University of Baghdad
College of Engineering
JE
JOURNAL OF ENGINEERING

Journal of Engineering

journal homepage: www.jcoeng.edu.iq

Volume 32 Number 4 April 2026

A New Artificial Intelligence Algorithm for Solving the Integrated Model of Aggregate Production Planning and Scheduling Problem

Karrar Emad Abd Al Sada *, **Bayda Atiya Kalaf**

Department of Mathematics, College of Education for Pure Science (Ibn Al Haitham), University of Baghdad, Baghdad, Iraq

ABSTRACT

In recent years, the integration of production planning and control scheduling has become a significant factor in complex and dynamic manufacturing environments. Therefore, this paper addresses this challenge by introducing an integrated model that simultaneously optimizes both aggregate production planning and single machine scheduling. The primary objectives of the developed model are to minimize overall production cost, makes pan, setup costs, and product delivery delays as key performance indicators. To solve the integrated model, two hybrid metaheuristic algorithms are applied with a novel hybridization ratio (40%-60%): the Hybrid Whale Algorithm with the Fruit Fly optimization algorithm and the Whale Algorithm with the Grey Wolves Optimization Algorithm. Large set of computational experiments are conducted on 10 instances with 20 to 200 jobs and 100 planning periods, with 30 independent runs. The results show that both hybrid algorithms are highly effective in finding near optimal solutions to the integrated complex problem. Comparative analysis demonstrates that the hybrid algorithms outperform common algorithms (Whale, Genetic, Fly, and grey wolf) and the dynamic programming method in terms of solution quality, convergence speed, and computational stability in most tested scenarios. These findings provide practitioners with robust decision-support tools to improve operational efficiency in modern manufacturing systems.

Keywords: Fly algorithm, Greywolf algorithm, Aggregate production planning, Scheduling, Genetic algorithm.

1. INTRODUCTION

Companies rely almost entirely on scheduling machines and planning and arranging production to achieve the highest level of accuracy **(Guzman et al., 2022)**. Production planning and scheduling are the cornerstones of any commercial project and are crucial for understanding the requirements necessary for its success. Additionally, they help increase efficiency, streamline the manufacturing process, and lower costs **(Mustajab, 2023)**. Aggregate production planning and scheduling establish a plan for the quantity of production and scheduling for a period ranging from three to eighteen months **(Dauzere-**

*Corresponding author
Peer review under the responsibility of University of Baghdad.
https://doi.org/10.31026/j.eng.2026.04.04

Article received: 19/11/2025
Article revised: 12/03/2026
Article accepted: 18/03/2026
Article published: 01/04/2026

Peres and Lasserre, 1994; Yu et al., 2022). So, during this period, the expected demand is fully met. It also helps determine the appropriate number of employees, expected orders, undisclosed agreements, product arrangement, storage of the remainder, and proper handling of it **(Armstrong, 2006)**. In addition, combining production planning and scheduling has received increasing and significant attention in the concentrated economic community. As a result, the issue of combining production planning and scheduling is receiving increasing attention in today's complex and changing economic climate **(Lohmer and Lasch, 2021; Ouelhadj and Petrovic, 2009)**. This resulted from an integrated model that inspired research into the topic **(Schwindt-Bayer and Mishler, 2005)**. **(Shobrys and White, 2002)** developed a comprehensive integrated production planning and scheduling model and solved it hierarchically, investigated issues in multi-period, multi-workshop production planning and scheduling, and devised a method to maximize both simultaneously using nonlinear mixed integer programming.

(Li and McMahon, 2007) employed integrated optimization in planning and scheduling can help you make better judgments about how to execute operations and proposed a new method for breaking things down called two-level optimization. **(Wang et al., 2017)** developed a two-level integrated model for production planning and scheduling using the State-Task Network representation. **(Vogel et al., 2016)** proposed a hierarchical and integrated system that combined aggregate production planning and master production scheduling. Computational testing shows that the integrated model is solvable and outperforms the existing hierarchical technique in all cases **(Černe et al., 2019). (Hassani et al., 2018)** created a novel model that combines planning and scheduling to account for the fact that the capacity limitation did not accurately represent the number of resources available. They also considered the constraint of resource availability. The goal of this model was to reduce the total cost of a single-level workshop **(Yao et al., 2022)**.

Multi-objective problems are significant because they depict real-world situations where decision-makers must balance various performance metrics to achieve optimal outcomes **(Zheng et al., 2019; Salih et al., 2020; Khalaf et al., 2021; Shir and Emmerich, 2024; Khraibet et al., 2025)**.

Furthermore, they widen core difficulties by incorporating several, often conflicting, goals. This increases operational efficiency and meets a wide range of stakeholder expectations, making it an important area of study **(Olkkonen, 2015; Mechaacha et al., 2025)**. As a result, the scientific community strives to address difficult issues by employing metaheuristic algorithms **(Tomar et al., 2024; Turgut et al., 2023)**. While metaheuristic algorithms are effective for tackling complex real-world problems, the no-free-lunch theorem states that there is no algorithm to solve all problems **(Joyce and Herrmann, 2018)**. As a result, modern concepts such as self-adaptive algorithm modification or hybrid algorithms enable the selection of a suitable algorithm to overcome the implicit limitation of metaheuristics. In addition, hybrid approaches combining *Whale Optimization* Algorithm with Firefly Algorithm **(Tian et al., 2024)** and WOA with Grey Wolf Optimizer Algorithm **(Korashy et al., 2019)** have been successfully applied in various optimization problems.

