}.(\overrightarrow{D_\beta}) \\ \overrightarrow{X_3} &= \overrightarrow{X_\delta} - \overrightarrow{A_3}.(\overrightarrow{D_\delta}) \end{aligned} \tag{5.2}$$

Based on the preceding equations, the integration of GWO and PSO variations is executed by modifying the equations for velocity and position in the following manner:

$$v_i^{k+1} = w \times (v_i^k) + c_2 \times r_2 \times (X_2 - x_i^k) + c_3 \times r_3 \times (X_3 - x_i^k) \tag{5.3}$$

The particles position is then updated using equation 2.11 for the next iteration cycle. This forms the basis of the update of the positions of the alpha, beta and delta wolves in the modified hybrid GWOSCAPSO algorithm given by:

$$\overrightarrow{X}(i+1) = \overrightarrow{X_1} + v_i^{k+1} \tag{5.4}$$

The pseudo-code for the GWOSCAPSO algorithm is highlighted in **Algorithm 1**.

Algorithm 1: Pseudo-code for the Hybrid GWOSCAPSO algorithm

Optimize Fitness function subject to $lb, ub, dim, fobj$;
Where lb - lower bound, ub - upper bound, dim - problem dimension, fobj - objective function;
Initialize the search agents or population $X_i(i = 1, 2, ...n)$;
Calculate the fitness of each search agent;
X_α = best search agent;
X_β = second best search agent;
X_δ = third best search agent ;
while $t<$ *MaxIteration* **do**
    Update position of best search agent and fitness function for each search agent;
    Update the values of a, A, and C using equations 2.3 and 2.4;
    Calculate the fitness of all search agents ;
    if $rand() < 0.5$ **then**
        Update the values of X_α using first part of equation 2.23 and $\overrightarrow{X}_1$ using equation 2.24 ;
    else
        $rand() \geq 0.5$ Update the values of X_α using second part of equation 2.23 $\overrightarrow{X}_1$ using equation 2.24;
    end
    Update the values of X_β and X_δ using equations 5.1 and 5.2;
    Update the value of $\overrightarrow{X}(i+1)$ using equation 5.4;
end
Check for out of bound of search agents and rectify;
Calculate the fitness of each search agent;
Update X_α if current result better than initial value;
Result: return X_α

5.3 Performance evaluation of the developed GWOSCAPSO optimization algorithm

This section presents the metrics which aid the evaluation of hybrid GWOSCAPSO algorithm with respect to the computation complexity in time and the significant statistically tests which help in the comparative assessment of the algorithm.

5.3.1 Time Complexity

The time complexity refers to the run time required by an algorithm when it is used to solve a problem by the corresponding executable program. It is an important metric when evaluating an algorithm's performance and computational efficiency. The worst time complexity of the hybrid GWOSCAPSO algorithm is calculated and compared with that of the original GWO in terms of the big-O notation by analyzing its pseudo-codes.

The computational complexity involved in both the GWOSCAPSO and GWO is determined in the five subsequent stages namely, population initialization, parameters updating, wolf position updating, evaluation of objective function and wolf leaders updating. The time complexity of an algorithm relays on the population size N and the maximum number of iterations T. The time complexity of the GWOSCAPSO algorithm is given by:

1. For population initialization, the time complexity is indicated as $O(N)$ for both the GWOSCAPSO as well as GWO.
2. The time complexity for parameter updating for both the algorithms is also indicated as to be $O(N)$.
3. For the position updating stage of the algorithm, for both GWOSCAPSO and GWO the time complexity is given by $O(TN)$.
4. For the calculation of the objective functions, the time complexity for GWOSCAPSO is given as $O(TN)$, which is the same as that of GWO.
5. The time complexity related to the updating of the wolf leaders position is also given as $O(TN)$ for both the GWOSCAPSO and GWO algorithms.

From the above analysis, it is evident that for the GWOSCAPSO algorithm the time complexity is given as $O(TN)$, which is the same as that for the GWO. This means that the computational complexity involved in the developed algorithm does not exceed the complexity involved in the base algorithm, i.e. the GWO algorithm.

5.3.2 Statistical Performance Analysis

For reliable and quality performance comparison of the developed GWOSCAPSO algorithm, it is to be tested against the performance of other algorithms to be selected based on established metaheuristic algorithms. This is accomplished by evaluating the efficiency of the algorithms by comparison with well-established benchmark functions. Each problem from this category of benchmark functions is run for a minimum of 30 times and the result of these runs are stored. Based on the results, the best value (min), mean or average value, worst (max) and standard deviation of each cost function is reported. In order to further establish statistical significance, we perform the Friedman's test.

5.3.2.1 Friedman's Test

The non-parametric Friedman Test will be performed to rank optimization techniques and calculate the p value, which is a statistical metric employed to assess the importance of outcomes in hypothesis testing. A p value greater than 0.05 indicates rejection of the null hypothesis. The test is a multiple comparison test utilized to ascertain the significant disparity among the algorithms, and is comparable to the repetitive tests and measurements employed in ANOVA. The null hypothesis of the Friedman test states that there is no difference in the quality of medians among the data sets. The rejection of the null hypothesis confirms the establishment of the alternative hypothesis, demonstrating that it is non-directional.

