INJ INTERNATIONAL NEUROUROLOGY JOURNAL

A Multiobjective Seagull Optimization Algorithm For Solving Flow Shop Scheduling Problem – A Case Study

M. K. Marichelvam[1]* and G. Ayyavoo[2]
[1]Department of Mechanical Engineering, Mepco Schlenk Engineering College, Sivakasi, Tamilnadu, India – 626 005.
[2]Department of Mechanical Engineering, PTR College of Engineering and Technology, Madurai, Tamilnadu, India – 625 005

Abstract

In the manufacturing industries, the most complicated scheduling crisis is the flowshop scheduling problem (FSP) and is proven to be an NP-hard problem in real-world cases. In FSP, the chief issue in this problem is handing over jobs in every stage to machines and choosing the jobs processing order allotted to each machine. This crisis comprises three sub-decisions: assigning jobs to factories, picking suitable machines for jobs, and establishing the processing order on each machine. In this paper, a FSP has been investigated by considering objective functions like makespan, mean flow time and machine idle time. By reducing the make span, mean flow time and the machine idle time the energy consumption of the company could be reduced. To show the outcome of the proposed methodology, it is implemented in MATLAB by taking real-world scenarios. With the benefit of the seagull optimization procedure, the manufacturing field has the power to determine the processing order on each machine by appropriately selecting jobs on specific machines. The mathematical model assists the proposed model in achieving an efficient outcome meanwhile maintaining the energy constraint model.

Keywords: Flow shop; scheduling; makespan; mean flow time; seagull optimization algorithm

Introduction

The energy demand is rapidly increasing worldwide because of the sudden economic development and fast growth in globalization. The official reports show that the energy demand is overgrowing at the constant rate of 56%. Especially in manufacturing factories, energy resources are playing an essential role in manufacturing the products. So that the industry holders need to deal with various kinds of problems like reducing the cost of production, achieve the satisfaction of customers, and worldwide environment's protection [1]. Most of these problems made an impact on all kinds of manufacturing units. The single sector manufacturing units are deal with these kinds of issues in a typical manner. These manufacturing units are the best example for the traditional flow shops [2]. Here, the manufacturing process is accomplished based on customer requirements and satisfaction. Due to the rapid changes in the markets, these manufacturing units face lots of process scheduling problems, and it is named as flow shop scheduling problem (FSP) [3]. It is nothing but sequencing the group of products that travel through various manufacturing machines that can be in the same order. But the fact is that all the machines are available and can't be applicable in real-time manufacturing units. This is due to preventive maintenance or breakdown [4]. This problem will cause an essential impact on the different kinds of performance features like reliability, profitability, and productivity. Here, the maintenance planning is not interconnected with the scheduling process, which generates problems in flow shop scheduling. Therefore, the maintenance and scheduling process should be performed together to stabilize the availability and utilization of machines [5]. The Flow shop scheduling problem has some categorize based on their operations, and they are named as no-wait FSP (NWFSP), no-idle FSP (NIFSP), non-smooth FSP (NSFSP), permutation FSP (PFSP), and blocking FSP (BFSP), etc.

Generally, *n* numbers of products are manufacturing in various stages of machines in the standard FSP, which follows the same manufacturing layout, and each stage has single machines for the process [6]. In this case, the efficiency of the manufacturing units is not satisfied the manufacturers and also customers. Therefore, the manufacturing layouts are move towards the flow shop (FS) layout to maximize the manufacturing efficiency. FS is a developed version of FS which allows the product to manufacture parallelly with the utilization of any single machine in the manufacturing units during a specified stage of process [7]. It is a universal layout that is implemented on many manufacturing units such as iron and steel, furniture, textile, paper industries, electronics industries, etc., to minimize the impact of the bottleneck. Meanwhile, it maximizes the capacity of the production. This FS layout is also named multiprocessor flow shop, flow shop with the parallel machine or flexible flow shop. It obtains similar significance to both combinatorial optimization and production management [8]. Every processing stage in the FS has a minimum of one machine in parallel, and importantly every stage must have more than one machine in the respected stage. Here the scheduling problem makes an

ISSN:2093-4777 | E-ISSN:2093-6931
DOI: 10.5123/inj.2024.3.inj3
Vol. 28 Iss. 3 (2024)

impact on the processing stage [9]. It is a common problem in the flow shop layout, and it is one of the types of flow shop scheduling problems mentioned above. The FSP is complicated to solve compared to the standard flow shop scheduling problem. It considers the scheduling and process assignment for the manufacturing flow shop [10]. These problems are motivated by researchers to design a novel method to identify these problems. The research considers the processing time as constant because, in actual manufacturing units, the manufacturing process times are different based on the various level of laborers, their learning skills and so on.

The current research works commonly identify the energy-efficient flow shop scheduling and multi-objective flow shop scheduling [11]. Usually, the main target of the scheduling process is to allocate the minimum level of resources to manufacturing the product with a respected time, and it is utilized to optimize specified objectives. Due to the enormous demand for energy resources, the current research concentrates on optimizing energy consumption in manufacturing scheduling problems [12]. Therefore, there are lots of heuristics and meta-heuristics optimization techniques such as genetic algorithm (GA), artificial bee colony (ABC), iterated greedy (IG), cuckoo search algorithm (CSA), particle swarm optimization (PSO), ant colony optimization (ACO), Harmonic search (HS), parallel tabu search algorithm (PTSA), bi-layer optimization approach (BLO), etc., are developed and implemented to optimize the flow shop scheduling problem. Some metaheuristic optimization algorithm has better worldwide search capacities. Many research works are utilizing multi-objective optimization algorithms to optimize the flow shop's scheduling problems in recent days. Various multi-objective optimization algorithms are generated, like NSGA-II, NSGA-III, and Multiobjective Energy-Aware and Decomposition (MOEA/D) [13]. Most of the existing multi-objective optimization algorithms are focusing on continuous optimization problems. But there is a minimum level of research works concentrating on solving FSP by utilizing a multi-objective optimization algorithm [14]. This case motivates the presented work to implement a multi-objective optimization algorithm with enhanced optimization methods to solve the FSP. Utilizing the improved optimization methods in the multi-objective optimization algorithm will bring the best solutions for the manufacturing units' FSPs. The manuscript is organized as follows; section 2 carries the literature review part. Section 3 demonstrates the problem statement. Section 4 elaborates the design of the proposed methodology. Section 5 discusses the result and discussion part. The final section offers the conclusion.