Therefore, in this study, we proposed two new hybrid algorithms by combining Whale Optimization with Grey Wolf Optimizer and Whale Optimization with Fruit Fly Optimization to solve the new integrated model of aggregate production planning and scheduling problem, as shown in **Fig. 1.**

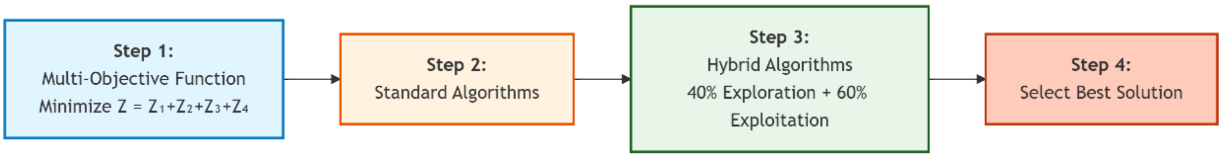

Figure 1. Workflow of the proposed methodology

2. MATHEMATICAL MODEL

The mathematical model of multi-objective optimization for an integrated model for aggregate production planning and single machine scheduling problem (MOIASM) is formulated in this section to minimize four conflicting objectives: total weighted completion time, maximum tardiness, range of lateness, and total workforce changes**.** We presumed that industrial manufacturing. The company produces n types of products to satisfy market demands over a planning time horizon T.

2.1 Notation

The notations utilized in the formulation of MOIASM are as follows:
$\sum w_j c_j$: minimize the total weighted completion time
C_{max} : Maximum completion time
c_j : The completion time of the job j
d_j : Due to the data of the job j
F_t : Workers fired for a period t
H_t : Workers hired for a period t
i : Quantity of production $\quad i = 1,2, \dots, n$
L_{max}: Maximum lateness
L_{min} : Minimum lateness
p_j : processing time of the job j
R_L : Range of lateness
T_y : The tardiness of the job y
T_{max}: Maximum Tardines
t : Number of periods in the planning $\quad t = 1,2, \dots, t$
w_j : weight of the job j

2.2 Objection Function

Multi-objective approaches help achieve more than one objective when choosing the appropriate course of action. In this study, four objectives were addressed simultaneously: cost, makespan, setup costs, product delivery delays, and workforce reduction.

$$Z = \sum_{j=1}^{n} w_j c_j + T_{max} + R_L + \sum_{j=1}^{n} H_j + F_j \tag{1}$$

$$minZ_1 = \sum_{j=1}^{n} w_j c_j \tag{2}$$

minimize the total weighted completion time, Jobs are sequenced in non-decreasing order $of \frac{p_j}{w_j}$ (Shortest Weighted Processing Time rule) to minimize total weighted completion time.

Subject to:

$c_\gamma(1) \geq p_\gamma(1)$
$c_\gamma(j) = c_\gamma(j-1) + p_\gamma(j) \qquad j = 2, \dots, n$
$w_\gamma(j) \geq 0 \qquad j = 1,2, \dots, n$
$$minZ_2 = T_{max} \tag{3}$$

This function minimizes the maximum tardiness.
For the maximum tardiness objective (Z_2), jobs are sequenced according to the Earliest Due Date (EDD) rule, where jobs with earlier due dates are processed first.
Subject to:

$c_\gamma(1) \geq p_\gamma(1)$
$c_\gamma(j) = c_\gamma(j-1) + p_\gamma(j) \qquad j = 1,2, \dots, n$
$T_\gamma(j) \geq c_\gamma(j) - d_\gamma(j) \qquad j = 1,2, \dots, n$
$T_\gamma(j) \geq 0$
$$minZ_3 = R_L = L_{max} - L_{minx} \tag{4}$$

This function minimizes the range of Latenesy
$c_\gamma(1) \geq p_\gamma(1) \qquad j = 1,2, \dots, n$
$c_\gamma(j) = c_\gamma(j-1) + p_\gamma(j) \qquad j = 2, \dots, n$
$T_\gamma(j) = c_\gamma(j) - d_\gamma(j) \qquad j = 1,2, \dots, n$
$R_L = L_{max} - L_{min}$
$$minZ_4 = \sum_{j=1}^{n} H_j + F_j \tag{5}$$

This function minimizes total workforce changes,
$W_{t-1} + H_t - F_t = W_t$
We can write $W_t - W_{t-1} - H_t + F_t = 0$
If $t = 1$ $W_1 - W_0 - H_1 + F_t = 0, \qquad W_0$ →Is initial worker

2.3 Normalization of Objection Function

The units of measurement used by the four objective functions are different, i.e., time in terms of hours for Z1, Z2, and Z3, and the number of employees for Z4. The direct application of the weighted sum method to the objectives might allow the objectives with the largest values to influence the optimization process, irrespective of the values of the weights. To overcome the aforementioned difficulty, the Min-Max normalization method is used to normalize all the objectives to a common interval, i.e., [0, 1]. The normalized value of each objective can be calculated as follows:
$Z_{i,norm} = (Z_i - Z_{i,min}) / (Z_{i,max} - Z_{i,min})$
Here, Zi denotes the original value of the objective; $Z_{i,min}$ denotes the ideal point of the objective, and $Z_{i,min}$ denotes the nadir point of the objective. The ideal and nadir points can be calculated by generating a payoff table, where each objective function is optimized separately, and the values of all the objectives at the optimal points are determined. The minimum values of the objectives correspond to the ideal points, and the maximum values correspond to the nadir points. After normalizing the values of the objectives, the overall objective function can be formulated as follows:

Minimize $F = \propto_1 Z_{1,\text{norm}} + \propto_2 Z_{2,\text{norm}} + \propto_3 Z_{3,\text{norm}} + \propto_4 Z_{4,\text{norm}}$
subject to the constraints $\sum \propto_i = 1$ and $\propto_i \geq 0$

2.4 Special Case Analysis for Problem

This subsection analyzes several special cases of the integrated production planning and scheduling problem. Each case represents a specific combination of job parameters (processing times, due dates, weights) that leads to a simplified scheduling rule. Understanding these special cases provides insights into the behavior of the problem and validates the mathematical formulation.

