Penerbit Akademia Baru

Journal of Advanced Research Design

Journal homepage:
https://akademiabaru.com/submit/index.php/ard
ISSN: 2289-7984

Minimizing Total Production Cost in Hybrid Flow Shop Scheduling using Taguchi Enhanced Particle Swarm Optimization Algorithm

Wasif Ullah[1], Mohd Fadzil Faisae Ab. Rashid[2,*], Muhammad Ammar Nik Mu'tasim[2]

[1] Faculty of Manufacturing and Mechatronics Engineering Technology, Universiti Malaysia Pahang Al-Sultan Abdullah, 26600 Pekan, Pahang, Malaysia
[2] Faculty of Mechanical and Automotive Engineering Technology, Universiti Malaysia Pahang Al-Sultan Abdullah, 26600 Pekan, Pahang, Malaysia

ARTICLE INFO

Article history:
Received 27 January 2025
Received in revised form 21 February 2025
Accepted 2 May 2025
Available online 23 May 2025s

Keywords:
Cost optimization; PSO tuning; hybrid flow shop; metaheuristics; Taguchi design

ABSTRACT

This study uses metaheuristic optimization algorithms to minimize the total production cost (TPC) in a hybrid flow shop scheduling (HFS) environment. Scheduling jobs in manufacturing systems is vital for fulfilling customer demands and improving efficiency. In this research, four well-established metaheuristic algorithms, namely Tuned Particle Swarm Optimization (TPSO), Standard particle swarm optimization (PSO), Sine cosine algorithm (SCA) and Arithmetic optimization algorithm (AOA), were explored for TPC optimization in HFS environment. Through experimental analysis, TPSO consistently provided the best solutions regarding mean fitness, outperforming other algorithms in a maximum of 12 benchmark test problems. Taguchi's Design of Experiment (DOE) was utilized to identify the most influential parameter configurations for PSO. The findings highlight the effectiveness of TPSO in minimizing production costs and improving productivity in HFS. This research contributes to production scheduling and offers insights for organizations striving to optimize manufacturing systems utilizing the HFS environment.

1. Introduction

Production scheduling refers to assigning jobs to available machines in manufacturing systems to optimize single or multiple objectives. The scheduling of jobs is essential to meet the customers' demands and avoid delays while enhancing the efficiency of the manufacturing system. Scheduling can be classified into job shop and flow shop categories, depending on the flow of resources. One of the well-established flow shops is known as a hybrid flow shop (HFS) [1]. The HFS is a manufacturing setup comprising multiple stages, each with numerous parallel machines. Each job must pass through the stages in the same order, being processed on one of the machines at each stage. The HFS poses a complex combinatorial optimization challenge, especially when it comes to determining the sequence of job processing and the assignment of jobs to machines. Both of these factors serve as design variables in the standard HFS [2].

* *Corresponding author*
E-mail address: ffaisae@umpsa.edu.my

https://doi.org/10.37934/ard.132.1.4151

When the cost is the objective function in HFS, it is known as the cost-based hybrid flow shop (CHFS). Many researchers have contributed to the CHFS optimization field with various optimization objectives related to cost in HFS production systems. In the literature, the dominant optimization objectives in CHFS were production and energy costs [3]. In some problems, the makespan was somehow optimized in multi-objective cost criterion problems [4]. In addition to these primary costs, other factors such as material, tardiness and storage costs were also taken into account [5,6]. Moreover, specific papers have also considered transportation, rejected jobs, operating, setup, overtime, adjustment and resource allocation costs [7-10].

Organizations strive to optimize total production cost (TPC) to increase productivity and reduce the economic imbalance [11]. The main expenses encountered in manufacturing are the material, transportation, electrical energy, maintenance, tardiness, accessories, labour and late penalty [12]. This paper covered four significant costs: labour costs, the electrical energy cost of the machine in operating condition, the preventive maintenance cost of the running machines to produce parts and late penalty costs when jobs are not completed on their due dates. Accumulating these four costs leads to the TPC, which is the optimization objective in this paper.

High-level optimisation techniques are used to optimize the objective function in CHFS, known as the metaheuristic's algorithms. The metaheuristics algorithms are used to find the optimal solution for a given problem under constraints. Metaheuristics are divided into three categories: nature-inspired, population-based and intelligent swarm. In this paper, four popular and well-established metaheuristics algorithms named Tuned particle swarm optimization (TPSO), Standard particle swarm optimization (PSO), Sine cosine algorithm (SCA) and Arithmetic optimization algorithm (AOA) were selected for optimization of TPC. The algorithm that consistently gives the best solution over the number of iterations and optimization runs even for large and small-scale optimization problems is declared the best optimization algorithm.

