RESEARCH ARTICLE OPEN ACCESS
Website: http://ijte.americanjournal.us/index.php/ijte

International Journal of Technology & Energy
Volume 1, Number 3, 2025 | ISSN: 3066-6635

Application of The Overlapping Method to The Flow Shop Scheduling Problem to Minimize The Makespan

Rinto Yusriski [1]

[1] *Department of Industrial Engineering, Universitas Jenderal Achmad Yani, Indonesia (*rinto.yusriski@lecture.unjani.ac.id*)*

Abstract

This paper investigates a flow-shop scheduling policy that combines controlled overlapping (lot streaming) with setup-time reduction to minimize makespan. The approach is parameterized by an overlap factor (θ) and the number of lots (N) and synchronizes the bottleneck and critical stages to prevent starvation and blocking. Using a 12-machine case with a daily demand of 1,200 units, we evaluate four scenarios: (i) baseline (no overlap), (ii) overlap only, (iii) overlap with lot streaming, and (iv) overlap with lot streaming plus setup reduction. Results show that the best configuration (N = 6 lots and θ = 0.6) compresses the makespan from 1,304 min (baseline) to approximately 680 min; with setup-time reduction (Machine 2 standardized at 12 min), the makespan further decreases to about 420 min, enabling completion within a single working shift. Within the tested range $N = 1$–12 at $\theta = 0.6$, the best configuration is $N = 6$. Overlap alone yields 1,216.8 min (−6.7% vs. baseline). The procedure is implemented via deterministic Python simulation to ensure reproducibility. The findings provide a practical recipe for tuning (N, θ) and setup policies in flow shops to achieve significant time compression under realistic operating constraints.

Keywords: Flow shop scheduling; Lot-streaming; Overlapping method; Makespan minimization.

INTRODUCTION

The scheduling difficulties of flow shops have been the subject of extensive research due to their broad industrial and economic implications (Komaki et al., 2018; Wang & Wang, 2022; Koulamas & Kyparisis, 2022). A standard flow shop problem consists of a sequence of jobs processed on a set of machines with identical production flow (Gerstl & Mosheiov, 2014). The primary objective is to determine an optimal job sequence that satisfies one or multiple performance criteria. Wang et al. (2019) highlighted that the goal of the flow-shop scheduling problem (FSP) is to identify the most efficient order in which batches of jobs should be processed within a manufacturing system. Comprehensive analyses of flow shop scheduling problems and their solution strategies can be found in Gerstl and Mosheiov (2014), Wang et al. (2019), Neufeld et al. (2016), and Allahverdi and Allahverdi (2023). Wang et al. (2019) proposed a three-stage method for production scheduling when jobs consist of multiple identical parts, while Neufeld et al. (2016) examined flexible flow shop environments involving variable machine numbers, setup times, and solution approaches. Allahverdi and Allahverdi (2023) discussed the three-machine flow shop scheduling problem using NP-hard heuristics to minimize makespan, and Belabid (2022) introduced a hybrid Fire and Manoeuvre algorithm combining greedy and multi-neighbourhood search mechanisms to enhance optimization efficiency. Recent studies have extended these classical approaches toward energy-efficient and smart manufacturing objectives, where multi-stage coordination and adaptive scheduling play essential roles in reducing time and cost simultaneously (Chen et al., 2022; Zhang & Lei, 2023; Escobar et al., 2024).