Case 1: If $p_j = p_i$, $d_i = d_i$ and $w_j = 0$, $\forall j \in N$. Then these conditions give SPT rule is an optimal solution for the problem: $\sum w_j c_j + T_{max} + R_L$
Proof: If $c_j + p_j > d_j$ $(\, i.e\ jop_j\ is\ late)$ and $w_j = 0$

$$\sum w_j c_j + T_{max} + R_L = \sum c_j + T_{max} + R_L = c_j + p_j - d_j + c_j + p_j$$
$$= c_j + p_j - d_j + c_j + p_j$$
$$= 2c_j + 2p_j - d_j$$

Since i is the late then j is late so

$$\sum c_j + T_{max} + R_L = c_j + p_i + p_j - d_j + c_j + p_i + p_j$$
$$= 2c_j + 2p_j + 2p_i - d_j$$

From 1 and from 2 we get

$$4c_j + 4p_i + 2p_j - 2d_j$$

Thus SPT rule gives an optimal solution

Case 2: if $w_j = w\ and\ p_j = p\ and\ d_j = w + p_j$ $\forall j \in N$
then $WSPT$ rule gives the optimal solution for the problem
proof: then $c_1 = p_1$, $c_j = max\{c_j - 1\} + p_j$
$\forall j \in N\ then$ $c_j \geq d_j$, j is either
Late or just in time (JIT) that is $RL = 0$.
The reduction to problem $1//\sum w_j c_j + T_{max}$.
Then $WSPT$ rule provides the optimal solution.

Case 3: if $d_j = d$ and $w_j = 0$ then $\gamma = SPT$ rule gives optimal solution for the problem.
Proof: since $d_j = d\ and\ w_j = 0$ $\forall j \in N$
Then $L_{max}(\gamma) = C_{max}(\gamma) - d$ and
$L_{min}(\gamma) = P_\gamma(1) - d$ then
$R_L = L_{max}(\gamma) - L_{min}(\gamma)$
$= C_{max}(\gamma) - P_\gamma(1)$, if $d < C_{max}(\gamma)$ then $T_{max}(\gamma) = C_{max} - d$ which is
$\sum c_j(\gamma) + T_{max}(\gamma) + R_L(\gamma) = \sum c_j(\gamma) + C_{max}(\gamma) - P_\gamma(1) + C_{max}(\gamma) - d$
Then SPT rule will give an optimal solution.
Case 4: if $w_j = 0$ then SPT rule is an optimal solution for the problem if $c_j > d_j$ $\forall j \in N$.
Proof: let π be a schedule orderly according to SPT rule such that $c_j > d_j$,$j = 1,2 \ldots\ldots, n$

Then for each job_j $(j \in \pi)$, j is late job_j and

$$min \sum c_j + T_{max} + R_L = min 2 \sum c_j - \sum d_j$$

Then $min 2 \sum c_j - d_j$ depend on $\sum c_j$

So π is the optimal solution for the problem.

Case 5: if $\pi = EDD$ sequences give an optimal solution for the problem if

$$c_\pi(j) \le d_\pi(j) \ \ \forall j \in N.$$

Proof: if $c_\pi(j) \le d_\pi(j)$ $\forall j \in N$, j is either late or (JIT) that is $T_{max}(\pi) = 0$
$L_{max}(\pi) \le 0$ and $L_{min}(\pi)$

Then $R_L(\pi) = L_{max}(\pi) - L_{min}(\pi)$

The minimum R_L for the optimal sequences.

Since $c_\pi(j) \le d_\pi(j), \forall j \in N$ with $T_{max}(\pi) = R_L(\pi) = 0$

Then π is gives an optimal solution for the problem.

3. METHODOLOGY

This section introduces the optimization Algorithms considered in this study. To establish context, we first review several well-known algorithms: the Whale Optimization Algorithm (WOA), the Grey Wolf Optimizer (GWO), the Fruit Fly Optimization Algorithm (FOA), Dynamic Programming (DP), and the Genetic Algorithm (GA). These algorithms have been applied widely in engineering, operations research, and artificial intelligence. After this overview, two proposed hybrid algorithms are described in detail: WOAGWO and WOAFOA, which form the central contribution of this research.

3.1 Whale Optimization Algorithm

The Whale Optimization Algorithm was introduced by **(Mirjalili and Lewis, 2016)**, is inspired by the bubble-net feeding strategy of humpback whales. In nature, whales circle their prey in a spiral motion while moving gradually closer. In WOA, this behaviour is translated into two mechanisms for updating solutions: spiral convergence toward the best candidate (exploitation) and random movement away from it (exploration). The position update equations are:

Encircling Behavior: $D = | \ C \cdot X^* - X \ |, X(t+1) = X^* - A \cdot D$

Spiral Behavior: $X(t+1) = D \cdot e^{bl} \cdot \cos(2\pi l) + X^*$

Where b defines the spiral shape, $l \in [-1,1]$, random number, and $A = 2a \cdot r - a$, $C = 2r$, $r \in [0,1]$. The algorithm has been used for structural design, scheduling, training neural networks, and managing energy. Its strength is that it can balance global search, which often leads to results that are better than those of other metaheuristics **(Dhiman and Kumar, 2019; Mohammadzadeh et al., 2024).** However, it suffers from slow convergence speed and poor local exploitation in later iterations, often getting trapped in local optima for complex multimodal problems **(Suman and Udmale, 2026).**

3.2 Grey Wolf Optimizer

The first grey wolf optimizer was developed by Mirjalili as an optimization algorithm that emulates grey wolves' hunting behavior. For example, the grey wolf community is divided into a hierarchical structure with three main roles: alpha (α), beta (β), and delta (δ) wolves, while the whole pack converges to it, and omega is the rest of the pack converging to the leader. Such a structure, as well as hunting behavior, such as predating a scared and slowly-accelerating animal by encircling it from different directions and converging slowly, permits the achievement of a balance between exploring and exploiting. GWO has been successfully used in control system design, image processing, and power system optimization. It is valued for its simplicity, efficiency, and ability to avoid premature convergence **(Mirjalili et al., 2014; Saremi et al., 2017)**.

3.3 Fruit Fly Optimization Algorithm

The Fruit Fly Optimization Algorithm was proposed by Pan in and is inspired by the exploitation method of fly insects that follow two phases during a search. These tiny animals can lead search targets utilizing their sense of smell and then have a rapid vision target refinement. The algorithm operates in a dual phase: smell-based exploration in the first phase and sight-based exploitation in the second phase. FOA has been widely used in various tasks such as function optimization, pattern recognition, and clustering. Variants inspired by FOA have made the algorithm useful in industrial and engineering applications due to increased convergence accuracy **(Bekdaş et al., 2015; Pan, 2012).**

3.4 Dynamic Programming

Dynamic Programming was formulated by Bellman in 1950 as a highly deterministic approach to solving complex problems. It is based on breaking down the problems into overlapping subproblems that are solved once and their solutions are reused, which minimizes computational redundancy. DP operates under the principle of optimality, which posits that the optimal solution of the entire problem depends on the optimal solutions of its subproblems. It finds widespread applications in operations research, economics, engineering, and artificial intelligence. The chief advantage of DP is the ability to guarantee accurate optimal solutions. On the other hand, as the problem size increases, DP inevitably suffers the curse of dimensionality **(Bertsekas, 2012; Bellman, 1957)**.