PSO is a population-based metaheuristic inspired by the collective behaviour observed in bird flocks. It can optimize complex combinatorial optimization problems. However, some standard controlling input parameters in PSO influence the objective function. These parameters are the inertia weight w, personal weight $c1$ and global weight $c2$. The $c1$ and $c2$ are also known as correction factors or constant of acceleration. Changing the values of these parameters in their respective ranges is called tuning [13]. In PSO, parameters such as w, $c1$ and $c2$, population size and number of iterations directly influence efficiency and reliability. The orthogonal array technique, alongside signal-to-noise (S/N) ratios, is employed to pinpoint the optimal parameter configurations, enhancing performance characteristics. This paper used the Taguchi design of experiment (DOE) to investigate the best parameter settings for PSO, which returns the best fitness value [14].

Besides PSO, two other widely recognized metaheuristics named SCA and AOA were chosen based on recent advancements to evaluate their performance against TPSO. The AOA iteratively adjusts the values of the decision variables to minimize an objective function, using simple arithmetic operations like addition, subtraction, multiplication and division at each step [15]. Natural phenomena inspire it and evolve the solution towards an optimal value over successive iterations. The SCA is a metaheuristic optimization algorithm that uses sinusoidal waves and randomly generated numbers to develop candidate solutions toward the optimal value [16]. It initializes a random population of solutions and then calculates new solutions by exploring the current best solution using the sine and cosine functions. Over successive iterations, it converges toward the global optimum.

The paper is structured as follows: Section two elaborates on the standard PSO and its operational principles. Section three details the Taguchi DOE. Section four concerns the experimental setup and testing of the benchmark test problems using TPSO, PSO, AOA and SCA. Section five of the paper presents the results and discussions. Finally, the paper will end up with conclusions.

2. Particle Swarm Optimization (PSO)

The PSO algorithm begins by initializing a population of random candidate solutions, referred to as particles. Each particle possesses a position and velocity, representing a potential solution within the search space. Particles iteratively fly through the multi-dimensional search space to find the optimal position according to a fitness function. Critical PSO concepts include particle position (*Xi)* and particle velocity (*Vi)*. Some other variables are the personal best (*Pbest*) and global best (*Gbest*) [17].

Xi represents a candidate solution in the multi-dimensional problem space. Each axis represents one dimension of the search space and an exact position on that axis represents one element of the overall solution. The position is updated over-optimization as the particle flies through the parameter space. The final position represents an approximate or exact solution for the objective function.

Vi dictates position change, representing the step size and direction that a particle in the swarm will move in to update its position. High velocity leads to more global exploration, while low velocity supports more precise local exploitation. Pbest and Gbest influence the velocity found so far by the swarm. Setting appropriate velocities facilitates the exploration of the search space.

Each particle keeps track of the best solution it has found so far individually at a particular position known as the personal best or *Pbest* solution for that particle. Comparing the current solution to the *Pbest* enables improvement based on an individual's exploration through the search space over the algorithm iterations.

At each iteration, the global optimum solution found so far across all particles in the swarm is broadcast to every other particle. This global best-known position, *Gbest*, allows information sharing between particles. Particles can use the *Gbest* to improve their movement through the search space. Communication of the swarm's overall best solution improves convergence. In PSO, each particle with inertia weight *w* is manipulated based on Eq. (1) and Eq. (2):

$$Vi(t+1) = wVi(t) + c1r1 * (Pbest - Xi(t)) + c2r2(Gbest - Xi(t)) \tag{1}$$

$$Xi(t+1) = Xi(t) + Vi(t+1) \tag{2}$$

Where $Vi(t+1)$ and $Xi(t+1)$ are the new velocities and new positions for the particles over the defined iterations. $Vi(t)$ is the velocity of a particle at its current position and $Xi(t)$ is the current position of the particle. Where *r1* and *r2* are the random numbers taken in range *1<r1, r2<0*, once the velocity for the particle has been updated, the position can be updated using Eq. (2) to find the new solution for the candidates [13].

3. Taguchi Design of Experiment (DOE) for PSO Parameters Settings

Taguchi design is a statistical approach using orthogonal array analysis to determine the best parameter setting (factors and levels). A suitable orthogonal array is chosen according to the number of parameters and levels under investigation. The Taguchi orthogonal arrays dictate the specific combinations of parameter levels used in each experimental run [18]. Data analysis is then done to calculate signal-to-noise (*S/N*) ratios, which indicate the factor levels that maximize robustness to noise. The optimal levels are interpreted to define the best parameter settings [13].