Literature Review

Numerous existing techniques based on flow-shop scheduling problems are analyzed. The drawbacks of each method stood as motivation to design a novel optimization strategy by employing a multi-objective function. Some of the recent studies are elaborated as follows. Santucci et al. [15] have presented a simple algebraic method for combinatorial search space. This method was suggested to the differential evolution algorithms. The author derived the discrete differential evolution algorithm for permutation problems and implemented it to the issue, which impacted permutation flow shop scheduling and the total flowtime criterion. The suggested algorithm is only implemented for permutation, and it fails to implement other genetic operators such as crossover, classical mutation, etc. Quan-Ke Pan et al. [16] have presented an optimization method for flow shop scheduling problems with flow time criterion and it is associated with local search and existing approaches of both objectives of makespan and total flow time. The performance can be supervised by particle swam optimization and it minimizes the permutation flow shop sequence problem and is therefore strongly in NP-hard and it analyse the worst-case ratio bound for several heuristics. Finally, it minimizes the maximum job completion time(makespan). Christos Koulamas et al. [17]] have presented three stage assembly flow shop scheduling problem and used to minimize the makespan associated with NP-hard. Then analyse worst case ratio bound for several heuristics. Afterwards the problem can be derived by the maximum job completion time (makespan).Jiang et al. [18] have suggested an enhanced decomposition-based multi-objective evolutionary algorithm to rectify the bi-objective problems effectively. Here, the permutation flow shop scheduling problem and setup times depending on the sequence were decomposed into various sub-problems by utilizing the decomposition technique. The solutions, which relied on the sub-problems in every generation, were mated by designing the dynamic strategy. But the author fails to analyze the properties of the problems to develop a novel knowledge guided for search operators.

Betul Yagmahan et al [19] have proposed the flow shop scheduling problem with associated with makespan and total flowtime. In this criteria NP-hard type used to solve the problem. Scheduling problem can be solved using multi-objective ant colony system algorithm (MOACSA), and it is incorporated with a local search strategy with multi-objective heuristics. The computational results show that proposed algorithm is more efficient and better than other methods compared. Jiang et al. [20] have investigated the scheduling problem in the flow shop along with the low number of buffers, and it depended on energy. The presented work suggested the decomposition-based multi-objective evolutionary algorithm to rectify the Ni-hard problems in the flow shop. It did not apply to the real-time flow shop scheduling problem. Li et al. [21] have presented a two-level imperialist competitive algorithm to examine the Energy-efficient hybrid flow shop problem. These two levels of the algorithm were considered as the most substantial empire and other empires correspondingly. In every search stage, the revolution and assimilation were variously implemented in the realms to obtain the best results. Hao et al. [22] have suggested a novel hybrid brainstorming optimization algorithm to rectify the distributed FSP. Here, the author utilized distributed Nawaz-Encore-Ham method for construct the heuristic in the presented work. The proposed work not concentrated on the optimization problems in the wireless network placed in the flow shop. Shao et al. [23] have introduced two algorithms to rectify the distributed FSP. Then, the jobs were assigned to industries by utilizing a greedy iteration algorithm. But the author failed to load balancing in every machine and also not focused on the production cost. Zhang et al. [24] have implemented a discrete whale swarm algorithm to recognize the near-optimal results in a FSP. The encoding and decoding methods were mainly contributed to keeping away from the infeasible results. Based on the insight of existing studies on scheduling problem, some of the research gaps are explored in manufacturing systems which are listed along with the proposed research work findings,

- The existing techniques mentioned in the review work only focused on the time-oriented constraint. Recently, energy-aware limitations have considered being a promising constraint for achieving sustainable manufacture.
- Uncertainties haven't received enough concentration while investigating scheduling problems in real-world production systems.
- Meta-heuristic algorithmic procedures, consisting of simulated annealing algorithms, genetic algorithms, and tabu search algorithms, were used to deal with scheduling problems. However, their functionalities to multi-objective energy-conscious scheduling problems are limited, and, especially for a complex and uncertain scheduling model, their advantages cannot be fully explored. Population-based meta-heuristics are considered a helpful tool for tackling multi-purpose optimization problems due to their robust search capabilities.

Problem Formulation

The FSP from the presented work is briefly defined in this section. The flow shop layout contains the manufacturing stages in a series manner, and each stage has several machines located in a parallel manner. The machines which are located in the manufacturing units are may be uniform, unrelated, or identical. Here, every stage must have more than one machine. Every product is manufacturing by utilizing one machine at a time in each stage. The traditional FSP consists of three types based on a parallel machine. (a) Identical parallel machines FSP for example, in parallel machines, every job has similar time for manufacturing at each stage. (b) Uniform parallel machines FSP for example, the manufacturing speed of the machine is inversely proportional to the similar product's processing time on any parallel machines at every stage. (c) Unrelated parallel machines FSP for example, the manufacturing product's processing time is inappropriate at every stage on any parallel machine. Still, it is based on the matching degree among machines and jobs. FS problems are considered more significant and complex NP-hard problems, and they are commonly found in most real-world manufacturing factories. Therefore, the presented work considers the o number job's set ($K = \{1, 2, \dots, o\}$), and stages perform it in a similar layout. Every stage$(j = 1, 2, \dots, t)$ contains a set of uniform parallel machines$(N_{j,1}, N_{j,2}, \dots, N_{j,}NJ)$, here $N_{j,}$denoted as l-th machine at stage j. Every job $k \in K$ contains the process chain $\left(P_{1,k}, P_{2,k}, \dots, P_{j,k}, \dots, P_{t,}\right)$ where the process of the job k is denoted as $P_{j,k}$at the stage j. Here, one uniform parallel machine processing the one process $P_{j,k}$at a time. The requirement of manufacturing Qj is attained by each process $P_{j,k}$. When the process P_j is allocated to the parallel machine $N_{j,l}$and its speed is$u_{j,l}$, then it is derived as $q_{j,k,l} = q_{j,k}/u_{j,l}$and that has a time in units to accomplish. The main objective of this work is to reduce the makespan, mean flow time and idle time of the machine based on simulation, and it is relevant to the consumption of energy. The formulation of the problems mentioned above as given below,

Makespan is defined as the completion time of the last job to leave the system. Makespan is important for effective utilization of resources and is expressed as,

$$g1 = D(\pi_0, n)$$

Mean flow time is the average time spent by the job in the system. Mean flow time is vital to minimize the work-in-process inventories.

$$g2 = \sum_{j=1}^{o} D(\pi_0, n)$$

Machine Idle Time represents how the count of the production machine is not working or the count of the machine in an idle state. Its mathematical expression is offered as follows,

$$g3 = \{D(\pi_1, k-1) + \sum_{j=1}^{0}\{\max\{D(\pi_j, k-1) - D(\pi_{j=1,}k), 0\}\} \,|k = 2, \dots, n\}$$

Based on the problem statement, in this research, assumptions made are depicted as follows,

1) Every job can be processed in one machine at its corresponding process time,
2) Every machine in the manufacturing units is only processing one job at a particular process time,
3) The manufacturing time of every product is considered as a constant value,and all jobs are independent.
4) The transportation time between every accomplished stage are assumed as negligible,
5) Every setup time for each machine is also assumed negligible, and each operation cannot be interrupted once it starts.
6) The buffer and storage in the respected workstation are considered to be unlimited,
7) Exclusive products are independent and available for manufacturing with processing time zero; preventive maintenance and machine breakdowns are not considered.
8) Each machine will be turned off after completing the last job, but intermediately there may be idle times for machines.

