Journal of Theoretical and Applied Information Technology
15th April 2021. Vol.99. No 7

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

WHALE OPTIMIZATION ALGORITHM APPLIED TO THE 0/1 KNAPSACK PROBLEM

RAJA MASADEH

The World Islamic Sciences and Education University, Department of Computer Science, Amman, Jordan

Email: raja.masadeh@wise.edu.jo

ABSTRACT

This study investigated the 0/1 knapsack problem. A new technique is suggested to find the best solution that maximize the overall carried values without exceeding the capacity of the knapsack using whale optimization algorithm and clustering algorithm. The whale optimization algorithm is considered as one of the most recent metaheuristic optimization algorithms that mimics the hunting behavior of humpback whales. Moreover, it has an effective performance in solving engineering optimization problems and continuous problems. The evaluated performance is satisfying. The simulation results showed great compatibility between experimental and theoretical results.

Keywords: 0/1 *Knapsack Problem, Metaheuristic, Optimization, Whale Optimization Algorithm.*

1. INTRODUCTION

The 0-1 Knapsack problem (0/1KP) is considered a part of NP-hard combinatorial optimization problem. Moreover, it plays a significant role in resource allocation, computer science, cutting stock problems, production planning problem, scheduling problems and project selection (1-4). This study focused on standard 0/1 knapsack. Given N items, where each item has its own weight (Wi) and profit (Pi). The main target of this problem is to maximize the profit of the chosen objects for the knapsack. However, the total weights of these objects should less than or equal to the capacity (C) of the knapsack. This problem is mathematically described as follows:

$$Maximize\ f(x) = \sum_{i=1}^{N} P_i X_i \qquad (1)$$

$$Subject\ to\ \sum_{i=1}^{N} X_i W_i \leq C \qquad (2)$$

X_i indicates whether item i is selected and included the knapsack or not. More precisely, X_i takes a value 1 or 0 that represents the rejection or selection of the item.

Many studies have suggested many techniques to solve 0/1 KP. These techniques are divided into two types, namely, approximate algorithms and exact algorithms. The exact algorithms can give optimal and precise solutions such as branch-and bound algorithms and dynamic programming. Dragonfly Algorithm, Cuckoo Search, firefly algorithm, flower pollination algorithm and Particle Swarm Optimization algorithms are considered as approximate algorithms which give near optimal solutions. However, the exact techniques give optimal and accurate solutions, but in case the size of the problem is increased, the time complexity will be increased exponentially. Thus, many studies towards the approximate algorithms are raised for gaining approximate solutions in a reasonable time [5-9].

The motivation of this study is to present an approach for solving the 0/1 KP as the existing techniques that suffered from lack of variety and local optima. The whale optimization algorithm (WOA) is applied to solve the 0/1 knapsack problem. This study tested to find the applicability of WOA on solving 0/1 knapsack problem. The suggested technique provided global solutions rather than local solutions.

The rest of this paper is organized as follows. In Section 2, the literature review and related work of 0/1 Knapsack problem and metaheuristic optimization algorithms are introduced. In Section 3, the description of the basic WOA is provided. In Section 4, the proposed algorithm (0/1 Knapsack-WOA) is presented. The simulation and comparison outcomes are given in Section 5. Finally, we conclude this research in Section 6.

2. RELATED WORK

A few studies are introduced in order to solve 0/1 knapsack problem. Some techniques search to find optimal solutions by taking into consideration reasonable solution times. While other algorithms search to find near optimal solutions.

The study [10] introduced a novel technique to detect the best solution which maximizes the overall loaded values without exceeding a limited capacity applying Grey Wolf Optimization (GWO) algorithm and K-means clustering mechanisms. K-means clustering technique is employed in order to classify each 5-12 agents together into one group according to the constraints of GWO algorithm. The simulation outcomes proved satisfied compatibility between experimental and theoretical outcomes.

The researchers [11] proposed a novel technique which based on chemical reaction optimization and greedy strategy algorithm in order to solve 0/1 knapsack problem. Moreover, they presented explanation of the operator design and parameter turning techniques for the proposed algorithm. In addition, they used a new function that merge a random selection and greedy strategy in order to repair the infeasible solutions. The results proved the superior performance of the proposed algorithm compared to traditional genetic algorithm, quantum-inspired evolutionary algorithm and ant colony optimization technique.

