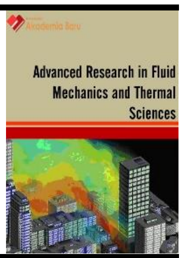
Penerbit Akademia Baru

Journal of Advanced Research in Fluid Mechanics and Thermal Sciences

Journal homepage: www.akademiabaru.com/arfmts.html
ISSN: 2289-7879



Methodology for Modified Whale Optimization Algorithm for Solving Appliances Scheduling Problem

Open Access

Mohd Faizal Omar[1,2,*], Noorhadila Mohd Bakeri[3], Mohd Nasrun Mohd Nawi[3], Norfazlirda Hairani[4], Khalizul Khalid[5]

[1] School of Quantitative Sciences, College of Arts and Sciences, Universiti Utara Malaysia, 06010 Sintok, Kedah, Malaysia
[2] Centre for Testing, Measurement and Appraisal (CeTMA), Universiti Utara Malaysia, 06010 Sintok, Kedah, Malaysia
[3] School of Technology Management and Logistic, College of Business, Universiti Utara Malaysia, 06010 Sintok, Kedah, Malaysia
[4] Faculty of Entrepreneurship and Business, Universiti Malaysia Kelantan, 16100 Pengkalan Chepa, Kelantan, Malaysia
[5] Faculty of Management and Economics, Sultan Idris Education University, 35900 Tanjung Malim, Perak, Malaysia

ARTICLE INFO

Article history:
Received 13 May 2020
Received in revised form 10 September 2020
Accepted 13 September 2020
Available online 19 October 2020

ABSTRACT

Whale Optimization Algorithm (WOA) is considered as one of the newest metaheuristic algorithms to be used for solving a type of NP-hard problems. WOA is known of having slow convergence and at the same time, the computation of the algorithm will also be increased exponentially with multiple objectives and huge request from n users. The current constraints surely limit for solving and optimizing the quality of Demand Side Management (DSM) case, such as the energy consumption of indoor comfort index parameters which consist of thermal comfort, air quality, humidity and vision comfort. To address these issues, this proposed work will firstly justify and validate the constraints related to the appliances scheduling problem, and later proposes a new model of the Cluster based Multi-Objective WOA with multiple restart strategy. In order to achieve the objectives, different initialization strategy and cluster-based approaches will be used for tuning the main parameter of WOA under different MapReduce application which helps to control exploration and exploitation, and the proposed model will be tested on a set of well-known test functions and finally, will be applied on a real case project i.e. appliances scheduling problem. It is anticipating that the approach can expedite the convergence of meta-heuristic technique with quality solution.

Keywords:
Scheduling Problem; Swarm Intelligence; Whale Optimization Algorithm

1. Introduction

The optimization of energy consumption schedules of appliances can be regarded as a classical scheduling problem. However, the user comfort level is often overlooked as an important constraint [1-2]. On the other hand, the goal of DSM is to successfully monitor the energy schedule to ensure the minimum electricity charge and the maximum Indoor Comfort Index (ICI) is achieved. ICI can be

* *Corresponding author.*
E-mail address: faizal_omar@uum.edu.my

https://doi.org/10.37934/arfmts.76.2.132143

distributed into four basic parameters, namely thermal comfort, air quality comfort, humidity comfort, and visual comfort. In Shaikh *et al.,* [3], the user comfort index is regarded as a significant issue due to the high portion of people's time is allocated by staying in the buildings. As a result, high energy consumption is required to obtain a convenient indoor condition.

Previous study has suggested the various approaches to overcome the appliances scheduling problems. Rahim [4] has proposed three optimization techniques i.e. Genetic Algorithm (GA), Binary Particle Swarm Optimization (BPSO) and Ant Colony Optimization (ACO). These methods were utilized to reduce the electricity charge, PAR, execution period and increase consumer comfort, while the integration of RES was solved.

Authors in Muralitharan *et al.,* [5] and Zafar *et al.,* [6] have implemented a Multi-Objective Evolutionary Algorithm (MOEA) and Harmony Search Algorithm (HSA), respectively. Both studies were trying to improve energy consumption and consumer comfort in the scope of a waiting period for appliances execution. However, they are only focused on consumer satisfaction in the context of the waiting period for the energy scheduling, but the indoor comfort index parameters are often ignored [4-6].

Over the last three years ago, meta-heuristic discipline has experienced a rapid growth of Swarm Intelligence (SI) techniques such as Multi-Verse Optimizer (MVO) [7], Grey Wolf Optimizer (GWO) [8], Sine Cosine Algorithm (SCA) [9], and Whale Optimization Algorithm (WOA) [10]. WOA is powerful and easy to conduct because only a small number of variables need to be controlled. WOA also shows excellent potential to bypass the local optima [11]. Nevertheless, most of the SI technique suffers from low convergence accuracy and convergence rate when solving complex optimization problems [12-13]. Hence, WOA also inherits the disadvantages of SI with a slow convergence rate [14].

The complexity of the problem is also simultaneously increased with the number of objectives and users demand in real-time. Thus, it is desirable to distribute the optimization tasks /agents to several clusters in order to minimize the time for convergence. Hence, to the best knowledge of the authors, in the corpus of literature, there is no attempt to minimize the problem of slow convergence of Multi Objective WOA particularly for scheduling problem.

