

Article

Statistical Performances Evaluation of APSO and Improved APSO for Short Term Hydrothermal Scheduling Problem

Muhammad Salman Fakhra^{1,†}, Syed Abdul Rahman Kashif^{1,*,†} , Noor Ul Ain^{1,†},
Hafiz Zaheer Hussain^{2,†}, Akhtar Rasool^{3,†}  and Intisar Ali Sajjad^{4,†} 

¹ Department of Electrical Engineering, University of Engineering and Technology, Lahore 54890, Pakistan; salmanfakhra@uet.edu.pk (M.S.F.); noor-ul-ain@uet.edu.pk (N.U.A.)

² Power & Mechanical Division, National Engineering Services Pakistan (NESPAK) Pvt. Limited, Lahore 54000, Pakistan; engr.hafizzaheer@gmail.com

³ Sharif College of Engineering and Technology, Lahore, Pakistan; akhtar@sabanciuniv.edu

⁴ Department of Electrical Engineering, University of Engineering and Technology, Taxila, Pakistan; intisar.ali@uettaxila.edu.pk

* Correspondence: abdulrahman@uet.edu.pk; Tel.: +92-345-409-2528

† These authors contributed equally to this work.

Received: 22 April 2019; Accepted: 11 June 2019; Published: 14 June 2019



Featured Application: Short-term hydrothermal scheduling is a physically existing problem that deals with the combined operation of hydro and thermal generators. The two types of generations are so dispatched that they do not violate the constraints related to thermal power and water reservoirs related to hydro power, while serving the load demand and covering the transmission power losses. It is a type of economic dispatch problem. The research is still in progress to find algorithms that help in such an economic dispatch such that the fuel cost, related to the thermal generators, is minimized and the water reservoirs are also not completely depleted, which are primarily required for irrigation purposes. This research article is a presentation of an effort to find an elegant metaheuristic optimization algorithm that robustly finds the solution of two out of several types of short-term hydrothermal thermal scheduling problems, while considering the standard test problems as already discussed in the literature. The accelerated particle swarm optimization and its improved version are thus presented in this research article, which are serve the defined purpose.

Abstract: The Accelerated Particle Swarm Optimization (APSO) algorithm is an efficient and the easiest to implement variant of the famous Particle Swarm Optimization (PSO) algorithm. PSO and its variant APSO have been implemented on the famous Short-Term Hydrothermal Scheduling (STHTS) problem in recent research, and they have shown promising results. The APSO algorithm can be further modified to enhance its optimizing capability by deploying dynamic search space squeezing. This paper presents the implementation of the improved APSO algorithm that is based on dynamic search space squeezing, on the short-term hydrothermal scheduling problem. To give a quantitative comparison, a true statistical comparison based on comparing means is also presented to draw conclusions.

Keywords: improved APSO; dynamic search space squeezing; independent sample *t*-test

1. Introduction

The PSO algorithm has gained much popularity among the metaheuristic optimization algorithms in the recent past due to its ease of implementation and promise towards finding good approximates to global optimum solutions of complex optimization problems [1]. APSO is a simpler yet brilliant variant of PSO, and it has been proven to find good approximates of global optimum solutions in less time and fewer iterations as compared to PSO algorithms [2]. Dynamic search space squeezing has been applied on PSO to make yet another variant of PSO named improved PSO, and it gives better performance for some optimization problems as compared to PSO itself [3]. Short-term hydrothermal scheduling is a non-linear and multi-modal optimization problem, which has many forms. These can be of a non-cascaded form to a cascaded form. In the non-cascaded form, there is only one reservoir of water, whereas in the cascaded form, there is a series of downstream reservoirs. It can be single objective or multi-objective. In the single-objective Short-Term Hydrothermal Scheduling (STHTS) problem, the objective is only to minimize the cost of the operation of the thermal generating units, whereas in the multi-objective STHTS problem, the other objectives are to reduce the emission of CO_x, NO_x, and SO_x gases, as well. The works in [3–27] discussed all these types of the STHTS problem and the implementations of conventional and non-conventional metaheuristic optimization algorithms on STHTS problems and presented the superiority of one type of algorithm (usually metaheuristic) over other types. Improved PSO, PSO, cuckoo search optimization, quantum behaved PSO, teaching-learning-based optimization, the multi-objective differential algorithm, the gravitational search algorithm, the artificial bee colony algorithm, the genetic algorithm, civilized swarm optimization algorithms, and gravitational search algorithms are better choices for those types of STHTS problems as presented in [3–19]. The works in [20–27] discussed specifically the non-cascaded and single-objective STHTS problems in which one hydel and one equivalent thermal unit of a number of thermal units are dispatched to supply the power demand. PSO and APSO have shown the best results for such problems. Specially, according to [24,25], the APSO algorithm has outperformed the other previously-implemented deterministic and metaheuristic optimization algorithms on non-cascaded and single-objective STHTS problems.

The works in [19–25] discussed specifically the non-cascaded problems in which one hydel and one thermal unit are dispatched to supply the power demand. PSO and APSO have shown the best results for such problems. Especially, according to [24,25], the APSO algorithm has outperformed the other previously-implemented deterministic and metaheuristic optimization algorithms.

Due to the stochastic nature, it is required that there must be a true statistical comparison between the implementations of various algorithms on one type of optimization problem, which was not presented in previous works [3–27]. The work in [28] also presented this thesis while applying the PSO algorithm using digital pheromones. In the STHTS problem, it is therefore also required to present the quantitative analysis by applying statistical hypothesis tests.

This paper is an extended work of [24,25], already published by the authors, and presents the implementation of the improved APSO algorithm on the non-cascaded STHTS problems (two cases) and gives its statistical comparison with the previous APSO implementations on the same problems, by the authors of this paper, in [24,25]. Independent sample *t*-tests will be performed on the dataset of the two implementations to check the superiority of one of the two APSO variants on the same STHTS problems, as discussed in [24,25].

2. Accelerated Particle Swarm Optimization and Its Improved Version

Particle swarm optimization is now a very famous metaheuristic optimization algorithm that has shown promising results when implemented on many types of complex, non-linear, and multi-modal optimization problems. The canonical version of PSO is given in Equation (1) as:

$$\begin{aligned} v_i^{n+1} &= \theta v_i^n + \alpha \epsilon_1 \odot [g^* - x_i^n] + \beta \epsilon_2 \odot [x_i^* - x_i^n] \\ x_i^{n+1} &= x_i^n + v_i^{n+1} \end{aligned} \quad (1)$$

In Equation (1), v_i^{n+1} is the velocity update of particle i , at iteration $n + 1$. This velocity is added to the particle position x_i^n of particle i , at iteration n , to get the updated position x_i^{n+1} of particle i . x_i^* is the local best position of particle i , in iteration history, and g^* is the global best among all the particles, at iteration n . ϵ_1 and ϵ_2 are random number ranging from 0–1.

There are almost more than two dozen variants of this canonical version of the PSO algorithm, as discussed in [29,30]. Accelerated particle swarm optimization is its variant, presented in [1,2], which is very elegant, i.e., easy to understand, the easiest to program, and proven itself to give good approximates to global optimum solutions. Its single step particle update equation is given as Equation (2):

$$x_i^{n+1} = (1 - \beta)x_i^n + \beta g^* + \alpha(\epsilon - 0.5) \quad (2)$$

As can be seen, this single step update equation has ignored the use of the velocity update and did not use the local best position in the updating of particle i . The typical values of α and β are 0.2 and 0.5, respectively. ϵ is a random variable, and its value is between zero and one, as given in [1,2]. This variant has been proven to find the global optimum of highly complex multi-modal functions like the Michaelwics2D function, as presented in [1,2]. There can be many possible variants of Equation (2), as discussed in [1,2]. The variant discussed in this paper is related to the dynamic search space squeezing method in which, at every iteration, the search space of the particles, in all dimensions, is readjusted by taking the influence of the global best particle, i.e., of g^* , each iteration. This is how the chance of each particle to oscillate in the search space is decreased, and there is a lesser chance to get stuck to local optima. By reducing the search space, the range of particles at each dimension is reduced, i.e., the constraints are readjusted, and new constraints are bounded by the original constraints of the optimization problem, by using the dynamic search space squeezing Equations (3)–(7), as presented in [25]. The work in [25] has applied the concept of the dynamic search squeezing technique on the canonical form of PSO, i.e., on Equation (1), on the velocity update equation. These equations help to squeeze the search space dynamically from the given constraints to newer constraints from one iteration to the next iteration. This paper presents this concept of dynamic search space squeezing on the APSO algorithm, i.e., Equation (2), to further enhance its performance. Therefore, rather than squeezing the search space of velocity terms, the search space of particles themselves will be directly squeezed. The dynamic search space squeezing phenomenon is shown in Figure 1.

$$\Delta_{low,i}^n = \frac{g_i^{*n} - x_{i,min}^n}{x_{i,max}^n - x_{i,min}^n} \quad (3)$$

$$\Delta_{high,i}^n = \frac{x_{i,max}^n - g_i^{*n}}{x_{i,max}^n - x_{i,min}^n} \quad (4)$$

$$\Delta_{low,i}^n + \Delta_{high,i}^n = 1 \quad (5)$$

$$x_{i,min}^{n+1} = x_{i,min}^n + (g_i^{*n} - x_{i,min}^n)(\Delta_{low,i}^n) \quad (6)$$

$$x_{i,max}^{n+1} = x_{i,max}^n + (x_{i,max}^n - g_i^{*n})(\Delta_{high,i}^n) \quad (7)$$



Figure 1. Dynamic search space squeezing under the influence of g^* at iteration $n + 1$. The arrows indicate the squeezing of the search space from iteration $n - n + 1$.