The study [12] introduced a modified whale optimization algorithm to solve 0–1 knapsack problem with different scales. The researchers added a penalty function to the estimation function. Thus, the feasible solutions' fitness can outperform the fitness of the infeasible ones. The proposed algorithm gave a better trade-off between diversification and condensation through employing two mechanisms: Local Search Strategy and Lévy Flight Walks. The experimental outcomes proved that the suggested algorithm is effective and robust for solving 0/1 knapsack problem.

The research [13] suggested a new global harmony search algorithm to solve 0/1 knapsack. Moreover, the suggested technique contains two significant processes; location updating and genetic mutation with a small eventuality. The first operation enables the worst harmony of harmony memory to proceed to the best global harmony quickly in each iteration, and the second operation can effectively prevent the proposed algorithm from colliding with the local optimum. The computational experiments proved that the proposed algorithm can be an effective technique for solving knapsack problem.

The researchers [14] suggested a novel algorithm based on chemical reaction optimization algorithm and a greedy strategy to solve 0-1 knapsack problem. The proposed technique inspiring the chemical reaction operation is employed to implement both local and global search. A repair operator is used to merge a greedy strategy and random selection in order to reform the infeasible solutions. The experimental outcomes showed that the novel algorithm had better performance than other well-known algorithms.

3. WHALE OPTIMIZATION ALGORITHM

Whale Optimization Algorithm (WOA) is a recently proposed randomly optimization algorithm by Mirjalili and Lewis in 2016 [15]. This algorithm purposes to determine the global optimum for the problem by using a population of search agents (whales). The search process is the first step. It begins with generating a collection of candidate solutions that are selected randomly for the given problem. Then, it ameliorates this collection during many numbers of iterations until the satisfaction of an end condition. In fact, whales' imitative private hunting technique was called bubble-net feeding method as reported in [15]. The following steps are the main steps of WOA [15]:

- *Initialization phase*: In this stage, the population of search agents W_i (i= 1, 2, 3..., k) is randomly created.
- *Fitness calculation phase*: is used to calculate the fitness function. The best search agent W* is chosen based on evaluation.
- *Encircling prey phase*: In this phase, the position of a prey is assumed to be fixed. Thus, the prey is surrounded by humpback whales assuming that the current solution is the best solution (prey). Other whales (search agents) update their locations based on the current best agent, which is represented as in Equation (3).

$$\vec{P} = \left|\vec{C}\,.\overrightarrow{W^*}\,(t) - \vec{W}\,(t)\right| \tag{3}$$

Where the current iteration is denoted as i, $\overrightarrow{W^*}\,(t)$ indicates the position vector of the best solution, and $\vec{W}\,(t)$ represents the position vector of a search agent. Thus, the other search agents update their positions based on the best search agent using Equations (4), (5), and (6). The value of $\vec{r}$ decreases from 2 to 0.

$$\vec{W}(t+1) = \overrightarrow{W^*}\,(t) - \vec{R}\,.\vec{P} \tag{4}$$

Where $\vec{C}$ and $\vec{R}$ indicate the coefficient vectors.

$$\vec{R} = 2\vec{r}\,.\vec{n} - \vec{r} \tag{5}$$

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

$$\vec{C} = 2.\vec{n} \quad (6)$$

- *Exploitation phase*: In this stage, two basic techniques will be chosen. The first technique is called shrinking encircling. In this technique, the algorithm calculates the new position of a search agent. The value of a search agent $\vec{r}$ should be a value between [-1, 1]. The new value should be calculated depending on the initial location of the search agent and the current best agent. The second technique is spiral updating position. It updates the position of agents in the spiral method using Equation (7).

$$\vec{W}(t+1) = \left|\overrightarrow{W^*}(t) - \overrightarrow{W(t)}\right|.e^{bl}.\cos(2\pi l) + \overrightarrow{W^*}(t) \quad (7)$$

Where l is a value in [-1, 1], and b is a constant for determining the shape of the logarithmic spiral.

- *Exploration phase*: The position of the search agent will be updated according to a randomly selected search agent using Equations (8) and (9).

$$\vec{P} = \left|\vec{C}.\overrightarrow{W_{rand}} - \vec{W}\right| \quad (8)$$

Where $\vec{W}$ indicates the position vectors of other search agents except $\overrightarrow{W_{rand}}$ which represents the random position vector.

$$\vec{W}(t+1) = \overrightarrow{W_{rand}} - \vec{R}.\vec{P} \quad (9)$$

- *Termination phase*: In case a search agent exits in the search region, the value of the best search agent will be updated, and the next iteration begins. This process will stop when the best solution is found.