The Friedman test involves transforming the test statistics into rankings. The steps involved in the process are :

1. To find the result of the 100 runs of all the algorithms under comparison.
2. For each of the benchmark problem, rank values to all of the algorithms from 1 (indicating best algorithm) to k (worst algorithm).
3. The final rank of the algorithm is determined form the rank of average ranks in the 100 runs.

As a result the test ranks each benchmark problem's algorithms individually based on which run produced the best cost function values, and then calculates the average rank (for 100 such runs) against one benchmark function.

5.3.3 Benchmark Functions

The benchmark functions, including their mathematical models and ranges are implemented in MATLAB [91]. On 23 traditional benchmark problems, the novel

Table 5.1: Results of simulation experiments conducted with Unimodal Benchmark Functions F1-F7

Function Name	Metrics	GWO	PSO	SCA	ALO	MFO	WOA	GWOSCAPSO
F1	Best	3.76E-29	0.90	0.008	2.25E-4	0.41	1.36E-84	7.79E-146
	Worst	6.71E-27	40000	77.54	0.005	10002	1.47E-71	1.82E-142
	Mean	1.21E-27	10341	5.31	0.002	1345.81	5.08E-73	3.22E-145
	Std Dev	1.65E-27	9992	14.14	0.001	3453.68	2.69E-72	0
F2	Best	1.46E-17	30.16	3.93E-5	5.91	0.17	1.46E-17	1.97E-59
	Worst	4.01E-16	120	0.083	237.44	90.0	4.01E-16	4.35E-49
	Mean	1.14E-16	76.40	0.013	43.07	29.46	1.14E-16	2.91E-50
	Std Dev	1.04E-16	23.22	0.016	54.22	18.67	1.04E-16	9.59E-50
F3	Best	5.50E-8	19147	628.5	754.62	4667	19822	2.28E-128
	Worst	1.93E-4	115014	28957	8526	31908	76318	1.39E-124
	Mean	7.64E-7	39.30	34.45	16.31	66.67	44.24	2.95E-68
	Std Dev	3.63E-5	21152	5582	1870	7256	13216	0
F4	Best	5.96E-8	23.68	13.52	5.72	51.92	1.96	1.70E-69
	Worst	3.14E-6	52.40	47.11	25.29	82.31	89.25	1.21E-67
	Mean	7.64E-7	39.30	34.45	16.31	66.67	44.24	2.95E-68
	Std Dev	7.61E-7	7.39	8.28	4.63	7.64	27.53	3.049E-68
F5	Best	26.13	74.03	30.23	19.77	194.22	27.30	27.27
	Worst	28.74	1.55	727138	1374	8.01E7	28.77	28.88
	Mean	27.25	1.32	74068	266.29	2.68E6	27.98	28.19
	Std Dev	0.85	3.62	144847	345.64	1.46E7	0.39	0.35
F6	Best	0.24	1.18	4.32	9.09E-5	0.66	0.12	3.11
	Worst	1.48	30101	100.14	0.017	10120	1.13	4.20
	Mean	0.85	9680	15.63	0.002	2014.31	0.47	3.64
	Std Dev	0.31	8113	23.02	0.003	4081.11	0.25	0.25
F7	Best	4.04E-4	0.02	0.013	0.12	0.056	4.37E-5	1.99E-6
	Worst	0.005	61.96	0.62	0.40	40.67	0.01	2.93E-4
	Mean	0.002	23.71	0.10	0.23	3.02	0.002	9.46E-5
	Std Dev	0.001	20.77	0.11	0.07	7.92	0.003	8.38E-5

hybrid approach's statistical performance with respect to the mean value, numerical values that represent the best and worst outcomes, as well as standard deviation, which are of interest in this context were assessed. The three sections of these classical benchmark test functions—Unimodal, Multimodal, and Fixed Dimension Multimodal—as listed in literature [231, 265] have been considered in the analysis in the current work. Due to certain advantages (also popular) in solving engineering optimization problems the GWO [60, 83] , SCA [65], PSO [96], ALO[17], MFO[89], and WOA [194] are chosen as reference for testing the performance of GWOSCAPSO. The experiments are conducted on a computer system with an Intel Core i7 CPU, 16 GB RAM, and running Windows 10.