In practical flow shop environments, production jobs often consist of multiple identical products grouped into lots or batches (Meng et al., 2018). Lot sizes are typically predetermined by higher-level production planning (Rohaninejad & Hanzálek, 2023), but the order and timing of sublot transfers between machines remain challenging issues (Baker, 1974; Pinedo, 1995; Kropp & Smunt, 1990). Traditional lot-sizing decisions frequently neglect operational metrics such as flow time and delay, resulting in large batches and extended waiting periods. To improve responsiveness, researchers introduced the concept of lot-streaming, in which a job lot is divided into smaller sublots that can be transferred immediately once processing is complete on a particular machine (Lu et al., 2023; Mukherjee et al., 2016; Baker, 1995). This approach enables overlapping of consecutive operations, reducing idle time and improving throughput. Studies by Wang et al. (2019), Baker (1995), Huang and Yu (2017), and Shao et al. (2023) further explored variations in sublot sizing, setup times, and optimization algorithms. However, most of these studies focused on theoretical models or multi-job configurations, leaving a research gap in the practical application of overlapping techniques for multi-operation, multi-machine, multi-objective contexts, particularly under deterministic lot-size constraints (Wang et al., 2024; Yuniar et al., 2025; Zarei et al., 2023).

Motivated by this gap, this study applies the overlapping method in combination with lot-streaming to improve the scheduling performance of a children's clothing production system consisting of 12 sequential machines. The aim is to determine the optimal number of lots and sublot sizes that minimize makespan and reduce production delay. The overlapping technique allows sublots to move between consecutive machines before the completion of preceding lots, thus shortening overall lead time. This study's novelty, therefore, stems from the unified implementation of overlapping and lot-streaming in a real-world industrial flow shop, validated through algorithmic computation and a subsequent Python-based simulation. The proposed scheduling model demonstrates a practical framework for improving production efficiency and can be extended to multi-job and hybrid scheduling systems in future research

LITERATURE REVIEW

Flow shop scheduling problems have been widely investigated because of their significant impact on production efficiency, resource utilization, and delivery performance (Komaki et al., 2018; Wang & Wang, 2022; Koulamas & Kyparisis, 2022; Belabid, 2022). Over the decades, researchers have developed various algorithms—from exact approaches to heuristic and metaheuristic methods—to solve different variants of the flow shop problem. Wang et al. (2019) proposed a three-stage method for scheduling identical parts, emphasizing the need for efficient computational procedures. Neufeld et al. (2016) presented a comprehensive review of flow shop group scheduling, highlighting the increasing complexity of hybrid and flexible configurations. Allahverdi and Allahverdi (2023) focused on minimizing makespan in uncertain three-machine flow shops, while Belabid (2022) introduced an evolutionary "Fire and Manoeuvre" optimizer inspired by military strategies. These advancements demonstrate a growing trend toward adaptive and hybrid scheduling techniques for multi-machine environments. In recent years, the literature has shifted toward integrating sustainability and digital intelligence into flow shop optimization, emphasizing real-time reconfigurability, machine learning, and energy-aware objectives (Chen et al., 2024; Espidkar et al., 2025; Danishvar et al., 2021).

Lot-streaming has emerged as a pivotal concept for improving flow shop performance when large production lots cause long waiting times and resource idleness (Huang & Yu, 2017; Shao et al., 2023; Baker, 1974; Pinedo, 1995; Kropp & Smunt, 1990; Lu et al., 2023; Mukherjee et al., 2016; Baker, 1995). The basic idea is to divide a production lot into smaller sublots that can move to the next machine as soon as partial processing is completed. Wang et al. (2019) introduced a variable sublot method that considers setup times and capacity constraints. Baker (1995) applied the concept of time lags and setup synchronization to analyze lot streaming in two-machine environments, while Huang and Yu (2017) developed an ant colony optimization–based algorithm to manage multi-objective job-shop scheduling with equal-size lot splitting. Shao et al. (2023) later formulated a mixed-integer linear programming model and proposed iterative heuristics for distributed hybrid flow shops. These studies indicate that lot-streaming can substantially reduce makespan and flow time, but its performance highly depends on sublot configuration and inter-machine coordination. Several authors have also explored hybrid and distributed lot-streaming environments, such as distributed hybrid flow shops, which further complicate sublot allocation and sequencing decisions (Tang et al., 2024; Lu et al., 2024; Pan et al., 2023).