2. Meta-Heuristic Algorithm for the Scheduling Problem

Authors in Baker [15], Morton *et al.,* [16], and Pinedo [17] defined that "scheduling is a process associated with the allocation of sources to perform the task within a period of time". It has been widely utilized on the various field in real-world problem including manufacturing and services [18], satellite broadcast scheduling problem [19], academic scheduling problems [20], energy management [21], and engineering field [22]. Scheduling also involves many components including the distribution of available resources for tasks or activities over time. Meanwhile, this problem can be modeled as a big group of combinatorial optimization problems.

However, in many scenarios, the optimization problem provides the most problematic and it becomes NP-hard problem [18]. NP-hard stands for a non-deterministic polynomial-time hardness where the problem has no known polynomial algorithm, then, the finding solution time is increased simultaneously with problem size. In order to obtain optimum results, there is a need to find the most appropriate method in solving these problems. Therefore, the design of the efficient meta-heuristic algorithm is seen as a great choice.

2.1 The Progress of Swarm Intelligence for Scheduling Problem

Swarm Intelligence (SI) is a nature-inspired meta-heuristic algorithm motivated by biological or physical phenomena. SI plays an important role in solving any type of optimization issue, while Swarm-based technique shows the most popular approach in SI [10]. Originally, the SI term was implemented in the scope of cellular robotic systems only [23]. But today, SI is widely utilized as a division of Computational Intelligence (CI) [24]. There are five principles of SI paradigm as defined by [25] which are proximity, quality, diverse reaction, stability, and adaptability.

The stable expansion of SI studies prove that it has excellent potential in the research area of CI [26]. According to the definition above, the taxonomy of SI can be divided into several groups which consists of insect, bird, fish, group hunting, bacteria, mammal, frog, and others [26]. Particle Swarm Optimization (PSO) [27] and Ant Colony Optimization (ACO) [28] were some of the well-known algorithms in SI.

PSO which developed in the year 1995 [27] is ideally coming from the social nature of bird flocking or fish schooling. PSO also shows much likeness to the evolutionary computation method such as the Genetic Algorithm. The process of PSO was initialized with the random population to generate the system. Then, the system will use the first solution to continually update the generation until the optimal solution is found.

In the past several decades, PSO shows outstanding performance in various fields and practices. This is due to the advantages of PSO such as high convergence speed and low cost compared to other optimizers. Here are some of an example of PSO practices, Ravagnani *et al.,* [29] utilized PSO for solving the shells problem. Han *et al.,* [30] and Yu *et al.,* [31] implemented PSO to solve the problem of rolling fin-tube and plate fin heat exchanger, respectively.

Meanwhile, ACO is developed in the year 1992 [28]. ACO initially introduced as the metahueristic algorithm that usually use to solve the combinatorial optimization problems. Started with the basic algorithm of ACO [32-33], ACO is spread in many forms of variations and implementation i.e., traveling salesman person (TSP) [34], vehicle routing problem [35], set covering problem [36], and graph coloring [37].

2.2 Whale Optimization Algorithm

Recently, many new, robust, and improved SI algorithms have been introduced. One of the latest algorithms under swarm-based technique is Whale Optimization Algorithm (WOA). WOA is firstly introduced by Mirjalili and Lewis in the year 2016 [10]. This algorithm is inspirited by the bubble net of a humpback whale as their hunting strategy [38]. WOA popularly come with the two ways; 1) exploitation and 2) exploration.

Meanwhile, under the exploitation strategy exists another two processes called encircling prey and spiral updating position. At first strategy, the best global or prey (in terms of whales) is investigated. While the exploration strategy is utilized to randomly explore the prey in order to avoid local optima. WOA is assumed had an impressive potential in exploring the best global solution and bypassing local optima in a good time length. This is because of their high capacity to maintain a stable process between exploitation and exploration. It's already confirmed by the previous research in solving many real-life problems i.e., optimal sizing of renewable resources for loss reduction in distribution systems [39], feature selection [40], sizing optimization for skeletal structures [41] and data clustering [42]. Since the superiorities of WOA are undeniable, many researchers choose WOA rather than the other algorithm in dealing with complex optimization problems.

2.3 Current Issues in WOA

However, WOA shows a similar deficiency compare to other SI algorithms where it is slow in terms of convergence rate. Khalil *et al.,* [14] asserted that the WOA performance degrades when it is applied to a large-size problem due to the need for a massive computational workload. Yan *et al.,* [12] also reported a similar weakness of WOA in water resource allocation optimization problem. WOA shows a slow convergence rate by the increment number of iterations. They have started the iteration with a set of 180 times to an increase of 2000 times and does not yield the expected convergence rate [12]. Therefore, further research is needed to overcome the drawbacks of the low convergence accuracy and convergence rate before the WOA is applied to the more difficult optimization problem.