Seagull Optimization Algorithm- Background knowledge

The section elaborates on the inspiring behavior of developing a seagull optimization technique. Additionally, its mathematical modeling is offered. From the nature of the biological concept of seagull behavior, it is developed scientifically named Laridae. Typically, seagull is a sea bird and is found anywhere on the planet.These types of birds possess varying nature of weight and height.It is dependent on the characteristics of the omnivorous kind, and they can eat fishes, amphibians, insects, earthworms, reptiles etc. The structural bodyparts of the seagull are enclosed with feather-type plumage substance and arenoted to be an intelligent bird. For attacking prey, bread crumbs are utilized by a seagull and making a noise like rain sound with the help of its feet. This can be done by having food on searing earthworms situated inside the ground. Both the fresh and saltwater are drunk by this bird because so many animals cannot perform this task.

Nevertheless, a seagull carries a pair of glands above its eyes on the right side that can flush salt through its functionality. They are available in colonies and utilize their knowledge to perform discovering and attacking prey. For learning food inadequate amount of time, migration procedure is to be followed. The migration phase in the seagull optimization algorithm is determined as the seasonal appearance of the seagull changing its place to another place. The main properties of the seagull optimization procedure are described as subsequently,

- ✓ For avoiding collision in its initial population, seagulls are supposed to present in the varying initial position. While the migration process takes over, the respective seagull rides in the group.
- ✓ Seagulls in the group tend to fly ina particular direction to locate optimal fitness function by following survival seagull oriented with the best fitness rate.
- ✓ By following the best fitness seagull, other seagulls can update their corresponding position. While the birds migrate in the sea to shift from one place to another, the seagull starts its attacking behavior.
- ✓ At the stage of attacking, the seagull makes spiral shape movement. This behavior is formulated for finding objective functions.

Mathematical framework

This section detailly explains the migration and attacking strategies of the seagull in forming a mathematical framework.

Migration or Exploration

Here, the seagull optimization algorithm demonstrates the movements of the seagull from one position to another position. As per this statement, the seagull should accomplish three situations:

Collisions avoiding: In this condition, variable B is additionally applied for the new search agent position's calculation to prevent the collision among neighbor seagulls,

$$\overrightarrow{D_t} = B \times \overrightarrow{Q_t}(y) \qquad (1)$$

The search agent's position is denoted as $\overrightarrow{D_t}$, and it doesn't make a collision with neighbor search agents. The search agent's present position is represented as $\overrightarrow{Q_t}$, the current iteration is denoted as y, and the search agent's movement behavior is designated as B in a given search space and the equation to calculate the variable B is given below:

$$B = g_d - (y \times (g_d / Max_{iter}))$$

Here: $y = 0, 1, 2, \ldots, Max_{iter}$ (2)

The applying variable B's frequency is controlled by utilizing g_d, and it is decreasing linearly from g_d to 0. The presented work set the value of g_d to 2.

The forward movement to the direction of best neighbors: the search agents are moving forwardly to the best neighbor's direction. This is performed after avoiding the collision and the neighbors.

$$\overrightarrow{N_t} = C \times (\overrightarrow{Q_{ct}}(y) - \overrightarrow{Q_t}(y)) \quad (3)$$

The search agent $\overrightarrow{Q_t}$'s position is denoted as $\overrightarrow{N_t}$, and $\overrightarrow{Q_{ct}}$ represents the best-fit search agent. The activities of C control the appropriate equalizing among exploitation and exploration. The following equation is utilized to calculate the value of C,

$$C = 2 \times B^2 \times rand \quad (4)$$

Where random number is denoted as $rand$, which lies in the range of [0,1]

Endure close to the best search agent: at last, the search agent is updating its position concerning the best search agent.

$$\overrightarrow{E_t} = |\overrightarrow{D_t} + \overrightarrow{N_t}| \quad (5)$$

Where the distance among the best-fit search agent and search agent is denoted as $\overrightarrow{E_t}$.

Attacking or Exploitation

In this process, the search operation's history and experience are exploited. Here, the attacking angle of the seagull can be changed continuously and changing the speed when the process of migration. The seagull utilizes their weight and wings to control their altitude. The spiral movement attacking strategy of the seagull is performed in the air while attacking the prey. The behavior of these attacking strategies in the x, y, and z plane is derived below:

$$x' = s \times \cos(l) \quad (6)$$
$$y' = s \times \sin(l) \quad (7)$$
$$z' = s \times l \quad (8)$$
$$s = v \times f^{lw} \quad (9)$$

Where the radius of spiral's each turn is denoted as s, a random number is indicated as l in range of $[0 \le l \le 2\pi]$, the shape of the spiral is marked as and w, and the natural logarithm's base is denoted as f. The following equation is utilized to calculate the search agent's updated position.

$$\overrightarrow{Q_t}(y) = (\overrightarrow{E_t} \times x' \times y' \times z') + \overrightarrow{Q_{ct}}(y) \quad (10)$$

The best results are saved, and the other search agent's position is updated by utilizing $\overrightarrow{Q_t}(y)$.
At first, the proposed seagull optimization algorithm is developing the populations in a random manner. When the iteration process started, the position of the search agent is updated for the best search agent. Variable B is linearly reduced from g_d to 0. Here, variable C takes control of the smooth transition among the exploration and exploitation. Therefore, the proposed SOA is utilizing in the worldwide optimization process because of its capability of exploration and exploitation.

4.1. The Proposed Model

An energy-efficient Multi-objective SOA algorithm is developed to solve HFSP. The chief goal of this research work is to offer a solution for overcoming delivery delays of jobs in manufacturing industries. Hence, in the

DOI: 10.5123/inj.2024.3.inj3
ISSN:2093-4777 | E-ISSN:2093-6931
Vol. 28 Iss. 3 (2024)

INJ INTERNATIONAL NEUROUROLOGY JOURNAL

proposed research, aligning jobs among machines depends on its priority and is solved using a seagull optimization strategy. Generally, seagulls are available in villages, and with their knowledge, they can be capable of tracking and attack the prey for the food source. The main behavior followed by seagulls is its migration and attacking behavior. This, in turn, make several orders to be processed in a parallel manner among various production units. This can be done by solving mathematical models by focusing on energy-oriented constraints by scheduling the production. Henceforth the model is developed under the basis of the energy-efficient seagull Optimization method to wrap the uncertain nature of the processing time parameter. The layout of the proposed industrial machine arrangement is displayed in Figure 1.