3. Short-Term Hydrothermal Scheduling Problem

Short-term hydrothermal scheduling is a type of economic dispatch problem in which a parallel operation of hydel power generating units is combined with thermal power generating units on a generation bus. This problem can have several forms varying from single objective to multi-objective, cascaded to non-cascaded, and pumped storage to non-pumped storage problems. The generation model of the STHTS problem is presented in Figure 2, as taken from [24]. The complexities of these problems can be further enhanced by considering the valve point loading effects of thermal generating units. The detailed analysis of all these problem types was presented in [3–25]. This paper considers two types of these problems as case studies, as already considered in [19–25], to implement the improved APSO algorithm and its comparison with the previous implementations of APSO algorithm in [24,25], in a formal quantitative hypothesis testing. The two problem cases belong to the following two types of STHTS problem.

1. Non-cascaded pumped storage short-term hydrothermal scheduling [24].
2. Non-cascaded non-pumped storage short-term hydrothermal scheduling while considering transmission losses [25].

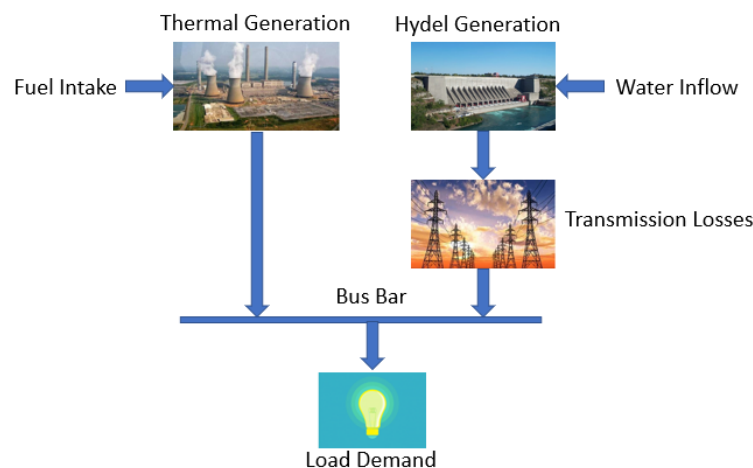


Figure 2. Generation model of the Short-Term Hydrothermal Scheduling (STHTS) problem [25].

It seems more logical to check the performance of one type of algorithm by implementing it on as many types of problems as possible. However, according to the “no free lunch theorem”, as given in [29–34], if one algorithm performs best on a Type A problem, it does not necessarily mean that it performs the best for a Type B problem, i.e., an algorithm cannot be proven to be the best optimization algorithm for all types of problems. Therefore, it is required to find a good performing algorithm for every type of optimization problem. Problems 1 and 2 belong to the same class of single objective and non-cascaded STHTS problems and are different in terms of structure from the remaining types of STHTS problems. Any algorithm, providing the best optimum to these problems, will perform well on all similar types of STHTS problems.

APSO algorithms are the simplest and most elegant metaheuristic algorithms, which, without having a lot of tuning parameters and having just a single-step particle update equation, are able to solve non-linear and multi-modal objective functions like Michaelwicz 2D functions, as given in [1,2]. Therefore, the elegance and robustness of these APSO algorithms make them the first choice out of many metaheuristic algorithms to implement on optimization problems. The other types of STHTS problems, for example the cascaded STHTS of a single objective-type problem, requires metaheuristic algorithms with more tuning parameters instead of the APSO algorithm, like PSO and improved PSO. Another type is the multi-objective STHTS problem, which requires variants of metaheuristic

algorithms specially designed for multi-objective problems, like multi-objective PSO, ant colony and genetic algorithms, cuckoo search, the gravitational search algorithm, teaching-learning-based algorithms, and water cycle algorithms [3–19].

The problems taken into consideration in this paper have been solved using the APSO algorithm by the authors of this paper already in [23,24], which outperformed the other algorithms implemented on the same problems given in [22,26,27]. In this enhanced research, the improved APSO algorithm has been implemented on the same problems that used dynamic search space squeezing, and this modified APSO has shown even better results than the APSO algorithm, as presented in the Results Section. Figure 2 presents the basic generation model of the STHTS problem, which was also considered in [23,24].

4. Hypothesis Testing

Due to their stochastic nature, metaheuristic algorithms are destined to give different results for different trials on the same optimization problem. It is therefore required to test the performance of the implementations by performing a statistical analysis on a dataset of the results obtained. To check whether Algorithm A is superior to Algorithm B in terms of performance, there are hypothesis tests that quantitatively establish the superiority. The works in [3–27] have not taken into consideration this point, and though the performance of algorithms presented in those references was the best, the quantitative proof has not been established. This proof can be established by performing hypothesis testing to compare the means of the performances. For this purpose, a null hypothesis is established. The null hypothesis states that there is no difference in the performance of the two algorithms. By performing independent sample *t*-tests, the null hypothesis can either be accepted or rejected [28,35]. If the null hypothesis is rejected, this means that there is a significant difference between the performances of the two algorithms. If the null hypothesis is accepted, this means that, statistically, there is no difference between the performances of the two algorithms [28,35].

5. Methodology

According to [19–27], the non-cascaded hydrothermal scheduling problem can be described by Equations (8)–(14).

$$\min(f) = \sum_{j=1}^N n_j F_j \quad (\text{cost function or objective function}) \quad (8)$$

subject to:

$$P_{hyd} + P_T = P_{Demand} + P_{Losses} \quad (\text{Power balance equality constraint}) \quad (9)$$

$$\sum_{j=1}^N n_j Dis_j = Dis_{total} \quad (\text{Discharge rate equality constraint}) \quad (10)$$

$$V_{min} < V_j < V_{max} \quad (\text{Water reservoir volume limits}) \quad (11)$$

$$Dis_{min} < Dis_j < Dis_{max} \quad (\text{Discharge rate limits}) \quad (12)$$

$$P_{T,min} < P_{T,j} < P_{T,max} \quad (\text{Thermal power limits}) \quad (13)$$

$$P_{hyd,min} < P_{hyd,j} < P_{hyd,max} \quad (\text{Hydro power limits}) \quad (14)$$

The reservoir's volume and the discharges are balanced by the continuity Equation (15) given as:

$$V_j = V_{j-1} + n_j(R_j - Dis_j - S_j) \quad (15)$$

To implement APSO or improved APSO algorithms on the non-cascaded STHTS problem, the following steps can be implemented;

1. Randomly initialize the volume vectors (particles) within the given volume constraints. In this paper, uniform random number generators have been used.
2. Calculate the water discharge rate, hydro power, and thermal power using the values in Step 1.
3. Check the limits in the constraints. If the limits are not met, restart from Step 1. If the limits are met, proceed to Step 4.
4. Find the total cost using the thermal power values found in Step 2, against each particle.
5. Take the minimum cost value and its corresponding volume vector. That volume vector will be the global best particle.
6. Update all the particles using APSO/improved APSO updating “(10)”.
7. Dynamically squeeze the volume constraints by applying search space squeezing equations for the improved APSO algorithm
8. Iterate from Steps 2–6 till the stopping criterion is reached.
9. Get the results.

6. Results and Discussions

This paper presents the essence and importance of true statistical quantitative tests to check the superiority of one type of algorithm over another type of algorithm. This is because, the metaheuristic optimization algorithms are stochastic, as well as deterministic in nature. Due to the stochastic nature, there is an in-built randomness, and therefore, no metaheuristic optimization algorithm the gives same redundant results when implemented on one type of problem. Therefore, to check if one algorithm is performing that same on the same problem every time, the performance is judged by standard deviations among the trials. To check the performance of two different types of algorithms, on the same problem, the data taken from an equal number of trials are hypothetically tested for significant differences in the means of all the trials. This type of testing is known as independent sample *t*-tests. This *t*-test was performed with the software SPSS, i.e., Statistical Package for Social Sciences. To test the performance of APSO and improved APSO, two test cases have been taken, as already discussed in [24,25].

6.1. Case 1: Non-Cascaded Pumped Storage Hydrothermal Scheduling [24,27]

This is a two-unit hydrothermal scheduling problem as given in [24]. The total number of scheduling periods was six, out of which three periods were pumping, i.e., discharged water is pumped back to the reservoir, while three periods were generating. The thermal unit characteristics are given in Equation (16):

$$F_T = 3877.5 + 3.9795P_T + 0.00204P_T^2 \text{ } \$/h \quad (16)$$

$$200\text{MW} \leq P_T \leq 2500\text{MW} \quad (17)$$

The generating and pumping models are given by Equations (18) and (19), respectively.

$$Dis(P_{hyd}) = 200 + 2P_{hyd}(\text{acre.ft/h}) \text{ for } 0 \leq P_{hyd} \leq 300 \text{ MW} \quad (\text{Generating characteristics}) \quad (18)$$

$$Dis(P_{H,pump}) = -600(\text{acre.ft/h}) \text{ with } P_{H,pump} = -300 \text{ MW} \quad (\text{Discharging characteristics}) \quad (19)$$

The starting and ending volume of the reservoir = 8000 (acre-ft/h). The load demand at every interval is given in Table 1.