3. Cluster based Computation for WOA

WOA operates well on a single machine and has proven to outperform another Evolutionary Algorithm (EA) [36, 39-43]. However, as mentioned earlier, the WOA faces the major drawback against low convergence speed when it comes to complicated issues. This gives a negative impact on the performance of WOA towards time consumption and computational workload [12-14]. The computational task in WOA is the potential to enhance and expanded into clusters of WOA agents to produce a faster and feasible solution.

On the other hand, the multi objective or constraint/cost computation are also possible to be implemented in different/distributed machines or clusters considering the size, complexity and scalability of the datasets. In order to support enormous request at the same time, parallels and distributed approaches are promising to be develop in WOA framework. Some related works of other approaches includes Message-Passing Interface (MPI) [44], Open Multi-Processing (OpenMP) [45], Spark and Hadoop MapReduce [46-47].

3.1 MapReduce Framework

MapReduce is a programming model that help to simplify the growth of scalable and fault-tolerant parallel applications in a distributed condition. This model is basically formed from the two main functions, called Map and Reduce, where they drive together in a divide-and-conquer component. Parallelization in MapReduce occurs through distributing the workload among a cluster of heterogeneous commodity machines.

Meanwhile, Apache Hadoop [48-49] is an open-source implementation of the MapReduce structure written in Java. Previously, researchers are frequently using the Apache Hadoop due it is containing the characteristic of MapReduce that make it easy to use in a various range of disciplines including text mining, machine learning, bioinformatics, etc. Besides that, MapReduce also get a high concern in distributing EAs. Basically, the builder only needs to construct the main algorithm in a Map and Reduce structure [50-51]. This will encourage them to only pay attention to the algorithm but not too worried about the management of distributed implementations.

3.2 Spark Distributed Computation

However, when compared between Hadoop MapReduce and Spark, Spark dominantly shows an excellent performance in several perspectives rather than Hadoop MapReduce. Spark is well-known as an open-source of huge data structure with the strength of high speed and more common-function data processing engines rather than Hadoop. Compared to a Hadoop MapReduce, Hadoop provide an open-source structure, but it only limited to writing purposes. It is also developed to run the big volume of data on a group of commodity hardware and is able to operate the data in batch mode [52].

Concurrently, in the spec of Spark area, it is conquered the several areas in showing the outstanding performances which are iterative processing, near real-time processing, good of graph processing, etc. However, when comes to the biggest data sets, Hadoop is preferable to take it over since it needs a high frequency for the shuffling and sorting process [52]. Table 1 shows a comparison between Apache Spark and Hadoop MapReduce.

In this situation, Spark and MapReduce have their own advantages and disadvantages. Spark able to conduct any kind of specifications including batch, interactive and streaming, while MapReduce only restricted on batch processing. Hence, it is worth to discover which methods are expected to have a high tendency to increase the convergence rate of WOA in solving a real-world problem.

Table 1
Comparison between Apache Spark and Hadoop MapReduce [53]

No.	Characteristic	Apache Spark	Hadoop MapReduced
1.	Processing speed	100 times faster in memory and while running on disk.	Slower than Apache Spark because if I/O disk latency.
2.	Data processing	Batch Processing as well as Real Time Data Processing.	Only for Both Processing.
3.	Category	Data Analytics Engine.	Data Processing Engine.
4.	Scalability	Scalable	Scalable
5.	Machine learning	Apache Spark have inbuilt API's to Machine Learning.	More compatible with Apache Mahout while integrating with Machine Learning.
6.	Real-time analysis	Excellent	Fail
7.	Compatibility	Can integrate with all data sources and file formats supported by Hadoop cluster.	Majorly compatible with all the data sources and file formats.
8.	Scheduler	Have own scheduler.	Dependent on external Scheduler.
9.	Ease of use	Easier to use because of Rich API's.	Bit complex comparing Apache Spark because of JAVA API's.
10.	Fault tolerance	Uses resilient distributed dataset (RDD) and various data storage models.	Uses replication for fault tolerance.
11.	Duplicate elimination	Able to process every record exactly once hence eliminated duplication.	Do not support this feature.
12.	Latency	Much faster comparing MapReduce Framework.	Very high latency.
13.	SQL	Supports through Spark SQL.	Supports through Hive Query Language
14.	Complexity	Easy to write and debug.	Difficult to write and debug codes.
15.	Security	More evolving and getting matured.	More secured compared to Apache Spark.
16.	Costs	More costlier because of large amount of RAM.	Less costlier comparing Apache Spark.

Additionally, there have other cluster-based technologies, namely Message Passing Interface (MPI) and OpenMP. Message Passing Interface (MPI) is introduced by researchers and implemented in a wide range of computing architectures field [54]. While OpenMP is used as a supporter of the multi-platform shared memory multiprocessing programming in C, and C++ [55]. As a conclusion, Spark and Hadoop MapReduce can be categorized in a cluster-based model in solving large datasets of problems. Both technologies show the different pros and cons of architecture, speed rate, data processing, compatibility, and others.

4. Multiple Restart Strategy

Multi-restart strategy is defined as the initialization process where requires rework operations constructively. The rework operation is considered successful when the best result is obtained and selected for the execution. Glover [56] proposes a model/structure to build a few networks to multiple-restart techniques to upgrade the initialization solution. Each model provides the process to produce the initialization value for the parameters, while it's also used to produce perturbation values from another initialization point. The changing of perturbation rules should involve the local search method for generating re-start. Thus, to expedite the convergence process, there is a need to control the initial randomization of solution.