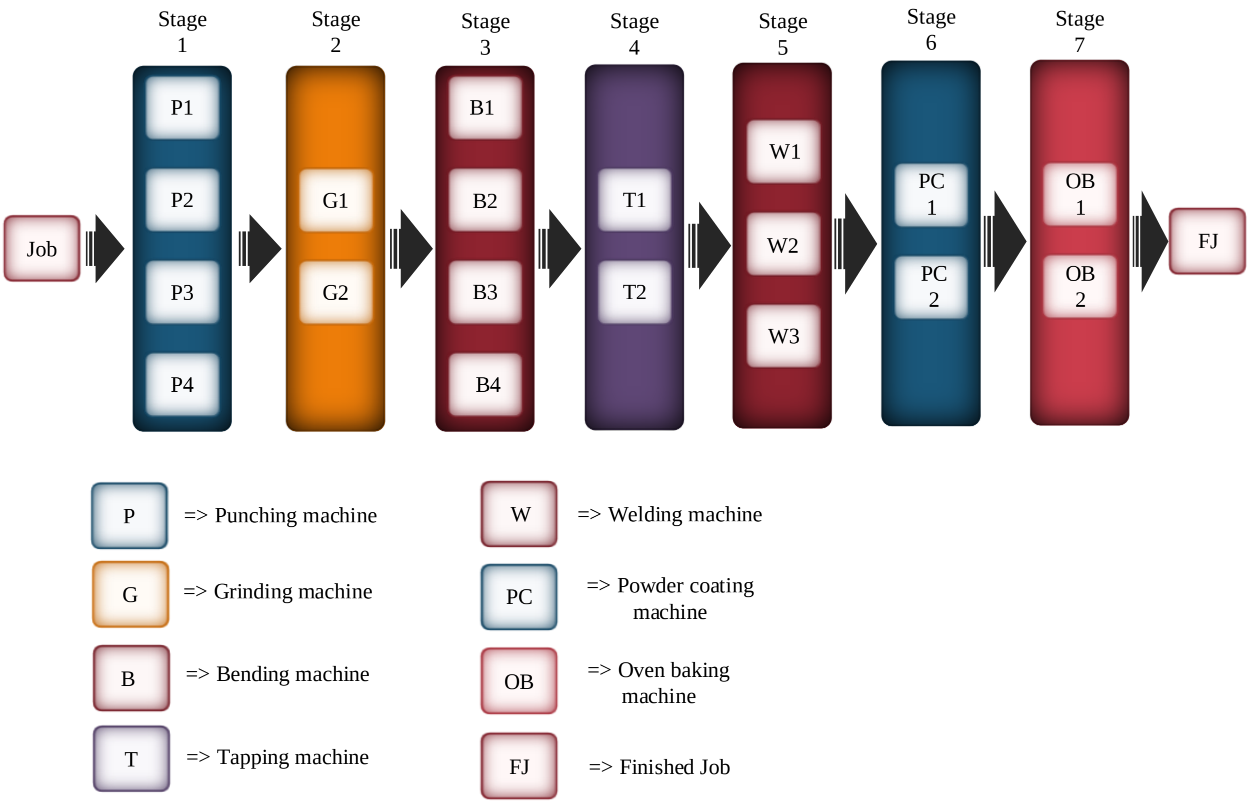

Figure 1 HFS layout and process flow

Due to the demand for an energy source in production and manufacturing firms, and energy-efficient multi-objective seagull optimization algorithm for HFS is introduced. Figure 1illustrates the real-world scenario by containing seven stages with several different jobs. This type of energy-intensive industry should possess an energy constraint mechanism for manufacturing and production activities. For maintaining energy crisis in this type of infrastructure, it is necessary to schedule the activities based on time-oriented criteria. With this insight, multi-objective functions are determined using the seagull optimization procedurein HFSP under energy constraint criteriaby considering the minimization problem. The step-wise procedure of the proposed optimization algorithm is described as follows.

Step 1:The total dimension by carrying the number of jobs in the real-world manufacturing firm is considered a population matrix. In this research, for aligning job positions, the Largest Rank Value (LRV) is determined for the proposed HFSP.Thesolution representation is provided based on$o$ number jobs$(K = \{1, 2, \dots, o\})$ in stage $j(j = 1, 2, \dots, t)$ and is represented as follows,

Dimension (20×7)							
Jobs/ Stages	j_1	j_2	j_3	j_4	j_5	j_6	j_7
$\boldsymbol{o_1}$	y_{11}	y_{12}	y_{13}	y_{14}	y_{15}	y_{16}	y_{17}
$\boldsymbol{o_2}$	y_{21}	y_{22}	y_{23}	y_{24}	y_{25}	y_{26}	y_{27}

DOI: 10.5123/inj.2024.3.inj3
ISSN:2093-4777 | E-ISSN:2093-6931
Vol. 28 Iss. 3 (2024)

⋮	
o_{20}	…

The above population matrix establishes the individual LRV function for the corresponding dimensions using a multi-objective function. Based on the above equation, the initialization is done without collision, and it follows the procedure from equation (1-5).

Step 2: To progress productivity and minimize delivery delay issues in manufacturing firms, it is necessary to obtain an optimal decision, so that customer satisfaction gets raised. With that concern, multi-objective functions are framed on behalf of the minimization function. It is mathematically formulated as follows,

$$Z_{HFSP} = \min \left\{ g1 = D(\pi_0, n) \middle\| g2 = \sum_{j=1}^{o} D(\pi_0, n) \middle\| g3 \right.$$

$$\left. = \{D(\pi_1, k-1) + \sum_{j=1}^{0} \{\max\{D(\pi_j, k-1) - D(\pi_{j=1,} k), 0\}\} \right\}$$

The minimization function assists the proposed model to eventually schedule the jobs with its respective stages based on LRVoperation.

Step 3: Each seagull updates its position based on the associated multi-objective function. The objective function of the seagull depends on the attacking behavior for a food source in finding prey, and it is determined using (10). The following mathematical notation$\overrightarrow{Q_t}(y) = (Z_{HFSP} \times x' \times y' \times z') + \overrightarrow{Q_{ct}}(y)$ is expressed to solve the manufacturing systems' HFSP.Thus, the proposed solution seems to determine the optimal sequence of jobs processing on the machine in a flow shop scheduling procedure with an energy constraint mechanism with proper scheduling factors.The energy-efficient multi-objective seagull optimization algorithmic procedure for sustainable manufacturing is depicted in Figure 2.