A humpback whale generates ambush with shifting in a spiral route around the victims, and then generates bubbles all the way ahead. Thus, this search process is the major inspiration of the whale optimization algorithm. In addition to that, the encircling method is another simulated technique in WOA. The humpback whales surround around the victims in order to begin hunting, then uses the foraging mechanism which is called the bubble- net technique. Assume that there is a probability (p) of 50% to choose between either the shrinking encircling mechanism or the spiral model to update the position of whales during optimization. This behavior is represented mathematical by Equation (2) from [15].

$$\vec{X}(t+1) = \begin{cases} \vec{X}*(t) - \vec{A}.\vec{D} & if\ p < 0.5 \\ D'.e^{bl}.Cos\,(2\,\pi\,l) + \vec{X}*(t) & if\ p \geq 0.5 \end{cases} \quad (10)$$

Where p is a random number in [0, 1], b is a constant for determining the shape of the logarithmic spiral, and l indicates to a random number in [−1, 1], t represents the current iteration, D' = |X∗ (t) − X (t)| which mentions the distance between the ith whale and the victim, X^* indicates the position vector of the best solution, X is the position vector and A indicates to coefficient vector.

In other words, the first phase is presented in this equation is the foraging mechanism that mimics the encircling technique, whereas the second phase simulates the bubble-net mechanism. The variable p exchanges between these two phases with similar probability. The potential cases using these equations are shown in Figure 1 [15].

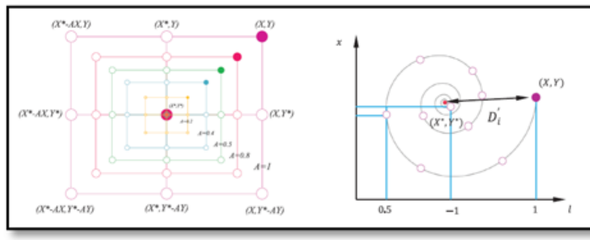


Figure 1: Mathematical Models for Prey Encircling and Bubble-Net Hunting, Where (X, Y) is The Location of The Whale and (X,Y*) is The Location of The Prey.*

The main two phases of the optimization algorithm by using population-based algorithms are exploration and exploitation phases. Whereas both ensured in WOA by adaptively setting a and c in the major equation.

In case, a problem is given, the WOA begins optimizing this problem by generating collection of random solutions. In each iteration, the search agents update their location depends on the randomly chosen search agent or the best search agents that will gain so far. To assure the exploration phase, the other agents update their positions based on the best solution that represents the pivot point when |A|>1. In other case, when |A|<1 the best solution plays another role with the pivot point. The flowchart of the WOA is shown in Figure 2 [15].