Basically, randomization is regulated in two ways [56]. Firstly, the most popular method in heuristic is called a random restart approach. In this way, the random value is inserted as a first-generation for initializing the earliest starting point. Secondly, this method is called a random shakeup procedure. It systematically produces a randomized series of moves that lead the heuristic from its customary path into a region it would not otherwise reach.

Initialization is the assignment of an initial value for a data object or variable. Initialization plays a very important role in meta-heuristic process and then encourages many studies in this field to improve initialization techniques. The various approach has been proposed with the main goal of increasing the search space area. Authors in Kimura and Matsumura [57], Ma and Vandenbosch [58], and Kazimipour *et al.,* [59] state that, the wide range of search space will increase the probability to gain the near-optimal solution in both aspects of maximizing global optima searching and minimize the computational costs.

Generally, most of the metaheuristic algorithms is utilize the random initialization at the starting process i.e., Genetic Algorithm [60], Particle Swarm Optimization (PSO) [27], Artificial Bee Colony Algorithm (ABC) [61], Grey Wolf Optimizer (GWO) [62], Artificial Fish Swarm Algorithm (AFSA) [63], and Whale Optimization Algorithm (WOA) [10]. As mentioned earlier, a random solution is key to enhancing the convergence performance in order to balance the diversity in the population [64].

Xiang Li *et al.,* [65] had present their method in producing the initial population of a Genetic Algorithm (GA) and investigate their effect on the effectiveness of overall GA in the scope of computational time and convergence. Shupeng Gao *et al.,* [66] had proposed a novel pheromone initialization (NPI) approach for Ant Colony Optimization (ACO) algorithm to counter the Travel Salesman Problem (TSP). Thus, the proposed method shows an improvement in the quality of robustness and the end solution. Meanwhile, Gai-Ge Wang *et al.,* [67] using the equal partition and two random distributions (F and T distribution) to upgrade the Monarch Butterfly Optimization (MBO) achievement.

The basic principle of multiple restart strategy is to mix between the random start (population initialization) and finding the best approach of between stochastic and deterministic approach. Other related technique for deterministic approach is the use of Chaos Theory which is promising to be used as a strategy for initialization of WOA.

5. Methodology: Appliances Scheduling Problem using Modified Whale Optimization Algorithm

There are three phases involves in our research methodology. We normally follow the method defined from our previous research [68-70].

5.1 Phase 1: Information Gathering and Theoretical Framework

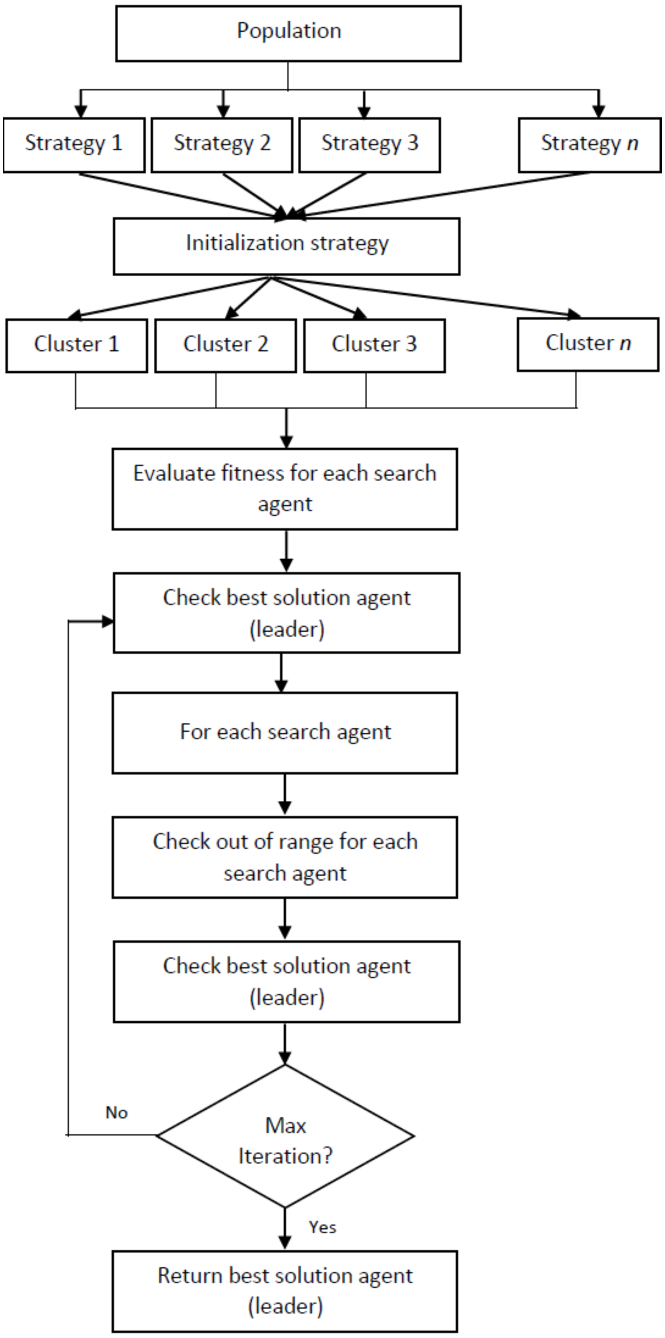
Related Activities in Phase 1 includes (i) Literature review on appliances scheduling problem and WOA and (ii) semi-structured interview to gather constraints related to appliances scheduling problem.