3.5 Genetic Algorithm

One of the earliest and most impactful evolutionary algorithms, the Genetic Algorithm, was introduced by Holland in the 1970s. Based on natural selection and genetic principles, the GA defines potential solutions in the form of chromosomes. The three fundamental operators include selection, which determines the solutions of higher fitness, crossover, and mutation, which offer random changes to retain diversity. The GA has been relatively engaged in scheduling, optimization of engineering systems, as well as machine learning and artificial intelligence. Its advantage is the balance between exploration and exploitation, rendering the GA robust to address nonlinear, multi-modal, and large-scale tasks. Although in some instances, GA demonstrates slow convergence, multiple modifications and hybrids have increased the GA efficiency and scalability **(Moon et al., 2008; Duan et al., 2025; Chao et al., 2026).**

3.6 Proposed Hybrid: Whale and Grey Wolf Algorithms (WOAGWO)

As a hybrid algorithm, the first is WOAGWO integration. This combines WOA's potential for superior searching capabilities in the exploration stage: using the first 60% of iterations to search intensely. In the remaining 40% of iterations, the position update follows the GWO, demonstrating WOA's ability to discover with a leader following the wolves to ensure that the same has been optimized effectively. This model is ideal for decreasing the local-optimum effect because it explores for 60% of the iterations, thus being less sensitive to local optimum, and the remaining 40% is exploitative, following a guiding wolf. The following steps illustrate how the algorithm operates:

Step 1: Generate Initial Population: A set of initial populations, denoted by N, is randomly generated within the given problem constraints.

Step 2: Execute WOA– Exploration Phase (60% of iterations): Perform global search and cover large areas of the solution space. the total number of iterations WOA will run for 60%.

- Encircling prey: Whales update position towards the best-known solution.
- Spiral update: Simulates the bubble-net attacking mechanism.
- WOA Position Update Rules:

Encircling behavior: $D = | C \cdot X^* - X |, X(t+1) = X^* - A \cdot D$
where: X^*: best solution, $A = 2a \cdot r - a$, decreases over iterations, $C = 2r, r \in [0,1]$
Spiral update: $X(t+1) = D \cdot e^{bl} \cdot \cos(2\pi l) + X^*$
where: $l \in [-1,1]$, random number, b: constant to define spiral shape

- Random Switch: With 60% probability, apply either encircling or spiral update.
- After WOA Phase: Select the best agent X^*_{WOA} to pass to the next phase.

Step 3: Handover Mechanism: All the population is transferred from the WOA to the GWO without any modification, and the best solution is maintained as the alpha wolf.

Step 4: Execute GWO—Exploitation Phase (40% of iterations): Run for 40 % of the total number of iterations.
GWO Mechanism: Maintain the best three solutions: α: best, β: second-best, δ: third-best

- Position Update Equation:

For each agent X_i, update using:
$D_\alpha = | C_1 \cdot X_\alpha - X_i |, X_1 = X_\alpha - A_1 \cdot D_\alpha$
$D_\beta = | C_2 \cdot X_\beta - X_i |, X_2 = X_\beta - A_2 \cdot D_\beta$
$D_\delta = | C_3 \cdot X_\delta - X_i |, X_3 = X_\delta - A_3 \cdot D_\delta$
Then: $X_i(t+1) = \frac{X_1 + X_2 + X_3}{3}$
Control Parameters: $A = 2a \cdot r - a$, decreases linearly, $C = 2r, r \in [0,1]$

Step 5: Compare Best Solutions: The best solution obtained from the WOA and GWO is compared, and the best solution is maintained.

Step 6: Return Final Solution: The best solution obtained is printed as the final solution obtained by the WOGWO. **Fig. 2** illustrate the flowchart of the proposed hybrid Algorithm (WOAGWO).

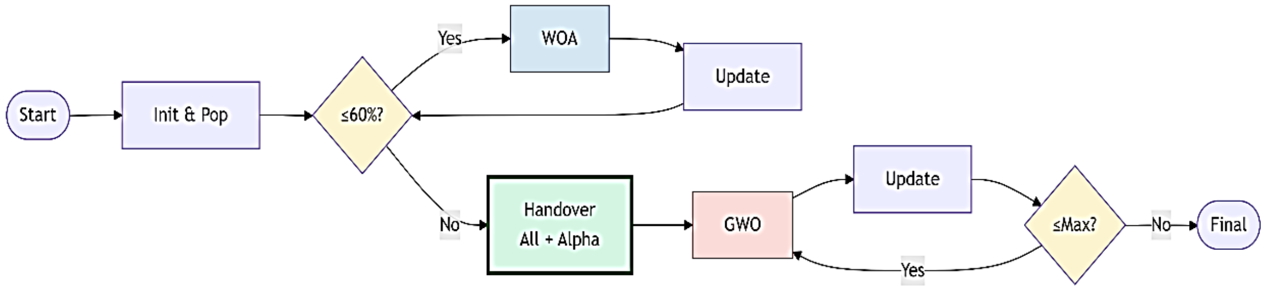

Figure 2. Flowchart of the WOAGWO

The hybrid algorithm used 60% and 40% of the total iterations for the WOA and GWO algorithms, respectively. These percentages were obtained through extensive calibration experiments. Different ratios of exploration and exploitation were considered for the integrated production planning and scheduling problem, i.e., 80%–20%, 70%–30%, 60%–40%, 50%–50%, and 40%–60%. These ratios were compared in terms of the quality of the solution obtained, rate of convergence, and stability of the solutions obtained over 30 independent runs for each ratio. The results obtained in the calibration experiments are presented in **Table 1.**

Table 1. Calibration results for different exploration-exploitation splits

Split (WOA: GWO)	Average Objective Value	Standard Deviation	Convergence Behavior
80%:20%	0.58	0.12	Fast initial, premature convergence
70%:30%	0.52	0.09	Good
60%:40%	0.47	0.06	Balanced, best overall
50%:50%	0.51	0.08	Moderate
40%:60%	0.55	0.10	Slow convergence

From the results presented in **Table 1**, it can be observed that increasing the exploration phase beyond 60% does not yield any significant results, and increasing the exploration phase beyond 40% does not help in refining the promising regions obtained during the exploration phase for the given problem class. Hence, the optimal ratio for the integrated production planning and scheduling problem is 60% and 40% for the WOA and GWO algorithms, respectively.