As the Taguchi method is applied to PSO in this paper, the parameters to be considered are inertia weight *w*, personal weight *c1* and global weight *c2*. These are the potential parameters which influence the fitness/objective function. This paper's fitness is TPC, so the TPC was calculated for all nine experiments (*L9*) using the associated parameter settings. The optimum parameters were declared corresponding to the best fitness value. The proposed methodology to conduct the Taguchi (DOE) is as follows:

The fitness function to be calculated is the TPC, expressed in Eq. (3):

$$Min\ f(x) = CL + CE + CR + CP \tag{3}$$

The fitness was calculated for five optimization runs and the results were noted. The formulas to calculate associated costs are expressed below.

Labour cost, C_L: The labour cost represents the wages paid to workers for a particular time period. In this paper, C_L is taken constant based on average cost per hour. Additionally, only one worker can operate one machine. C_L is calculated by multiplying all machines' total operating time and hourly wage rates. The operating times used are expressed in minutes. Parameters used include the jobs *J*, the overall production stages *S* and the number of machines *M*. *Pjms* additionally represents the minute processing time for job *j* on machine *m* at stage *s*. Eq. (4) is used to calculate the labour cost. The α_{jsm} is a binary variable with constraints representing the operational condition of machines and can be seen in Eq. (4):

$$C_L = \left(\sum_{j=1}^{J} \sum_{s=1}^{S} \sum_{m=1}^{M} t_{jsm} \cdot \alpha_{jsm}\right) \times \left(\frac{\text{Hourly pay rate}}{60}\right) \tag{4}$$

$$\alpha_{jsm} = \begin{cases} 1, \text{if job } j \text{ is processed on machine } m \text{ at stage } s \\ 0, \text{otherwise} \end{cases}$$

Energy cost, C_E: The electricity cost is the energy utilized by machines while operating the jobs. Electricity utilized for other purposes, such as lighting and ventilation, is excluded. Additionally, standby power consumption when machines are idle is not factored into the electricity cost. The calculation is based on the power rating of each machine and its total run time. Specifically, the machine processing time of machines is multiplied by machines power ratings to determine the power consumption of machines. This paper considers non-identical machines, so the power ratings for machines in the same production stage may vary, impacting the total energy usage. In Eq. (5), the first term calculates the total energy used across all machines in watt-minutes. The second term converts this into *kWh* and multiplies it by the average electricity rate to determine the total electricity cost. *Psm* represents the power rating in watts for each machine *m*. To calculate the C_E, Eq. (5) was used,

$$C_E = \left(\sum_{j=1}^{J} \sum_{s=1}^{S} \sum_{m=1}^{M} t_{jsm} \cdot p_{sm} \cdot \alpha_{jsm}\right) \times \left(\frac{\text{Average electricity tariff}}{60 \times 1000}\right) \tag{5}$$

Maintenance cost, C_R: The maintenance cost refers to expenses incurred to keep production assets in proper working order [19]. Maintenance is typically categorized as either preventive or corrective. This paper only considers scheduled-based maintenance costs that are tied to machine usage. Maintenance scheduling will be determined according to a maximum recommended operating duration tailored to each machine model. The maintenance needs and expenses per machine will vary depending on its operating time accrued and durability specifications. The required number of maintenance sessions is calculated by dividing the total run time of machine *m* in stage *s* (represented by *Tsm*) by the advised maximum operating duration *trsm* for that machine and then rounding up, as shown in Eq. (6). This determines how many complete maintenance cycles need to be performed to cover the total usage:

$$C_R = \sum_{s=1}^{S} \sum_{m=1}^{M} \left\lceil \frac{T_{sm}}{tr_{sm}} \right\rceil \times r_{sm} \tag{6}$$