The problem is implemented using both the APSO algorithm and improved APSO algorithm. The work in [24] has already discussed the implementation of APSO on this problem; however, this paper presents its comparison with the improved APSO algorithm on hypothetical testing. The improved APSO algorithm has shown even more promising results as compared to APSO algorithm. Table 2 presents the comparison between the two implementations.

Both algorithms have been tested on the Case 1 problem for sets of a hundred and two hundred trials each to have a normally-distributed dataset. Their means were compared using the hypothesis testing independent sample *t*-test. The results obtained from the SPSS software run are given in Table 3.

Table 1. Load demand for each scheduling interval of four hours each.

Interval	1	2	3	4	5	6
Demand MW	1600	1800	1600	500	500	500

Table 2. Comparison of some performance parameters between APSO and improved APSO on the pumped storage STHTS problem.

Performance Parameter or Algorithm	APSO	Improved APSO
Minimum Cost	269,642.4001	269,642.4000
Average Cost	269,642.5318	269,642.4752
Maximum Cost	269,643.2336	269,642.402
No. of acceptable convergences	4 out of 100 trials	97 out of 100 trials
Standard deviation	6.2557539	0.0088976
Average No. of iterations	7	8
Average computation time *	0.462 s	0.506 s

* Average computation time is according to MATLAB 2015 on a Core i5 second-generation processor.

Table 3. SPSS results of independent sample *t*-test showing the comparison between simple APSO and improved APSO on the pumped storage non-cascaded STHTS problem.

Independent Samples <i>t</i> -Test									
Levene's Test for the Equality of Variances					<i>t</i> -Test for the Equality of Means				
Comparison _APSO	F	Sig.	t	df	Sig. (2-tailed)	Mean Difference	Std. Error Difference	95% Confidence Interval of the Difference	
								Lower	Upper
Equal variances assumed (100 instances)	62.424	0.000	5.333	198	0.000	3.3361210	0.6255760	2.1024741	4.5697679
Equal variances assumed (200 instances)	89.787	0.000	−5.768	398	0.000	−1.086	0.1883	−1.4564	−0.7159
Equal variances not assumed (100 instances)			5.333	99.00	0.000	3.3361210	0.6255760	2.0948425	4.5773995
Equal variances not assumed (200 instances)			−5.768	200.638	0.000	−1.08622	0.1883	−1.4575	−0.718

The null hypothesis was that there was no difference in the performance of the two algorithms on the pumped storage STHTS problem. This hypothesis was tested on a confidence interval value of 95%, giving a significance value (2-tailed) equal to 0.000, i.e., for both sets of a hundred and two hundred trials lesser than 0.05 or 5% [35]. This value suggests that the null hypothesis is rejected, and the statistical test quantitatively suggests that the improved APSO algorithm has performed significantly differently from the simple APSO algorithm. Furthermore, Levine's test of the equality of variances suggest that for a significance value equal to 0.000, which is less than 0.05, variances are also statistically different, significantly. Table 2 further elaborates the difference in performance by

establishing the superiority of the improved APSO over the simple APSO algorithm. Table 2 also presents that though the minimum values achieved by the two algorithms were the same, the improved APSO algorithm converged to the acceptable result, i.e., this paper has considered an acceptable cost value less than 269,642.4001, 97% of the time as compared to the simple APSO algorithm, which converged to acceptable results 4% of the time. The number of iterations and computation time were very low for both algorithms; however, due to the dynamic search space squeezing, improved APSO took a slightly higher average time to converge to the solution as compared to the simple APSO algorithm. However, the difference in computation time was not very significant. Although the APSO algorithm has given promising results, the improved APSO algorithm has proven itself better for the pumped storage non-cascaded STHTS problem, statistically. Table 4 presents the best results of the improved APSO implementation on the non-cascaded pumped storage STHTS problem.

Table 4. Power flow and cost optimization with the improved APSO algorithm implementation on the non-cascaded pumped storage STHTS problem.

Interval	Demand (MW)	PT (MW)	Phyd (MW)	Dis (ac-ft/h)	V (ac-ft)	Total Cost (\$)
1	1600	1449.99	150	500	5999.95	269,642.4
2	1800	1500	300	800	2799.819	
3	1600	1450	149.97	500	800	
4	500	800	−300	−600	3200	
5	500	800	−300	−600	5600	
6	500	800	−300	−600	8000	

The convergence characteristics of the simple APSO and improved APSO are given in Figures 3 and 4, respectively. Figure 5 presents the convergence characteristics of the improved APSO algorithm on the non-cascaded pumped STHTS problem for five different instances. The algorithm has shown good performance in reaching the good approximates of the global optimum solution for all the instances. These figures are taken as the successful trials for both algorithms. The number of particles was taken to be equal to 200 for both implementations. The convergence characteristics of both algorithms were quite similar and showed the general nature of PSO variants. The algorithms, as their names suggest, are accelerated in nature, i.e., they are fast at finding good approximates of global optima.

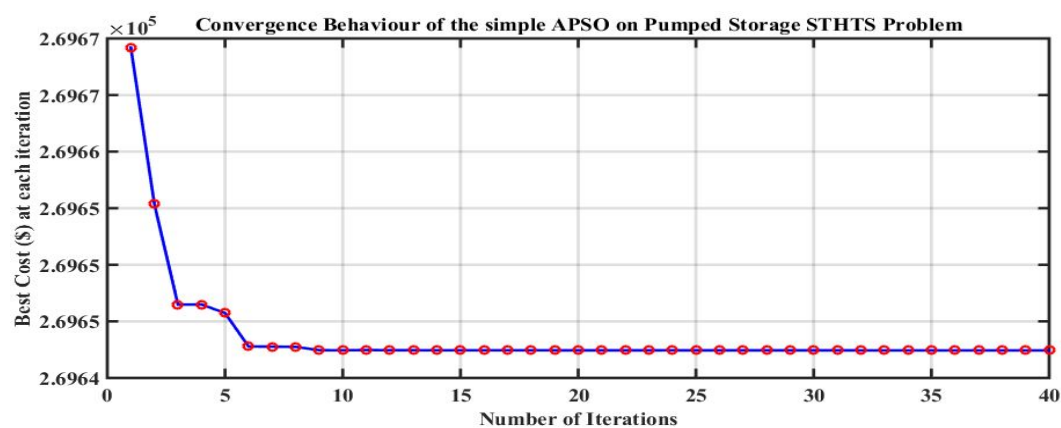


Figure 3. Convergence characteristics of the simple APSO algorithm on the non-cascaded pumped STHTS problem.

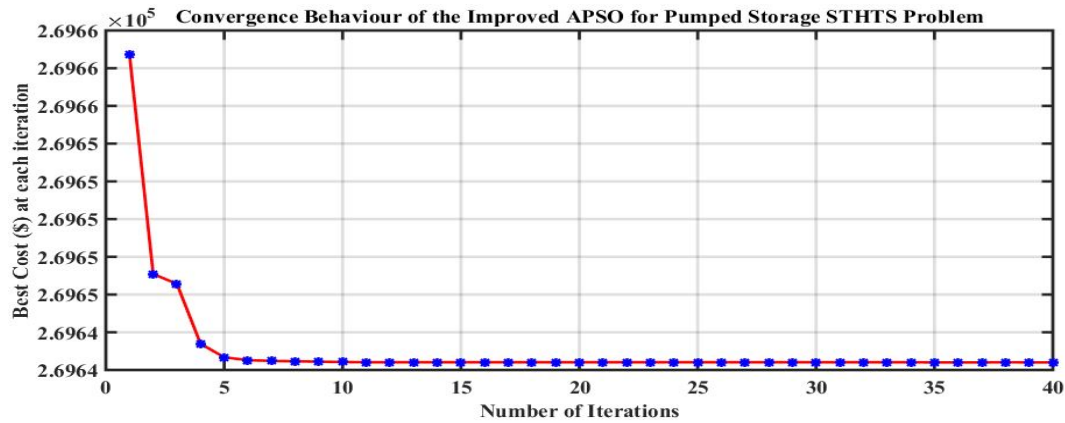


Figure 4. Convergence characteristics of the improved APSO algorithm on the non-cascaded pumped SHTS problem.

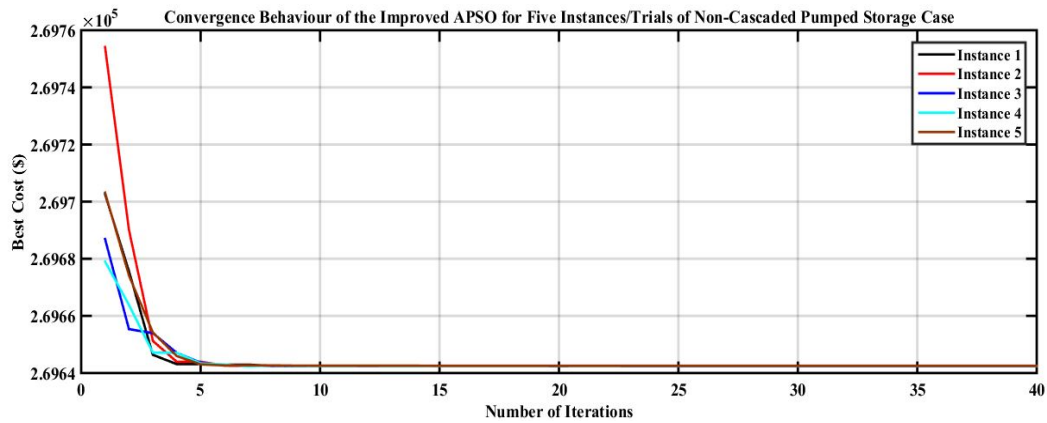


Figure 5. Convergence characteristics of the improved APSO algorithm on the non-cascaded pumped SHTS problem for five different instances.