Table 3.1 (Chapter 3, section 3.4) presents the optimization algorithms' control settings. For each method, we used the optimal parameter combinations as recommended in the primary paper of those algorithms to evaluate peak performance. The algorithms' population size is set at 30, while the maximum iterations is set at 100. The relevant parameters of the GWOSCAPSO are set as per their

Table 5.2: Results of simulation experiments conducted with Multimodal Benchmark Functions F8-F13

Function Name	Metrics	GWO	PSO	SCA	ALO	MFO	WOA	GWOSCAPSO
F8	Best	-7158	-8629	-4305	-8278	-10706	-12569	-9056
	Worst	-5028	-4316	-3337	-5417	-6759	-6752	-5001
	Mean	-5952	-6451	-3721	-5653	-8640	-10488	-7097
	Std Dev	572.30	1009	224.84	628.07	993.12	1843	1003
F9	Best	0	114.94	0.37	43.77	52.96	0	0
	Worst	25.71	261.38	160.86	155.21	257.91	0	0
	Mean	2.92	190.26	45.63	82.74	153.46	0	0
	Std Dev	6.02	39.26	37.66	26.13	45.48	0	0
F10	Best	5.77E-14	14.50	0.16	1.50	0.20	8.88E-16	4.44E-15
	Worst	1.50E-13	20.38	20.36	12.37	19.96	7.99E-15	4.44E-15
	Mean	1.05E-13	18.98	13.16	5.57	12.79	4.20E-15	4.44E-15
	Std Dev	2.06E-14	1.808	9.23	3.08	8.03	1.85E-15	0
F11	Best	0	0.86	0.18	0.01	0.45	0	0
	Worst	0.02	270.86	1.66	0.13	90.47	0.21	0
	Mean	0.001	108.91	1.01	0.06	3.917	0.007	0
	Std Dev	0.005	76.12	0.34	0.03	16.34	0.04	0
F12	Best	0.23	0.31	0.74	6.36	1.82	0.004	0.012
	Worst	0.44	2.56E8	1.17E6	24.85	2.56E8	0.13	0.082
	Mean	0.33	3.414E7	95273	12.69	8.53E6	0.02	0.038
	Std Dev	0.04	8.85E7	265709	4.57	4.67E7	0.025	0.018
F13	Best	1.94	0.76	3.75	0.12	4.79	0.06	0.20
	Worst	2.48	4.10E8	2.16E7	58.18	27516.57	1.28	0.92
	Mean	2.32	4.10E7	890072	25.93	1036.46	0.49	0.59
	Std Dev	0.13	1.25E8	3.97E6	17.73	5023.10	0.29	0.21

base algorithm. To acquire statistical findings, each algorithm in the current study is run for 100 times, and for each technique, the optimum fitness function value is recorded in each run. The findings are tabulated in table 5.1 for the Unimodal benchmark functions, table 5.2 for the Multimodal benchmark functions, while the results of the comparison for the fixed dimension Multimodal benchmark functions are presented in table 5.3. In the current study GWOSCAPSO algorithm uses the parameters of the primary paper of the algorithms involved for ease of implementation.

From table 5.1 it is evident that the GWOSCAPSO algorithm performs better than the rest of the algorithms compared to all the algorithms in the study except function F6. In the case of the multimodal functions as seen in table 5.2, we observe better performances with the GWOSCAPSO algorithm in term of the statistical parameters considered. For function F8, where the performance of GWO is slightly better when we consider the metric for comparison to be standard deviation. The MFO and WOA both give better results for function F8 with respect to mean values as well as best and worst values for the function.

For the set of functions F14-F23, (from Table 5.3) which represent the

Table 5.3: Results of simulation experiments conducted with Fixed Dimension Multimodal Benchmark Functions F14-F23

Function Name	Metrics	GWO	PSO	SCA	ALO	MFO	WOA	GWOSCAPSO
F14	Best	0.99	0.99	0.998	0.99	0.99	0.99	0.99
	Worst	12.67	3.96	2.98	6.90	10.76	10.76	10.76
	Mean	4.65	1.22	1.92	2.05	3.20	3.48	2.82
	Std Dev	4.01	0.67	1.00	1.39	2.27	3.57	2.81
F15	Best	3.07E-4	0.002	3.41E-4	5.65E-4	7.44E-4	3.08E-4	3.14E-4
	Worst	0.02	0.022	0.002	0.020	0.008	0.002	0.0018
	Mean	0.003	0.014	9.87E-4	0.002	0.002	7.96E-4	4.58E-4
	Std Dev	0.008	0.009	3.79E-4	0.005	0.002	5.39E-4	3.25E-4
F16	Best	-1.03	-1.03	-1.031	-1.031	-1.031	-1.031	-1.031
	Worst	-1.03	-1.03	-1.031	-1.031	-1.031	-1.031	-1.031
	Mean	-1.031	-1.031	-1.031	-1.031	-1.031	-1.031	-1.031
	Std Dev	2.46E-8	1.62E-5	4.30E-5	6.61-14	0	4.24E-10	0
F17	Best	0.39	0.39	0.39	0.39	0.39	0.39	0.39
	Worst	0.39	1.94	0.41	5.04	0.39	0.39	0.39
	Mean	0.39	0.55	0.39	0.55	0.39	0.39	0.39
	Std Dev	2.49E-4	0.47	0.003	0.84	9.42E-14	2.46E-5	0
F18	Best	3.00	3	3	3	3	3	3
	Worst	3.00	91.81	3.00	3	3	3.02	3
	Mean	3.00	6.86	3.00	3.00	3	3.00	3
	Std Dev	5.27E-5	16.78	5.25E-5	5.99E-6	8.02E-13	0.005	4.4E-15
F19	Best	-3.86	-3.86	-3.86	-3.86	-3.86	-3.86	-3.86
	Worst	-3.85	-3.51	-3.85	-3.86	-3.86	-3.84	-3.85
	Mean	-3.86	-3.81	-3.85	-3.85	-3.86	-3.85	-3.86
	Std Dev	0.002	0.12	0.003	0.002	0	0.004	3.64E-13
F20	Best	-3.32	-3.32	-3.26	-3.26	-3.32	-3.32	-3.32
	Worst	-2.84	-2.43	-1.45	-1.45	-2.99	-3.13	-3.20
	Mean	-3.24	-2.95	-2.79	-2.75	-3.23	-3.23	-3.28
	Std Dev	0.12	0.25	0.50	0.40	0.093	0.058	0.057
F21	Best	-10.15	-10.15	-7.13	-10.15	-10.15	-10.15	-8.00
	Worst	-3.37	-0.49	-0.49	-2.63	-2.63	-2.62	-0.88
	Mean	-8.74	-6.66	-2.04	-6.55	-6.81	-7.76	-4.75
	Std Dev	2.39	3.66	1.89	3.32	3.49	2.80	1.14
F22	Best	-10.40	-10.40	-6.17	-7.19	-10.40	-10.40	-10.40
	Worst	-2.76	-0.75	-0.90	-2.97	-1.83	-1.84	-5.12
	Mean	-7.45	-7.03	-3.34	-4.8	-6.17	-7.07	-10.22
	Std Dev	3.05	3.54	1.48	0.69	3.16	3.43964	0.96
F23	Best	-6.02	-10.53	-9.04	-10.53	-10.53	-10.53	-10.53
	Worst	-3.50	-1.67	-0.94	-1.67	-2.42	-1.67	-10.53
	Mean	-4.84	-6.72	-4.16	-6.97	-8.33	-7.49	-10.53
	Std Dev	0.51	3.91	1.55	3.72	3.43	3.62	8.14E-4