5.2 Phase 2: Development Model of Appliances Scheduling by Using A Cluster-Based Multi-Objective WOA With Multiple Restart Strategy

This phase concentrates on developing the new objective function with a set of constraints related to scheduling of appliances. Next, the research will focus on finding an initial feasible solution based on an ordering strategy where the meta-heuristics used are adapted. At this stage, the multiple initialization/restart strategy will be applied with a mixture of stochastic and deterministic methods. Subsequently, a new intelligent Multi-Objective WOA which is constructed through the implementation of the cluster-based platform such as MapReduce i.e. Hadoop and Spark to improve the quality of the appliances schedule. Related Activities in Phase 2 includes (i) Development of Multi-Objective of WOA (MOWOA), (ii) Development of MOWOA in multiple restart strategies and (iii) Development of Cluster based MOWOA using MapReduce. The pseudo-code of a general WOA algorithm is illustrated in Algorithm 1.Due to the limitation of SI, we modified the original WOA. The flow chart of the New Cluster-Based WOA with Multiple Restart Strategy is illustrated in Figure 1.

6. Conclusion and Future Works

This project is expected to assist the stakeholders and agencies to manage the energy industry efficiently. This work is in line with the United Nation's Sustainable Development Goals (SDG17) that is to ensure access to affordable, reliable, sustainable, and modern energy for all. In Malaysia, a quality schedule with feasible solutions for energy-efficiency which involves multiple households with variety of appliances in a residential or commercial area is perceived as an important national agenda. On the other hand, we attempt to improve the current SI by improving WOA as an example and has the potential to replicate to other SI algorithms. In future we will continue with algorithm development and experimentations.



Algorithm 1. Pseudo-code of the WOA algorithm

Initialize the whales population $X_i (i = 1,2, \ldots, n)$
Calculate the fitness of each search agent
X^* = the best search agent
while ($t <$ maximum number of iterations)
  for each search agent
    Update a, A, C, l and p
      if1 ($p < 0.5$)
        if2($|A| < 1$)
          Update the position of the current search agent by the Eq. (1)
        else if2 ($|A| \geq 1$)
          Select a random search agent (X_{rand})
            Update the position of the current search agent by Eq. (8)
        end if2
          else if1 ($p \geq 0.5$)
            Update the position of the current search by the Eq. (5)
      end if1
  end for
Check if any search agent goes beyond the search space and amend it
Calculate the fitness of each search agent
Update X^* if there is a better solution
$t = t + 1$
end while
return X^*

Fig. 1. Cluster Based WOA with Multiple Restart Strategy

Acknowledgement

This research was funded by a grant from Ministry of Education Malaysia (FRGS Grant; SO Code: 14379). Our gratitude also goes to Research and Innovation Centre and The Library of Universiti Utara Malaysia for giving us an opportunity to conduct a case study for this project.

References

[1] Morsy, M., M. Fahmy, H. A. Elshakour, and A. M. Belal. "Effect of thermal insulation on building thermal comfort and energy consumption in Egypt." *Journal of Advanced Research in Applied Mechanics* 43, no. 1 (2018): 8-19.

[2] Muhieldeen, M. W., and Y. C. Kuang. "Saving energy costs by combining air-conditioning and air-circulation using CFD to achieve thermal comfort in the building." *Journal of Advanced Research in Fluid Mechanics and Thermal Sciences* 58, no. 1(2019): 84-99.