3.7 Proposed Hybrid: Whale and Fruit Fly Optimization Algorithm (WOAFOA)

The second hybrid algorithm is referred to as WOAFOA and combines WOA and FOA. The idea is to start with WOA's weak spiral movement to conduct early broad exploration, followed by FOA's much stronger smell-and-sight stage to work in more depth with the obtained data. The motive of such a design is to counteract WOA's lack of local optimization efficiency and basal FOA's tendency to converge rather quickly. More specifically, the approach is divided into three stages: an initial untargeted exploration by WOA, a targeted or exploitative stage by FOA, and the "fine-tuning" stage, which correctly perturbs solutions with minimal baseline perturbations in the correct direction. The resulting algorithm possessed increased domain adaptability, which allowed for the utilization of the same framework for multiple diverse problems. The result was an increase in overall robustness and accuracy of the algorithm as opposed to its single-constituent relatives. The numerous applications, such as design engineering or general artificial intelligence problems, have been reported to date.

- **Start With Random Solutions**

A set of possible solutions is randomly created within the limits of the problem.

- **Run WOA for the 60% of iterations**

WOA explores the search space using mechanisms inspired by whales' bubble-net hunting strategy. It focuses on diversification, searching widely to avoid premature convergence. WOA is run for almost 60% of the total number of iterations. It is based on humpback whales' bubble- net hunting strategy, and comprises two main behaviors:
Encircling the Prey: Updating the position of each whale by:

- **Encircling Behavior**: $D = | C \cdot X^* - X |, X(t+1) = X^* - A \cdot D$
- **Spiral Behavior**: $X(t+1) = D \cdot e^{bl} \cdot cos\ (2\pi l) + X^*$

Where b defines the spiral shape, $l \in [-1,1]$*, random number, and* $A = 2a \cdot r - a$, $C = 2r$, $r \in [0,1]$

- With 60% probability, apply either encircling or spiral update.

Update the population and store the best solution X^*_{WOA}.

- **Switch to FOA for the 40%**

Once WOA finishes, FOA takes over. It searches locally around promising areas using a "smell-based" behaviour, refining what WOA discovered.
To intensify the search for the best solution, WOA FOA will be running for the remaining 40% of total iterations as:

1. Take the best solution from WOA.
2. For each new candidate: $X_{new} = s + rand \cdot (ub - lb) \cdot search\_radius$
3. Clip the new solution within bounds: $X_{new} = max\ (min\ (X_{new}, ub), lb)$
4. Evaluate fitness, keep the best nagents.
5. Repeat this process for the specified number of FOA iterations.

- **Pick the Best Result**

After both phases, the best solution, whether from WOA or FOA, is selected based on the objective value.

- **Output the Result**

The chosen solution is returned as the final output of the hybrid algorithm.
Fig. 3 illustrates the flowchart of the proposed hybrid Algorithm (WOAFOA).

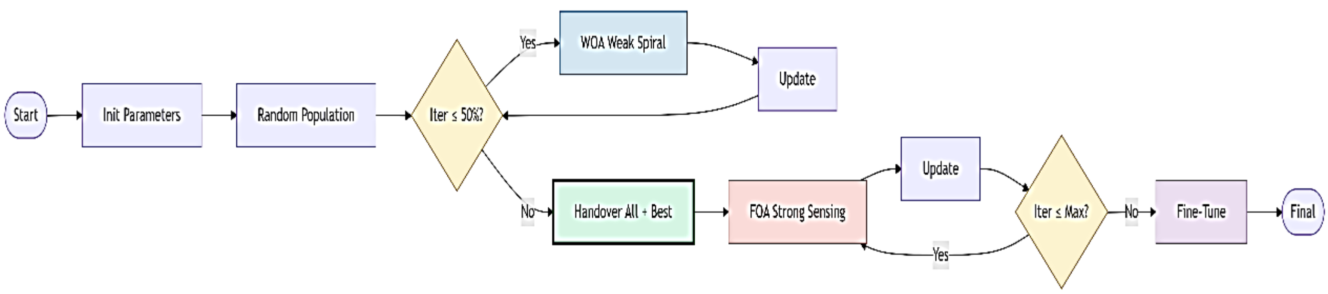

Figure 3. Flowchart of the WOAFOA

3.8 Connection Between Algorithms and the Objective Function

All algorithms are designed to minimize the integrated objective function ($Z = Z_1 + Z_2 + Z_3 + Z_4$) as discussed in Section 3. At each step of the optimization process, all solutions, i.e., production schedules, are evaluated by calculating their objective function value. This value acts as the fitness function for the optimization process, where solutions having better fitness, i.e., smaller objective function values, are considered better and affect the update of other solutions. Therefore, the objective function acts as the feedback function for the optimization process.