Late penalty cost, C_P: A late penalty fee is charged to the manufacturer if they fail to fulfil order quantities by the delivery deadline agreed upon with the customer. In this research, a penalty will be applied daily if the requested number of job units is not completed by the specified due date. The longer an order is overdue, the greater the late penalty cost accrued. The term C_j represents the actual completion date when the job order is finished production while D_j is the due date. The term Y_j in Eq. (7) calculates the lateness factor in days by finding the difference between C_j and D_j. C_P in Eq. (8) determines the overall late penalty cost, which increases with the longer the lateness factor Y_j. The lateness factor Y_j is multiplied by penalty charges to calculate C_P using Eq. (8):

$$Y_j = \left(\frac{C_j}{\text{Working time per day in minutes}}\right) - D_j \tag{7}$$

$$Y_j = \begin{cases} y_j & \text{if } y_j > 0 \\ 0 & \text{else} \end{cases}$$

$$C_P = \sum_{j=1}^{J} Y_j \times \text{Daily penalty charges} \tag{8}$$

The PSO parameter settings are depicted in Table 1. The three fundamental potential parameters (*w, c1, c2*) were set in their ranges and labelled A, B and C. Throughout the experiment, consistency was maintained by keeping parameters constant, including 300 iterations, a population size of 30 and conducting 5 optimization runs.

Table 1
Factors and levels settings for PSO

Parameter	Level		
	1	2	3
A(*w*)	0.8	1.4	1.8
B(*C1*)	1.2	1.6	2
C(*C2*)	1.2	1.6	2

The Taguchi experimental setup *L9* orthogonal array is depicted in Table 2.

Table 2
Taguchi experimental setup: Orthogonal array *L9*

DOE	A	B	C	Mean f (MYR)
1	0.8	1.2	1.2	1136.78
2	0.8	1.6	1.6	1320.016
3	0.8	2	2	1213.799
4	1.4	1.2	1.6	1151.133
5	1.4	1.6	2	1157.508
6	1.4	2	1.2	1087.167
7	1.8	1.2	2	996.0775
8	1.8	1.6	1.2	1057.524
9	1.8	2	1.6	1149.695

After conducting the Taguchi method and evaluating the fitness, the means and S/N ratio plots were framed (Figure 1 and 2).

