Optimizing Flow Shop Scheduling for Energy Efficiency Using Dandelion Algorithm

Nine Dwi Milleniawatia), Annisa Kesy Garsideb), Thomy Eko Saputroc), and Amelia Khoidird)

Industrial Engineering Department, University of Muhammadiyah Malang, Malang, Indonesia

b) Corresponding author: <annisa@umm.ac.id>

a) <ninedwimilleniawati@gmail.com>

c) <thomysaputro@umm.ac.id>

d) [ameliakhoidir@umm.ac.id](mailto:ameliakhoidir@umm.ac.id)

**Abstract.**  The Permutation Flow Shop Scheduling Problem (PFSSP) faces serious challenges in optimizing job sequences over numerous machines since it necessitates balancing multiple parameters to get optimal scheduling solutions. Specifically, reducing overall energy consumption while increasing operational efficiency is still a major concern for manufacturing systems. This study aims to evaluate the effectiveness of the Dandelion Algorithm (DA) in reducing energy consumption compared to traditional scheduling approaches. The DA was inspired by the natural dispersal patterns of dandelion seeds and is employed to schedule 50 jobs in a manufacturing scenario, with implementation using MATLAB software. The study compares the DA's performance to the First Come, First Served (FCFS) technique, particularly a focus on energy consumption evaluations and efficiency improvements. The results show that the DA reduces energy consumption by over 2% compared to the FCFS approach, with a significant drop of 22.95 kWh in total energy consumption. The results presented highlight the DA's potential to improve energy efficiency in production scheduling, particularly in industries where energy costs are a critical factor to reduce operational costs and meet sustainability goals.

**Keywords:** Flow Shop Scheduling, Dandelion Algorithm, Energy Consumption.

# INTRODUCTION

The efficient scheduling of production processes is a fundamental challenge in modern manufacturing, particularly as industries attempt to balance both productivity with sustainability [[1](#_ENREF_1)]. Among the numerous scheduling problems encountered in industrial operations, the Permutation Flow Shop Scheduling Problem (PFSSP) becomes of particular importance. The Permutation Flow Shop Scheduling Problem (PFSSP) requires scheduling 𝑛 jobs across 𝑚 machines, with each job following the identical sequence of operations [[2](#_ENREF_2)]. The problem has an immediate impact on the energy consumption of production systems and operational efficiency [[3](#_ENREF_3)]. As industries face increasing pressure to stay in line with serious environmental standards while maintaining competitive production levels, optimizing scheduling techniques within the PFSSP becomes essential. The rising complexity of manufacturing environments highlights the limitations of traditional scheduling approaches, such as the First Come, First Serve (FCFS), in terms of energy efficiency and operational performance [[4](#_ENREF_4)]. This emphasizes the necessity for innovative methods that incorporate energy considerations into scheduling.

The search for novel approaches that address both energy consumption and scheduling efficiency has become essential for striking the right balance in today's manufacturing scene, establishing the way for more successful and environmentally responsible scheduling techniques. A review of current literature suggests that different heuristic and metaheuristic methods have been investigated to address the PFSSP, such as Genetic Algorithms [[5](#_ENREF_5)], Simulated Annealing [[3](#_ENREF_3)], and Particle Swarm Optimization [[6](#_ENREF_6)]. In addition, several nature-inspired metaheuristic algorithms have also been developed to optimize the PFSSP [[7-10](#_ENREF_7)]. However, nature-inspired metaheuristic algorithms have some limitations [[11](#_ENREF_11)]. For instance, the performance is sensitive to parameter settings and might struggle with scalability challenges while applied to large-scale problems. Furthermore, these studies frequently rely on traditional optimization techniques, which do not completely address the most recent energy efficiency criteria as well as leverage sophisticated metaheuristic approaches.