[3] Shaikh, P. H., Nor, N. M., Nallagownden, P., and Elamvazuthi, I. "Intelligent multi-objective optimization for building energy and comfort management." *Journal of King Saud University-Engineering Sciences* 30, no. 2 (2018): 195-204. https://doi.org/10.1016/j.jksues.2016.03.001
[4] Rahim, Sahar. "Performance Evaluation of Heuristic Algorithms in Smart Grids." Ms. Thesis, COMSATS Institute of Information Technology, Islamabad, 2015.
[5] Muralitharan, K., Rathinasamy Sakthivel, and Yan Shi. "Multiobjective optimization technique for demand side management with load balancing approach in smart grid." *Neurocomputing* 177 (2016): 110-119. https://doi.org/10.1016/j.neucom.2015.11.015
[6] Zafar, Ayesha, Samia Shah, Rabiya Khalid, Sardar Mehboob Hussain, Hassan Rahim, and Nadeem Javaid. "A meta-heuristic home energy management system." In *2017 31st International Conference on Advanced Information Networking and Applications Workshops (WAINA)*, pp. 244-250. IEEE, 2017. https://doi.org/10.1109/WAINA.2017.118
[7] Mirjalili, Seyedali, Seyed Mohammad Mirjalili, and Abdolreza Hatamlou. "Multi-verse optimizer: a nature-inspired algorithm for global optimization." *Neural Computing and Applications* 27, no. 2 (2016): 495-513. https://doi.org/10.1007/s00521-015-1870-7
[8] Mirjalili, Seyedali, Seyed Mohammad Mirjalili, and Andrew Lewis. "Grey wolf optimizer." *Advances in engineering software* 69 (2014): 46-61. https://doi.org/10.1016/j.advengsoft.2013.12.007
[9] Mirjalili, Seyedali. "SCA: a sine cosine algorithm for solving optimization problems." *Knowledge-based systems* 96 (2016): 120-133. https://doi.org/10.1016/j.knosys.2015.12.022
[10] Mirjalili, Seyedali, and Andrew Lewis. "The whale optimization algorithm." *Advances in engineering software* 95 (2016): 51-67. https://doi.org/10.1016/j.advengsoft.2016.01.008
[11] Niu, P. F., Z. L. Wu, Y. P. Ma, C. J. Shi, and J. B. Li. "Prediction of steam turbine heat consumption rate based on whale optimization algorithm." *CIESC J* 68, no. 3 (2017): 1049-1057.
[12] Yan, Zhihong, Jinxia Sha, Bin Liu, Wei Tian, and Jipan Lu. "An ameliorative whale optimization algorithm for multi-objective optimal allocation of water resources in Handan, China." *Water* 10, no. 1 (2018): 87. https://doi.org/10.3390/w10010087
[13] Kaur, Gaganpreet, and Sankalap Arora. "Chaotic whale optimization algorithm." *Journal of Computational Design and Engineering* 5, no. 3 (2018): 275-284. https://doi.org/10.1016/j.jcde.2017.12.006
[14] Khalil, Y., M. Alshayeji, and I. Ahmad. "Distributed whale optimization algorithm based on MapReduce." Concurrency and Computation: Practice and Experience 31, no. 1 (2019): 4872. https://doi.org/10.1002/cpe.4872
[15] Baker, Kenneth R. *Introduction to sequencing and scheduling*. John Wiley & Sons, 1974.
[16] Morton, Thomas, and David W. Pentico. *Heuristic scheduling systems: with applications to production systems and project management*. Vol. 3. John Wiley & Sons, 1993.
[17] Pinedo, M.L. *Scheduling: Theory, Algorithm and Systems*. 3rd Edition, Springer-Verlag, New York, 2008.
[18] Adhi, Antono, Budi Santosa, and Nurhadi Siswanto. "A meta-heuristic method for solving scheduling problem: crow search algorithm." In *IOP Conf. Ser. Mater. Sci. Eng.*, vol. 337, no. 1. 2018. https://doi.org/10.1088/1757-899X/337/1/012003
[19] Salman, Ayed A., Imtiaz Ahmad, and Mahamed GH Omran. "A metaheuristic algorithm to solve satellite broadcast scheduling problem." *Information Sciences* 322 (2015): 72-91. https://doi.org/10.1016/j.ins.2015.06.016
[20] Teoh, Chong Keat, Antoni Wibowo, and Mohd Salihin Ngadiman. "Review of state of the art for metaheuristic techniques in Academic Scheduling Problems." *Artificial Intelligence Review* 44, no. 1 (2015): 1-21. https://doi.org/10.1007/s10462-013-9399-6
[21] Mwayuli, Constance K., and Cyrus Wekesa. "Genetic Algorithm-Based Residential Appliance Scheduling for Optimal Power Management." *International Journal of Innovative Research in Science, Engineering and Technology* 6, no. 4 (2017).
[22] Maher, R. A., and A. M. Al Hadidi. "Engineering project management planning and scheduling." *International Journal of Civil Engineering and Technology* 8, no. 1 (2017):140-148.
[23] Beni, Gerardo, and Jing Wang. "Swarm intelligence in cellular robotic systems." In *Robots and biological systems: towards a new bionics?*, pp. 703-712. Springer, Berlin, Heidelberg, 1993. https://doi.org/10.1007/978-3-642-58069-7_38
[24] Engelbrecht, Andries P. *Computational intelligence: an introduction*. John Wiley & Sons, 2007.