4. RESULTS AND DISCUSSION

Simulation is used to investigate the effectiveness of the suggested two methods in solving the multi-objective integrated model of aggregate production planning and scheduling problem. Experiments were conducted on 10 instances with n = 20 to 200 jobs, T = 100 periods. The algorithms have been implemented using MATLAB R2022a, running on an Intel Core i7-10700K processor with a CPU speed of 3.80 GHz, along with 32 GB of RAM, using a Windows 10 Pro operating system with 1000 iteration. The combined algorithms show promising results in both quality and execution time. Regarding the objective value, the ability of the combined algorithms to find a high-quality solution is clear. They can generate high-quality solutions or solutions at least no worse than the best solutions of traditional algorithms. This is particularly apparent in small and medium cases, where the difference from an optimal value is minimal, as proven by the DP index. Execution time is represented by comparing the average runtime of implemented algorithms. WOAGWO and WOAFOA have outstanding performance, taking tens of times to execute better than basic algorithms. This implies that the combination finds an easy way to search, converging to an optimal solution. Therefore, the results of **Table 2** indicate that the hybrid algorithms WOAGWO and WOAFOA achieved superior performance compared to the individual algorithms (GA, GWO, FOA, WOA), and it can be said that the hybrid algorithms offer a tradeoff between computational accuracy and rapidity, which makes them more applicable in practice, especially for large-scale problems or in cases when a result is needed as soon as possible. Thereby, such a solution is promising as it promotes the development of mercury intelligence methods through combinations with Whale algorithms. The objective value of the hybrid algorithms (WOAGWO and WOAFOA) is significantly lower than that of WOA, GA, GWO, and FOA for all n. The performance of WOAGWO and WOAFOA is always very similar. Examples: At n = 125: WOA = 544454, GA = 513892, GWO = 509844, FOA = 1073327, WOAFOA = 443045, WOAGWO= 419374. We observe that the hybrid yields improved and more stable results compared to all individual algorithms. Often: WOAGWO time ≈ 0.0164399 seconds. Individual algorithm time: GA sometimes reaches 0.03948 seconds, GWO up to 0.03732 seconds, WOA approximately 0.1287328 seconds. This indicates that the combination does not add significant time complexity, but rather improves performance without additional computational cost. In addition, the hybrid algorithms (WOAFOA and WOAGWO) outperformed all individual algorithms across all test problems, achieving average improvements of 22.8% and 26.6%, respectively. The WOAGWO algorithm demonstrated the best overall performance.

Table 2. The outcome of DP, GA, GWO, FOA, WOA, and WOAGWO and WOAFOA depend on MSE.

n		DP	WOA	GA	GWO	FOA	WOAFOA	WOAGWO
20	**obj**	772	89757	90048	87445	88488	89548	89130
	Time	0.0025	0.1477	0.1789	0.17752	0.12033	0.04972	0.05775
30	**obj**	905	133644	133965	132014	137446	134395	133654
	Time	0.0003	0.1227	0.2214	0.1801	0.159	0.0538	0.0658
40	**obj**	885	170330	172070	170399	181247	170550	170592
	Time	0.0001	0.1161	0.2294	0.1786	0.1744	0.0565	0.0727
50	**obj**	1053	212840	219289	211792	239597	212781	212666
	Time	0.0001	0.1187	0.2691	0.1879	0.1823	0.0677	0.0702
75	**obj**	1154	313540	324466	313067	404234	313870	313495
	Time	0.0002	0.1485	0.3407	0.206	0.1618	0.0647	0.0864
100	**obj**	1293	411390	430121	411013	722519	411005	411045
	Time	0.0002	0.146	0.461	0.2506	0.2038	0.0831	0.1191
125	**obj**	1256	510415	537744	509678	1054896	510363	510396
	Time	0.0004	0.1514	0.5432	0.2636	0.2292	0.0824	0.1043
150	**obj**	1304	600908	655343	600854	1747434	600914	600660
	Time	0.0004	0.1812	0.6784	0.2953	0.2494	0.0928	0.1186
175	**obj**	1367	694914	800466	695332	2351138	694660	694912
	Time	0.0004	0.1604	0.8096	0.3194	0.2732	0.1056	0.1311
200	**obj**	1320	797864	903721	800875	3219646	797376	797490
	Time	0.0007	0.217	0.9569	0.3799	0.2969	0.1144	0.155

5. CONCLUSIONS

This study aimed to design and formulated an innovative combined model, which covers and connects aggregate production planning and single-machine scheduling to reduce overall production expenses, makes pan, setup, and time delays in product delivery. To address and handle such an intricate model, new hybrid metaheuristic algorithms inspired by nature, termed Whale Optimization and Grey Wolves Optimization Algorithm and Whale Optimization and Fly Optimization Algorithm, were developed. The overall experimental outcomes confirmed and testified to the efficiency of the proposed approaches in reaching the target values. The benchmark tests conducted by comparing the proposed method with traditional metaheuristics, such as Genetic, FLY, GWO, and WOA, and with dynamic programming approaches, confirmed and supported that hybrid approaches possess high convergence robustness and optimality. To be more specific, it was ascertained that Whale Optimization and Grey Wolves Optimization Algorithm (WOAGWO) overperformed all approaches by demonstrating high potential in overcoming local optimum points and by achieving optimal values with high average performance. However, the limitation of the current work is the fixed 60%-40% ratio between exploration and exploitation phases. In order to overcome the limitation of the current work, in the future, adaptive exploration-exploitation strategies will be used where the decision regarding the switch between exploration and exploitation will be made based on real-time feedback from the search process. Finally, it is highly recommended to apply the WOAGWO algorithm, which is considered an effective decision-making assistant method, in production planning and scheduling systems.

Credit Authorship Contribution Statement

Karrar Emad Abd Al Sada: Conceptualization, Methodology, Formal analysis, Investigation, Writing – original draft, Visualization, Writing – review & editing. Bayda Atiya Kalaf**.** Conceptualization, Methodology, Writing – review & editing, Project administration, Supervision.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

REFERENCES

Armstrong, M., 2006. *A Handbook of Management Techniques: A Comprehensive Guide to Achieving Managerial Excellence & Improved Decision Making (3rd ed.)*. Kogan Page. ISBN 07494 47664.

Bekdaş, G., Nigdeli, S. M. and Yang, X.-S. 2015. Sizing optimization of truss structures using flower pollination algorithm. *Applied Soft Computing*, 37, pp. 322–331. https://doi.org/10.1016/j.asoc.2015.08.037.

Bellman, R. 1957. *Dynamic Programming (1st ed.)*. Princeton University Press. ISBN:9780691146683.