6.2. Case 2: Non-Cascaded Non-Pumped SHTS Considering Transmission Losses [25]

This is a two-unit and non-pumped hydrothermal scheduling problem, as given in [25]. The total number of scheduling periods is six, and the discharging characteristics of the thermal unit are given in Equations (20) and (21); where the fuel cost is 1.15 \$/MBTU.

$$F_T = 500 + 8P_T + 0.0016P_T^2 \quad (\text{MBTU}/h) \quad (20)$$

$$150 \text{ MW} \leq P_T \leq 1500 \text{ MW} \quad (21)$$

The water discharging constraints are given in Equations (22)–(25):

$$D_{is}(P_{hyd}) = 330 + 4.97P_{hyd} \quad (\text{acre} - \text{ft}/h) \quad (22)$$

for:

$$0 \text{ MW} \leq P_{hyd} \leq 1000 \text{ MW} \quad (23)$$

$$D_{is}(P_{hyd}) = 5300 + 12(P_{hyd} - 1000) + 0.05(P_{hyd} - 1000)^2 \quad (\text{acre} - \text{ft}/h) \quad (24)$$

for

$$1000 \text{ MW} \leq P_{hyd} \leq 1100 \text{ MW} \quad (25)$$

The transmission losses are given by Equation (26):

$$(P_{loss}) = 0.0008P_{hyd}^2 \text{ (MW)} \quad (26)$$

The discharge rate characteristics are given in Equation (27):

$$5300 \text{ (acre-ft/hr)} \leq D_{is} \leq 7000 \text{ (acre-ft/hr)} \quad (27)$$

The load demand in every interval is given in Table 5.

Table 5. Load demand for each scheduling interval (each interval is 12 hours, making a total scheduling period of 3 days).

Interval	1	2	3	4	5	6
Demand MW	1200	1500	1100	1800	950	1300

The reservoir's water volume flow constraints as given in [25,27] are:

1. 100,000 acre-feet constitute the volume of water in the reservoir prior to the first scheduling period.
2. 60,000 acre-feet constitute the volume of water in the reservoir in the last period.
3. Volume limit constraints in the first iteration are in acre-ft: $60,000 \text{ (acre-ft)} \leq V \leq 120,000 \text{ (acre-ft)}$
4. Continuous incoming flow into the reservoir is 2000 acre-feet/hour throughout the scheduling period.
5. The continuity equation as given in Equation (15) is to be met.

The problem was implemented using both APSO algorithm and the improved APSO algorithm. The work in [25] already discussed the implementation of APSO for this problem; however, this paper presents its comparison with the improved APSO algorithm on hypothetical testing. The improved APSO algorithm has shown even more promising results as compared to the APSO algorithm. Table 6 presents the comparison between the two implementations.

Table 6. Comparison of some performance parameters between APSO and the improved APSO on non-cascaded non-pumped STHS problem.

Performance Parameter or Algorithm	APSO	Improved APSO
Minimum Cost	727,870.00	727,855.8
Average Cost	728,422.7	728,539.4
Maximum Cost	730,103.3	732,386.8
No. of acceptable convergences	15 out of 100 trials	15 out of 100 trials
Standard deviation	413.35	539.34
Average No. of iterations	6	7
Average computation time *	1.035 s	1.227 s

* Average computation time is according to MATLAB 2015 on a Core i5 second-generation processor.

Both algorithms have been tested on the Section 6.2 Case 2 problem for sets of a hundred and two hundred trials each to have a normally-distributed dataset. Their means were compared using the hypothesis testing independent sample *t*-test. The results obtained from the SPSS software run are given in Table 7.

For this (Section 6.2 Case 2) problem, the value of significance of two-tailed *t*-test showed that both algorithms were not significantly different in terms of performance because the value was greater than 5%. This means that for more than 25% of the time, the results were somewhat repeated. Therefore, at the 95% confidence level, the null hypothesis could not be rejected, and the means were not statistically different. However, the improved APSO has been able to find a better approximation to the global best optima, as given in Table 6. The number of iterations and computation time were very low for both of these algorithms; however, due to the dynamic search space squeezing, improved APSO took a

slightly higher average time to converge to the solution as compared to the simple APSO algorithm. However, the difference in computation time was not very significant.

Table 7. SPSS results of the independent sample *t*-test showing the comparison between simple APSO and improved APSO on the non-cascaded non-pumped STHTS problem.

Independent Samples <i>t</i> -Test									
Levene's Test for the Equality of Variances			<i>t</i> -Test for the Equality of Means						
Comparison APSO	F	Sig.	t	df	Sig. (2-tailed)	Mean Difference	Std. Error Difference	95% Confidence Interval of the Difference	
								Lower	Upper
Equal variances assumed (100 instances)	4.674	0.032	−1.133	198	0.259	−76.9991850	67.9525842	−211.0028687	57.0044987
Equal variances assumed (200 instances)	14.501	0.000	1.605	398	0.109	77.2827	48.1474	−17.3722	171.93
Equal variances not assumed (100 instances)			−1.133	185.468	0.259	−76.9991850	67.9525842	−211.0585712	57.0602012
Equal variances not assumed (200 instances)			1.1605	367.693	0.109	77.2827	48.1474	−17.3960	171.9615

Therefore, if some more tuning parameters were properly adjusted, both algorithms would have the capability to find robust approximations to global optimum solutions with low standard deviations, for the non-cascaded non-pumped STHTS problem while considering transmission line losses, improved APSO having more ability to find a better approximation to global minima. Table 8 gives the best cost result of improved APSO's performance.

Table 8. Power flow and cost optimization with the improved APSO algorithm implementation.

Interval	PT (MW)	Phyd (MW)	Ploss (MW)	Dis (acre-ft/h)	V (acre-ft)	Total Cost (\$)
1	833.91	377.4	11.399	2206	97,527.06	727,855.8
2	950.66	575.86	26.52	3192	83,222.43	
3	813.76	293.10	6.87	1786.7	85,781.40	
4	1091.94	753.47	45.41	4074.7	60,884.22	
5	733.34	220.54	3.89	1426.1	67,770.92	
6	851.08	466.313	17.395	2647.57	60,000	

The convergence characteristics of the simple APSO and improved APSO are given in Figures 6 and 7, respectively, on the non-cascaded non-pumped STHTS problem considering transmission losses. These figures are taken for the successful trials for both the algorithms. Figure 8 presents the convergence characteristics of the improved APSO algorithm on the non-cascaded non-pumped STHTS problem for five different instances. The algorithm has shown good performance in reaching good approximates of the global optimum solution for four instances, whereas the algorithm could not reach a good solution for one instance. The number of particles was taken to be equal to 200 for both implementations. The convergence characteristics of both the algorithms were quite similar and showed the general nature of PSO variants. The algorithms, as their names suggest, are accelerated in nature, i.e., they are fast at finding good approximates of global optima.

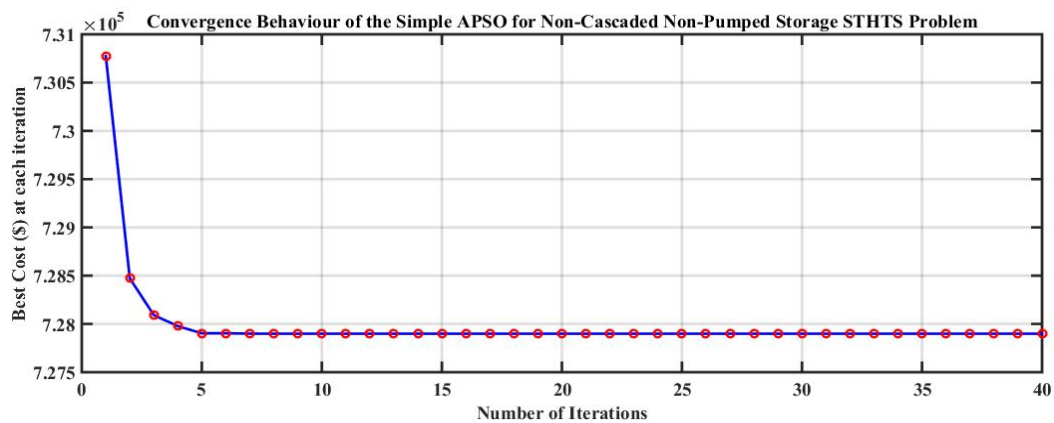


Figure 6. Convergence characteristics of the simple APSO algorithm on the non-cascaded non-pumped SHTS problem taken from [25].