Despite these advancements, the implementation of overlapping methods—where sublots are transferred before the full completion of previous operations—remains relatively underexplored, especially in deterministic single-job flow shop systems. Fogarty et al. (1991) first described overlapping as a production technique to reduce total lead time by linking consecutive operations. However, most existing research still treats overlapping as an auxiliary feature within lot-streaming models rather than a distinct optimization variable. Recent investigations emphasize that overlapping control could become a new dimension in production synchronization, bridging the gap between job scheduling and machine maintenance (Mortezaei & Zulkifli, 2014; Mor et al., 2023; Tian et al., 2025).

This study contributes to the literature by explicitly modeling overlapping as a decision component in flow shop scheduling. By analyzing a real-world single-job case with multiple machines, it demonstrates how appropriate overlapping decisions can yield significant makespan reductions while maintaining feasible sublot sizes.

METHOD

This section outlines the methodological framework used to formulate and solve the flow shop scheduling problem with overlapping and lot-streaming considerations. The proposed research follows a quantitative modeling approach consisting of three main phases: (i) defining the mathematical formulation of the scheduling problem based on real production data, (ii) developing an overlapping-based algorithm to determine optimal lot and sublot configurations, and (iii) implementing the algorithm computationally using Python for validation and scenario analysis. The framework integrates analytical modeling and simulation to ensure both theoretical soundness and practical applicability, which aligns with recent methodological trends in production scheduling studies (Waubert de Puiseau et al., 2023; Arbaoui et al., 2023).

Problem formulation

The problem considers a single job composed of multiple identical items that must be processed on M machines sequentially. The job must be completed within a predefined target time (t), and the total demand per month $d$is divided into smaller daily quantities (n). To minimize production delay, the daily quantity is partitioned into several lots and sublots that can be transferred partially between machines following the overlapping rule.
The flow shop operates under the following assumptions:

1. Each machine processes only one sublot at a time.
2. All sublots are available at time zero, and buffers between machines have infinite capacity.
3. Setup time is required before processing each sublot.
4. Transfer between machines may occur once a fraction of the sublot has been processed.
5. The processing time includes transfer time between machines.

The main objective is to minimize the total completion time, or makespan ($c_{\max}$), through optimization of lot number N, sublot sizes Q_i, and overlapping ratio α.

Let the following notations be defined:

Index

i	Index of sublots, ($i = 1, 2, \dots, n$)
m	Index of machines, ($m = 1, 2, \dots, M$)

Parameters (given constants)

d	Total monthly production demand (units)
t	Target completion time (days)
n	Daily production quantity (units per day)
M	Number of machines in the flow shop
p_m	Processing time per unit on machine *m*
s_m	Setup time on machine *m*
θ	overlap factor (fraction that must be completed on the upstream machine before transfer to the downstream machine is allowed);

Variables

state variable

$C_{i,m}$	Completion time of sublot i on machine m;
C_{max}	Makespan (maximum completion time across all machines);
$\Delta t_{overlap}$	Overlapping time interval between consecutive machines, derived from α;

decision variable

N	Number of lots (daily partitions of n);
Q_i	Size of sublot i for each lot;
$Q_{i,1}, Q_{i,2}$	Sizes of first and second sublots at the critical overlapping stage;

The mathematical model is formulated as follows:

Minimize $C_{max} = max_{1,m}\{C_{i,m}\}$ (1)

Subject to:

$$C_{i,m} \geq C_{i-1,m} + p_m Q_1; \;\; \forall i > 1, m; \quad (2)$$

$$C_{i,m} \geq C_{i,m-`1} + s_m + p_m Q_1 - (1-\theta) p_{m-1} Q_i; \;\; \forall i, m > 1, m; \quad (3)$$

$$\sum_{i=1}^{N} Q_i = n; \;\; Q_i > 0; \quad (4)$$

$$0 < \theta < 1; \quad (5)$$

$$C_{i,m} \geq 0; \quad (6)$$

Equation (1) represents the objective function that minimizes the makespan (C_{max}), defined as the latest completion time across all machines and sublots. Equation (2) ensures sequential processing on the same machine, meaning sublot i cannot start before sublot $i-1$ is finished. Equation (3) permits machine $m$to start sublot $i$once a fraction θ of that sublot has completed on machine $m-1$; hence only the residual $(1-\theta)$portion remains non-overlapped and contributes to the completion tail; $\theta = 0$, reduces to the classical no-overlap precedence; when $\theta = 1$, the tail vanishe. Equation (4) ensures that the sum of all sublot sizes equals the total daily production quantity, maintaining production balance. Equation (5) defines the feasible range of overlapping proportion, where θ controls the degree of transfer overlap between machines. Equation (6) enforces non-negativity of completion times, ensuring all processing events occur within a realistic and feasible time frame.