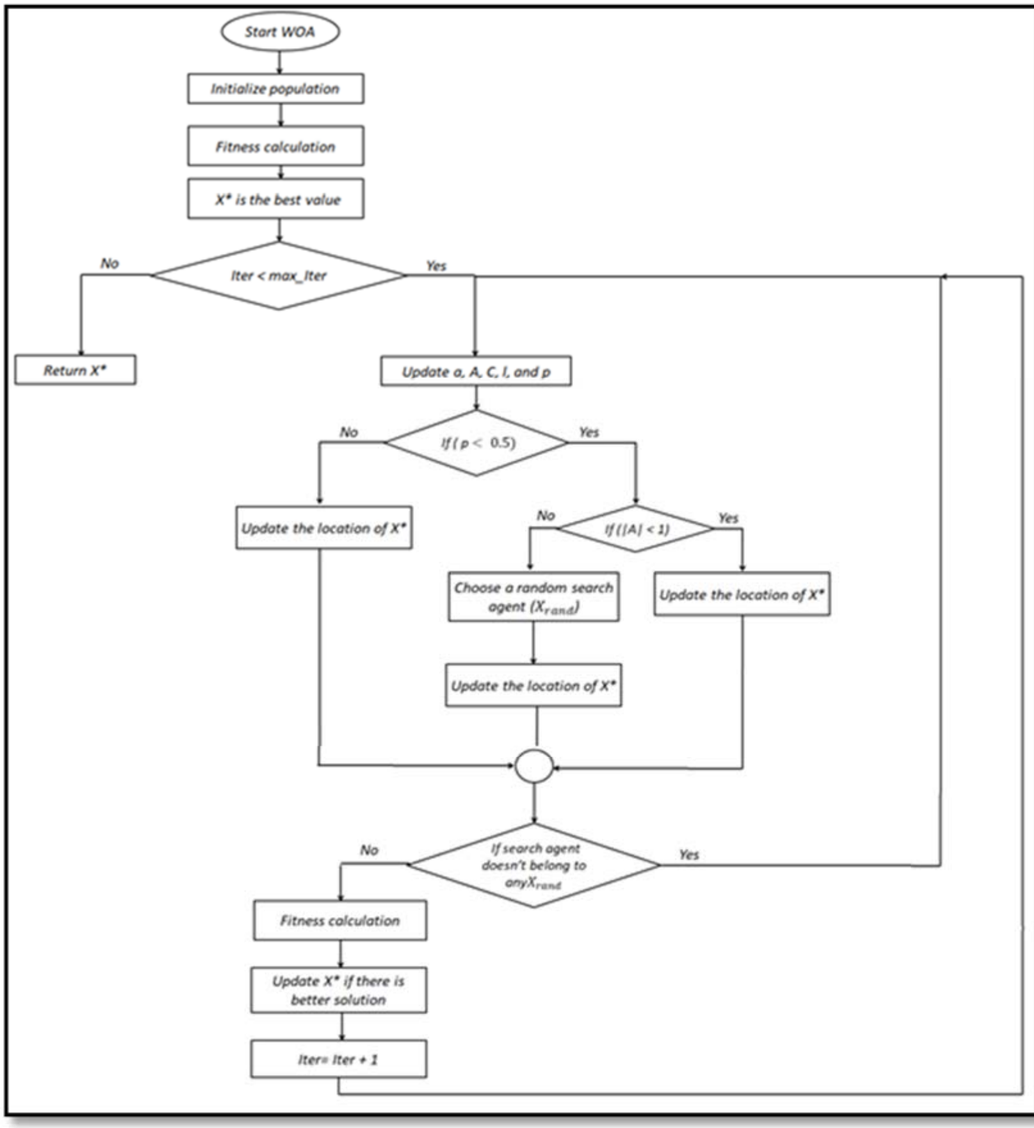


Figure 2: Flowchart of WOA

4. SUGGESTED ALGORITHM

The study applied WOA [15] and clustering technique [16] on 0/1 knapsack problem searching for optimal or near optimal solutions. Figure 3 illustrated the pseudocode of the proposed algorithm.

A pack of humpback whales indicates to several possible solutions to optimize specific problem such as knapsack problem, in which each possible solution is called a search humpback whale. As shown in Figure 4 (initialization function), initialize the population of search agents (Whales), random vectors (a, A, C and P), overall allowed weight, and both weights and values for each whale (search agent). One of these whales is chosen randomly to represent the prey while other whales indicate the sources.

The clustering problem is solved with WOA by [16]. The solution [16] proposed that search agents indicate cluster center R which has been predetermined. Each search agent Xi is written as follows:

$$X_i = (Z_{i1}, Z_{i2}, \dots, Z_{iR}) \quad (4)$$

Where Z_{ij} indicates the central vector of cluster (j) and search agent (i) in cluster R_{ij}. Thus, a pack of humpback whales indicates several candidates group for vectors from datasets. Computing of distance among clusters is considered a fitness function employed to scale the distance between the cluster center (R) and vector data from the same cluster according to the following Equations (5) and 6. Therefore, each object is designated to its nearest cluster.

$$d(s,z) = \left(\sum_{i=1}^{n} \sum_{R=1}^{R} Wh_{iR} \|X_i - Z_R\|^2\right)^{1/2} \quad (5)$$

$$Wh_{iR} = \{1 \; if \; \|X_i - Z_R\|^2 = min_{1\le j \le n} \|X_i - Z_R\|^2\}, \{0, Otherwise\}$$
(6)