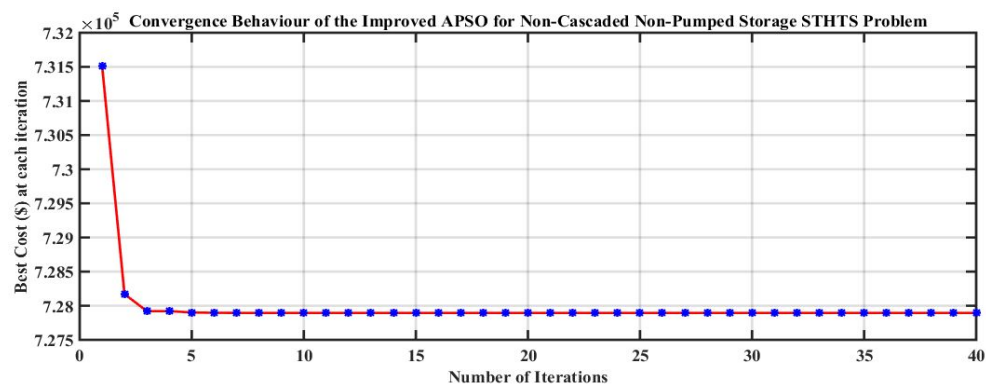


Figure 7. Convergence characteristics of the improved APSO algorithm on the non-cascaded non-pumped SHTS problem.

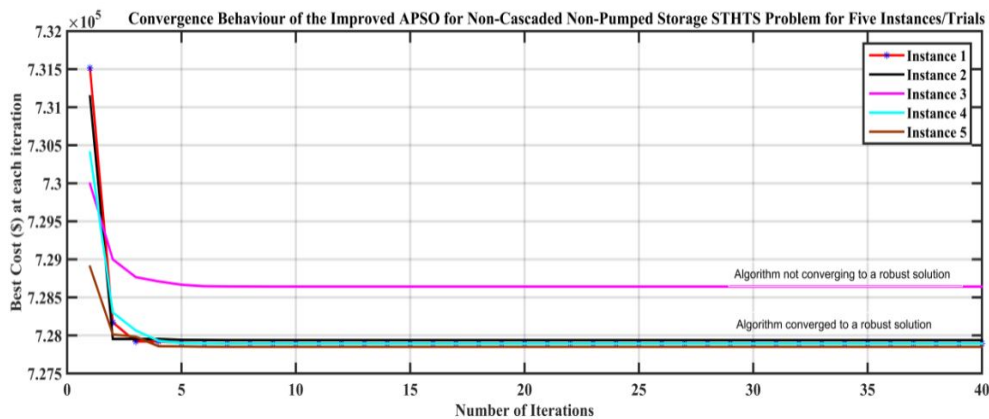


Figure 8. Convergence characteristics of the improved APSO algorithm on the non-cascaded non-pumped SHTS problem for five different instances.

7. Additional SHTS Problems (as Test Cases)

In this research paper, the two problems of interest were presented as Section 6.1 Case 1 and Section 6.2 Case 2. These two problems were already solved by other algorithms, as can be seen in [24,25]. For the interested reader, two new SHTS problems were added, which are quite similar to the Section 6.1 Case 1 and Section 6.2 Case 2 problems discussed in the previous section. These problems have not been solved in the literature and are very close to practical SHTS problems, as are Section 6.1 Case 1 and Section 6.2 Case 2. The simple APSO and improved APSO algorithms have

been implemented on these two problems, taking α and β equal to 0.5, and the statistical results are presented.

7.1. Case 1: Non-Cascaded Pumped-Storage STHTS Problem Considering Valve Point Loading

In steam turbine-based thermal generators, valves control the steam entering the turbine through separate nozzle groups. Each nozzle group achieves best efficiency when operated at full output. Therefore, when increasing the output, valves are opened in sequence in order to achieve the highest possible efficiency for a given output. This causes a rippled cost curve, which is usually a sinusoidal wave riding on the quadratic cost function [19]. A valve point loading-based STHTS problem was given to test the effectiveness of the proposed improved APSO algorithm in comparison with the already tested simple APSO algorithm in the literature. The system taken was that of Case 1 with a valve point loading-based cost function. The cost function is now as given in (28):

$$F_T = 3877.5 + 3.9795P_T + 0.00204P_T^2 + 500 \sin(0.085(200 - P_T)) \text{ } \$/h \quad (28)$$

The rest of the problem data are the same as Section 6.1 Case 1 (Equations (17)–(19)) of the previous section. The statistical results of both the simple APSO and improved APSO algorithms are shown in Table 9. Table 10 shows the results of the SPSS results of the independent sample *t*-test performed for the comparison of the simple APSO and improved APSO algorithms on the non-cascaded pumped-storage STHTS problem considering valve point loading. Table 11 shows the results of power flow and cost optimization with the improved APSO algorithm implementation on the non-cascaded pumped-storage STHTS problem considering valve point loading. Convergence results for the simple APSO and improved APSO are shown in Figures 9 and 10, respectively.

The results show that there is no significant difference between the performances of these two algorithms, statistically; however, the improved APSO algorithm has reached acceptable solutions more quickly than the simple APSO algorithm. If the tuning parameters of both algorithms are adjusted properly, both algorithms can help in finding a good approximate of the global optimum solution.

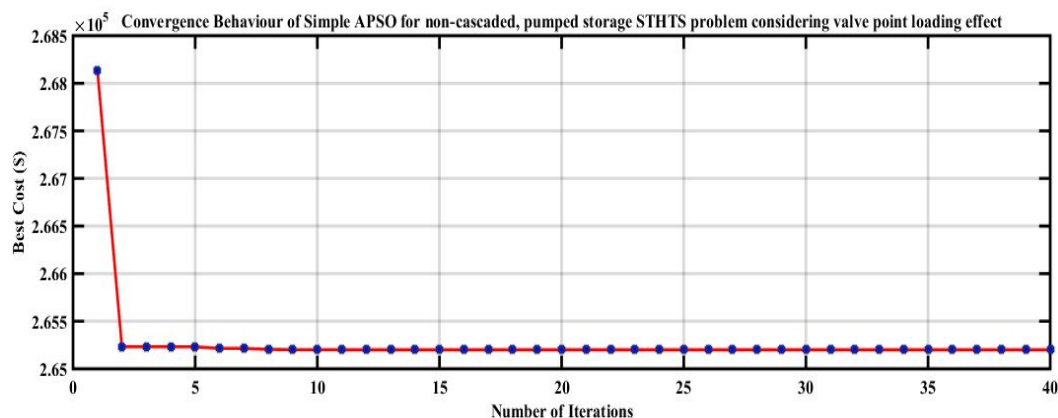


Figure 9. Convergence characteristics of the simple APSO algorithm on the non-cascaded pumped-storage STHTS problem considering valve point loading.

Table 9. Comparison of some performance parameters between APSO and improved APSO on the pumped storage STHTS problem considering valve point loading.

Performance Parameter or Algorithm	APSO	Improved APSO
Minimum Cost	265,200.2082	265,200.2082
Average Cost	265,433.4813	265,459.0192
Maximum Cost	265,781.3508	269,539.6085
No. of acceptable convergences	53 out of 100 trials	59 out of 100 trials
Standard deviation	174.6004	454.7182
Average No. of iterations	7	7
Average computation time *	0.5 s	0.6 s

* Average computation time is according to MATLAB 2015 on a Core i5 second-generation processor.

Table 10. SPSS results of the independent sample *t*-test showing the comparison between simple APSO and improved APSO on the non-cascaded pumped-storage STHTS problem considering valve point loading.

Independent Samples <i>t</i> -Test									
Levene's Test for the Equality of Variances				<i>t</i> -Test for the Equality of Means					
Comparison _APSO	F	Sig.	t	df	Sig. (2-tailed)	Mean Difference	Std. Error Difference	95% Confidence Interval of the Difference	
								Lower	Upper
Equal variances assumed (100 instances)	1.672	0.198	0.524	198	0.601	25.53785	48.70872	−70.51659	121.59230
Equal variances not assumed (100 instances)			0.524	127.571	0.601	25.53785	48.70872	−70.84376	121.91947

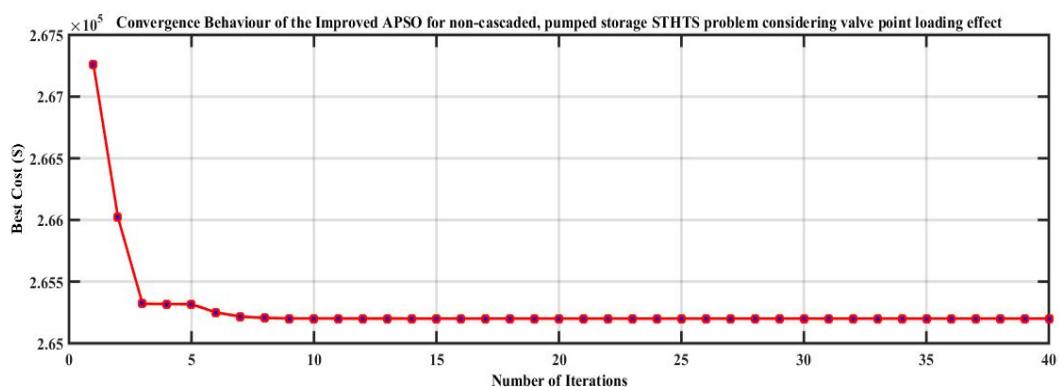
**Figure 10.** Convergence characteristics of the improved APSO algorithm on the non-cascaded pumped-storage STHTS problem considering valve point loading.

Table 11. Power flow and cost optimization with the improved APSO algorithm implementation on the non-cascaded pumped-storage STHTS problem considering valve point loading.

