Artificial Fish Swarm Algorithm for Single Machines Scheduling with release date

Tahani Jabbar Khraibet[1,*], Nagham Muosa Neamah[2]

[1] Department of Mathematics Thi-Qar Directorate of Education, Ministry of Education, Thi-Qar, Iraq

[2] Department of Mathematics, College of Science for Women, University of Baghdad, Baghdad, Iraq

[*]Corresponding Author: tahani@utq.edu.iq

Received 3 Nov. 2025, Accepted 12 Dec. 2025, published 30 Dec. 2025.
DOI: 10.52113/2/12.02.2025/158-165

Abstract: The present paper addresses the problem of scheduling jobs on single machines to minimize the total weight of completion times and late work, considering release dates. This problem, denoted by $1 / r_j / \sum_{j=1}^{n}(w_j c_j + V_j)$, is NP-hard and, as far as we know, unstudied. In view of the NP-hardness of the problem, an Artificial Fish Swarm Algorithm approach is proposed to solve it. The Computational results indicate that the proposed algorithm is capable of finding near optimal solutions (approximate solutions) to the problems up to 4000 jobs. These results establish the AFSA as a revolutionary instrument for addressing intricate scheduling challenges in both theoretical and practical contexts. Arithmetic results are calculated computational results were obtained using MATLAB 2023a.

***Keywords*:** Scheduling; Single machines; the total weighted of completion times; the late work with release date; Artificial Fish Swarm Algorithm

1. Introduction

Scheduling, in general, means giving machines jobs so that all jobs can be done within the constraints set. [1] Single machine scheduling challenges are about figuring out how to schedule n jobs in a way that minimizes some functions of the weight completion times and the total late work. When preemption of jobs is not allowed, problems break down into two smaller ones. We analyze a class of problems that involve additive objective functions, where all jobs in the relevant part of the schedule are arranged based on specific priority rules. This work considers online scheduling over single machines with the goal of minimizing the overall weighted completion time. It is widely known that the optimal deterministic online algorithm with the competitive ratio has been provided for $1 // \sum_{j=1}^{n} w_j c_j$ is optimally solved by Smith's jobs in order of shortest weighted processing times $\frac{p_i}{w_i}$.[2] Rudek [3] studied the problem of $1 // \sum_{j=1}^{n} w_j c_j$ to be strongly NP-hard using the BAB method. Chen et.al [4] claimed that heuristic algorithm to solve problem $\sum_{j=1}^{n} w_j c_j$ up to 40 jobs. Kis and Györgyi [5] used the Approximation algorithms to solve the problem $\sum_{j=1}^{n} w_j c_j$. Chen et.al [6] studied the problem the total weighted completion time and then solved these problems by the heuristic algorithm. Lin et.al [7] used the mixed integer linear programs (MILPs) to solve minimize the total weighted completion time. The problem $1 // \sum_{j=1}^{n} V_j$ is NP-complete problem [8] the jobs in order of earliest due date (EDD) rule1 (i.e., by ordering the jobs according to non-decreasing due dates. Mosheiov et.al [9] introduced pseudo-polynomial dynamic programming solution algorithms to solve the problem $1//\sum_{j=1}^{n} V_j$. Mor and Shabtay [10] used the approximation algorithms to solve the total late work. Neamah and Kalaf [11] used a Exact and Heuristic Methods to solve the minimize multi-objective function $\sum C_j + \sum V_j + E_{max}$. A lot of researchers have looked into the heuristic method in depth and written about. [12-13] During the last few decades, SMSP challenges have become increasingly difficult to address. This approach to solving such problems is useful for optimization and solving scheduling problems with clear solutions. As a result of the complexity that arises from many objectives, it is very challenging to use traditional methods to single-machine scheduling problems. To obtain best or near-optimal solutions to a problem, metaheuristic algorithms are introduced. [14] The remaining sections of this paper are organized as follows: Section 2 presents the mathematical model. Section 3, presents metaheuristic algorithms (Artificial Fish Swarm Algorithm). Results and discussion are presented in Section 4. Finally, conclusions are given in Section 5.

2. Mathematical model

This section introduces a collection of multi-objective functions, each with n jobs, to solve the single machine scheduling problem. We analyzed these functions and discovered that they were constantly available and capable of completing or carrying out their tasks.

2.1. Notations

Some symbols used to create the multi-objective and multi-criteria models and rules that would be employed in this work paper.