Bertsekas, D. P. 2012. *Dynamic Programming and Optimal Control (4th ed., Vol. 2)*. Athena Scientific. ISBNs**:** 1-886529-43-4.

Černe, B., Duhovnik, J. and Tavčar, J. 2019. Semi-analytical flash temperature model for thermoplastic polymer spur gears with consideration of linear thermo-mechanical material characteristics. *Journal of Computational Design and Engineering*, 6(4), pp. 617–628. https://doi.org/10.1016/j.jcde.2019.03.001

Chao, Y., Zhuang, C., Guo, H. and Liu, J., 2026. A genetic programming hyper-heuristic with whale optimization algorithm for the dynamic resource-constrained multi-project scheduling problems. *Expert Systems with Applications*, *295*, P. 128881. https://doi.org/10.1016/j.eswa.2025.128881

Dauzere-Peres, S., and Lasserre, J.B. 1994. *An Integrated Approach in Production Planning and Scheduling (1st ed. 1994, Vol. 411)*. Springer-Verlag Berlin and Heidelberg GmbH & Co. K. ISBNs**:** 9783642468049

Dhiman, G. and Kumar, V. 2019. Seagull optimization algorithm: Theory and its applications for large-scale industrial engineering problems. *Knowledge-Based Systems*, 165, pp. 169–196. https://doi.org/10.1016/j.knosys.2018.11.024

Duan, Y., Li, M., Shi, L., Li, L., Zhao, X. and He, L., 2025. A genetic-operator-integrated whale optimization algorithm for solving the flexible job shop scheduling problem. *Journal of King Saud University Computer and Information Sciences*, *37*(9), pp. 1-25. https://doi.org/10.1007/s44443-025-00283-0

Guzman, E., Andres, B. and Poler, R. 2022. Models and algorithms for production planning, scheduling and sequencing problems: A holistic framework and a systematic review. *Journal of Industrial Information Integration*, 27, P. 100287. https://doi.org/10.1016/j.jii.2021.100287

Hassani, Z.I.M., El Barkany, A., Jabri, A., El Abbassi, I. and Darcherif, A.M. 2018. New approach to integrate planning and scheduling of production system. *International Journal of Engineering Research in Africa*, 39, pp. 156–169. https://doi.org/10.4028/www.scientific.net/JERA.39.156

Joyce, T. and Herrmann, J. M. 2018. *A Review of No Free Lunch Theorems, and Their Implications for Metaheuristic Optimisation.* In: Yang, XS. (eds) Nature-Inspired Algorithms and Applied Optimization. Studies in Computational Intelligence, 744. Springer, Cham. https://doi.org/10.1007/978-3-319-67669-2_2

Khalaf, B.A., Khalaf, W.S. and Mansor, H.K. 2021. An integrated model for solving production planning and capacity problems. *Periodicals of Engineering and Natural Sciences,* 9(3), pp. 715–724. https://doi.org/10.21533/pen.v9i3.2283

Khraibet, T.J., Kalaf, B.A. and Rehman, E. 2025. Solving multi-objective machine scheduling problem using the Meerkat Clan Algorithm. *Journal of Economics and Administrative Sciences*, 31(147), pp. 158–169. https://doi.org/10.33095/0ss7v861

Korashy, A., Kamel, S., Jurado, F., and Youssef, A. R. 2019. Hybrid whale optimization algorithm and grey wolf optimizer algorithm for optimal coordination of direction overcurrent relays. *Electric Power Components and Systems*, *47*(6-7), pp. 644-658. https://doi.org/10.1080/15325008.2019.1602687

Li, W. D. and McMahon, C. A. 2007. A simulated annealing-based optimization approach for integrated process planning and scheduling. *International Journal of Computer Integrated Manufacturing*, 20(1), pp. 80 -95. https://doi.org/10.1080/09511920600667366.

Lohmer, J. and Lasch, R. 2021. Production planning and scheduling in multi-factory production networks: a systematic literature review. *International Journal of Production Research*, 59(7), pp. 2028–2054. https://doi.org/10.1080/00207543.2020.1797207.

Mechaacha, A., Belkaid, F. and Brahimi, N., 2025. Multi-objective multi-product process planning with reconfigurable machines: Exact and metaheuristic approaches. *Expert Systems with Applications*, p.129591. https://doi.org/10.1016/j.eswa.2025.129591

Mirjalili, S. and Lewis, A. 2016. The whale optimization algorithm. *Advances in Engineering Software*, 95, pp. 51–67. https://doi.org/10.1016/j.advengsoft.2016.01.008

Mirjalili, S., Mirjalili, S. M. and Lewis, A. 2014. Grey wolf optimizer. *Advances in Engineering Software*, 69, pp. 46–61. https://doi.org/10.1016/j.advengsoft.2013.12.007

Moon, S., Lee, S. and Bae, H. 2008. Genetic algorithms for job shop scheduling problems. *International Journal of Production Research*, 46(10), pp. 2695–2705. https://doi.org/10.1080/00207540701244820

Mustajab, D. 2023. Effective production planning and scheduling: Literature review. *Vifada Management and Social Sciences*, *1*(2), pp. 54–72. https://doi.org/10.70184/h9wy6f46

Mohammadzadeh, A., Chhabra, A., Mirjalili, S. and Faraji, A., 2024. Use of whale optimization algorithm and its variants for cloud task scheduling: A review. *Handbook of Whale Optimization Algorithm*, pp.47-68.

Olkkonen, L. 2015. *Stakeholder expectations: Conceptual foundations and empirical analysis*. University of Jyväskylä. ISBN: 978-951-39-6386-6.