class of fixed dimension multimodal benchmark functions, we can clearly observe that the proposed GWOSCAPSO algorithm is able to outperform the other algorithms. The performance of PSO and SCA is better for function F14, however for all the other functions, the proposed algorithm performs better. It is seen that the developed algorithm outperforms in 20 out of 23 test functions across different types of benchmarks and as such, it can be concluded that the proposed GWOSCAPSO algorithm can be implemented for solving global engineering optimization problems.

5.4 Application of hybrid GWOSCAPSO for solving Renewable Energy Optimization Problems

In this section the optimal design of a DC-DC boost is carried out using the developed GWOSCAPSO optimization algorithm. The algorithm is also utilized for the design optimization of a dc-dc boost converter for MPPT application in PV systems and for design optimization of the hybrid PV Hydro system. To study the efficacy of the algorithm in solving the design optimization problems, it is compared against six popular and established algorithms and the results are tabulated and discussed. The results obtained indicate that the GWOSCAPSO algorithm gives better performance and can serve as a suitable algorithm for the said design optimization problems.

5.4.1 Design Optimization of DC-DC Boost Converter using GWOSCAPSO

The problem's constraints and the range of design parameters were taken from table 3.2 which were also selected in chapter 3, section 3.4.

With the designed GWOSCAPSO optimization technique the optimal combination of design parameters is determined for the DC-DC boost converter in a manner that total operational losses are at a minimum. The parameters for algorithms are considered to be same as in the preceding section 5.3.3 for uniformity in the analysis. Each of the algorithms is run for a minimum of 100 iterations. The comparison made is based on the statistical performance metrics that include the best value, mean or average value, worst value standard deviation (SD) and computational time.

From table 5.4, the best value and optimized solution to the DC-DC converter is 1.75388 W (P_{BOOST}). And therefore, the overall efficiency of the converter designed is 91.94 %. It is also observed that all the optimization algorithms reach this result which indicates that the global optimum has been attained. The results obtained are best for the GWOSCAPSO considering the standard deviation as well as requiring the minimum computational time for 100 iterations for each run of the algorithms. While the best values are found to be equal for the GWO, MFO and PSO, however in terms of SD within the 100 runs considered, GWOSCAPSO

Table 5.4: Optimized P_{BOOST} values of the DC-DC boost converter by ALO, MFO, PSO, WOA, GWO, SCA and GWOSCAPSO

Algorithm	Rank	Best Value	Mean Value	Worst Value	Standard Deviation	Computational Time (sec)
ALO	7	1.76158	1.76428	1.78782	0.0064	14.6096
MFO	2	1.75388	1.75565	1.77729	0.00567	0.76824
WOA	6	1.75403	1.75519	1.83733	0.00856	0.373068
PSO	3	1.75388	1.75548	1.76988	0.00408	0.612471
SCA	5	1.75389	1.75443	1.76507	0.00184	0.620821
GWO	4	1.75389	1.75433	1.7746	0.00233	0.685787
GWOSCAPSO	1	1.75388	1.75388	1.75388	0	0.509022

clearly outperforms MFO and PSO.