DOI: [10.5123/inj.2024.3.inj3](https://doi.org/10.5123/inj.2024.3.inj3)
ISSN:2093-4777 | E-ISSN:2093-6931
Vol. 28 Iss. 3 (2024)

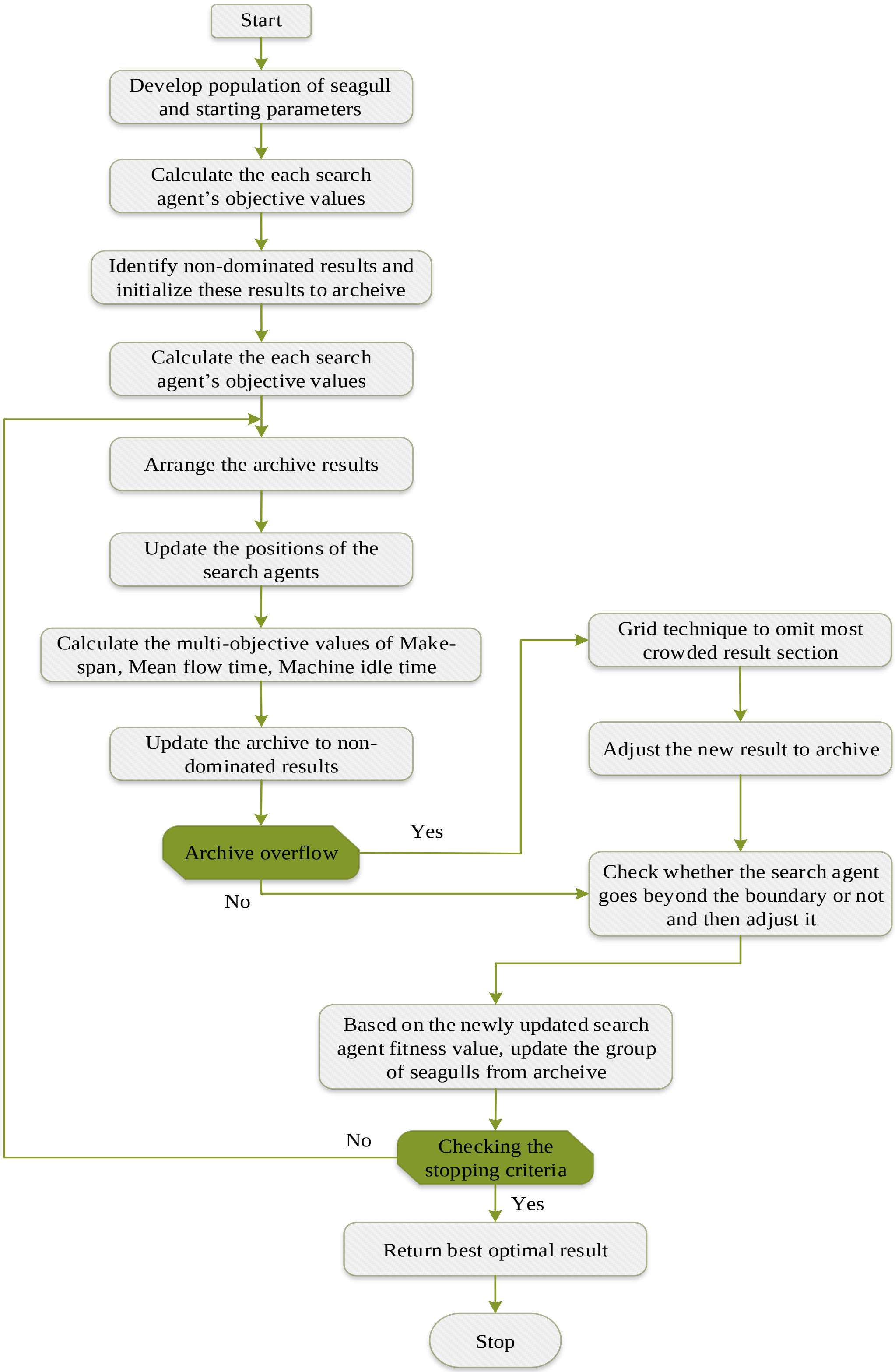

Figure 2 Layout of the optimization algorithm

The proposed scheduling solution can provide the interaction between production cost and energy consumption to grasp as stainable production practice.

With the executed procedure, intelligent manufacturing with energy constraint scheduling problems is done in modern manufacturing systems. This scheduling model can formulate an interface between the energy consumption and the production cost to realize an efficient and sustainable production process.

INJ INTERNATIONAL NEUROUROLOGY JOURNAL

Case study and discussion

In this work, a mathematical framework is developed. The suggested multi-objective seagull optimization is analyzed by utilizing a real-world manufacturing unit, which is manufacturing sheet metal components of solar panels, UPS, etc. This manufacturing unit is located in Hosur, India. Entire algorithms are coded by using MATLAB version 2020a. These experimental analyses are executed on system configuration with Intel i3 Processor having RAM -8GB in Operating System Windows 10.

5.1. Performance analysis

The case study is based on the real-world manufacturing unit manufacturing the components for the solar panels, UPS, etc., as shown in Figure 3. Each partis traveling through seven different types of process, and each process stage has several machines located in a parallel manner. Here, the transportation of the components between each process is performed by utilizing the trolley with workforce or forklift. In this manufacturing unit, sheet metal is being used as a raw material with different type's thicknesses and grades, based on the application of the product. Some related data are provided in Table 1 and Table 2, which contain processing time, job no, and mass of each component.

Figure 3 Manufacturing component which is processed in the studied industry

Table 1 Mass of the components

Job no	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**
Mass(kg)	2.43	2.22	2.13	2.84	5.06	1.56	1.56	1.65	4.23	1.95
Job no	11	12	13	14	15	16	17	18	19	20
Mass(kg)	1.95	2.41	2.69	4.95	1.99	1.99	1.58	5.54	2.50	2.53

Table 2 Process time (sec) of each component with process name

Job no	**Process**						
	Punching	Grinding	Bending	Tapping	Welding	Powder coating	Oven baking
1	45	8	16	18	0	0	0
2	69	8	16	0	0	0	0
3	21	1	8	7	0	0	0
4	82	9	16	21	240	65	1200
5	78	13	32	0	0	0	0
6	44	7	24	7	0	0	0
7	44	7	24	11	178	47	1200
8	35	6	16	11	0	0	0
9	80	14	32	4	0	0	0
10	44	8	24	0	162	59	1200
11	44	8	24	0	162	59	1200

DOI: 10.5123/inj.2024.3.inj3
ISSN:2093-4777 | E-ISSN:2093-6931
Vol. 28 Iss. 3 (2024)

INJ INTERNATIONAL NEUROUROLOGY JOURNAL

12	52	9	4	18	0	0	0
13	50	8	12	0	0	0	0
14	28	5	0	0	0	0	0
15	61	11	20	0	0	0	0
16	35	6	8	7	0	0	0
17	72	15	32	14	130	42	1200
18	48	8	12	0	0	0	0
19	65	12	16	0	90	31	1200
20	30	5	0	4	0	0	0

From the above Table, the processing time of each component is provided in seconds. Here, punching, grinding, bending, tapping, welding, powder coating, and oven baking are the process which is taken from the manufacturing units. The data mentioned above are collected from the real-time manufacturing process of the manufacturing factory. The performance of the proposed multi-objective seagull optimization is validated by optimizing the settings of the parameter. Here, the pilot analysis is taking place due to the restrictions in spaces.