Ouelhadj, D. and Petrovic, S. 2009. A survey of dynamic scheduling in manufacturing systems. *Journal of Scheduling*, 12(4), pp. 417–431. https://doi.org/10.1007/s10951-008-0090-8

Pan, W.-T. 2012. A new Fruit Fly Optimization algorithm: Taking the financial distress model as an example. *Knowledge-Based Systems*, 26, pp. 69–74. https://doi.org/10.1016/j.knosys.2011.07.001

Salih, M.A., Mahmoud, A.H. and Khalaf, B.A. 2020. Six Sigma-supported expert system in construction projects. *IOP Conference Series: Materials Science and Engineering*, 881(1), 012168. https://doi.org/10.1088/1757-899X/881/1/012168

Saremi, S., Mirjalili, S. and Lewis, A. 2017. Grasshopper optimisation algorithm: theory and application. *Advances in Engineering Software*, 105, pp. 30–47. https://doi.org/10.1016/j.advengsoft.2017.01.004

Schwindt-Bayer, L. A. and Mishler, W. 2005. An integrated model of women's representation. *The Journal of Politics*, 67(2), pp. 407–428. https://doi.org/10.1111/j.1468-2508.2005.00323.x

Shir, O.M. and Emmerich, M., 2024. Multi-objective mixed-integer quadratic models: A study on mathematical programming and evolutionary computation. *IEEE Transactions on Evolutionary Computation*. https://doi.org/10.1109/tevc.2024.3374519

Shobrys, D. E. and White, D. C. 2002. Planning, scheduling and control systems: why cannot they work together. *Computers & Chemical Engineering*, 26(2), pp. 149–160. https://doi.org/10.1016/S0098-1354(01)00737-2

Suman, A., and Udmale, S. S. 2026. Metaheuristic algorithms: A benchmark-driven functional taxonomy and performance analysis. Computer Science Review, 60, P. 100884. https://doi.org/10.1016/j.cosrev.2025.100884

Tian, T., Liang, Z., Wei, Y., Luo, Q., & Zhou, Y. 2024. Hybrid whale optimization with a firefly algorithm for function optimization and mobile robot path planning. *Biomimetics*, *9*(1). https://doi.org/10.3390/biomimetics9010039

Tomar, V., Bansal, M. and Singh, P. 2024. Metaheuristic algorithms for optimization: A brief review. *Engineering Proceedings*, 59, 238 RAiSE-2023, 238. https://doi.org/10.3390/engproc2023059238

Turgut, O. E., Turgut, M. S. and Kırtepe, E. 2023. A systematic review of the emerging metaheuristic algorithms on solving complex optimization problems. *Neural Computing and Applications*, 35(19), pp. 14275–14378. https://doi.org/10.1007/s00521-023-08481-5

Vogel, T., Almada-Lobo, B. and Almeder, C. 2017. Integrated versus hierarchical approach. *OR Spectrum*, 39, pp. 193–229. https://doi.org/10.1007/s00291-016-0450-2

Wang, Y., Su, H.Y., Shao, H.S., Lu, S. and Xie, L. 2017. Integration of production planning and scheduling. *Journal of Zhejiang University (Engineering Science)*, 51, pp. 57–67.

Wang, Z., Dai, D., Zeng, Z., He, D. and Chan, S., 2024. Multi-strategy enhanced grey wolf optimizer for global optimization and real world problems. *Cluster Computing*, *27*(8), pp. 10671-10715.

Yao, X., Almatooq, N., Askin, R. G. and Gruber, G. 2022. Capacity planning and production scheduling integration: improving operational efficiency via detailed modelling. *International Journal of Production Research*, 60(24), pp. 7239–7261. https://doi.org/10.1080/00207543.2022.2028031

Yu, V.F., Kao, H.C., Chiang, F.Y. and Lin, S.W. 2022. Solving aggregate production planning problems. *Applied Sciences*, 12(14), 6945. https://doi.org/10.3390/app12146945

Zhang, T., Wang, Y., Jin, X. and Lu, S. 2019. Integration of production planning and scheduling based on RTN representation under uncertainties. *Algorithms*, 12(6), 120. https://doi.org/10.3390/a12060120

Zheng, J. H., Kou, Y. N., Jing, Z. X. and Wu, Q. H. 2019. Towards many-objective optimization: Objective analysis, multi-objective optimization and decision-making. *IEEE Access*, 7, pp. 93742–93751. https://doi.org/10.1109/ACCESS.2019.2926493

خوارزمية ذكاء اصطناعي جديدة لحل نموذج متكامل لتخطيط الإنتاج الكلي ومشكلة الجدولة

كرار عماد عبد الساده *، بيداء عطيه خلف

قسم الرياضيات، كلية التربية للعلوم الصرفة (ابن الهيثم)، جامعة بغداد، بغداد، العراق

الخلاصة

في السنوات الأخيرة، أصبح دمج تخطيط الإنتاج وجدولة التحكم عاملاً أساسياً في بيئات التصنيع المعقدة والديناميكية. تتناول هذه الورقة هذا التحدي من خلال تقديم نموذج متكامل يُحسّن في آنٍ واحد تخطيط الإنتاج الكلي وجدولة الآلة الواحدة. تتمثل الأهداف الرئيسية للنموذج المُطوّر في تقليل تكلفة الإنتاج الإجمالية، ومدة التصنيع، وتكاليف الإعداد، وتأخيرات تسليم المنتج كمؤشرات أداء رئيسية. لحل هذا النموذج المتكامل، طُوّرت خوارزميتان هجينتان استلاليه: خوارزمية الهجينة الحوت مع خوارزمية تحسين الذباب، وخوارزمية الحوت مع خوارزمية تحسين الذئب الرمادية. أُجريت مجموعة كبيرة من التجارب الحسابية على هاتين الخوارزميتين المقترحتين. تُظهر النتائج أن كلتا الخوارزميتين الهجينتين فعالتان للغاية في إيجاد حلول شبه مثالية للمشكلة المعقدة المتكاملة. تُظهر التحليلات المقارنة تفوق الخوارزميات الهجينة على الخوارزميات الشائعة (خوارزمية الحوت، والخوارزمية الجينية، وخوارزمية الذباب، وخوارزمية الذئب الرمادي) وعلى طريقة البرمجة الديناميكية من حيث جودة الحل، وسرعة التقارب، والاستقرار الحسابي في معظم السيناريوهات المختبرة. تُوفر هذه النتائج للمختصين أدوات دعم قرار فعّالة لتحسين الكفاءة التشغيلية في أنظمة التصنيع الحديثة.

الكلمات المفتاحية: خوارزمية الذباب، تخطيط الإنتاج الكلي، الجدولة، الخوارزمية الجينية .