Figure 5.1 demonstrates the convergence traits for each algorithm's best runs. It is evident that the GWOSCAPSO takes the lowest number of iterations to reach the minima, thus establishing its superiority with other algorithms in terms of convergence characteristics. From the results of the Friedman's test, we see that with regards to the ranks, GWOSCAPSO outperforms all the other algorithms (Table 5.4). If the performance metric considered for analysis is the computational time, then only WOA outperfroms the developed GWOSCAPSO algorithm. Thus, we observe that collectively considering the performance parameters like minimum standard deviation, best value of objective function, lower iteration time and fast convergence characteristics, the developed GWOSCAPSO gives superior performance when compared against established benchmark optimization algorithms like GWO,SCA, PSO, ALO, MFO and WHO.

Table 5.5 compares the optimal design parameters achieved by various optimization methods. The best optimized result is attained with the GWOSCAPSO with lowest power loss and highest efficiency in the converter design. The more significant finding is that the optimized design uses the lowest value of filter inductor, among the algorithms which is often the limiting criterion when considering the converter economics [215]. The obtained results also indicate an improvement in the design when compared with the work of [202] from three aspects namely, the optimized power loss in the converter, the efficiency and filter inductor. This indicates the efficacy of GWOSCAPSO for finding an improved optimized result for minimizing the power loss by 11.11 %, which is a significant improvement.

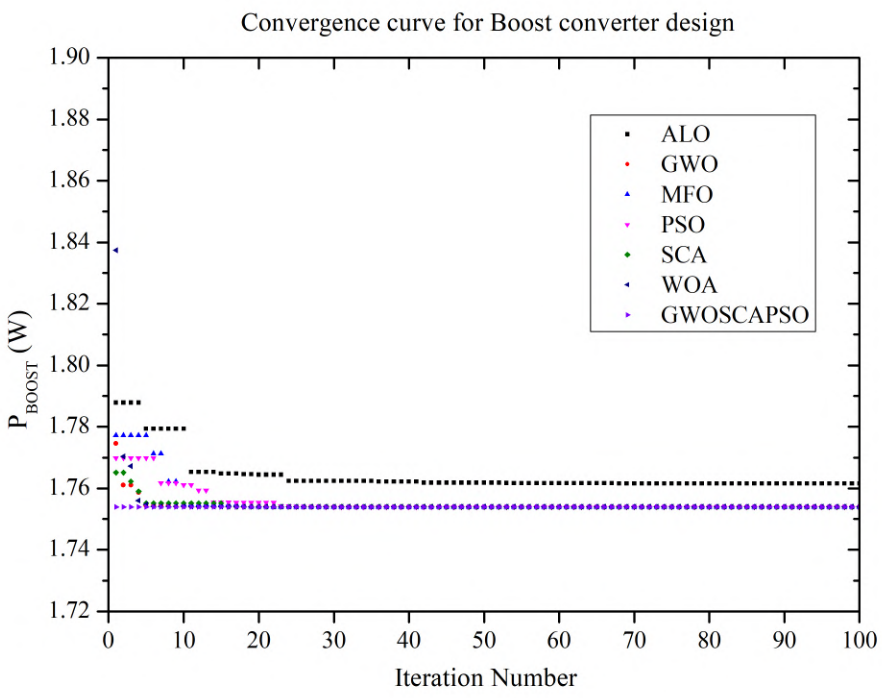

Figure. 5.1: Convergence Curve of the ALO, MFO, WOA, PSO, SCA, GWO and GWOSCAPSO for the boost converter

Table 5.5: Optimized design parameter values for the boost converter obtained by ALO, MFO, WOA, PSO, SCA, GWO and GWOSCAPSO

Algorithm	L (mH)	C (μF)	Fs (kHz)	P_{BOOST}(W)	η(%)
ALO	5.73	81.94	127.95	1.764	91.89
MFO	9.98	99.97	100.04	1.755	91.93
WOA	8.29	97.70	109.65	1.755	91.93
PSO	10	100	100.	1.755	91.93
SCA	7.3	100	100.28	1.754	91.93
GWO	6.16	99.99	100.02	1.754	91.93
GWOSCAPSO	10	89.88	100	1.753	91.94

5.4.2 Optimized design of the Boost Converter for MPPT applications for standalone PV systems using GWOSCAPSO

In this current section, we discuss the application of the GWOSCAPSO optimization technique to find the optimal design parameters for the DC-DC boost converter with an aim to be used for MPPT applications in standalone systems. The algorithms are run in a similar manner as discussed in section 5.4.1. Table 5.6 gives the statistical performance of the algorithms which includes the best value, mean or average value, worse value, standard deviation and iteration time.