Despite advancements in scheduling algorithms, there remains a gap in addressing the specific challenges of reducing energy consumption throughout the PFSSP. A notable gap in current research is the inability to implement the Dandelion Algorithm to schedule problems such as PFSSP. Although several metaheuristic approaches have been investigated, the Dandelion Algorithm has yet to be applied to this particular situation. This study aims to fill this gap by applying the DA to optimize scheduling with a focus on decreasing energy consumption. Furthermore, this study contributes to the field of production scheduling by introducing the Dandelion Algorithm (DA) as an innovative approach to solving the Permutation Flow Shop Scheduling Problem (PFSSP) with a focus on energy efficiency. The Dandelion algorithm was first introduced by [Li, et al. [12]](#_ENREF_12) as a metaheuristic optimization method inspired by the foraging behavior of dandelion seeds. The algorithm leverages the concept of seed dispersal to navigate complex search environments, therefore being particularly beneficial to address multidisciplinary optimization problems. A study by [Zhao, et al. [11]](#_ENREF_11) employed the Dandelion Optimizer algorithm to precisely determine the parameters of photovoltaic models demonstrating its effectiveness for optimizing the performance of solar panels. The Dandelion algorithm was implemented by [Ali, et al. [13]](#_ENREF_13) and demonstrates the effectiveness of the algorithm to combine automatic voltage regulation and load frequency control in multi-area. Additionally, [Zhu, et al. [14]](#_ENREF_14) presented a self-adaptive and efficient version of the DA for feature selection in credit card fraud detection. The application demonstrates the algorithm's versatility and efficiency in identifying pertinent features. Thus, these findings highlight the significance of further study into the Dandelion algorithm.

# METHODS

## PERMUTATION FLOW SHOP SCHEDULING PROBLEM (PFSSP)

Several key assumptions are used to solve the Permutation Flow Shop Scheduling Problem (PFSSP): (1) each machine is capable of processing only one job at any given time. (2) A job can only be processed on one machine at a time. (3) Jobs are processed in a strict sequence, moving from the first machine to subsequent machines in a predefined order. (4) all jobs share a common completion time, denoted by 𝐶. (5) Particular attention is given to identifying the optimal permutation of job sequences to ensure the most efficient scheduling.

The notation used in this PFSSP problem is as follows [[15](#_ENREF_15)]:

|  |  |
| --- | --- |
| : Number of jobs | : Power consumption for machine |
| : Number of machines | : Completion time for job on machine |
| : Number of machines | : Idle time for machine |
| : Processing time for job on machine | : Completion time of the last machine |
| : Processing speed coefficient at speed level | TEC: Total energy consumption |
| : Energy transformation coefficient at speed level | : Idle energy transformation coefficient for each job processed on machine |
|  |  |

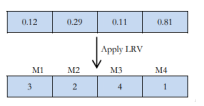
The PFSSP problem to minimize the energy consumption is formulated as follows:

Objective Function:

|  |  |  |
| --- | --- | --- |
|  | | (1) |
| Subject to : | | (2) |
|  | | (3) |
|  | | (4) |
|  | | (5) |
|  | | (6) |
|  | | (7) |
|  | | (8) |
|  | | (9) |
|  | | (10) |
|  |  | (11) |
|  | | (12) |

## LARGEST RANK VALUE (LRV)

The Largest Ranked Value (LRV) is a pivotal concept in converting continuous values into discrete sequences, particularly in the context of the Permutation Flow Shop Scheduling Problem (PFSSP), which is an NP-Hard problem [[16](#_ENREF_16)]. The Dandelion Algorithm (DA), inherently a continuous optimization method, necessitates periodic reduction of the search space to ensure efficient exploration of the solution space. LRV serves as a critical tool in this process, enabling the transformation of continuous variables into discrete values that can be effectively processed by DA, as seen in **FIGURE 1**. This integration is essential for tackling the PFSSP, where discrete sequences are essential for scheduling tasks efficiently. By leveraging LRV, DA can efficiently navigate the vast solution space of PFSSP, leading to more accurate and efficient scheduling solutions [[17](#_ENREF_17)].