Interval	Demand (MW)	PT (MW)	Phyd (MW)	Dis (acre-ft/h)	V (acre-ft)	Total Cost (\$)
1	1600	1392.80	207.13	614.27	5542.813	265,200.2
2	1800	1540.4	259.51	719.03	2666.47	
3	1600	1466.6	133.34	466.68	800	
4	500	800	−300	−600	3200	
5	500	800	−300	−600	5600	
6	500	800	−300	−600	8000	

7.2. Case 2: Non-Cascaded and Non-Pumped Storage STHTS Problem While Not Considering Transmission Losses

This problem is similar to the Section 6.2 Case 2 problem of the last section. However, in this new system, the transmission line losses are considered as zero. The thermal and hydro units have the following characteristics.

Thermal system: Equations (29) and (30):

$$F_T = 700 + 4.8P_T + 0.0005P_T^2 \text{ ($/h)} \quad (29)$$

$$200 \text{ MW} \leq P_T \leq 1200 \text{ MW} \quad (30)$$

Hydro system: Equations (31)–(33):

$$D_{is}(P_{hyd}) = 260 + 10P_{hyd} \text{ (acre-ft/h)} \quad (31)$$

for:

$$0 \text{ MW} \leq P_{hyd} \leq 350 \text{ MW} \quad (32)$$

$$D_{is}(P_{H,pump}) = 0 \text{ (acre-ft/h) with } P_{hyd} = 0 \text{ MW} \quad (33)$$

The starting and ending volume of the reservoir were 16,000 (acre-ft) and 12,000 (acre-ft) with a maximum allowed volume of 18,000 (acre-ft) and a minimum allowed volume of 12,000 (acre-ft). A constant water inflow of 2000 (acre-ft/h) was to be maintained in the reservoir for each scheduling interval. It was considered that there were no electrical transmission power losses. The load demand during each interval of four hours each is given in Table 12. Table 13 shows the SPSS results of the independent sample *t*-test showing the comparison between simple APSO and improved APSO on the non-pumped storage non-cascaded STHTS problem without transmission losses. Table 14 presents the power flow and cost optimization with the improved APSO algorithm implementation on the selected problem. Table 15 shows the comparison of some performance parameters between APSO and improved APSO on the non-cascaded and non-pumped storage STHTS problem while not considering transmission losses. Convergence results for the simple APSO and improved APSO for this case are shown in Figures 11 and 12, respectively.

Table 12. Load demand for each scheduling interval of 4 hours each.

Interval	1	2	3	4	5	6
Demand MW	600	1000	900	500	400	500

The results from the *t*-test show that for this problem type, both the simple APSO and improved APSO algorithms performed equally well, statistically. However, the improved APSO algorithm was able to find a better approximate to the global minimum solution as compared to the simple APSO algorithm. Moreover, the improved APSO algorithm has found the acceptable solution more times as

compared to the simple APSO algorithm. If the tuning parameters of both the algorithms are adjusted properly, both algorithms can help in finding a good approximate of the global optimum solution.

Table 13. SPSS results of the independent sample *t*-test showing the comparison between simple APSO and improved APSO on the non-pumped storage non-cascaded STHTS problem without transmission losses.

Independent Samples <i>t</i> -Test									
Levene's Test for the Equality of Variances				<i>t</i> -Test for the Equality of Means					
Comparison _APSO	F	Sig.	t	df	Sig. (2-tailed)	Mean Difference	Std. Error Difference	95% Confidence Interval of the Difference	
								Lower	Upper
Equal variances assumed (100 instances)	0.158	0.691	−0.853	198	0.395	−1.27093	1.48974	−4.20872	1.66686
Equal variances not assumed (100 instances)			−0.853	199.946	0.395	−1.27093	1.48974	−4.20872	1.66686

Table 14. Power flow and cost optimization with the improved APSO algorithm implementation on the non-cascaded non-pumped-storage STHTS problem while not considering transmission losses.

Interval	PT (MW)	Phyd (MW)	Ploss (MW)	Dis (acre-ft/h)	V (acre-ft)	Total Cost (\$)
1	600	475.8	124.11	1501.10	17,995.58	72,658.09
2	1000	699.4	299.50	3255.49	12,973.58	
3	900	702.7	197.96	2239.63	12,015.06	
4	500	348.6	151.94	1779.47	12,897.18	
5	400	282.7	117.46	1434.66	15,158.53	
6	500	246.4	252.96	2789.63	12,000.00	

Table 15. Comparison of some performance parameters between APSO and improved APSO on the non-cascaded and non-pumped storage STHTS problem while not considering transmission losses.

Performance Parameter or Algorithm	APSO	Improved APSO
Minimum Cost	72,662.81	72,658.09
Average Cost	72,684.09	72,682.82
Maximum Cost	72,708.60	22,717.87
No. of acceptable convergences	72 out of 100 trials	80 out of 100 trials
Standard deviation	10.62	10.44
Average No. of iterations	50	50
Average computation time *	0.512 s	0.506 s

* Average computation time is according to MATLAB 2015 on a Core i5 second-generation processor.

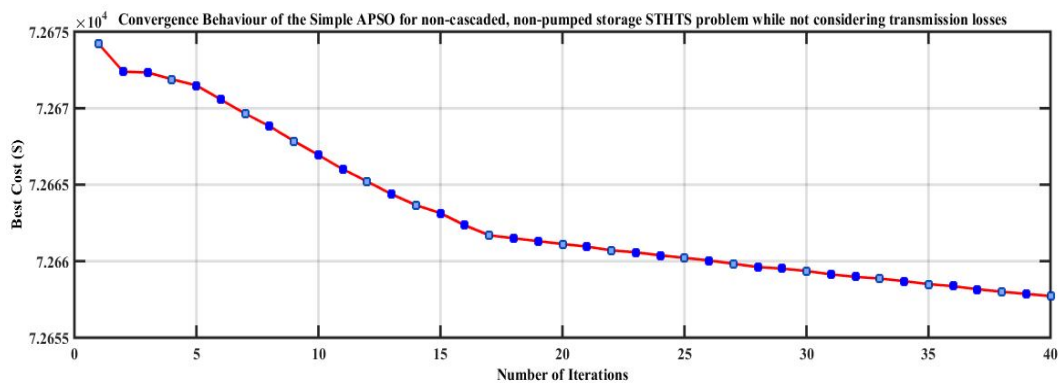


Figure 11. Convergence characteristics of the simple APSO algorithm on the non-cascaded non-pumped SHTS problem without transmission loss.

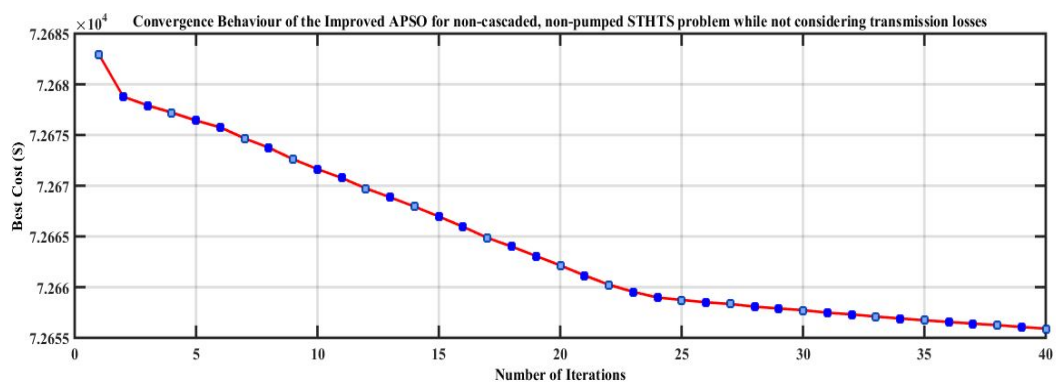


Figure 12. Convergence characteristics of the improved APSO algorithm on the non-cascaded non-pumped SHTS problem without transmission loss.

8. Optimization of Some General Deterministic Non-Linear Test Functions Using the Proposed Improved APSO Algorithm

To check the performance of the proposed improved APSO algorithm, some hard and general deterministic optimization problems have been solved using the APSO and improved APSO algorithms. The problem functions are given in Table 16 and are plotted in Figure 13. Table 17 presents the results of the implementations of APSO and the proposed improved APSO algorithm on these complex optimization functions, showing the capability of these two algorithms to solve such non-linear and multi-modal objective functions.

Table 16. Selected functions for the validation of the proposed improved APSO.

Optimization Function	Michaelwics 2D function
Mathematical form	$f(x, y) = -\sin(x) \sin^{20}(\frac{x}{\pi}) + \sin(y) \sin^{20}(\frac{2y^2}{\pi})$ $(x, y) = [0, 4] \times [0, 4]$
Nature of the problem and its global optimum value	Highly non-linear and multi-modal Global optimum approximately at $(x, y, z) = (2.20319, 1.57049, -1.801)$
Optimization Function	Rosenbrock 2D function (Banana function)
Mathematical form	$f(x, y) = (1 - x)^2 + 100 \times (y - x^2)^2$ $(x, y) = [-5, 5] \times [-5, 5]$
Nature of the problem and its global optimum value	Non-linear and convex Global optimum at $(x, y, z) = (1, 1, 0)$
Optimization Function	De Jong 2D function
Mathematical form	$f(x, y) = x^2 + y^2$ $(x, y) = [-5, 5] \times [-5, 5]$
Nature of the problem and its global optimum value	Non-linear and convex Global optimum at $(x, y, z) = (0, 0, 0)$
Optimization Function	Egg crate 2D function
Mathematical form	$f(x, y) = x^2 + y^2 + 25[\sin^2(x) + \sin^2(y)]$ $(x, y) = [-5, 5] \times [-5, 5]$
Nature of the problem and its global optimum value	Highly non-linear and multi-modal Global optimum at $(x, y, z) = (0, 0, 0)$

It can be seen that both the APSO and improved APSO algorithms have shown promising results to find the good approximates to the determined or known global minima of the highly non-linear and multi-modal test functions. The performances of both APSO and improved APSO were good; however, the improved APSO had the capability to find closer results to the determined or known values of the minima of these functions.