Table 5.6: Statistical parameters for the Optimized P_{BOOST} values of the DC-DC boost converter for MPPT application in standalone PV systems by ALO, MFO, PSO, WOA, GWO, SCA and GWOSCAPSO

Algorithm	Rank	Best Value (W)	Mean Value (W)	Worst Value (W)	Standard Deviation	Iteration Time (sec)
ALO	7	7.04505	7.05488	7.11271	0.01852	8.623
MFO	2	7.04505	7.04505	7.04505	0	0.465
WOA	6	7.04505	7.09523	7.44597	0	0.383
PSO	3	7.04505	7.04505	7.04505	0	0.48
SCA	5	7.04505	7.04505	7.04505	0	0.4583
GWO	4	7.04505	7.04505	7.04505	0	0.4177
GWOSCAPSO	1	7.04505	7.0457	7.07431	0.00395	0.550

From table 5.6, its is seen that the best optimized design of the boost converter for the objective function, i.e. P_{BOOST} is equal to 7.045 W which gives the converter efficiency of 94.37 (%). It is also observed that all the compared algorithms namely, ALO, MFO, WOA, PSO, SCA, GWO and GWOSCAPSO reach this result indicating that the global optimum has been reached. Although the GWOSCAPSO algorithm gives comparable performance, in terms of computation time, it performs only better than the ALO. Considering standard deviation as well, the GWOSCAPSO algorithm gives comparable performance with respect to all the algorithms. If we compare the best values, we see that all the values of the algorithms are the same, but for the mean values we see GWOSCAPSO outperforms ALO and WOA, and gives similar results with the rest of the algorithms. From the results of the Friedman's test, we see that in order of the ranks, GWOSCAPSO outperforms all the algorithms, which points towards its superior performance even though computationally its performance is comparable with the rest of the algorithms.

Figure 5.2 for the convergence curve indicates that the algorithm is able to reach the optimum result in the least number of iterations among all the compared algorithms, proving its superiority in reaching the optima.

From table 5.7 it is seen that the converter design parameters when considering both the filter inductor and capacitor are smaller compared to the rest of the algorithms, except ALO. This indicates that the GWOSCAPSO algorithm is indeed capable of performing quite well for the optimization problem at hand.

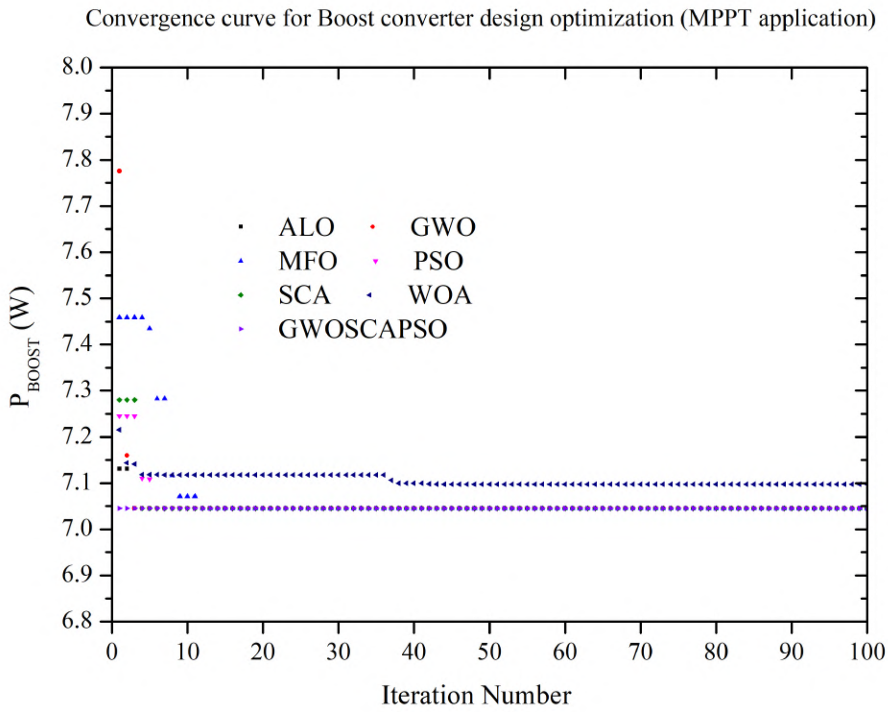

Figure. 5.2: Convergence Curve of the ALO, MFO, WOA, PSO, SCA, GWO and GWOSCAPSO for the Boost converter for MPPT Applications with PV systems

Table 5.7: Optimized design parameters for the boost converter for MPPT application in standalone PV systems obtained by ALO, MFO, WOA, PSO, SCA, GWO and GWOSCAPSO

Algorithm	L (mH)	C (μF)	Fs (kHz)	P_{BOOST}(W)	η(%)
ALO	10	37.34	100	7.04505	94.37
MFO	10	92.28	100	7.04505	94.37
WOA	10	95.07	100	7.04505	94.37
PSO	10	92.26	100	7.04505	94.37
SCA	10	94.49	100	7.04505	94.37
GWO	10	92.58	100	7.04505	94.37
GWOSCAPSO	10	65.25	100	7.04505	94.37

5.4.3 Optimal Sizing of Hybrid PV Hydro Renewable Energy System(HRES) using GWOSCAPSO

The results and convergence plot of the evaluated algorithm for the minimized LCOE for optimal sizing of the hybrid PV Hydro system is presented in figure 5.3 and table 5.8. In the figure 5.3, the convergence characteristics of all the algorithms do not reach the best optimal results.