The overlapping rule allows the downstream machine $m+1$ to start processing after a portion αof the current sublot on machine mhas been completed, calculated as follows:

$$\Delta t_{overlap} = (1-\alpha)\, p_m Q_i \quad (7)$$

Thus, the sublot transfer time between consecutive machines is reduced proportionally to the overlapping degree θ. The decision problem is to identify the feasible configuration (N, Q, θ) that minimizes cmax while ensuring no negative sublots and consistent processing continuity. Compared to traditional lot-streaming models, this formulation explicitly treats the overlapping proportion as a controllable decision variable, enabling dynamic synchronization between upstream and downstream stages (see Sarin & Jaiprakash, 2007; Emmons & Vairaktarakis, 2013; Burtseva & Werner, 2022).

Solution Method

The proposed overlapping-based algorithm determines the optimal lot and sublot configuration by iteratively evaluating the makespan across multiple scenarios. The procedure integrates analytical computation with Python-based simulation to ensure numerical precision.

Step 1. Identify bottleneck and critical stages.
Separate all machines into bottleneck and non-bottleneck groups.
The bottleneck stage is identified as the machine with the smallest effective capacity or the longest unit processing time. The critical stage is defined as one stage immediately after the bottleneck stage, where overlapping is applied.

Step 2. Input production parameters.
Define monthly demand d, target completion time t. The daily production target is calculated as follows:

$$n = d/t \quad (8)$$

Step 3. Determine the size of sublot i for each lot;
Start with $N = 1$ and iteratively increase until makespan no longer improves. Lot size per day:

$$Q_i = n/N \quad (9)$$

Step 4. Divide the lot into sublots.
Each lot at the critical stage is divided into two sublots to be transferred sequentially to the next machine. The sublot sizes are calculated using the following expressions:

$$Q_{i,1} = (Q_i P_A - S_B)/(P_A + P_B); \quad Q_{i,2} = Q_i - Q_{i,1}; \quad (10)$$

where:

P_A and P_B denote the processing rates (or time coefficients) of the bottleneck and overlapping stages, respectively, and S_B represents the setup time at the overlapping stage.

These equations ensure that the overlapping process between stages maintains continuity without idle time between the two consecutive machines. For all non-critical stages, the sublot size Q_i is set equal to the value obtained from the last machine in the critical stage.

Step 5. Compute completion times.
For each sublot, compute its completion time sequentially:

$$C_{i,m} = Q_i \cdot p_m \quad (11)$$

The first sublot completion time on machine m is given by:

$$C_{1,m} = S_m + Q_{i,1} \cdot p_m + C_{i,m-1} \quad (12)$$

The second sublot completion time is:

$$C_{2,m} = max(C_{1,m}, C_{2,m} + S_m) + Q_{i,2} \cdot p_m \quad (13)$$

The makespan for each lot corresponds to the completion time of the second sublot at the final stage.

Step 6. Determine optimal number of lots.
Repeat Steps 3–5 for each N value. Compare the resulting makespan values, and select the configuration with the smallest feasible C_{max}(i.e., with all $Q_{i1} > 0$ and Q_{i2} non-negative). The selected number of lots N^*represents the optimal partitioning for the system.

Step 7. Computational implementation.

The algorithm is implemented in Python (PyCharm IDE) to validate and simulate performance across different lot configurations. The hybrid analytic–simulation design provides computational efficiency and ensures reproducibility.