**FIGURE 1.** Application of LRV

## DANDELION ALGORITHM

The mathematical model of the Dandelion Algorithm is a metaheuristic algorithm inspired by the behavior of finding the optimal reproduction location when dandelion seeds mature that was proposed by [Li, et al. [12]](#_ENREF_12). They emphasized that the flight behavior of dandelion seeds is also essential in biological evolution. The dispersal process involves three stages: rising, descending, and landing. The stages are replicated in the algorithm to represent exploration and convergence within a search space. The following subsection describes how the Dandelion Algorithm mathematically models within the Dandelion Algorithm.

1. Rising stage

During the rising stage, dandelion seeds must reach a sufficient altitude to be efficiently disseminated away from the parent plant. The height attained by the seeds during this phase is influenced by various environmental factors, including wind speed and air humidity. The environmental conditions impacting the dandelion seed dispersal process are classified into two distinct scenarios.

Case 1 on a clear day, wind speeds are typically characterized by a lognormal distribution , leading to a higher chance for dandelion seeds to travel long distances. Consequently, the Dandelion Algorithm emphasizes exploration in such conditions. The wind's strength determines how high and far the seeds rise and scatter, with vortexes adjusting to create a spiral ascent. The mathematical formulations of dandelion seeds on the clear day of the rising stage are as follows:

|  |  |
| --- | --- |
|  | (13) |
|  | (14) |
|  | (15) |
|  | (16) |
|  | (17) |
|  | (18) |
|  | (19) |
|  | (20) |

The position of a dandelion seed during an iteration is denoted by 𝑥t. A randomly chosen position within the search space during the iteration is represented by 𝑥s. ln Y follows a lognormal distribution. The adaptive parameter used to adjust the step length of the search is 𝛼. The coefficients 𝑣𝑥 and 𝑣𝑦 represent the rising part of the dandelion's movement generated through various vortices.

Case 2 on a rainy day, dandelion seeds experience increased air resistance and higher humidity, which impede their ascent with the wind. Consequently, the seeds are confined to a more localized area and focus on exploiting their immediate surroundings rather than dispersing over long distances. The mathematical formulations of dandelion seeds on the rainy day of the rising stage are as follows:

|  |  |
| --- | --- |
|  | (21) |
|  | (22) |
|  | (23) |
|  | (24) |

Where, 𝑘 is applied to maintain the local search domain of the agent, and randn represents a random value that follows a standard normal distribution.

2. Descending Stage

During this stage, the DA algorithm focuses on exploration. Dandelion seeds fall steadily after reaching a set height, as mimicked in DA using Brownian motion. As Brownian motion follows a normal distribution, it enables individuals to traverse multiple search areas during iterations. The average position after the rising stage is used to reflect the stability of descent, guiding the population toward promising regions. The mathematical formulations of the descending stage are as follows:

|  |  |
| --- | --- |
|  | (25) |
|  | (26) |

Where 𝛽𝑡 denotes a randomly generated value derived from the standard normal distribution, which is used to model Brownian motion. This Brownian motion simulates the random movement observed in dandelion seeds. During the 𝑖-th iteration, represents the average location of the population according to Equation (25).

3. Landing Stage

Effective metaheuristic algorithms must balance exploration and exploitation. In the Dandelion Algorithm, exploration involves generating diverse solutions, while exploitation focuses on refining these solutions by searching within promising regions. The dandelion seeds land randomly but, through iterations, the algorithm converges on an optimal solution. This convergence is achieved by leveraging information from the best solutions to guide the search in local neighborhoods. The mathematical formulations of the landing stage are as follows:

|  |  |
| --- | --- |
|  | (27) |
|  | (28) |
|  | (29) |
|  | (30) |

Where, represents the best position of the agent in each iteration. denotes the Levy flight function, which is used in the Dandelion Algorithm to model the random steps of dandelion seeds during dispersal. Inspired by the Levy distribution, this function allows for large and unpredictable steps. By using the Levy function, the algorithm enhances its exploration capabilities, improving the chance of finding better or optimal solutions in complex search spaces.