5.2. Analysis of Makespan

In this section, the analysis of makespan is conducted, and it is plotted based on job numbers. The x-axis of the graph describes the number of jobs, and the y-axis represents the values of makespan. This graph is plotted in Figure 4. The job unit is based on quantity, and the makespan units take as seconds in time. Totally 20 number of jobs are taken from the manufacturing company. From the simulation results, job no 4 has the maximum makespan value of 1637 seconds and job no 3 has a minimum makespan value of 40 seconds. The F1 score of the proposed seagull optimization algorithm and the best score with iteration are provided in Figure 5 and Figure 6 respectively. Here, the x-axis is taking as the iteration from 10^0 to 10^3, and the y-axis is taking as the best score value for the proposed optimization algorithm. The best score value of makespan in the proposed optimization algorithm is 0 at 10^3 th iteration.

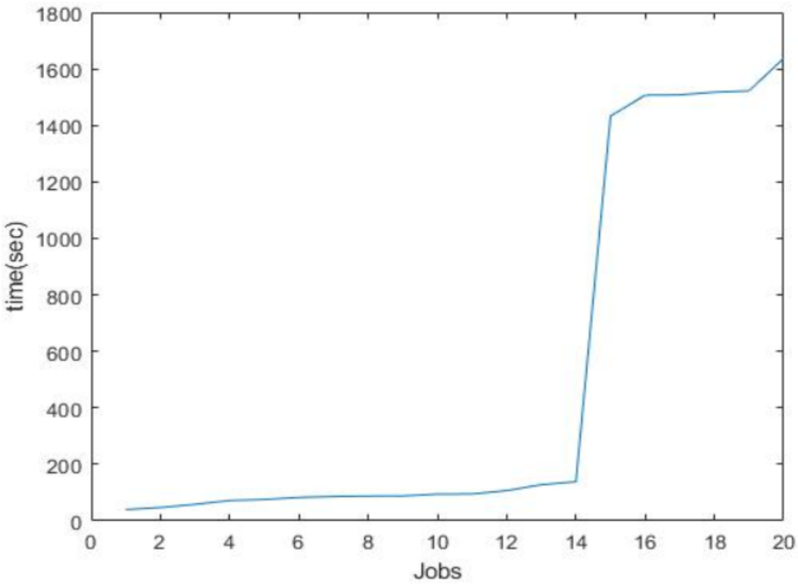

Figure 4 Graph plotted for the analysis of makespan

DOI: 10.5123/inj.2024.3.inj3
ISSN:2093-4777 | E-ISSN:2093-6931
Vol. 28 Iss. 3 (2024)

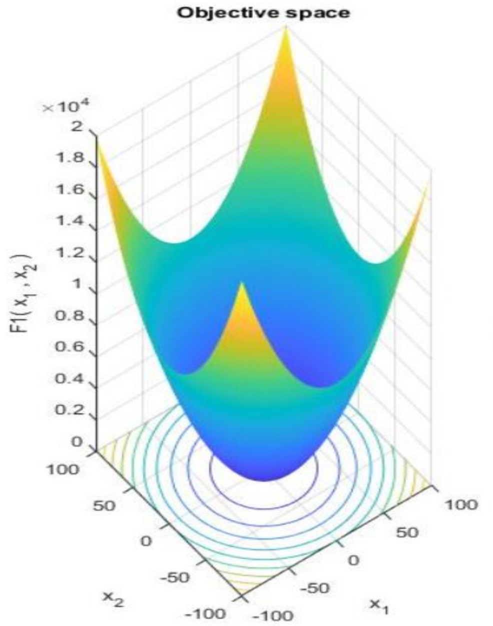

Figure 5 The F1 measure value of the proposed algorithm in makespan

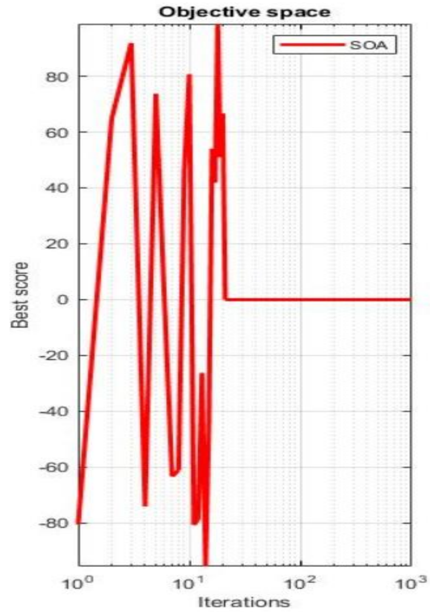

Figure 6 The best score value of proposed algorithm makespan

5.3. Analysis of Mean flow time

In this section, based on the number of jobs, the analysis of Mean flow time is performed and plotted. Figure 7 shows the graph, which contains the x-axis of the chart, the number of jobs, and the y-axis is the values of Mean flow time. The job unit is based on quantity, and the makespan units take as seconds in time. Totally 20 number of jobs are taken from the manufacturing company. From the simulation results, job no 4 has the maximum makespan value of 204.6 seconds and job no 3 has a minimum makespan value of 5 seconds. The F1 score of the proposed seagull optimization algorithm and the best score with iteration are provided in Figure8 and Figure 9 respectively. Here, the x-axis is taking as the iteration from 10^0 to 10^3, and the y-axis is taking as the best score value for the proposed optimization algorithm. The best score value of makespan in the proposed optimization algorithm is 0 at 10^3 th iteration.