Due to the random nature of the algorithms, the recorded best values over

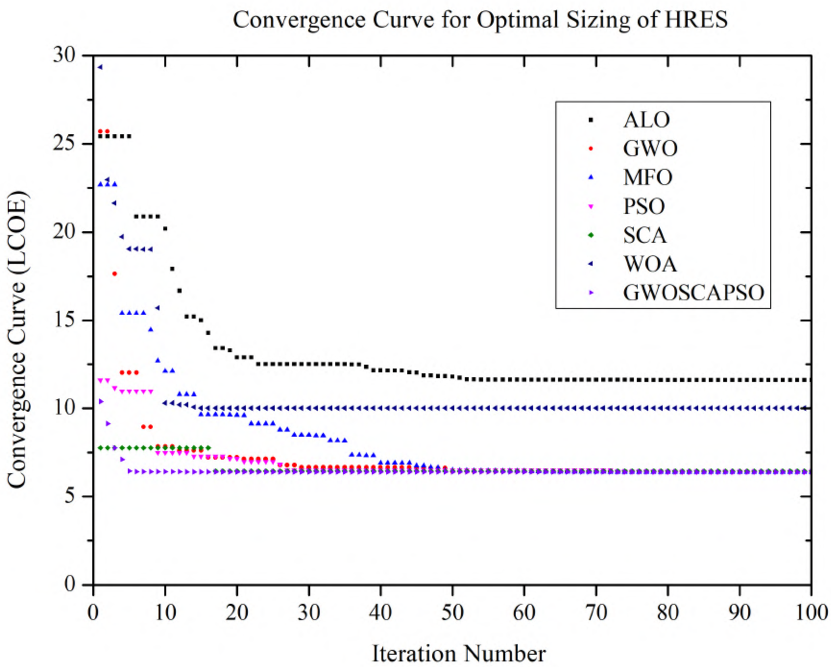

Figure. 5.3: Convergence characteristics of the ALO, MFO, WOA, PSO, SCA, GWO and GWOSCAPSO for the optimal sizing of the HRES

Table 5.8: Statistical parameters of the algorithms for minimized LCOE (INR/kWh) obtained by ALO, MFO, WOA, PSO, SCA, GWO and GWOSCAPSO

Algorithm	Rank	Best Value	Mean Value	Worst Value	Standard Deviation	Computational Time (sec)
ALO	6	6.408	7.67	14.97	1.636	3.71
MFO	3	6.408	6.409	6.441	0.005	4.51
WOA	7	6.506	10.371	18.636	3.360	2.81
PSO	4	6.412	6.488	7.570	0.124	4.70
SCA	5	6.435	6.746	7.288	0.21	2.84
GWO	2	6.408	6.409	6.437	0.006	4.80
GWOSCAPSO	1	6.408	6.409	6.435	0.006	2.87

100 iterations have to be considered to ascertain their performance over a number of trials. We observe that for the majority of the algorithms the fitness value of function, i.e. LCOE in our case starts far away from the final value ranging from 7.5 to 28 (INR/kW), which demonstrates the randomness of algorithms and best optimal values are reached around the 5-6th iteration for the GWOSCAPSO curve compared to the next best which is seen to take around the 20th iteration for the following fastest converging algorithms. The convergence characteristics (figure 5.3) show similar trends in most of the algorithms except the ALO and WOA algorithms, indicating that they are not suitable for application to the

optimization of the HRES problem. It is clearly evident that SCA and PSO starts very close to the final value as compared to the rest of the algorithms and also approaches its final value in a smaller number of iterations compared to most of the other algorithm. While the convergence curves are a good indicator of the overall performance to attain the best solution of the fitness value, however they are not the absolute indicators for the same. Due to inherent randomness in the metaheuristic algorithms, it is essential to carry out a statistical assessment to ascertain the overall performance of algorithms to solve an optimization problem.

From table 5.8 it is evident that the iteration time of the best algorithms are WOA, SCA and GWOSCAPSO respectively. However, upon closer reflection on the statistical parameters and figure 5.3 it is seen that the GWO, PSO and SCA produce similar results in terms convergence characteristics. However, the proposed GWOSCAPSO algorithm performs better than any other algorithms when measured against the convergence curve. Thus with respect to the best cost function values (the smallest LCOE value), mean values, and standard deviation, we see that the proposed GWOSCAPSO algorithm is demonstrating its superiority when compared with the rest of the algorithms. The statistical assessment of the algorithms is carried out, i.e. GWOSCAPSO is compared with ALO, MFO, WOA, PSO, SCA and GWO using the Friedman's Test. From table 5.8 the rank order of the GWOSCAPSO algorithm is one, establishing the algorithms superior performance when compared to all the algorithms considered in this study.

From the table 5.9 it is seen that best optimal size of HRES comes upto 31.86 kW in which hydro contributes 15.74 kW and PV contributes 16.12 kW respectively and optimal head is found to be 9 m and flow rate is 0.21 m^3/sec and optimized area has been found 95.58 m^2. This result is obtained in the case of all the algorithms compared here except the WOA algorithms.