The average value of the set of minima obtained for the case of the egg-crate function was high, as well as the value of standard deviation. The reason is that the egg-crate function had many local minima quite near the global minimum, as can be seen in Figure 13. Therefore, due to the dependency of the particle updating process, only on the g^* value, and due to the presence of fewer tuning parameters, both the simple APSO and improved APSO algorithms sometimes became stuck in the local optima. If some chaotic maps are introduced, the stochastic nature of APSO and improved APSO can be increased to allow the particles to escape from local peaks. However, the improved APSO algorithm has still achieved a higher number of successful results for the egg-crate function. Similarly, for the Michaelwicz 2D function, both improved APSO and APSO sometimes become stuck in the local minima. However, for most of the time, good and robust solutions were achieved. If the neighborhood topologies of particles and some stochastic term are dominantly added in the particle update equation, the number of successful achievements of global optimum solution can be increased to more than 90% of the time.

Table 17. Performance of the proposed improved APSO compared with available references.

Optimization Function	Michaelwics 2D Function
Minimum Using Improved APSO	$(x, y, z) = (2.2010, 1.5710, -1.8012)$ Average = -1.6932 Standard deviation = 0.31429 Successful trials = 15/20 (-1.7950 and less taken as successful values)
Iterations Using Improved APSO	15 iterations to achieve nearest results to three decimal places. Best result achieved after 70 trials.
Minimum Using Original Algorithm	APSO in [1] with $(x, y, z) = (2.20319, 1.57049, -1.801)$
Iterations Using Previous Algorithms	15 iterations to achieve nearest result. Best result achieved after 200 trials.
Optimization Function	Rosenbrock/Banana 2D
Minimum Using Improved APSO	$(x, y, z) = (1.0047, 1.0099, 0.000)$ Average = 0.09987 Standard deviation = 0.3389 Successful trials = 15/20 (0.001 and less taken as successful values)
Iterations Using Improved APSO	100 iterations to achieve nearest results to three decimal places.
Minimum Using Original Algorithm	Harmony search in [1] with $(x, y, z) = (1.005, 1.0605, 0.000)$
Iterations Using Previous Algorithms	2500 iterations to achieve nearest result.
Optimization Function	De-Jong 2D
Minimum Using Improved APSO	$(x, y, z) = (0.0015, 0.0042, 0.000)$ Average = 0.000315 Standard deviation = 0.00055 Successful trials = 20/20 (0.001 and less taken as successful values)
Iterations Using Improved APSO	10 iterations to achieve nearest results to three decimal places.
Minimum Using Original Algorithm	Simple APSO with $(x, y, z) = (0.0203, 0.0197, 0.0008)$
Iterations Using Previous Algorithms	10 iterations to achieve nearest result.
Optimization Function	Egg-Crate 2D
Minimum Using Improved APSO	$(x, y, z) = (0.0030, 0.0027, 0.0004)$ Average = 3.989945 Standard deviation = 5.0615 Successful trials = 12/20 (0.01 and less taken as successful values)
Iterations Using Improved APSO	10 iterations to achieve nearest results to three decimal places.
Minimum Using Original Algorithm	Simulated annealing in [1]. $(x, y, z) = (0, 0, 0)$
Iterations Using Previous Algorithms	2500 trials of program and unknown number of iterations, accurate to three decimal places.
Average computation time of improved and simple APSO is less than 1 s for each objective function.	

The results obtained from the implementation of the improved APSO and the results taken from [1] of the implementation of the simple APSO algorithm on the deterministic, non-linear, and multi-modal functions show that both of these algorithms are a brilliant choice to obtain robust solutions of the optimization problems. Moreover, the elegance of these APSO variants in terms of a few tuning parameters and a single-step particle update equation make them a preferred choice as optimization algorithms.

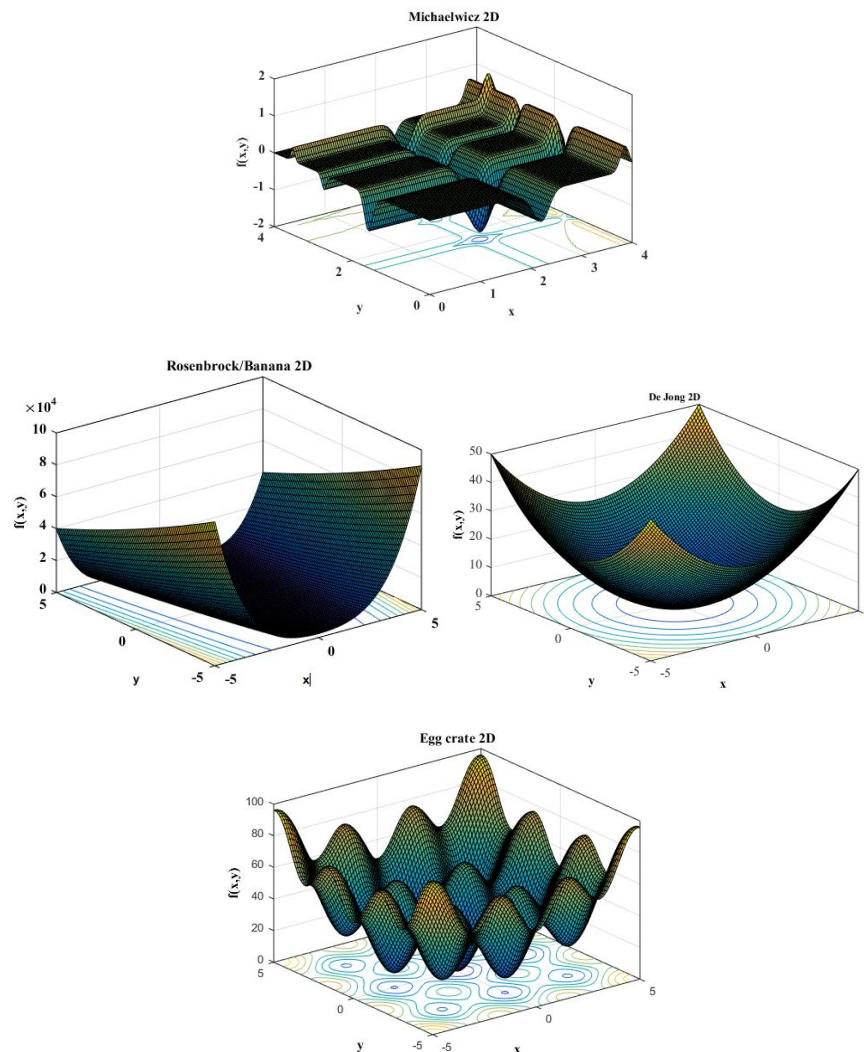


Figure 13. Plots of selected functions.

9. Conclusions

Establishing the superiority of one type of algorithm over another type of algorithm, while implementing them on one type of optimization problem, requires a proper statistical hypothesis testing like the *t*-test. This paper has presented another modification of particle swarm optimization as the improved accelerated particle swarm optimization algorithm, which dynamically squeezes the search space for the particles in every iteration. Two out of many types of STHTS problems were tested using simple APSO and improved APSO. The improved APSO algorithm has shown very promising results as compared to the simple APSO algorithm for the non-cascaded pumped storage STHTS case, whereas both algorithms have shown equivalent results in statistical terms for the non-cascaded non-pumped STHTS problem. However, the improved APSO has successfully found a better approximate to the global minimum for this problem. Further research can be done on the tuning of these variants of PSO to find even more promising results for these problems and for other types of STHTS problems.

Author Contributions: Conceptualization, M.S.F. and H.Z.H.; methodology, M.S.F.; software, M.S.F. and H.Z.H.; validation, M.S.F., S.A.R.K. and N.U.A.; formal analysis, M.S.F., S.A.R.K. and N.U.A.; investigation, A.R.; resources, I.A.S.; data curation, M.S.F.; writing—original draft preparation, M.S.F.; writing—review and editing, S.A.R.K.; visualization, A.R. and I.A.S.; supervision, S.A.R.K.; project administration, S.A.R.K., A.R. and I.A.S.

Funding: This research received no external funding.

Conflicts of Interest: The authors declare no conflict of interest.