4. Pseudocode of the Proposed Algorithm

***Input:*** *the population size pop, maximum number of iterations T and variable dimension Dim*

***Output:*** *the optimal dandelion seed and its fitness value*

*1: Initialize dandelion seeds X of DO*

*2: Calculate the fitness value f of each dandelion seeds*

*3: Select the optimum dandelion seed according to fitness values*

*4:* ***while*** *(t<T)* ***do***

***/\*Rise stage\*/***

*5:* ***if*** *randn() <1.5* ***do***

*6: Generate adaptive parameters using Eq. (13)*

*7: Update dandelion seeds using Eq. (12)*

*8:* ***else if do***

*9: Generate adptive parameters using Eq. (23)*

*10: Update dandelion seeds using Eq. (20)*

*11:* ***end if***

***/\*Decline stage\*/***

*12: Update dandelion seeds using Eq. (24)*

***/\*Land stage\*/***

*13: Update dandelion seeds using Eq. (26)*

*14: Arrange dandelion seeds from good to bad according to fitness values into a permutation sequence using LRV*

*15: Update*

*16: if*

*17:*

*18:* ***end if***

*19:* ***end while***

*20:* ***Return***  *and*

## DATA COLLECTION

The data for this study were collected through observations conducted at a PVC product manufacturing company located in Indonesia. The case study involves scheduling 50 jobs utilizing the permutation flow shop scheduling method. The energy consumption data are detailed in **TABLE 1**, while the speed parameters are provided in **TABLE 2**. Furthermore, the study parameters include a population size of 500 and 1000 iterations.

**TABLE 1**. Energy consumption data

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Machine  Code | Energy Consumption () (kw) | | | |
| Idle | Slow | Normal | Fast |
| 1 | 0,566 | 10,948 | 11,887 | 13,138 |
| 2 | 0,313 | 13,585 | 15,015 | 19,662 |
| 3 | 0,164 | 2,234 | 3,277 | 4,618 |

**TABLE 2**. Parameter Data

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Machine Code |  |  | | |  | | |
| *S* | *N* | *F* | *S* | *N* | *F* |
| 1 | 0,048 | 0,921 | 1 | 1,105 | 0,667 | 1 | 1,33 |
| 2 | 0,021 | 1,143 | 1,263 | 1,654 | 0,75 | 1 | 1,5 |
| 3 | 0,05 | 0,188 | 0,276 | 0,388 | 0,75 | 1 | 2 |

# RESULTS AND DISCUSSION

The company currently employs the First Come First Served (FCFS) method for production scheduling, where jobs are processed in the sequence they are received. The FCFS method consumes a total of 99,495.951 kWh as shown in **FIGURE 2**. The consumption of energy is an outcome of the method's essential method, which processes orders in the order they receive without regard for the processing time required for each order. The approach ensures that production is in sync with consumer demand, but it may not always maximize energy usage, leading to higher overall energy consumption than more complex scheduling algorithms. However, adopting the Dandelion Algorithm for scheduling with population sizes of 500 and 1000 iterations resulted in a slightly reduced overall energy consumption of 99,470 KWh. The DA method, implemented with MATLAB software, shows a slight gain in energy efficiency over the FCFS method. It indicates that the Dandelion Algorithm may provide a more optimal scheduling method, which could end in lower energy consumption during the production process. The findings show that even minor changes to scheduling algorithms can have a considerable impact on total energy efficiency.