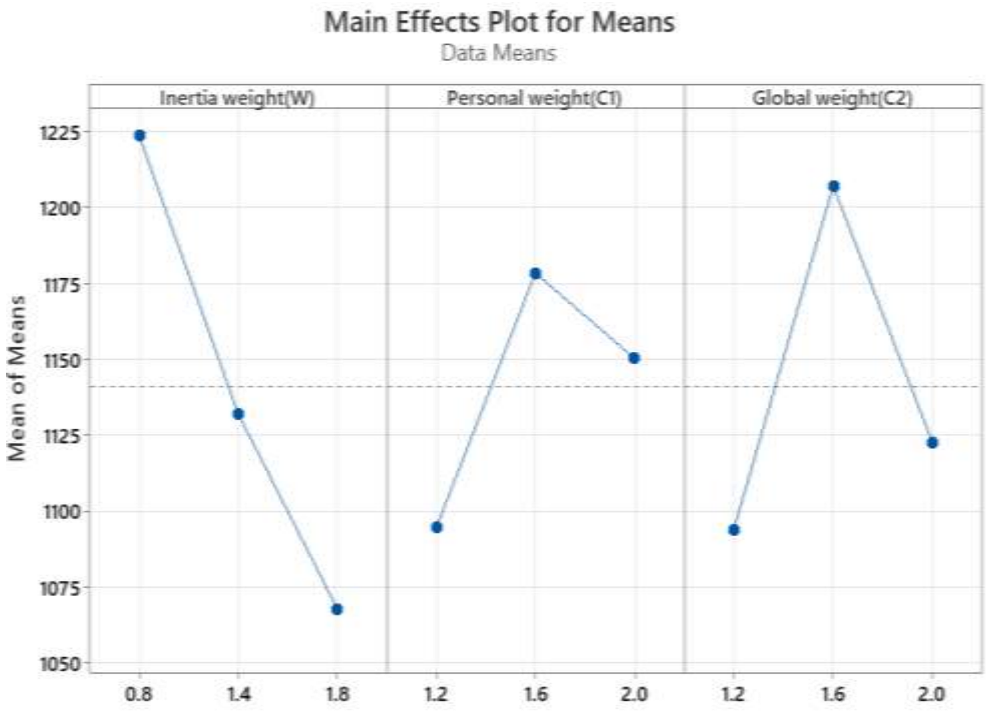

Fig. 1. Means plot

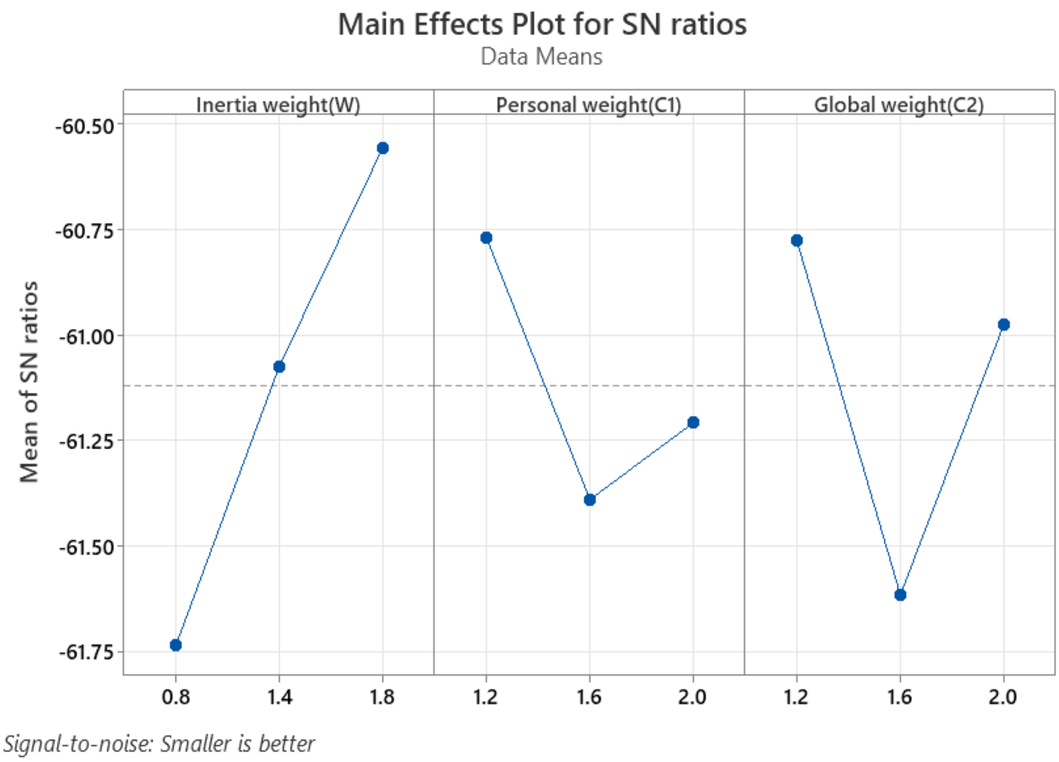

Fig. 2. *S/N* ratios plot

4. Total Production Cost Optimization

A computational experiment was conducted to optimize TPC, employing the TPSO, Standard PSO, SCA and AOA. The purpose of the computational experiment was to investigate the performance of the proposed algorithms while optimizing the TPC.

4.1 Computational Experiment

To optimize the TPC, a computational experiment was conducted using the proposed TPSO, PSO, AOA and SCA. Three hypothetical benchmark test problems were considered defied by *Carlier* and *Neron and* the proposed algorithms were implemented [20]. The benchmark problems are well-established popular hypothetical approaches to evaluate the HFS scheduling problems. The benchmark test problems and machine configurations are depicted in Table 3. The processing time for jobs is randomly generated within the range of {3,20}. The machine configuration indicates the quantity of machines at each stage.

Table 3
Benchmark test problems configurations

Test Problem	No. of Jobs	No. of Machines	Machines Configuration
J10C5a2	10	5	2 2 1 2 2
J10C5b1	10	5	1 2 2 2 2
J10C5c1	10	5	3 3 2 3 3

The performance of the proposed metaheuristics was compared to find the best optimal solution for each test problem. The algorithm which consistently provides the best fitness within maximum test problems was declared the best optimization algorithm in this paper. The PSO has already been utilized in many CHFS problems. However, this paper presents the TPSO to enhance the performance, so the experiment for the PSO was conducted based on the optimum parameter settings. After this, the PSO was tested for the same set of three problems using the default parameters settings. Finally, the benchmark problems were optimized using the AOA and SCA and the results were compared with TPSO. The population size was set to *50,* having iterations of *1000* with *5* optimization runs for all three benchmark problems.

5. Computational Results

The fundamental indicator was calculated using the proposed algorithms for all three benchmark problems. The indicators are the mean fitness, standard deviation (SD), maximum and minimum of average fitness values and the mean computational time (CPU time). These indicators identify the performance and efficiency of the proposed algorithms. The indicators for all three benchmark problems with the associated proposed algorithm are shown in Table 4.