Pseudocode of 0/1 KP-WOA
1. Call initializing function
2. While (t< Maximum number of iterations)
3. Call clustering function
4. For each cluster {
5. For each search agent {
6. Update a, A, C and P
7. If $p < 0.5$ then
8. If $A < 1$ then
9. Update search whales by Eq. (4)
10. Else if $A \ge 1$ then
11. Select search whale randomly
12. Update current search whale by Eq. (9)
13. End if
14. Else if $P \ge 0.5$ Then
15. Update current search whale by Eq. (7)
16. End if
17. End for
18. t = t + 1
19. End While
20. Return Wh* // return an optimal solution

Figure 3: Pseudocode of 0/1 knapsack-WOA

Initialization function
1. Initialize the population Wh_i (i = 1, 2, 3, … n) and give value R of the cluster center at random.
2. Initialize a, A, C and P
3. Initialize overall allowed weight (W)
4. For each search agent {
5. Initialize weight (W) and value (V) randomly.
6. End for

Figure 4: Initializing Function

Clustering function
1. Give value R of the cluster center at random.
2. For each search agent Wh {
3. For each data vector XP
4. Calculate Euclidean distance from XP to all cluster centers.
5. Specify XP to cluster R_{ij} by Eq. (6)
6. Calculate fitness value by using Eq. (5)
7. End for
8. End For
9. Set Global_Optimal= Max (All local optimal)
10. Assign search agent (Wh_i) to its nearest cluster

Figure 5: Clustering Function

When all clusters are created, each cluster has cluster head (leader) and many members that have their fitness values. The leader creates a cluster matrix using the hunting behavior of humpback whales, which has all fitness values of all cluster members. While the member that has the highest value, represents the nearest search agent to the prey in the cluster. Thus, the leader selects the maximum of all local optimal values to be considered the global solution of its cluster as illustrated in Figure 5.

5. SIMULATION RESULTS

The performance of 0/1KP-WOA algorithm was evaluated by a simulation program with GUI to expedite the usage of the tool. Moreover, ORACLE software was employed in order to simulate the suggested algorithm the major load on the DBMS and it is ORACLE 11g release 2 and it utilized to stock data and performing technique's work by procedures and functions, ORACLE Developer 6i was applied to build the GUI and link it with the database schema.

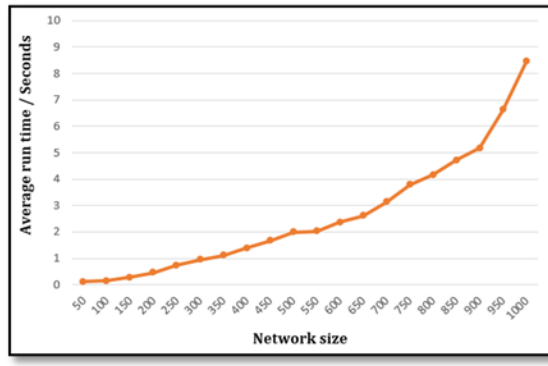
The datasets that used in this study were generated automatically by the simulator with respect to parameters and ranges of users. Various network' size was taken to evaluate 0/1 KP-WOA for solving 0/1knapsack problem with rising the network's size. Each scenario was repeated 10 times in this study, to generate more precise overcomes. The platform specifications that utilized to get these experiments are: Intel(R) Core (TM) i7-8550U CPU 2.00 GHz, 16 GB RAM and Windows 10. Table 1 displays the average run-time for different datasets that is calculated in seconds.

As shown in Figure 8, it is obvious that the time complexity of the proposed algorithm is quadratic polynomial that means it increases with rising the number of nodes in the graph. Furthermore, it is fairly a good performance. Depending on Figure 8 and Figure 9, it is clear that the simulation and theoretical results are very approximate.

Table 1: Average run time of "0/1 KP-WOA for various datasets

Dataset	Average run time/ Seconds	Dataset	Average run time/ Seconds
50	0.11	550	2.02
100	0.15	600	2.36
150	0.27	650	2.61
200	0.45	700	3.13
250	0.73	750	3.78
300	0.94	800	4.16
350	1.11	850	4.72
400	1.39	900	5.18
450	1.67	950	6.63
500	1.99	1000	8.46

Figure 8: Average running time for simulation results for "0/1 KP-WOA".