Abbreviations

The following symbols and abbreviations are used in this manuscript:

APSO	Accelerated Particle Swarm Optimization
x_i^{n+1}	Particle position at iteration $n + 1$
g^*	Global best particle
$\Delta_{low,i}^n$	Lower increment to be added in the i^{th} particle's lower limit for search space squeezing at iteration n
$\Delta_{low,j}^n$	Higher increment to be added in the i^{th} particle's upper limit for search space squeezing at iteration n
$x_{i,min}^{n+1}$	Lower limit of the i^{th} particle at iteration $n + 1$
$x_{i,max}^{n+1}$	upper limit of the i^{th} particle at iteration $n + 1$
f	Objective function/cost
F_j	Cost of the j^{th} generating unit
V_j	Volume of the reservoir at the j^{th} interval
Dis_j	Discharge rate of the reservoir at the j^{th} interval
$P_{T,j}$	Thermal power at the j^{th} interval
$P_{hyd,j}$	Hydro power at the j^{th} interval
R_j	Constant water inflow into the reservoir at the j^{th} interval
S_j	Spillage of water from the reservoir at the j^{th} interval
$P_{loss,j}$	Power loss at any scheduling j^{th} interval
α, β, ϵ	Tuning parameters of APSO equation. Their value ranges between 0 and 1.

References

1. Yang, X.S. *Engineering Optimization: An Introduction with Metaheuristic Applications*; John Wiley & Sons: Hoboken, NJ, USA, 2010. [\[CrossRef\]](#)
2. Yang, X.S.; Deb, S.; Fong, S. Accelerated particle swarm optimization and support vector machine for business optimization and applications. *Commun. Comput. Inf. Sci.* **2011**, *136*, 53–66. [\[CrossRef\]](#)
3. Hota, P.; Barisal, A.; Chakrabarti, R. An improved PSO technique for short-term optimal hydrothermal scheduling. *Electr. Power Syst. Res.* **2009**, *79*, 1047–1053. [\[CrossRef\]](#)
4. Mandal, K.K.; Basu, M.; Chakraborty, N. Particle swarm optimization technique based short-term hydrothermal scheduling. *Appl. Soft Comput.* **2008**, *8*, 1392–1399. [\[CrossRef\]](#)
5. Yu, B.; Yuan, X.; Wang, J. Short-term hydro-thermal scheduling using particle swarm optimization method. *Energy Convers. Manag.* **2007**, *48*, 1902–1908. [\[CrossRef\]](#)
6. Thang, T.N.; Dieu, N.V.; Anh, V.T. Cuckoo search algorithm for short-term hydrothermal scheduling. *Appl. Energy* **2014**, *132*, 276–287. [\[CrossRef\]](#)
7. Ahmadi, A.; Masouleh, M.S.; Janghorbani, M.; Manjili, N.Y.G.; Sharaf, A.M.; Nezhad, A.E. Short term multi-objective hydrothermal scheduling. *Electr. Power Syst. Res.* **2015**, *121*, 357–367. [\[CrossRef\]](#)
8. Chen, J.J.; Zheng, J.H. Discussion on short-term environmental/economic hydrothermal scheduling. *Electr. Power Syst. Res.* **2015**, *127*, 348–350. [\[CrossRef\]](#)
9. Sun, C.; Lu, S. Short-term combined economic emission hydrothermal scheduling using improved quantum-behaved particle swarm optimization. *Expert Syst. Appl.* **2010**, *37*, 4232–4241. [\[CrossRef\]](#)
10. Roy, P.K. Teaching learning based optimization for short-term hydrothermal scheduling problem considering valve point effect and prohibited discharge constraint. *Int. J. Electr. Power Energy Syst.* **2013**, *53*, 10–19. [\[CrossRef\]](#)
11. Zhang, H.; Zhou, J.; Zhang, Y.; Fang, N.; Zhang, R. Short-term hydrothermal scheduling using multi-objective differential evolution with three chaotic sequences. *Int. J. Electr. Power Energy Syst.* **2013**, *47*, 85–99. [\[CrossRef\]](#)
12. Farhat, I.A.; El-Hawary, M.E. Optimization methods applied for solving the short-term hydrothermal coordination problem. *Electr. Power Syst. Res.* **2009**, *79*, 1308–1320. [\[CrossRef\]](#)
13. Tian, H.; Yuan, X.; Ji, B.; Chen, Z. Multi-objective optimization of short-term hydrothermal scheduling using non-dominated sorting gravitational search algorithm with chaotic mutation. *Energy Convers. Manag.* **2014**, *81*, 504–519. [\[CrossRef\]](#)

14. Zhou, J.; Liao, X.; Ouyang, S.; Zhang, R.; Zhang, Y. Multi-objective artificial bee colony algorithm for short-term scheduling of hydrothermal system. *Int. J. Electr. Power Energy Syst.* **2014**, *55*, 542–553. [[CrossRef](#)]
15. Mandal, K.K.; Chakraborty, N. Differential evolution technique-based short-term economic generation scheduling of hydrothermal systems. *Electr. Power Syst. Res.* **2008**, *78*, 1972–1979. [[CrossRef](#)]
16. Mandal, K.K.; Chakraborty, N. Daily combined economic emission scheduling of hydrothermal systems with cascaded reservoirs using self organizing hierarchical particle swarm optimization technique. *Expert Syst. Appl.* **2012**, *39*, 3438–3445. [[CrossRef](#)]
17. Immanuel S.A. Civilized swarm optimization for multi-objective short-term hydrothermal scheduling. *Int. J. Electr. Power Energy Syst.* **2013**, *51*, 178–189. [[CrossRef](#)]
18. Narang, N. Short-term hydrothermal generation scheduling using improved predator influenced civilized swarm optimization technique. *Appl. Soft Comput.* **2017**, *58*, 207–224. [[CrossRef](#)]
19. Wong, K.P.; Wong, Y.W. Short-term hydrothermal scheduling part. I. Simulated annealing approach. *IEE Proc. Gener. Transm. Distrib.* **1994**, *141*, 497–501. [[CrossRef](#)] [[CrossRef](#)]
20. Padmini, S.; Rajan, C.C.A. Improved PSO for short-term hydrothermal scheduling. *SEISCON* **2011**, 332–334. [[CrossRef](#)]
21. Samudi, C.; Das, G.P.; Ojha, P.C.; Sreeni, T.S.; Cherian, S. Hydro thermal scheduling using particle swarm optimization. In Proceedings of the Conference on Transmission and Distribution Exposition, Chicago, IL, USA, 21–24 April 2008. [[CrossRef](#)]
22. Fakhar, M.S.; Kashif, S.A.R.; Saqib, M.A.; Hassan, T.U. Non cascaded short-term hydro-thermal scheduling using fully-informed particle swarm optimization. *Int. J. Electr. Power Energy Syst.* **2015**, *73*, 983–990. [[CrossRef](#)] [[CrossRef](#)]
23. Khandualo, S.K.; Barisal, A.K.; Hota, P.K. Scheduling of pumped storage hydro-thermal system with evolutionary programming. *J. Clean Energy Technol.* **2013**, *1*. [[CrossRef](#)]
24. Fakhar, M.S.; Kashif, S.A.R.; Saqib, M.A.; Mehmood, F.; Hussain, H.Z. Non-cascaded short-term pumped-storage hydro-thermal scheduling using accelerated particle swarm optimization. In Proceedings of the International Conference on Electrical Engineering, San Francisco, CA, USA, 23–25 October 2018.
25. Hussain, H.Z.; Haider, A.; Fakhar, M.S.; Ahmad J.; Butt, M.A.; Khokhar, K.S. Short-term scheduling of non-cascaded hydro-thermal system with transmission losses using accelerated particle swarm optimization algorithm. *Pak. J. Eng. Appl. Sci.* **2018**, *22*, 20–29.
26. Wood, A.J. *Power Generation Operation and Control*; Wiley: New York, NY, USA, 1996.
27. Wood, A.J.; Wollenberg, B.; Sheblé, G.B. *Power Generation Operation and Control*, 3rd ed.; Wiley, Hoboken, NJ, USA, 2013.
28. Kalivarapu, V.; Winer, E. A statistical analysis of particle swarm optimization with and without digital pheromones. In Proceedings of the Conference on Structures, Structural Dynamics, and Materials, Honolulu, HI, USA, 23–26 April 2007. [[CrossRef](#)]
29. Chopard, B.; Tomassini, M. *Particle Swarm Optimization*; Springer in Natural Computing Series; Springer: Cham, Switzerland, 2018. [[CrossRef](#)].
30. Kennedy, J. Particle Swarm Optimization. *Encyclopedia of Machine Learning*; Springer, New York, NY, USA, 2010.
31. Wolpert, D.H.; Macready, W.G. No free lunch theorems for optimization. *IEEE Trans. Evol. Comput.* **1997**, *1*, 67–82. [[CrossRef](#)] [[CrossRef](#)]
32. Ho, Y.C.; Pepyne, D.L. Simple explanation of the no-free-lunch theorem and its implications. *J. Opt. Theory Appl.* **2002**, *115*, 549–570, [[CrossRef](#)] [[CrossRef](#)]
33. Yu-Chi, H.; Pepyne, D.L. Simple Explanation of the No Free Lunch Theorem of Optimization. In Proceedings of the 40th IEEE Conference on Decision and Control, Orlando, FL, USA, 4–7 December 2001. [[CrossRef](#)]
34. Kifer, D.; Machanavajjhala, A. No Free Lunch in Data Privacy. In Proceedings of ACM SIGMOD International Conference on Management of Data, Athens, Greece, 12–16 June 2011. [[CrossRef](#)]
35. Field, A. *Doing Statistics Using SPSS*, 3rd ed.; SAGE Publications Ltd.: London, UK, 2009. [[CrossRef](#)]