The computational experiment shows that the TPSO algorithm consistently achieved the best optimization results regarding fitness across all three benchmark problems except problem J10c5a2 due to the higher SD shown in Table 4. Specifically, for the J10c5a2 problem, PSO obtained the lowest mean and minimum value of fitness average. However, the SCA recorded the best SD and maximum value, while the CPU time for PSO was outperformed for problem j10c5a2. Similarly, for the J10c5b1 and J10c5c1 problems, TPSO had superior results over other algorithms regarding mean fitness. Regarding the SD, SCA ranked first in all three benchmark problems. The CPU time was recorded as the minimum for the PSO in problems j10c5a2 and j10c5b1, while AOA ranked in problem j10c5c1.

Table 4
Computational results for benchmark test problems

Problem	Indicator	TPSO	PSO	AOA	SCA
J10c5a2	Mean	1085.915	1069.183	1234.443	1144.551
	SD	155.4929	162.3392	139.9214	80.26634
	Max	1322.375	1223.433	1414.111	1224.69
	Min	810.1266	817.9357	1064.12	1026.613
	CPU time	106.6707	69.51446	139.9797	72.1417
J10c5b1	Mean	953.4681	983.607	1005.838	1032.565
	SD	116.0176	178.2303	193.2568	84.1077
	Max	1095.072	1213.911	1257.546	1121.871
	Min	174.3345	734.1997	748.699	897.5344
	CPU time	147.203	69.80002	70.31076	77.9153
J10c5c1	Mean	453.4142	578.9981	469.0988	459.2085
	SD	5.495403	127.3499	3.166313	4.777544
	Max	465.6403	769.9084	472.6058	464.6857
	Min	445.2667	461.5053	465.8212	451.6955
	CPU time	107.6293	87.22908	86.53432	110.1811

Overall, the TPSO had dominated results across all three benchmark problems, achieving superior optimization performance over other algorithms. Meanwhile, the PSO and AOA algorithms had comparable or slightly faster CPU times on some problems. It was concluded from the main findings that the TPSO approach is the best and most effective algorithm among the three tested for solving these types of optimization problems. The consistent and superior performance of TPSO across multiple benchmark instances demonstrates its capabilities and potential as an optimization tool for real-world applications.

5.1 Discussions

The proposed methodology and algorithms' limitations and potential drawbacks must be acknowledged. Firstly, heavy reliance is placed on the availability and quality of input data, meaning any inaccuracies may impact performance and generalizability. Additionally, assumptions about data distribution and relationships were assumed, which may not always hold in real-world scenarios. Furthermore, challenges may arise due to the computational complexity of our algorithms, necessitating efficient optimization strategies for large-scale applications. Lastly, despite extensive experiments and evaluations, specific edge-case scenarios may exist where effectiveness is limited. Addressing these limitations and exploring potential solutions are important directions for future research.

Our research findings carry important implications and potential applicability across various industrial contexts. As our study only concentrated on a specific industry (PCB fabrication), the underlying principles and methodologies can be extended to similar industries. The insights from our research offer valuable guidance for decision-making processes, resource allocation and performance optimization in diverse industrial settings. For example, sectors such as electronics, automotive, aerospace, defence, Pharmaceutical and textile can utilize our findings to enhance operational efficiency and product quality and minimize the total production cost. Moreover, the methodologies developed in our study can be tailored to address specific challenges and requirements in different industrial contexts, facilitating the transferability and generalizability of our findings.

This study has successfully applied the algorithms in addressing the research problem, there are opportunities for further enhancements and modifications to improve their performance and effectiveness. One potential avenue for future research is to investigate alternative optimization algorithmic variations that may offer better computational efficiency or convergence properties. Additionally, incorporating additional data sources into the algorithms could enhance their predictive power and robustness. Furthermore, exploring hybrid approaches that combine multiple algorithms could improve accuracy and generalizability. Another area for improvement is the parameter tuning process, where more sophisticated methods, such as Bayesian optimization, can be explored to search for optimal parameter settings automatically. Finally, considering the dynamic nature of the problem, developing adaptive algorithms that can continuously learn and update their models in real time can strong future direction. Algorithms can improve performance and applicability by addressing these potential enhancements and modifications in various practical scenarios.

While our proposed approach demonstrates promising results in tackling the research problem, it is crucial to recognize the potential challenges and limitations that could emerge during its implementation in real-world manufacturing systems [21]. One key challenge is integrating our approach with existing manufacturing systems and processes. Implementing new methodologies and algorithms requires changes to the existing infrastructure, software systems and data collection mechanisms. This integration process may involve technical complexities, compatibility issues and potential disruptions to the ongoing production operations. Additionally, the availability and quality of data needed to feed into the algorithms can pose a challenge. Manufacturing systems generate vast amounts of data, but ensuring its accuracy, completeness and timeliness is crucial for the success of our approach. Data privacy and security concerns must also be addressed to protect sensitive information.