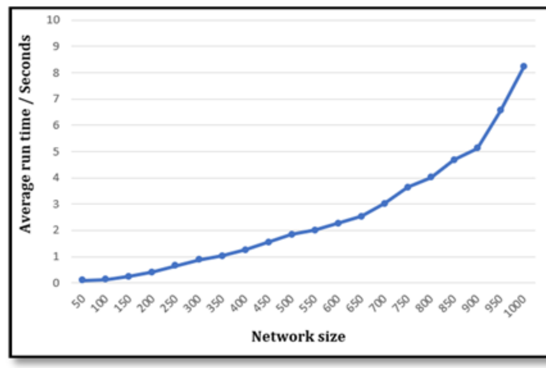


Figure 9: Theoretical runtime for "0/1 KP-WOA"

6. CONCLUSION

This study suggested a novel solution (0/1KP-WOA) to 01Knapsack problem using Whale Optimization Algorithm (WOA) to find the optimal solution. The time complexity of 0/1KP-WOA technique is proved theoretically to be O (n^2). Experimentally, time complexity is acquired to be quadratic polynomial. Therefore, there's a great convergent between theoretical and experimental results.

REFERENCES:

[1] Wang XZ, He YC (2017) A survey on evolutionary algorithms of solving knapsack problems. J Softw (Chinese) 281(1):1–16.

[2] De Vries S, Vohra RV (2003) Combinatorial auctions: a survey. INFORMS J Comput 15(3):284–309.

[3] Kellerer H, Pferschy U, Pisinger D (2004) Introduction to NP completeness of knapsack problems. Knapsack problems. Springer, Berlin Heidelberg, pp 483–493.

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

[4] Abdel-Basset M et al (2017) Solving 0–1 knapsack problems by binary dragonfly algorithm. In: International Conference on Intelligent Computing. Springer, Cham, pp 491–502.

[5] Srikanth K, Panwar LK, Panigrahi BK, Herrera-Viedma E, Sangaiah AK, Wang GG (2017) Meta-heuristic framework: quantum inspired binary grey wolf optimizer for unit commitment problem. Comput Electr Eng

[6] He YC, Wang XZ, He YL, Zhao SL, Li WB (2016) Exact and approximate algorithms for discounted {0–1} knapsack problem. Inf Sci 369: 634–647.

[7] Masadeh, R., Alzaqebah, A., Smadi, B., & Masadeh, E. (2020). Parallel Whale Optimization Algorithm for Maximum Flow Problem. Modern Applied Science, 14(3), 30-44.

[8] Masadeh, R., Hudaib, A., & Alzaqebah, A. (2018). WGW: A hybrid approach based on whale and grey wolf optimization algorithms for requirements prioritization. Advances in Systems Science and Applications, 18(2), 63-83.

[9] Masadeh, R., Mahafzah, B. A., & Sharieh, A. 20199). Sea lion optimization algorithm. International Journal of Advanced Computer Science and Applications, 10(5), 388-395.

[10] Yassien, E., Masadeh, R., Alzaqebah, A., & Shaheen, A. (2017). Grey wolf optimization applied to the 0/1 knapsack problem. International Journal of Computer Applications, 169(5), 11-15.

[11] Truong, T. K., Li, K., & Xu, Y. (2013). Chemical reaction optimization with greedy strategy for the 0–1 knapsack problem. Applied soft computing, 13(4), 1774-1780.

[12] Abdel-Basset, M., El-Shahat, D., & Sangaiah, A. K. (2019). A modified nature inspired meta-heuristic whale optimization algorithm for solving 0–1 knapsack problem. International Journal of Machine Learning and Cybernetics, 10(3), 495-514.

[13] Zou, D., Gao, L., Li, S., & Wu, J. (2011). Solving 0–1 knapsack problem by a novel global harmony search algorithm. Applied Soft Computing, 11(2), 1556-1564.

[14] Truong, T. K., Li, K., Xu, Y., Ouyang, A., & Nguyen, T. T. (2015). Solving 0-1 knapsack problem by artificial chemical reaction optimization algorithm with a greedy strategy. Journal of Intelligent & Fuzzy Systems, 28(5), 2179-2186.

[15] Mirjalili, S., & Lewis, A. (2016). The whale optimization algorithm. Advances in engineering software, 95, 51-67.

[16] Liauw, D., Khairuzzaman, M. Q., & Syarifudin, G. (2019, November). Whale Optimization Algorithm for Data Clustering. In 2019 7th International Conference on Cyber and IT Service Management (CITSM) (Vol. 7, pp. 1-6). IEEE.