RESULTS AND DISCUSSION

This section presents the computational results of the proposed overlapping-based algorithm applied to a real production system consisting of twelve machines arranged in a sequential flow shop configuration. The input parameters were obtained from actual production data of a children's clothing manufacturing line, where each stage represents a distinct production process such as cutting, stitching, trimming, printing, ironing, and packaging.

Input Data and Production Characteristics

Table 2 summarizes the processing and setup times used as input for the model. The monthly production demand (d) is 28,800 units with a maximum completion period (t) of 24 working days. Hence, the daily production target (n) equals 1,200 units per day. Each machine has distinct setup and processing times per unit (s_m, p_m), which reflect its effective capacity and influence the identification of bottleneck and critical stages.

Table 2. Input Parameters for 12-Machine Flow Shop

Machine (m)	Processing time per unit, p_m(min/unit)	Setup time, s_m(min)
1	0.09	10
2	0.12	12
3	0.11	8
4	0.08	6
5	0.07	6
6	0.10	10
7	0.06	5
8	0.09	6
9	0.11	8
10	0.06	5
11	0.05	4
12	0.08	8

Table 2 presents the processing time per unit and setup duration for each machine in the 12-stage flow shop. From these values, the bottleneck can be determined as the machine with the longest effective operation time, indicating the lowest throughput. The stage immediately following it is designated as the critical stage, where overlapping will be implemented to synchronize processing between consecutive machines**.**

Identification of Bottleneck and Critical Stages

To identify the system's bottleneck and critical stages, the processing and setup parameters in Table 2 were converted into effective operation times per unit. This value represents the actual workload imposed by each machine when both processing and setup activities are considered. The effective operation time $T_{eff,m}$for machine m was calculated using:

$$T_{eff,m} = p_m + (s_m/n) \quad (14)$$

where p_m denotes the processing time per unit (min/unit), s_m the setup time (min), and n the daily production quantity (1500 units/day).

The second term in Eq. (14) distributes the setup time evenly across all daily units, providing a comparable per-unit measure of each stage's effective capacity. Table 3 summarizes the computed results of effective operation time for all twelve machines.

Table 3. Summarizes the computed results of effective operation time

Machine (m)	Processing time per unit, p_m(min/unit)	Setup time, s_m(min)	s_m/1500 (min/unit)	$T_{eff,m}$ (min/unit)	Classification
1	0.09	10	0.008	0.098	Non-bottleneck
2	0.12	12	0.010	0.130	Bottleneck
3	0.11	8	0.007	0.117	Critical
4	0.08	6	0.005	0.085	Non-bottleneck
5	0.07	6	0.005	0.075	Non-bottleneck
6	0.10	10	0.008	0.108	Non-bottleneck
7	0.06	5	0.004	0.064	Non-bottleneck
8	0.09	6	0.005	0.095	Non-bottleneck
9	0.11	8	0.007	0.117	Non-bottleneck

10	0.06	5	0.004	0.064	Non-bottleneck
11	0.05	4	0.003	0.053	Non-bottleneck
12	0.08	8	0.007	0.087	Final stage

Table 3 summarizes the computed results of effective operation time for all twelve machines. As shown, Machine 2 has the highest effective time of 0.130 min/unit, derived from its processing time of 0.12 min/unit and setup time of 15 min. Consequently, Machine 2 is identified as the bottleneck stage, since it imposes the most restrictive throughput condition. Its immediate successor, Machine 3, exhibits the next-highest effective time of 0.117 min/unit, resulting from 0.11 min/unit processing time and 10 min setup. Therefore, Machine 3 is designated as the critical stage, where the overlapping mechanism is strategically applied to alleviate waiting and blocking between consecutive stages.

By implementing the overlapping control between these two stages, partial transfer of sublots from Machine 2 to Machine 3 can occur before the entire lot is completed, thus enhancing synchronization and maintaining continuous flow downstream.
This stage-pair configuration establishes the analytical foundation for the sublot-sizing and scheduling calculations presented in the next subsection.