[25] Millonas, Mark M. "Swarms, phase transitions, and collective intelligence." *arXiv preprint adap-org/9306002* (1993).
[26] Brezočnik, Lucija, Iztok Fister, and Vili Podgorelec. "Swarm intelligence algorithms for feature selection: a review." *Applied Sciences* 8, no. 9 (2018): 1521. https://doi.org/10.3390/app8091521
[27] Kennedy, James, and Russell Eberhart. "Particle swarm optimization." In *Proceedings of ICNN'95-International Conference on Neural Networks*, vol. 4, pp. 1942-1948. IEEE, 1995.
[28] Dorigo, M. "Learning and Natural Algorithms." Ph.D. Thesis, Politecnico di Milano, Milano, Italy, 1992.
[29] Ravagnani, Mauro ASS, Aline P. Silva, Evaristo C. Biscaia Jr, and Jose A. Caballero. "Optimal design of shell-and-tube heat exchangers using particle swarm optimization." *Industrial & Engineering Chemistry Research* 48, no. 6 (2009): 2927-2935. https://doi.org/10.1021/ie800728n
[30] Han, Wutao, Linghong Tang, Gongnan Xie, and Qiuwang Wang. "Performance comparison of particle swarm optimization and genetic algorithm in rolling fin-tube heat exchanger optimization design." In *Heat Transfer Summer Conference*, vol. 48487, pp. 5-10. 2008. https://doi.org/10.1115/HT2008-56213
[31] Yu, Xiaochun, Zhi-qin Cui, Ying Yu, and Yi-qiang Shi. "Fuzzy optimal design of the plate-fin heat exchangers by particle swarm optimization." In *2008 Fifth International Conference on Fuzzy Systems and Knowledge Discovery*, vol. 3, pp. 574-578. IEEE, 2008. https://doi.org/10.1109/FSKD.2008.313
[32] Dorigo, Marco, Vittorio Maniezzo, and Alberto Colorni. "The ant system: An autocatalytic optimizing process." (1991).
[33] Dorigo, Marco, Alberto Colorni, and Vittorio Maniezzo. "Distributed optimization by ant colonies." (1991).
[34] Dorigo, Marco, V. Maniezzo, and A. Colorni. *Positive feedback as a search strategy. Dipartimento di Elettronica, Politecnico di Milano*. Italy, Tech. Rep. 91-016, 1991.
[35] Bell, John E., and Patrick R. McMullen. "Ant colony optimization techniques for the vehicle routing problem." *Advanced engineering informatics* 18, no. 1 (2004): 41-48. https://doi.org/10.1016/j.aei.2004.07.001
[36] Lessing, Lucas, Irina Dumitrescu, and Thomas Stützle. "A comparison between ACO algorithms for the set covering problem." In *International Workshop on Ant Colony Optimization and Swarm Intelligence*, pp. 1-12. Springer, Berlin, Heidelberg, 2004. https://doi.org/10.1007/978-3-540-28646-2_1
[37] Costa, Daniele, and Alain Hertz. "Ants can colour graphs." *Journal of the operational research society* 48, no. 3 (1997): 295-305. https://doi.org/10.1038/sj.jors.2600357
[38] Nasiri, Jhila, and Farzin Modarres Khiyabani. "A whale optimization algorithm (WOA) approach for clustering." *Cogent Mathematics & Statistics* 5, no. 1 (2018): 1483565. https://doi.org/10.1080/25742558.2018.1483565
[39] Reddy, P. Dinakara Prasad, VC Veera Reddy, and T. Gowri Manohar. "Whale optimization algorithm for optimal sizing of renewable resources for loss reduction in distribution systems." *Renewables: wind, water, and solar* 4, no. 1 (2017): 3. https://doi.org/10.1186/s40807-017-0040-1
[40] Mafarja, Majdi M., and Seyedali Mirjalili. "Hybrid whale optimization algorithm with simulated annealing for feature selection." *Neurocomputing* 260 (2017): 302-312. https://doi.org/10.1016/j.neucom.2017.04.053
[41] Kaveh, A. "Sizing optimization of skeletal structures using the enhanced whale optimization algorithm." In *Applications of metaheuristic optimization algorithms in civil engineering*, pp. 47-69. Springer, Cham, 2017. https://doi.org/10.1007/978-3-319-48012-1_4
[42] Jadhav, Amolkumar Narayan, and N. Gomathi. "WGC: hybridization of exponential grey wolf optimizer with whale optimization for data clustering." *Alexandria engineering journal* 57, no. 3 (2018): 1569-1584. https://doi.org/10.1016/j.aej.2017.04.013
[43] Sipper, Moshe, Randal S. Olson, and Jason H. Moore. "Evolutionary computation: the next major transition of artificial intelligence?." *BioData Mining* (2017): 1-3. https://doi.org/10.1186/s13040-017-0147-3
[44] Schikuta, E. "Message-passing-interface-forum: MPI: a message-passing interface standard." Techn. Ber. Knoxville, Tennesee: University of Tennessee, 1994.
[45] BoardO. OpenMP application programinterface (version 4.5). Technical Report. OpenMP; 2015.