Furthermore, the scalability of our approach to larger manufacturing systems or multi-site operations is another consideration. Such scenarios' computational requirements and resource constraints may require optimization techniques, parallel processing capabilities or distributed computing architectures. Moreover, the human factor should not be overlooked. The successful implementation of our approach relies not only on the technological aspects but also on the willingness of the workforce to adapt to new processes, acquire new skills and embrace changes. Conducting pilot studies, collaborating with industry partners and involving key stakeholders in the implementation process can help identify and address potential obstacles. Future research can focus on developing frameworks or guidelines for successfully deploying our approach in diverse manufacturing environments. Considering these challenges and limitations, we can ensure a more realistic and practical implementation of our proposed approach in real-world manufacturing systems.

6. Conclusion

This study focused on minimizing the TPC in HFS scheduling. The performance of four established metaheuristic algorithms known as Tuned Particle Swarm Optimization (TPSO), Standard Particle Swarm Optimization (PSO), Sine Cosine Algorithm (SCA) and Arithmetic Optimization Algorithm (AOA) was assessed. The aim was to identify the most effective optimization algorithm for achieving TPC optimization in HFS and to determine the optimal PSO parameters setting. Through meticulous tuning of PSO using statistical software, key parameters such as inertia weight w, personal weight $c1$ and global weight $c2$ were adjusted, resulting in enhanced efficiency and reliability. The Taguchi Design of Experiment (DOE) was carried out to determine the optimal parameter settings for TPSO. The best parameter settings that yielded the lowest TPC were identified by conducting a list of experiments and calculating the TPC for different parameter combinations. The performance of TPSO was compared with three other algorithms: PSO, AOA and SCA. Overall, in most problems, TPSO showed promising results. TPSO outperformed in terms of TPC optimization across the maximum benchmark test problems. Moreover, TPSO obtained the minimum fitness value across all three benchmark test problems. The findings highlight the significance of metaheuristic algorithms, particularly TPSO, in addressing optimization problems in HFS scheduling. The results obtained from this study can provide valuable insights and guidance to organizations striving to minimize TPC and improve productivity in their manufacturing systems.

In summary, this research significantly contributes to production scheduling by showcasing the effectiveness of TPSO in minimizing TPC in HFS. The proposed methodology, combining metaheuristic algorithms and the Taguchi DOE, offers a practical approach to optimizing the performance of manufacturing systems. Future research can explore further enhancements to the algorithms and investigate their applicability in different industrial contexts.

Acknowledgement
The authors thank Universiti Malaysia Pahang Al-Sultan Abdullah for funding this research under the Research Grant (PPU230113).