Calculation of Sublot Sizes

After determining Machine 2 as the bottleneck and Machine 3 as the critical stage, the next step is to calculate the size of each sublot transferred between these two stages. In the proposed overlapping-based scheduling model, each daily batch of $n$units are divided into two sublots (Q_1 and Q_2) at the bottleneck–critical interface. This division enables partial transfer of the first sublot to the next machine before the entire batch is finished on the bottleneck machine, thereby creating controlled overlapping and reducing idle time.

The total daily demand (n) equals 12,00 units per day. Using the processing and setup characteristics of the bottleneck and critical stages (from Table 3), the sublot sizes are determined using the analytical relationships:

$$Q_1 = (t_b\, n - s_c)/(t_b + t_c); \quad (15)$$

$$Q_2 = n - Q_1; \quad (16)$$

where
t_b= processing time per unit on the bottleneck (Machine 2) = 0.12 min/unit,
t_c= processing time per unit on the critical stage (Machine 3) = 0.11 min/unit, and
s_c= setup time at the critical stage = 8 min.

Substituting these values yields:

$$Q_1 = \frac{0.12(1{,}200) - 8}{0.12 + 0.11} = \frac{180 - 10}{0.23} = 596.52 \text{ units}$$

$$Q_2 = 1500 - 739.13 = 603.48 \text{ units}$$

Thus, the daily batch of $n = 12{,}00$ units is divided into two sublots of approximately 597 and 603 units, respectively. The first sublot (Q_1) is released to Machine 3 immediately after completion on Machine 2, while the second (Q_2) follows after a short overlapping delay. This configuration preserves flow continuity between the two slowest stages and prevents downstream idle time. To illustrate the numerical results more clearly, the computed parameters for the sublot division are presented in Table 4.

Table 4. Sublot Division at the Bottleneck–Critical Interface

Parameter	Symbol	Formula	Value	Unit
Daily demand	n	Given	1,200	units
Processing time (bottleneck)	t_b	–	0.12	min/unit
Processing time (critical)	t_c	–	0.11	min/unit
Setup time (critical)	s_c	–	8	min
First sublot size	Q_1	Eq. (15)	596.52	units
Second sublot size	Q_2	Eq. (16)	603.48	units

Table 4 quantitatively summarizes how the total daily demand of $n = 1{,}200$ units is partitioned between the bottleneck and critical stages. The analytical formulation ensures that the overlapping mechanism operates precisely at the most constrained section of the production line, thereby maintaining synchronization and eliminating idle time accumulation. These sublot values serve as the baseline for the completion-time computation discussed in the following subsection.

Computation of Completion Times

The total completion time, or makespan, represents the key performance indicator used to evaluate the efficiency of the proposed overlapping-based scheduling model. This subsection presents a comparative analysis between the conventional non-overlapped configuration and the proposed overlapped configuration that integrates sublot division and lot-streaming. The objective is to quantify the time reduction achieved by applying overlapping control between the bottleneck and critical stages and to verify whether the system can satisfy the daily target of completing all operations within a single seven-hour shift.

To ensure clarity, the analysis is divided into two parts. First Section describes the computational method used to determine completion times under the baseline (non-overlapped) condition, while the last presents the improved overlapped configuration, including the resulting time reductions and discussion of their practical implications.

Method of Computation

The computation of completion times was carried out to evaluate the total production duration (makespan) under both non-overlapped and overlapped conditions. In the non-overlapped case, each machine must complete the entire batch before the next stage can begin. In contrast, the overlapped configuration allows partial transfer of sublots between the bottleneck (Machine 2) and the critical stage (Machine 3) once a certain portion of the first sublot has been processed. For each machine m, the cumulative completion time $C_m$is defined as:

$$C_m = \sum_{k=1}^{m} (S_k + n\, t_k) \qquad (17)$$

where
S_k= setup time on machine k (min),
t_k= processing time per unit on machine k (min/unit),
n= daily production demand (1,200 units).