**FIGURE 2.** Result of energy consumption

**TABLE 4** compares energy consumption for the First Come, First Served (FCFS) approach and the Dandelion Algorithm. The energy consumption of the FCFS approach is 99,495 kWh, while the Dandelion Algorithm is slightly lower at 99,470 kWh. Thus offers a difference of 22.95 kWh, indicating a 2% increase in energy efficiency with the Dandelion Algorithm. The Dandelion Algorithm offers better outcomes, due to its advanced optimization capabilities [[18](#_ENREF_18)]. Despite the FCFS method, which handles jobs sequentially, the Dandelion Algorithm offers a more dynamic approach. It employs a metaheuristic approach inspired by the natural distribution patterns of dandelion seeds, allowing it to investigate a broader range of task sequences and develop solutions more efficiently. The capacity enables the algorithm to eliminate idle times and optimize job scheduling, which leads to reduced consumption of energy. Consequently, the Dandelion Algorithm's ability to adapt and adjust the scheduling process leads to higher energy efficiency than the traditional FCFS approach.

**TABLE 4**. Comparison of energy consumption

|  |  |  |  |
| --- | --- | --- | --- |
| Method | Energy Consumption (kWh) | Deviation (kWh) | Efficiency |
| FCFS | 99495,951 | 22,95 | 2% |
| Dandelion Algorithm | 99470 |

The findings of this study have major consequences for the manufacturing industry, particularly in terms of sustainable production. The Dandelion Algorithm (DA) has demonstrated the ability to efficiently minimize energy consumption which makes it a significant tool for manufacturers to reduce operational expenses while meeting severe environmental regulations. The DA's success in optimizing energy utilization offers up opportunities for study into incorporating similar algorithms into other difficult scheduling problems, perhaps leading to broader applications in a variety of industrial sectors.

## SENSITIVITY ANALYSIS

Sensitivity analysis is used to assess the impact of changes in certain variables on the objective function. Specifically, the research investigates the effects of varying machine speeds, which were initially constant. Machine speeds are now determined by the processing time on each machine. The influence of these speed variations is observed through changes in energy consumption results. The sensitivity analysis reveals that altering machine speeds affects the objective function values. This analysis incorporates a combination of 27 speed scenarios, as presented in **TABLE 5**.

Based on **FIGURE 3**, a rising trend is observed in the data, indicating an upward movement in the graph. The highest energy consumption occurs in Scenario 27, while the lowest is found in Scenario 1. Scenario 27 leads to high energy consumption due to the use of high speeds across all machines, which shortens the job processing time and consequently increases energy usage. Conversely, Scenario 1 yields low energy consumption because it involves lower machine speeds, leading to longer processing times. Therefore, Scenario 1 represents the speed configuration that results in the lowest energy consumption.

**TABLE 5.** Scenario of sensitivity analysis

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Scenario | | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 |
| Machine | M1 | *Slow* | *Slow* | *Normal* | *Slow* | *Normal* | *Normal* | *Slow* | *Fast* | *Fast* |
| M2 | *Slow* | *Normal* | *Slow* | *Slow* | *Slow* | *Normal* | *Slow* | *Slow* | *Slow* |
| M3 | *Slow* | *Normal* | *Normal* | *Normal* | *Slow* | *Slow* | *Fast* | *Normal* | *Fast* |
| Energy Consumption | | 85390 | 86940 | 87520 | 88010 | 88450 | 88550 | 89450 | 89670 | 90290 |
| Scenario | | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 |
| Machine | M1 | *Slow* | *Normal* | *Normal* | *Slow* | *Fast* | *Slow* | *Slow* | *Normal* | *Fast* |
| M2 | *Normal* | *Normal* | *Slow* | *Fast* | *Normal* | *Normal* | *Fast* | *Fast* | *Slow* |
| M3 | *Slow* | *Normal* | *Fast* | *Normal* | *Normal* | *Fast* | *Slow* | *Fast* | *Slow* |
| Energy Consumption | | 90450 | 90500 | 90930 | 91770 | 92140 | 92150 | 92440 | 92900 | 93040 |
| Scenario | | 19 | 20 | 21 | 22 | 23 | 24 | 25 | 26 | 27 |
| Machine | M1 | *Normal* | *Fast* | *Slow* | *Fast* | *Fast* | *Normal* | *Fast* | *Normal* | *Fast* |
| M2 | *Fast* | *Fast* | *Fast* | *Normal* | *Fast* | *Fast* | *Normal* | *Normal* | *Fast* |
| M3 | *Normal* | *Normal* | *Fast* | *Slow* | *Slow* | *Slow* | *Fast* | *Fast* | *Fast* |
| Energy Consumption | | 93330 | 95810 | 98550 | 99150 | 99280 | 99310 | 99320 | 99470 | 99970 |