References

[1] Ruiz, Rubén and José Antonio Vázquez-Rodríguez. "The hybrid flow shop scheduling problem." *European journal of operational research* 205, no. 1 (2010): 1-18. https://doi.org/10.1016/j.ejor.2009.09.024
[2] Peres, Fernando and Mauro Castelli. "Combinatorial optimization problems and metaheuristics: Review, challenges, design and development." *Applied Sciences* 11, no. 14 (2021): 6449. https://doi.org/10.3390/app11146449
[3] Schulz, Sven. "A genetic algorithm to solve the hybrid flow shop scheduling problem with subcontracting options and energy cost consideration." In *Information Systems Architecture and Technology: Proceedings of 39th International Conference on Information Systems Architecture and Technology–ISAT 2018: Part III*, pp. 263-273. Springer International Publishing, 2019. https://doi.org/10.1007/978-3-319-99993-7_23
[4] Li, Jun-qing, Hong-yan Sang, Yu-yan Han, Cun-gang Wang and Kai-zhou Gao. "Efficient multi-objective optimization algorithm for hybrid flow shop scheduling problems with setup energy consumptions." *Journal of Cleaner Production* 181 (2018): 584-598. https://doi.org/10.1016/j.jclepro.2018.02.004
[5] Engin orhan and Batuhan Engin. "Hybrid flow shop with multiprocessor task scheduling based on earliness and tardiness penalties." *Journal of enterprise information management* 31, no. 6 (2018): 925-936. https://doi.org/10.1108/JEIM-04-2017-0051
[6] Fu, Yaping, Hongfeng Wang, Junwei Wang and Xujin Pu. "Multiobjective modeling and optimization for scheduling a stochastic hybrid flow shop with maximizing processing quality and minimizing total tardiness." *IEEE Systems Journal* 15, no. 3 (2020): 4696-4707. https://doi.org/10.1109/JSYST.2020.3014093
[7] Zohali, Hassan, Bahman Naderi, Mohammad Mohammadi and Vahid Roshanaei. "Reformulation, linearization and a hybrid iterated local search algorithm for economic lot-sizing and sequencing in hybrid flow shop problems." *Computers & Operations Research* 104 (2019): 127-138. https://doi.org/10.1016/j.cor.2018.12.008
[8] Jiang, Shenglong, Min Liu, Jinghua Hao and Wangping Qian. "A bi-layer optimization approach for a hybrid flow shop scheduling problem involving controllable processing times in the steelmaking industry." *Computers & Industrial Engineering* 87 (2015): 518-531. https://doi.org/10.1016/j.cie.2015.06.002
[9] Behnamian, J. and SMT Fatemi Ghomi. "Hybrid flowshop scheduling with machine and resource-dependent processing times." *Applied Mathematical Modelling* 35, no. 3 (2011): 1107-1123. https://doi.org/10.1016/j.apm.2010.07.057
[10] Lian, Xiaoyuan, Zhong Zheng, Cheng Wang and Xiaoqiang Gao. "An energy-efficient hybrid flow shop scheduling problem in steelmaking plants." *Computers & Industrial Engineering* 162 (2021): 107683. https://doi.org/10.1016/j.cie.2021.107683
[11] Abu, R., Muhammad Arif Ab Aziz and Zainura Zainon Noor. "Integrated Life Cycle Assessment, Life Cycle Costing and Multi Criteria Decision Making for Food Waste Composting Management." *Journal of Advanced Research in Technology and Innovation Management* 2, no. 1 (2022): 1-12. https://doi.org/10.37934/arbms.21.1.19
[12] Istokovic, David, Mladen Perinic, Maja Vlatkovic and Miran Brezocnik. "Minimizing total production cost in a hybrid flow shop: a simulation-optimization approach." *International Journal of Simulation Modelling* 19, no. 4 (2020): 559-570. https://doi.org/10.2507/IJSIMM19-4-525

[13] Sahoo, Shubhashree, Rabindra Kumar Dalei, Subhendu Kumar Rath and Uttam Kumar Sahu. "Selection of PSO parameters based on Taguchi design-ANOVA-ANN methodology for missile gliding trajectory optimization." *Cognitive Robotics* 3 (2023): 158-172. https://doi.org/10.1016/j.cogr.2023.05.002

[14] He, Yan, Wei Jin Ma and Ji Ping Zhang. "The parameters selection of PSO algorithm influencing on performance of fault diagnosis." In *MATEC Web of conferences*, vol. 63, p. 02019. EDP Sciences, 2016. https://doi.org/10.1051/matecconf/20166302019

[15] Abualigah, Laith, Ali Diabat, Seyedali Mirjalili, Mohamed Abd Elaziz and Amir H. Gandomi. "The arithmetic optimization algorithm." *Computer methods in applied mechanics and engineering* 376 (2021): 113609. https://doi.org/10.1016/j.cma.2020.113609

[16] Mirjalili, Seyedali. "SCA: a sine cosine algorithm for solving optimization problems." *Knowledge-based systems* 96 (2016): 120-133. https://doi.org/10.1016/j.knosys.2015.12.022

[17] Walczak, Beata and Federico Marini. "Particle swarm optimization (PSO). A tutorial." *Chemometrics and Intelligent Laboratory Systems* 149 (2015). https://doi.org/10.1016/j.chemolab.2015.08.020

[18] Cimbala, John M. "Taguchi orthogonal arrays." *Pennsylvania State University* (2014): 1-3.

[19] Ahmad, Tengku Noradeena Tengku and Siti Zaleha Abd Rasid. "Implementation of Building Maintenance Management System in an Organization." *Journal of Advanced Research in Technology and Innovation Management* 1, no. 1 (2021): 1-8.

[20] Carlier, Jacques and Emmanuel Neron. "An exact method for solving the multi-processor flow-shop." *RAIRO-Operations Research-Recherche Opérationnelle* 34, no. 1 (2000): 1-25. https://doi.org/10.1051/ro:2000103

[21] Chin, Thoo Ai and Liu Min. "The effect of supply chain risk management practices on resilience and performance: A systematic literature review." *Journal of Advanced Research in Technology and Innovation Management* 1, no. 1 (2021): 41-53.