[46] Lämmel, Ralf. "Google's MapReduce programming model—Revisited." *Science of computer programming* 70, no. 1 (2008): 1-30.
https://doi.org/10.1016/j.scico.2007.07.001
[47] Polato, Ivanilton, Reginaldo Ré, Alfredo Goldman, and Fabio Kon. "A comprehensive view of Hadoop research—A systematic literature review." *Journal of Network and Computer Applications* 46 (2014): 1-25.
https://doi.org/10.1016/j.jnca.2014.07.022
[48] Borthakur, Dhruba. "The hadoop distributed file system: Architecture and design." *Hadoop Project Website* 11, no. 2007 (2007): 21.
[49] White, Tom. *Hadoop: The definitive guide*. " O'Reilly Media, Inc.", 2012.
[50] Tan, Ying, and Ke Ding. "A survey on GPU-based implementation of swarm intelligence algorithms." *IEEE transactions on cybernetics* 46, no. 9 (2015): 2028-2041.
https://doi.org/10.1109/TCYB.2015.2460261
[51] Zhan, Zhi-Hui, Xiao-Fang Liu, Huaxiang Zhang, Zhengtao Yu, Jian Weng, Yun Li, Tianlong Gu, and Jun Zhang. "Cloudde: A heterogeneous differential evolution algorithm and its distributed cloud version." *IEEE Transactions on Parallel and Distributed Systems* 28, no. 3 (2016): 704-716.
https://doi.org/10.1109/TPDS.2016.2597826
[52] Bekker, Alex. "Spark vs. Hadoop MapReduce: Which big data framework to choose." (2017).
[53] Educba. "MapReduce vs Apache Spark- 20 Useful Comparisons To Learn."
[54] Gropp, William, Ewing Lusk, Nathan Doss, and Anthony Skjellum. "A high-performance, portable implementation of the MPI message passing interface standard." *Parallel computing* 22, no. 6 (1996): 789-828.
https://doi.org/10.1016/0167-8191(96)00024-5
[55] Gagne, A. S., and G. P. B. Galvin. *Operating system concepts (9th ed.).* Hoboken, N.J.: Wiley. pp. 181–182, 2012.
[56] Glover, Fred. "Heuristics for integer programming using surrogate constraints." *Decision sciences* 8, no. 1 (1977): 156-166.
https://doi.org/10.1111/j.1540-5915.1977.tb01074.x
[57] Kimura, Shuhei, and Koki Matsumura. "Genetic algorithms using low-discrepancy sequences." In *Proceedings of the 7th annual conference on Genetic and evolutionary computation*, pp. 1341-1346. 2005.
https://doi.org/10.1145/1068009.1068225
[58] Ma, Zhongkun, and Guy AE Vandenbosch. "Impact of random number generators on the performance of particle swarm optimization in antenna design." In *2012 6th European conference on antennas and propagation (EUCAP)*, pp. 925-929. IEEE, 2012.
https://doi.org/10.1109/EuCAP.2012.6205998
[59] Kazimipour, Borhan, Xiaodong Li, and A. Kai Qin. "A review of population initialization techniques for evolutionary algorithms." In *2014 IEEE Congress on Evolutionary Computation (CEC)*, pp. 2585-2592. IEEE, 2014.
https://doi.org/10.1109/CEC.2014.6900618
[60] Mitchell, Melanie. "An introduction to genetic algorithms." *Cambridge, Massachusetts. London, England* 1996 (1996).
[61] Karaboga, Dervis. *An idea based on honey bee swarm for numerical optimization*. Vol. 200. Technical report-tr06, Erciyes university, engineering faculty, computer engineering department, 2005.
[62] Mirjalili, Seyedali, Seyed Mohammad Mirjalili, and Andrew Lewis. "Grey wolf optimizer." *Advances in engineering software* 69 (2014): 46-61.
https://doi.org/10.1016/j.advengsoft.2013.12.007
[63] Li, X. L. "A new intelligent optimization-artificial fish swarm algorithm." PhD diss., Zhejiang University of Zhejiang, China, 2003.
[64] Goldenberg, David E. "Genetic algorithms in search, optimization and machine learning." (1989).
[65] Li, Xiang, Ningchuan Xiao, Christophe Claramunt, and Hui Lin. "Initialization strategies to enhancing the performance of genetic algorithms for the p-median problem." *Computers & Industrial Engineering* 61, no. 4 (2011): 1024-1034.
https://doi.org/10.1016/j.cie.2011.06.015
[66] Gao, Shupeng, Jiaqi Zhong, Yali Cui, Chao Gao, and Xianghua Li. "A novel pheromone initialization strategy of ACO algorithms for solving TSP." In *2017 13th International Conference on Natural Computation, Fuzzy Systems and Knowledge Discovery (ICNC-FSKD)*, pp. 243-248. IEEE, 2017.
https://doi.org/10.1109/FSKD.2017.8393155
[67] Wang, Gai-Ge, Guo-Sheng Hao, Shi Cheng, and Zhihua Cui. "An improved monarch butterfly optimization with equal partition and f/t mutation." In *International Conference on Swarm Intelligence*, pp. 106-115. Springer, Cham, 2017.
https://doi.org/10.1007/978-3-319-61824-1_12

[68] Omar, Mohd Faizal, Rosalina Abdul Salam, Nuraini Abdul Rashid, and Rosni Abdullah. "Multiple sequence alignment using genetic algorithm and simulated annealing." In *Proceedings. 2004 International Conference on Information and Communication Technologies: From Theory to Applications, 2004.*, pp. 455-456. IEEE, 2004.

[69] Omar, M. F., R. A. Salam, R. Abdullah, and N. A. Rashid. "Multiple sequence alignment using optimization algorithms." *International Journal of Computational Intelligence* 1, no. 2 (2005): 81-89.

[70] Abdul-Rahman, Syariza, Nur Suriani Sobri, Mohd Faizal Omar, Aida Mauziah Benjamin, and Razamin Ramli. "Graph coloring heuristics for solving examination timetabling problem at Universiti Utara Malaysia." In *AIP Conference Proceedings*, vol. 1635, no. 1, pp. 491-496. American Institute of Physics, 2014. https://doi.org/10.1063/1.4903627