.

**FIGURE 3.** Sensitivity analysis

# CONCLUSIONS

This study proposes the application of the Dandelion Algorithm (DA) as a solution approach for the Permutation Flow Shop Scheduling Problem (PFSSP), with the primary objective of minimizing energy consumption. The DA algorithm has been successfully developed and applied to scheduling problems within the framework of this research. The results indicate that the DA algorithm outperforms the traditional First Come, First Served (FCFS) method, achieving a 2% improvement in efficiency and a reduction in energy consumption by 22.95 kWh. These findings demonstrate the DA algorithm’s effectiveness in addressing the PFSSP with a focus on energy reduction. However, this study has some limitations. The scope of the study was confined to a specific manufacturing and may not fully capture the complexity of PFSSP in different industries or larger-scale operations. Additionally, the analysis was based on a fixed set of parameters and scenarios, which might not encompass all potential variations in machine speeds and job characteristics. Future research should explore the DA's applicability in diverse manufacturing environments and larger-scale problems to validate its effectiveness more broadly. Investigating different parameter settings, alternative metaheuristic algorithms, and hybrid approaches could also provide deeper insights into optimizing energy consumption and enhancing scheduling efficiency.

# References

1. M. Parente, G. Figueira, P. Amorim, and A. Marques, "Production scheduling in the context of Industry 4.0: review and trends," *International Journal of Production Research,* vol. 58, no. 17, pp. 5401-5431, 2020/09/01 2020, doi: 10.1080/00207543.2020.1718794.
2. M. Abdel-Basset, R. Mohamed, M. Abouhawwash, V. Chang, and S. S. Askar, "A Local Search-Based Generalized Normal Distribution Algorithm for Permutation Flow Shop Scheduling," *Applied Sciences*, vol. 11, no. 11*,* doi: 10.3390/app11114837.
3. B. Khurshid, S. Maqsood, M. Omair, B. Sarkar, I. Ahmad, and K. Muhammad, "An Improved Evolution Strategy Hybridization With Simulated Annealing for Permutation Flow Shop Scheduling Problems," *IEEE Access,* vol. 9, pp. 94505-94522, 2021, doi: 10.1109/ACCESS.2021.3093336.
4. E. O. Oyebode and O. A. Ojesanmi, "Comparative Evaluation of the Firefly, PSO and FCFS Scheduling Algorithms for Energy Management in Computer Data Center," *African Journal of Computing & ICT,* vol. 14, no. 2, 2021.
5. P. Zou, M. Rajora, and S. Y. Liang, "Multimodal Optimization of Permutation Flow-Shop Scheduling Problems Using a Clustering-Genetic-Algorithm-Based Approach," *Applied Sciences*, vol. 11, no. 8*,* doi: 10.3390/app11083388.
6. L. A. Bewoor, V. Chandraprakash, and S. Sapkal, "Evolutionary hybrid particle swarm optimization algorithm to minimize makespan to schedule a flow shop with no wait," *Journal of Engineering Science and Technology,* vol. 14, no. 2, pp. 609-628, 2019.