Table 5.10 and 5.11 present the GWOSCAPSO optimized design parameters of the Hydro and PV system respectively.

5.5 Discussion

This work presented in this chapter introduces a novel hybrid method that combines the Particle Swarm Optimization (PSO) and Sine Cosine Algorithm (SCA) with the Grey Wolf Optimizer (GWO). The proposed technique involves updat-

Table 5.9: Optimized Sizing of HRES system obtained by ALO, MFO, WOA, PSO, SCA, GWO and GWOSCAPSO

Algorithm	LCOE (INR/kWh)	Hydro plant size(kW)	Optimal flow rate(m^3/sec)	Effective Head (m)	PV plant size(kW)	Area for PV plant (m^2)	Total size of Hybrid system (kW)
ALO	6.408	15.74	0.210	9	16.128	95.62	31.87
MFO	6.40845	15.74	0.210	9	16.12	95.58	31.86
WOA	6.50642	15.74	0.210	9	16.12	95.58	31.86
PSO	6.41175	15.74	0.210	9	16.135	95.67	31.88
SCA	6.4353	15.74	0.210	9	16.17	95.87	31.91
GWO	6.40845	15.74	0.210	9	16.12	95.58	31.86
GWOSCAPSO	6.408	15.74	0.210	9	16.12	95.58	31.86

Table 5.10: Specification of the GWOSCAPSO Optimized Hydro System

Parameter	Specification
Optimum Size (kW)	15.74
Optimum Flow rate (m^3/sec)	0.21
Effective Head (m)	9.0
No of Units	3
Flow rate of each Unit (m^3/sec)	0.07
Turbine Type	Cross Flow @ 85% of Full load

Table 5.11: Specification of the GWOSCAPSO Optimized PV System

Parameter	Specification
Optimum Size (kW)	16.12
Total Area (m^2)	95.58
Efficiency (%)	16.94
No of Modules	43
Size of Each Module (Wp)	325
Type of module	Canadian Solar Max Power CS6X-325 Poly-crystalline

ing the location of the alpha grey wolf in the Grey Wolf Optimization (GWO) algorithm using the position update equation of the Sine Cosine Algorithm (SCA) and the position of the beta and delta wolves is updated using the position update equation of the Particle Swarm Optimization (PSO). The objective of the developed method is to enhance the exploration and exploitation capability of the GWO algorithm with possible faster convergence. The governing equations for hybridization and the philosophy of the inspiration behind the proposed effective improvement in the exploration and exploitation of the GWO algorithms is covered in detail. The performance metrics including the statistical techniques which establish the framework for comparison of algorithms utilized in the current study has also been discussed.

The performance of GWOSCAPSO algorithm was evaluated and compared with other meta-heuristic techniques, including PSO, GWO, ALO, WOA, MFO and SCA. A total of twenty-three classical functions were utilized in order to assess and validate the precision, optimal global solution, and efficiency of the recently devised methodology to arrive at the best optimal solution. The results obtained from the simulations demonstrate that the newly developed hybrid method exhibits a higher level of accuracy compared to the GWO, SCA, ALO. MFO, WOA and PSO algorithms. The hybrid GWOSCAPSO outperforms the rest of the algorithms 20 out of 23 times, statistically which shows that the algorithm can be used for solving engineering optimization problems.

In addition, GWOSCAPSO algorithm is applied to three design optimization problems namely the design optimization of DC-DC Boost converter, design optimization of a DC-DC Boost Converter for MPPT applications in standalone PV systems and the optimal sizing of the HRES with PV and Hydro power as the primary sources. The findings conclusively support the efficacy of the GWOSCAPSO algorithm in solving design engineering problems. The performance of the algorithm is evaluated using statistical parameters and the results indicate that the GWOSCAPSO technique distinctly outperforms all other methods.

In recent years, hybridization of algorithms has become a key area of research interest in the optimization studies. The work presented here can be further extended to compare with established hybrid algorithms and to test for efficacy in finding optimal solutions. Furthermore, the work is to be extended to real work engineering problems like design optimization related to fields of civil and mechanical engineering problems as well as other related areas that find increasing application of optimization approaches.

5.6 Summary

This study analyzed the performance of a novel hybrid GWOSCAPSO algorithm for solving optimization challenges in the field of renewable energy studies. A total of six established and popular metaheuristic algorithms are taken as reference and compared in solving the engineering optimization problems. A set of 23 standard benchmark functions are first solved using the algorithm and then the algorithm is applied to solve three design engineering problems. These include the design optimization of DC-DC boost converter at minimized loss, the optimal design of a DC-DC boost converter for MPPT application in PV systems and the size optimization problem of a HRES at minimized LCOE.

The GWOSCAPSO algorithm, when applied to three design engineering problems showed superior performance in the optimized results. Comparison between all the implemented algorithms show that the GWOSCAPSO algorithm is better than the other algorithms in terms of a range of performance metrics that includes best values of objective function, statistically significant results and fast convergence characteristics.