- $N = \{1,2,3, \dots, n\}$.
- p_j = processing time for job j.
- w_j = weight of the job j.
- d_j = due date of the job j.
- C_j = completion time of the job j.
- r_j = Release date.
- NP = Non-deterministic Polynomial-time hard.

2.2. Objective functions

This sub section introduces a collection of multi-objective functions, each with n jobs, to solve the single-machine scheduling problem. We analyzed these functions and discovered that they were constantly available and capable of completing or carrying out their tasks. In this study, objective functions were introduced to minimize the weighted completion time $1 / / \sum_{j=1}^{n} w_j c_j$ and the total late work $1 // \sum_{j=1}^{n} V_j$ with release date r_j $1 / r_j / \sum_{j=1}^{n}(w_j c_j + V_j)$ that denoted to the (TN)) problem, which can be defined as follows:

$$TN = min_{\sigma \in S}\{M_{(\sigma)}\} = min_{s \in S}\left\{\sum_{j=1}^{n}(w_j c_j + V_j)\right\}. \quad (1)$$

$$\left.\begin{array}{l} C_{(j)} \geq r_{(j)} + p_{\sigma(j)} \\ C_{(j)} = C_{j-1} + p_{\sigma(j)} \\ V_{\sigma(j)} \leq C_{\sigma(j)} - d_{\sigma(j)} \\ V_{\sigma(j)} \leq p_{\sigma(j)} \\ r_{(j)}, w_{\sigma(j)} > 1, V_{\sigma(j)} \geq 0, d_{\sigma(j)} > 0, \geq 0 \quad p_{\sigma(j)} > 0 \end{array}\right\} \left.\begin{array}{c} j\ from\ 1\ to\ n\ and \\ j = 2,3,\dots,n. \end{array}\right\} \quad (2)$$

3. Artificial Fish Swarm Algorithm (AFSA)

Fish may naturally discover places to eat based on whether they are searching alone or in a group. AFSA is one of the best swarm algorithms for finding approximate solutions. AFSA is inspired by fishes group movements and their various social behaviors. Fishes build their colonies depending on privative intelligence, social behavior and behave accordingly. This social behavior helps fishes to look for food, avoid danger sources and interact with each other. High convergence speed, flexibility and high accuracy are the main merits of AFSA. The AFSA main function is finding maximum amount of food. Computer scientists always face a challenging issue which is that solving the NP-complete problems, but the emergence of swarm intelligence algorithms have natively supported their ways in dealing with these problems. The fishes intelligence and social behavior have great effects on the algorithm global search and convergence speed. By following a single search or in groups, fishes can find food areas. Imitating the instinctual behaviors of fish in predation, swarming, and solitary or group searching to attain the global optimum. Li (2003).[15] AFSA is suggested by Li Xiao Lei .(2002)[16] The stochastic population-based algorithm is stimulated by fish group movements and intelligent behavior. Artificial fish (AF) is an imaginary aspect of real fish that is utilized to provide problem explanation and analysis.

The fundamental concept of AFSA [17-18] is to emulate fish behaviors, including swarming and predation, succeeded by localized searches of individual fish to achieve the global optimum. Many Researchers studied and written about AFSA algorithm including [19-21]. The Artificial Fish Swarm Algorithm (AFSA) is not only explained in general, but also particularly tailored to address the single machine scheduling problem. In this application, each artificial fish represents a conceivable job sequence, with the fish's position corresponding to a certain scheduling solution. The objective function is defined as minimizing the total weighted completion time and the total late work with release date . The artificial fish's behaviors, such as prey, swarming, and following, have been tuned to efficiently explore and exploit the search space. The optimum sequence is determined using iterative updates based on the fitness value associated with the scheduling aim.

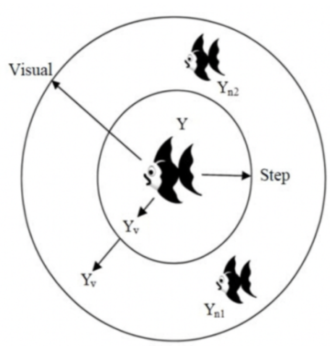

Fig (1): Artificial fish vision concept

The diagram in the figure 1 illustrates that AF recognizes external concepts through its visual knowledge. AF is now at vector $Y = (Y_1, Y_2, \dots, Y_n)$. The visibility domain of AF can show the visual, and Y_v shows where the AF wants to pick moving in a visual. The AF advances one step closer to Y_v if Y_n has a higher food density than the present position. This means that the AF goes from Y to X_{next}. If AF's current location is better than Y_v , it chooses a different position in its visual. The fitness value of position Y is the amount of food in that position, which is shown by $f(Y)$. The step shows how far the movement can go. The step represents the maximum length of the movement. The equation $Dis(ij) = ||Y_i - Y_j||$ shows how far apart two AFs are that are located at Y_i and Y_j. Let $Y = (y_1, y_2, \dots, y_n)$ and $Yv = (y_{v1}, y_{v2}, \dots, y_{vn})$ is discussed as follows:

$$Y_{vi} = y_i + Visual.rand\,() \,, i \in (0, n]$$

(3)

$$\mathrm{Y_{next}} = \mathrm{Y} + \frac{\mathrm{Y_v - Y}}{\|\mathrm{Y_v - Y}\|}.\mathrm{Step.rand()}.$$

(4)

Algorithm . Artificial fish swarm algorithm

Step1. Initialize the settings of the artificial fish, such as Step and Visual, as well as the number of exploratory tries, maximum number of iterations, and n randomly produced fish.

Step2. Set up a bulletin board to keep track of how each fish is doing right now, and choose the best value that was recorded.

Step3. Putting into action prey behaviour, swarm behaviour, and follow behavior.

Step4. The best value in the bulletin board has been changed.

Step5. If the termination condition is met, produce the result; if not, revert to step 2.

Because the companion Y_j state has a greater food concentration (a higher fitness function value) and a less crowded environment, it advances one step to the companion Y_j. Otherwise, carries out the preying behavior.

$$\mathrm{Y^{t+1}}_{\mathrm{i}} = \mathrm{Y^{t}}_{\mathrm{i}} + \frac{\mathrm{Y_j} - \mathrm{Y^{t}}_{\mathrm{i}}}{\| \mathrm{Y_j} - \mathrm{Y^{t+1}}_{\mathrm{i}} \|}.\mathrm{Step.rand}() \qquad (5)$$

4. Computational Results

To verify and evaluate the efficacy of the metaheuristic to solve the scheduling of multi-objective model linked to a SMSP. Molecular to large-scale problems with relatively uniform ranging from (5- 4000) jobs. We interact with the MSP, resulting in the evenly dispersed in with weights taken from the set has now become a common method for scheduling problems with due dates for a single machine. The are uniformly the is uniformly drawn from [P(1-TF-RDD/2),P(1-TF+RDD/2)]; where $P = \sum_{j=1}^{n} p_j$, the relative range of due dates (RDD) and the average tardiness factor (TF) affect this. The TF value is calculated from the values from 0.1 to 0.5, whereas the RDD value is calculated from 0.8 to 1.6. Here are the AFSA parameters: The fake fish population is 100, and it is positioned in the source of the water distribution network. The maximum number of iterations is 200, the visual range is 0.5, the number of tries is 5, and the congestion factor is 0.8. An investigation into the algorithm's performance indicated that the Matlab programming language translated and processed these data equally.

In Table 1, for instance, we present the completion times for AFSA. We observe that AFSA performs well across all work sizes. Its capacity to handle job counts of up to 4000 without losing significant performance demonstrates its efficiency and scalability.

Table 1: Data of cosine function.

EXN	**AFSA**	
	Av. of AFSA	**Av. of time**
5	1194	4.473
10	5563	5.384

15	8480	6.178
20	17545	7.246
25	18652	7.058
30	19355	8.220
40	213990	10.541
50	2441876	12.634
100	2610813	13.442
200	27411024	15.484
300	288464690	26.478
400	3011499852	27.649
500	31112672101	28.954
600	33423458970	29.488
700	36430898061	31.466
800	38901080256	34.675
900	41561071767	37.506
1000	431112501763	38.486
1100	461149952771	41.522
1200	47021071767	43.211
1500	512311250178	53.765
2000	549551098132	6.9543
4000	590234235394	8.9543

5. Conclusion

This paper presents a unique metaheuristic AFSA strategy for solving the multi-objective model of the single-machine scheduling problem. It is beneficial in some areas, such as the accuracy and the computational efficiency. The unique solution representation variety, search strategy adaptability, manageability of multi-objective optimization problems, and ability tojobs up to 4000 make AFSA stand out. However, there are some drawbacks in AFSA, such as high parameter tuning, high scalability, lack of convergence speed, and consumption of resources. For addressing these drawbacks and increasing the AFSA, its utilization and applicability in real-world scenarios can be made possible by developing automated and autonomous tuning techniques, highly scalable algorithms, utilization of dynamics in adaptations, and less resource consumption. Additionally, future research will focus on a multi-objective model for solving a single machine scheduling problem.