7. A. Abuhamdah, "Modified Hybrid Moth Optimization Algorithm for PFSS Problem," *SN Computer Science,* vol. 4, no. 3, p. 298, 2023/03/30 2023, doi: 10.1007/s42979-023-01743-y.
8. D. M. Utama, D. S. Widodo, M. F. Ibrahim, K. Hidayat, T. Baroto, and A. Yurifah, "The hybrid whale optimization algorithm: A new metaheuristic algorithm for energy-efficient on flow shop with dependent sequence setup," *Journal of Physics: Conference Series,* vol. 1569, no. 2, p. 022094, 2020/07/01 2020, doi: 10.1088/1742-6596/1569/2/022094.
9. X. Li and M. Yin, "A discrete artificial bee colony algorithm with composite mutation strategies for permutation flow shop scheduling problem," *Scientia Iranica,* vol. 19, no. 6, pp. 1921-1935, 2012/12/01/ 2012, doi: <https://doi.org/10.1016/j.scient.2012.10.034>.
10. P. Wu, Q. Yang, W. Chen, B. Mao, and H. Yu, "An Improved Genetic‐Shuffled Frog‐Leaping Algorithm for Permutation Flowshop Scheduling," *Complexity,* vol. 2020, no. 1, p. 3450180, 2020.
11. S. Zhao, T. Zhang, S. Ma, and M. Chen, "Dandelion Optimizer: A nature-inspired metaheuristic algorithm for engineering applications," *Engineering Applications of Artificial Intelligence,* vol. 114, p. 105075, 2022/09/01/ 2022, doi: <https://doi.org/10.1016/j.engappai.2022.105075>.
12. X. Li, S. Han, L. Zhao, C. Gong, and X. Liu, "New dandelion algorithm optimizes extreme learning machine for biomedical classification problems," *Computational intelligence and neuroscience,* vol. 2017, no. 1, p. 4523754, 2017.
13. T. Ali, S. A. Malik, A. Daraz, S. Aslam, and T. Alkhalifah, "Dandelion Optimizer-Based Combined Automatic Voltage Regulation and Load Frequency Control in a Multi-Area, Multi-Source Interconnected Power System with Nonlinearities," *Energies*, vol. 15, no. 22*,* doi: 10.3390/en15228499.
14. H. Zhu, M. Zhou, Y. Xie, and A. Albeshri, "A Self-Adapting and Efficient Dandelion Algorithm and Its Application to Feature Selection for Credit Card Fraud Detection," *IEEE/CAA Journal of Automatica Sinica,* vol. 11, no. 2, pp. 377-390, 2024, doi: 10.1109/JAS.2023.124008.
15. H. Öztop, M. F. Tasgetiren, D. T. Eliiyi, Q.-K. Pan, and L. Kandiller, "An energy-efficient permutation flowshop scheduling problem," *Expert Systems with Applications,* vol. 150, p. 113279, 2020/07/15/ 2020, doi: <https://doi.org/10.1016/j.eswa.2020.113279>.
16. D. M. Utama, "Minimizing Number of Tardy Jobs in Flow Shop Scheduling Using A Hybrid Whale Optimization Algorithm," *Journal of Physics: Conference Series,* vol. 1845, no. 1, p. 012017, 2021/03/01 2021, doi: 10.1088/1742-6596/1845/1/012017.
17. M. Abdel-Basset, R. Mohamed, M. Abouhawwash, R. K. Chakrabortty, and M. J. Ryan, "A Simple and Effective Approach for Tackling the Permutation Flow Shop Scheduling Problem," *Mathematics*, vol. 9, no. 3*,* doi: 10.3390/math9030270.
18. Y. Zhao, J. Dong, X. Li, H. Chen, and S. Li, "A binary dandelion algorithm using seeding and chaos population strategies for feature selection," *Applied Soft Computing,* vol. 125, p. 109166, 2022/08/01/ 2022, doi: <https://doi.org/10.1016/j.asoc.2022.109166>.