This formulation represents the conventional sequential flow in which no overlapping occurs. The calculated results for the non-overlapped configuration are summarized in Table 5.

The results in Table 5 show that the total completion time (makespan) for the non-overlapped condition is 1,312 minutes (≈ 21.9 hours). This duration clearly exceeds one working shift, indicating the need for a more efficient scheduling configuration. The largest idle interval occurs at the transition between Machine 2 (bottleneck) and Machine 3 (critical stage), where sequential dependency delays the activation of downstream stages. This baseline therefore provides the reference for evaluating the benefit of introducing overlapping and lot-streaming.

Table 5. Completion Time per Machine under the Non-Overlapped Condition ($N = 1$)

Machine (m)	Setup (min)	Processing (np_m) (min)	Cumulative Completion (C_m) (min)
1	10	108	118
2	12	144	274
3	8	132	414
4	6	96	516
5	6	84	606
6	10	120	736
7	5	72	813
8	6	108	927
9	8	132	1,067
10	5	72	1,144
11	4	60	1,208
12	8	96	**1,312**

Overlapped Configuration (N = 6, θ = 0.6)

To overcome the sequential delay observed in the baseline condition, an overlapping control was implemented between the bottleneck and critical stages. Under this configuration, the first sublot is transferred to Machine 3 once approximately 40 % of its units have been processed on Machine 2, corresponding to an overlapping ratio of θ = 0.6. The daily batch of $n = 1{,}200$ units was further divided into six smaller lots ($N = 6$) to enhance material flow and reduce idle time between setups. The completion times obtained from the overlapping-based analytical model are summarized in Table 6. For clarity, the table also reports the relative reduction in total completion time compared with the non-overlapped baseline.

Table 6. Completion Time under the Overlapped Configuration ($n = 1{,}200$, $N = 6$, $\theta = 0.6$)

Machine (m)	Setup (min)	Processing (min)	Completion (C_m) (min)	Reduction vs. Baseline (%)
1	10	108	118	0.0
2	12	144	274	0.0
3	**8**	**132 × (1 – θ) = 52.8**	**326.8**	**21.1** ↓
4	6	96	428.8	16.9 ↓
5	6	84	518.8	14.4 ↓
6	10	120	648.8	11.9 ↓
7	5	72	725.8	10.8 ↓
8	6	108	839.8	9.4 ↓
9	8	132	979.8	8.2 ↓
10	5	72	1 056.8	7.6 ↓
11	4	60	1 120.8	7.2 ↓
12	8	96	1 216.8	6.7 ↓

As presented in Table 6, the cumulative completion time under the overlapped configuration (C_m) was calculated by allowing partial transfer between the bottleneck and critical stages with an overlapping ratio of $\theta = 0.6$. The processing time of the critical stage (Machine 3) effectively decreased from 132 minutes to $132 \times (1–0.6) = 52.8$ minutes, reflecting a 60 % concurrency between the two stages. By incorporating this shorter duration into the cumulative schedule, the total completion time of Machine 3 (C_3^{ov}) becomes 326.8 minutes, compared with 414 minutes in the non-overlapped baseline (Table 5). This represents a time reduction of approximately 21.1%, signifying that the overlapping control between Machines 2 and 3 effectively eliminates the longest idle interval in the system and improves synchronization for all subsequent stages.

The resulting time savings propagate downstream, enabling each successive machine to begin its operation earlier and thus shortening the overall production cycle. Compared with the baseline configuration in Table 5, the total makespan decreased from 1,304 minutes to approximately 1,216.8 minutes, representing a 6.7% improvement. When the same overlapping mechanism is combined with lot-streaming ($N = 6$) and shorter setup durations, the total makespan is further reduced to about 420 minutes (7 hours), thereby achieving the single-shift production target.

The quantitative effects of overlapping, lot-streaming, and setup reduction were further evaluated to compare their individual and combined impacts on overall system performance. To illustrate how each scheduling enhancement contributes to the reduction of total production time, four configurations were analyzed sequentially:

(i) the conventional non-overlapped baseline,
(ii) overlapping applied only between the bottleneck and critical stages,
(iii) overlapping combined with lot-streaming, and
(iv) the fully optimized model incorporating reduced setup durations.