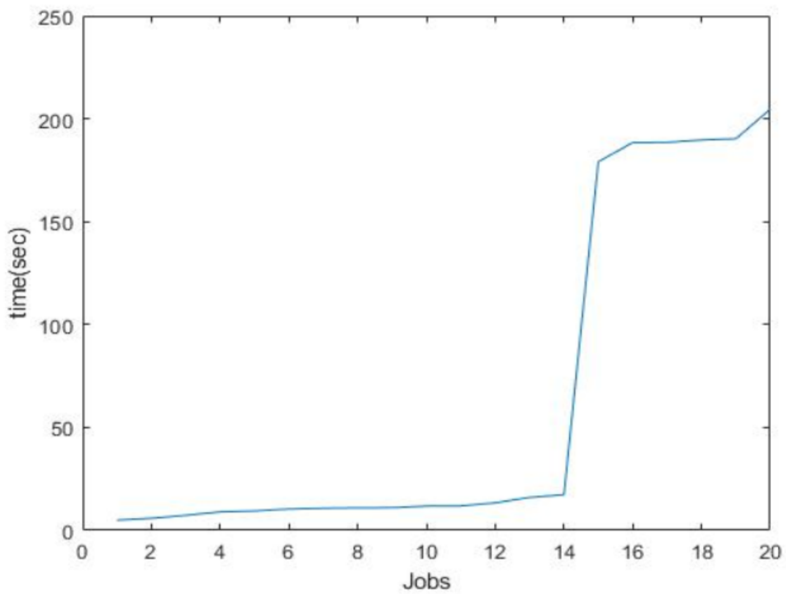

Figure 7 Graph plotted for the analysis of Mean flow time

DOI: 10.5123/inj.2024.3.inj3
ISSN:2093-4777 | E-ISSN:2093-6931
Vol. 28 Iss. 3 (2024)

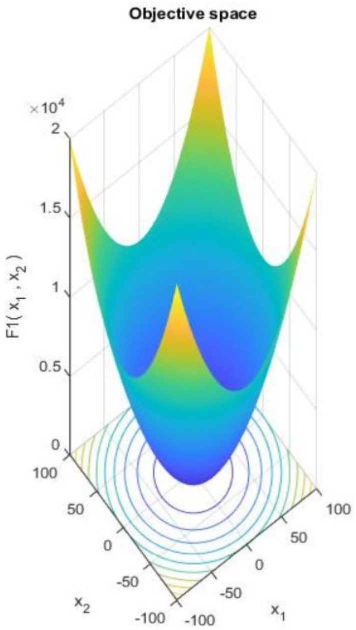

Figure 8 The F1 measure value of the proposed algorithm in mean flow time

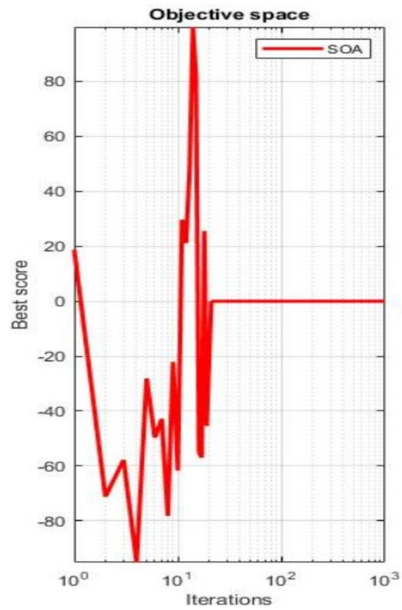

Figure 9 The best score value of the proposed algorithm in mean flow time

5.4. Analysis of machine idle time

In this section, the machine's idle time is analyzed based on job numbers and plotted as a graph, as showed Figure 10. The x-axis of the chart contains the number of jobs, and the y-axis is included the values of Mean flow time. The job unit is based on quantity, and the makespan units take as seconds in time. Totally 20 number of jobs are taken from the manufacturing company. The F1 score of the proposed seagull optimization algorithm and the best score with iteration are provided in Figure 11 and Figure 12. Here, the x-axis is taking as the iteration from 10^0 to 10^3, and the y-axis is taking as the best score value for the proposed optimization algorithm. The best score value of makespan in the proposed optimization algorithm is 0 at 10^3 th iteration.

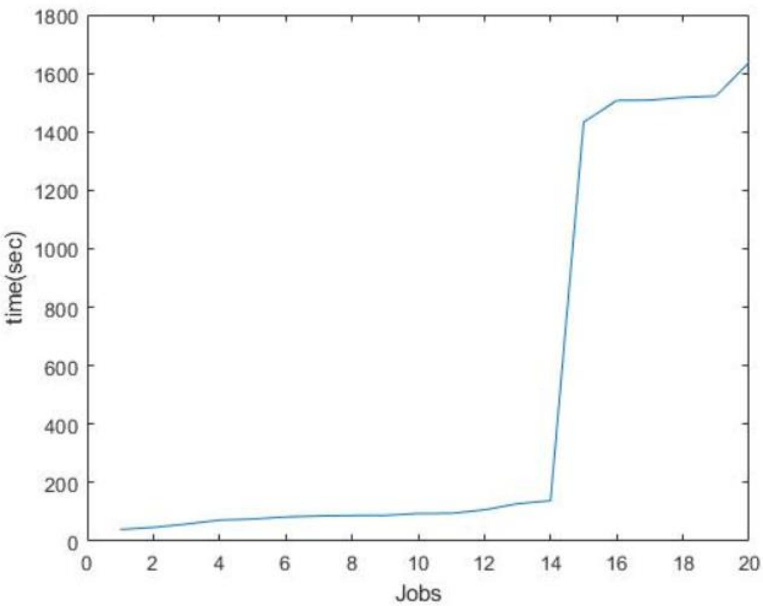

Figure 10 Graph plotted for the analysis of machine idle time

DOI: 10.5123/inj.2024.3.inj3
ISSN:2093-4777 | E-ISSN:2093-6931
Vol. 28 Iss. 3 (2024)

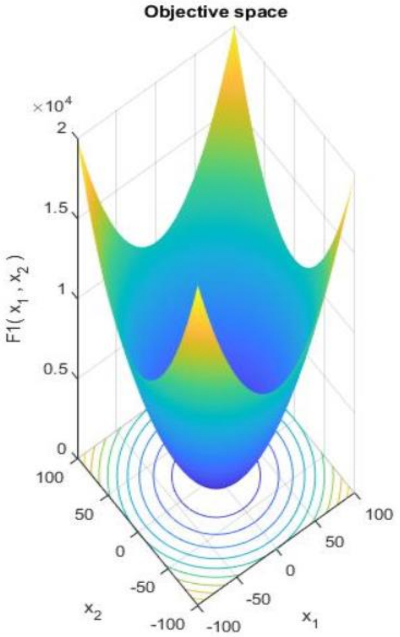

Figure 11 The F1 measure value of the proposed algorithm in machine idle time

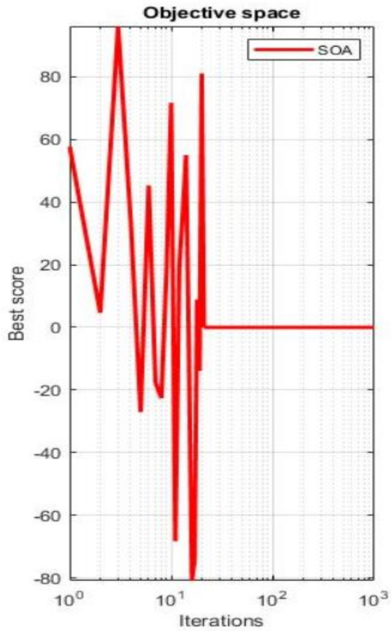

Figure 12 The best score value of the proposed algorithm in machine idle time

The outcome of the case study and discussion section reveals the advantage of executing the proposed multi-objective seagull optimization algorithm to solve the flow shop scheduling problems.