Acknowledgment

The authors thank the referees and editors for their constructive feedback, which helped improve the article.

References

[1] Ali, F.H., and Jawad, A.A., 2020, Minimizing the total completion time and total earliness time functions for a machine scheduling problem using local search methods, Iraqi Journal of Science., 2020, 126–133,.

[2] Smith, W.E., 1956, Various optimizers for single-stage production, Naval Research Logistics Quarterly, 3(1-2), 59–66,

[3] Rudek, ,R., 2017, The single machine total weighted completion time scheduling problem with the sum-of-processing time based models: strongly NP-hard," Applied Mathematical Modelling, 50, 565–575,

[4] Zhang, H., Yin, Y., and Cheng, Y., 2017, The single machine total weighted completion time scheduling problem with the sum-of-processing time based models: Strongly NP-hard, Appl. Math. Model., 50, 314–332,

[5] Györgyi, P., and Kis, T., 2019, Minimizing total weighted completion time on a single machine subject to non-renewable resource constraints," Journal of Scheduling, 22, 623–634,

[6] Chen, Y., Guan, Z., Tsai,Y.-C. and Chou, F.D., 2022, Single machine scheduling problem with a weight-modifying-activity to minimize the total weighted completion time," IEEE Access, 10, 45444–45456,

[7] Lin, B. M. T., Liu, S.W., and Mosheiov, G., 2024, Scheduling with a weight-modifying activity to minimize the total weighted completion time, Omega, 128, 103115,

[8] Potts, C.N., and Van Wassenhove, L.N., 1992, Approximation Algorithms for Scheduling a Single Machine to Minimize Total Late Work, Operations Research Letters.,11, 261–266.

[9] Mosheiov, G., Oron, D., and Shabtay, D., 2021.Minimizing total late work on a single machine with generalized due-dates, European Journal of Operational Research, 293, 837–846,

[10] Mor, B. and Shabtay, D., 2022, Single-machine scheduling with total late work and job rejection, Computers & Industrial Engineering, 169, 108168,

[11] Neamah, N.M. and Kalaf, B.A. 2024, Solving tri-criteria: total completion time, total late work, and maximum earliness by using exact, and heuristic methods on single machine scheduling problem, Iraqi Journal for Computer Science and Mathematics, 5, 14–25,

[12] Khraibet, T.J. Kalaf, B.A., and Mansoor, W., 2025, A new meta heuristic Algorithm for Solving multi-objective Single Machine Scheduling Problems, Journal of Intelligent Systems, 34, 20240373,

[13] Khraibet, T.J., Kalaf, B.A., and Jasim, A.A., 2024, A new approach for multiprocessor system-on-chip application scheduling using hybrid metaheuristic algorithms," *International Journal of Intelligent Engineering and Systems*, 17, 407–419.

[14] Gajpal, J.Li, Y. and Appadoo, S.S., 2020, Algorithms for A two-Agent Single Machine Scheduling Problem to Minimize Weighted Number of Tardy Jobs, Journal of Information and Optimization Sciences, 42, 785-811 ,

[15] Li, X.L., 2003, A new intelligent optimization—artificial fish swarm algorithm, Ph.D. dissertation, Zhejiang University, China, Jun.

[16] Li, L.X., Shao, Z.J., and Qian, J.X., 2002, An optimizing method based on autonomous animals: fish-swarm algorithm,"Systems Engineering Theory and Practice, 22, 32–38,

[17] Jiang, M.Y., and Yuan, D.F., 2006, Artificial fish swarm algorithm and its applications, in Proc. Int. Conf. on Sensing, Computing and Automation, 1782–1787,.

[18] Xiao, J.M., Zheng, X.M. and Wang, X.H., 2006, A modified artificial fish-swarm algorithm, in Proc. 6th World Congress on Intelligent Control and Automation (WCICA), Dalian, China, 3456–3460,

[19] Ali S.M., and Abdulqader, A.W., 2023.Diversity operators-based artificial fish swarm algorithm to solve flexible job shop scheduling problem, Baghdad Science Journal, 20, (Suppl.), 2067–2076.

[20] Zhang, H., Zhang,Y., Niu, Y., He, K., and Wang,Y., 2024, A grey wolf optimizer combined with artificial fish swarm algorithm for engineering design problems,"Ain Shams Engineering Journal, 15, 102797,

[21] Fathy, A., 2022, Enhanced artificial fish swarm algorithm for solving nonlinear optimization problems, Complex & Intelligent Systems, 8, 3903–3920,