List of Responses

Responds to the Anonymous Referee #1's comments:

Special thanks for your good comments which are very useful for us to improve the paper.

1. Response to comment: Please state the advantages of the both PSO and WSA algorithms, and their performance difference in detail, so that readers can know the motivation that you combine them to coevolve to solve the CNOP. Please use statistical method to demonstrate the better optimization performance of ACPW comparing with the PSO and the WSA in perspective of optimization time and accuracy.

Response: It is really true as Rreview1 suggested that we need to clarify the advantages of the both PSO and WSA algorithms and analyze the the better optimization performance of ACPW. Therefore we have illustrated this in the Section 4.1.

"To evaluate the advantages of the ACPW algorithm, we run the PSO, WSA and ACPW programs 10 times and then compare the maximum, minimum and mean objective values as well as the RMSE.

4.1 The advantages of the ACPW algorithm

Because the statistical analysis results are similar for the two TCs with the two resolutions, we only describe the analysis of Fitow at a resolution of 60 km. Table 3 presents the maximum objective value, the minimum objective value, the mean objective value and the RMSE of the 10 results.

Table 3: The analysis results of the PSO, WSA and ACPW methods.

Algorithm	Maximum Value	Minimum Value	Mean Value	RMSE
PSO	1034.192573	724.086002	900.7488578	**0.121400896**
WSA	1628.841294	323.7493169	930.9103862	0.431193448
ACPW	**2240.275956**	1243.377921	**1542.505251**	0.216750584

In Table 3, the maximum objective value is gained from the ACPW algorithm, and its mean value is also more than the other two algorithms. However, the RMSE of PSO is the smallest, which shows the best stability.

For additional analysis, we draw a box-plot of the 10 results for the PSO, WSA and ACPW algorithms, as shown in Fig. 3.

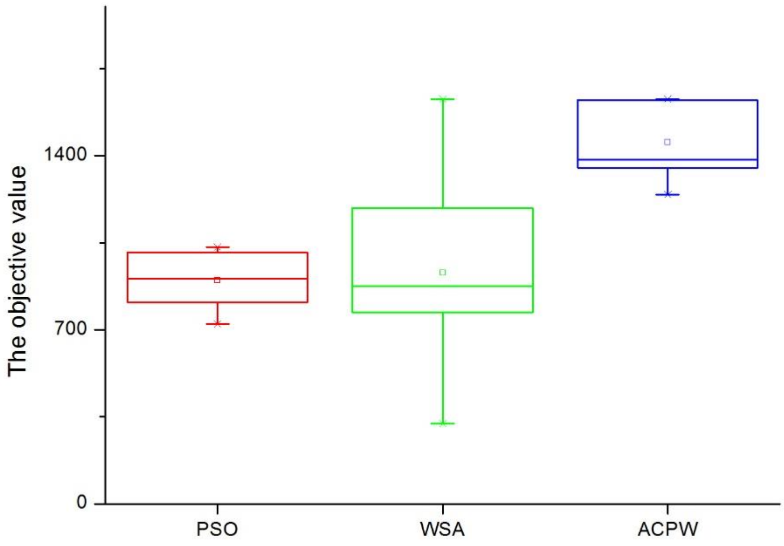

Figure 3: Box-plot of the PSO, WSA and ACPW methods for TC Fitow at 60 km resolution. The red box denotes PSO, the green box is for the WSA, and the blue box shows the results of the ACPW algorithm.

PSO has the narrowest range of values, although the objective values are smaller than the other two algorithms. The WSA has the widest range of values, although the objective values are also smaller than the ACPW algorithm. The ACPW algorithm has the second-best stability, although it has the best objective values. The experiments display the stability of PSO and the exploitation of the WSA. We combine the advantages of them and develop the ACPW algorithm to solve CNOPs. The analysis results demonstrate that the hybrid strategy and cooperation co-evolution is useful and effective."

2. Response to comment: There is a great difference at the operation rules of the WSA between the standard version given by Rui Tang et al. (2012) and the formula (6) of this study, please make explanation or correction.

Response: We are very sorry about errors in this paper and have corrected them in Page 5, line 2-9. "

$$\begin{cases} u_i^{k+1} = u_i^k + \theta \cdot r \cdot rand(\ ) & Prey \\ u_i^{k+1} = u_i^k + \theta \cdot s \cdot escape(\ ) & Escape \end{cases} \tag{6}$$

where the superscript k or $k+1$ is also the iterative step, θ is the velocity, r is the local optimizing radius, which is smaller than the global constraint radius δ, *rand( )* is the random function, whose mean value is distributed in [-1,1], *escape( )* is the function for calculating a random position, which is 3 times larger than r, and s is the step size of the updating individual.

As described in Eq. (6), the wolf has two behaviours, i.e., prey and escape. The prey behaviour uses the first sub-formula, and the second one is for the escape function, which happens in every iteration when the condition $p > p_a$ is satisfied, where p is a random number in [0,1], and p_a is the probability of individual escaping from the current position. ”

3. Response to comment: (1) Page 3, line 24, 26: The variants given in the propagation operator M should be uniform.

Response: As Rreview1 suggested that we rewritten this part in Page 3, line 25.

“$U_t = M_{\mathrm{t_0 \to t}}(U_0)$”

4. Response to comment: (2) Page 5, line 8-9: Please state in detail the rule setting adaptive subswarm coefficient a.

Response: As Rreview1 suggested that we have added the rule setting adaptive subswarm coefficient a in Page 5, line13-16.

“$\alpha = \begin{cases} \alpha + 0.05 \ , \ if \quad the \ \ bestvalue - \ \ current \ \ value < \quad \varepsilon \\ \qquad \alpha - 0.05, \qquad\qquad\qquad\qquad else \end{cases}$

In this paper, before we update the individuals, α is calculated, and then we divide the entire initial swarm into two subswarms according to the α value, i.e., the number of individuals depending on the PSO’s rule is $\alpha \times N$, and the other number is $(1 - \alpha) \times N$. We set the initial value of ε and α to 0.1 and 0.5, respectively. ”

5. Response to comment: (3) Page 5, line 17-19: It is better to delete these three lines since the description is unnecessary.

Response: We need to explain about this part. The reason for writing this part is to present the performance of our algorithms in this paper under those computer hardware environments. If the reader needs to compare with our results, they should have the same environments. Hence, we did not delete them.

In addition, we have improved the quality of our manuscript by American Journal Experts editing service and tracked the changes using revisions in the manuscript ‘Revised Manuscript with Track Changes’.