Conclusion

In the manufacturing unit, flow shop scheduling plays a significant role because of its excellence in scheduling plans that effectively achieve company productivity. Concerning this, a real flow shop scheduling is proposed using a multi-objective optimization procedure. The multi-objective functions related to the scheduling problem taken in our study are makespan, mean flow and idle time. Based on the derived mathematical model, the case study is carried out in the real-world scheduling environment. Experimental results show that the proposed mathematical model outperforms well to deal with the examined problem. The advancement of this investigation may adapt to other types of energy-efficient scheduling problems like blocked flush shop, no-idle flow shop and high-level integrated manufacturing problems. The future scope will focus on constructing aflow shopmeta-heuristics approach to resolve large-scale issues in a reasonable running time.

References

1. Ruiz, R., Maroto, C. and Alcaraz, J., 2006. Two new robust genetic algorithms for the flow shop scheduling problem. *Omega*, *34*(5), pp.461-476.
2. Gupta, J.N., 1971. A functional heuristic algorithm for the flow shop scheduling problem. *Journal of the Operational Research Society*, *22*(1), pp.39-47.
3. Ruiz, R. and Stützle, T., 2007. A simple and effective iterated greedy algorithm for the permutation flow shop scheduling problem. *European journal of operational research*, *177*(3), pp.2033-2049.
4. Naderi, B. and Ruiz, R., 2010. The distributed permutation flow shop scheduling problem. *Computers & Operations Research*, *37*(4), pp.754-768.
5. Dios, M., Fernandez-Viagas, V., & Framinan, J. M. (2018). Efficient heuristics for the hybrid flow shop scheduling problem with missing operations. Computers & Industrial Engineering, 115, 88-99.
6. Öztop, H., Tasgetiren, M. F., Eliiyi, D. T., Pan, Q. K., & Kandiller, L. (2020). An energy-efficient permutation flowshop scheduling problem. Expert systems with applications, 150, 113279.
7. Ruiz, R., Pan, Q. K., & Naderi, B. (2019). Iterated Greedy methods for the distributed permutation flowshop scheduling problem. Omega, 83, 213-222.
8. Yagmahan, B. and Yenisey, M.M., 2008. Ant colony optimization for multi-objective flow shop scheduling problem. *Computers & Industrial Engineering*, *54*(3), pp.411-420.
9. Lin, S. W., Cheng, C. Y., Pourhejazy, P., & Ying, K. C. (2021). Multi-temperature simulated annealing for optimizing mixed-blocking permutation flowshop scheduling problems. Expert Systems with Applications, 165, 113837.

10. Khalifa, A. E. W., Alodhaibi, S. S., & Kumar, P. (2021). Solving constrained flow-shop scheduling problem through multistage fuzzy binding approach with fuzzy due dates. Advances in Fuzzy Systems, 2021.
11. Abdel-Basset, M., Mohamed, R., Abouhawwash, M., Chakrabortty, R. K., & Ryan, M. J. (2021). A Simple and Effective Approach for Tackling the Permutation Flow Shop Scheduling Problem. Mathematics, 9(3), 270.
12. Zhao, F., Zhang, L., Cao, J., & Tang, J. (2021). A cooperative water wave optimization algorithm with reinforcement learning for the distributed assembly no-idle flowshop scheduling problem. Computers & Industrial Engineering, 153, 107082.
13. Huang, Y. Y., Pan, Q. K., Huang, J. P., Suganthan, P. N., & Gao, L. (2021). An improved iterated greedy algorithm for the distributed assembly permutation flowshop scheduling problem. Computers & Industrial Engineering, 152, 107021.
14. Meng, T., & Pan, Q. K. (2021). A distributed heterogeneous permutation flowshop scheduling problem with lot-streaming and carryover sequence-dependent setup time. Swarm and Evolutionary Computation, 60, 100804.
15. Santucci, V., Baioletti, M., & Milani, A. (2015). Algebraic differential evolution algorithm for the permutation flowshop scheduling problem with total flowtime criterion. IEEE Transactions on Evolutionary Computation, 20(5), 682-694.
16. Pan, Q.K., Tasgetiren, M.F. and Liang, Y.C., 2008. A discrete differential evolution algorithm for the permutation flow shop scheduling problem. *Computers & Industrial Engineering*, *55*(4), pp.795-816.
17. Koulamas, C. and Kyparisis, G.J., 2001. The three-stage assembly flow shop scheduling problem. *Computers & Operations Research*, *28*(7), pp.689-704.
18. Jiang, E. D., & Wang, L. (2019). An improved multi-objective evolutionary algorithm based on decomposition for energy-efficient permutation flow shop scheduling problem with sequence-dependent setup time. International Journal of Production Research, 57(6), 1756-1771.
19. Yagmahan, B. and Yenisey, M.M., 2010. A multi-objective ant colony system algorithm for flow shop scheduling problem. *Expert Systems with Applications*, *37*(2), pp.1361-1368.
20. Tseng, C.T. and Liao, C.J., 2008. A discrete particle swarm optimization for lot-streaming flow shop scheduling problem. *European Journal of Operational Research*, *191*(2), pp.360-373.
21. Li, M., Lei, D., & Cai, J. (2019). Two-level imperialist competitive algorithm for energy-efficient hybrid flow shop scheduling problem with relative importance of objectives. Swarm and Evolutionary Computation, 49, 34-43.
22. Hao, J. H., Li, J. Q., Du, Y., Song, M. X., Duan, P., & Zhang, Y. Y. (2019). Solving distributed hybrid flowshop scheduling problems by a hybrid brain storm optimization algorithm. IEEE Access, 7, 66879-66894.
23. Shao, W., Shao, Z., & Pi, D. (2020). Modeling and multi-neighborhood iterated greedy algorithm for distributed hybrid flow shop scheduling problem. Knowledge-Based Systems, 194, 105527.
24. Zhang, C., Tan, J., Peng, K., Gao, L., Shen, W., & Lian, K. (2021). A discrete whale swarm algorithm for hybrid flow-shop scheduling problem with limited buffers. Robotics and Computer-Integrated Manufacturing, 68, 102081.

ISSN:2093-4777 | E-ISSN:2093-6931
DOI: 10.5123/inj.2024.3.inj3
Vol. 28 Iss. 3 (2024)