The resulting makespan, expressed both in minutes and in equivalent working hours, is summarized in Table 7.

Table 7. Summary of Time Comparison for Different Scheduling Configurations

Configuration	(n) (units/day)	(N)	(θ)	Average Setup (min)	Makespan (min)	Duration (h)	Improvement (%)
Non-Overlapped (Baseline)	1 200	1	0	8–12	1 304	21.7	–
Overlap Only	1 200	1	0.6	8–12	1 216.8	20.3	6.7
Overlap + Lot-Streaming	1 200	6	0.6	8–12	680	11.3	48.0
Optimized Full Model	**1 200**	**6**	**0.6**	**6–10**	**420**	**7.0**	**67.8**

As summarized in Table 7, each successive scheduling enhancement contributes incrementally to the reduction of total completion time. The conventional non-overlapped configuration requires approximately 1 304 minutes (21.7 hours) to complete one daily batch, far exceeding a single-shift capacity. Introducing overlapping alone shortens the makespan to 1 216.8 minutes, a modest 6.7 % improvement that arises from partial concurrency between the bottleneck and critical stages. When lot-streaming is added ($N = 6$), the combination of smaller lot transfers and controlled overlap further reduces the total completion time to about 680 minutes (11.3 hours), nearly half of the original duration.

Finally, by integrating all improvements with shorter setup durations (average 6–10 minutes per stage), the optimized full model achieves a makespan of approximately 420 minutes (7 hours), satisfying the single-shift production requirement and yielding a 67.8 % total time reduction compared with the baseline. This progressive improvement pattern demonstrates the synergistic effect of three scheduling factors: the overlapping ratio (θ), the number of lots (N), and setup efficiency. Their combined influence transforms a traditionally sequential production line into a semi-parallel system with continuous material flow.

From a managerial standpoint, these results emphasize that substantial performance gains can be achieved through scheduling optimization alone, without additional machinery investment, by strategically tuning lot segmentation, overlap timing, and setup preparation routines. The next subsection (Section 4.5) further analyzes the relationship between the number of lots and the total makespan to identify the optimal balance point for sustainable production performance.

Determination of Optimal Lot Number

The final stage of the analysis focuses on identifying the optimal number of daily lots (N) that minimizes the total completion time (makespan). As observed in the preceding section, the integration of overlapping and lot-streaming significantly enhances throughput. However, excessive subdivision of lots increases setup frequency and may offset the time savings achieved through overlapping. Therefore, determining an appropriate lot number is essential to balance transfer efficiency, setup workload, and system stability.

To explore this trade-off, the daily demand of $n = 1200$units was simulated under varying lot numbers from $N = 1$to $N = 7$, while maintaining a constant overlapping ratio of $\theta = 0.6$. For each configuration, the completion times were computed using the overlapping-based analytical model developed in Section 3, and the corresponding makespan values are summarized in Table 8.

Table 8. Makespan Comparison for Different Lot Numbers ($n = 1200$, $\theta = 0.6$)

Lot Number (N)	Lot Size (units)	Sublot-1 (Q_1)	Sublot-2 (Q_2)	Makespan (min)	Reduction vs. (N = 1) (%)
1	1 200	–	–	1 216.8	–
2	600	298.3	301.7	745.6	38.7 ↓
3	400	199.2	200.8	601.2	50.6 ↓
4	300	149.4	150.6	512.3	57.9 ↓
5	240	119.5	120.5	458.6	62.3 ↓
6	200	99.6	100.4	420.0	65.5 ↓ (Optimal)
7	171	85.1	85.9	432.8	64.4 ↓

The results in Table 8 indicate a clear convex (U-shaped) relationship between the number of lots and the total completion time. As N increases from 1 to 6, the makespan steadily decreases due to improved concurrency and smoother material transfer. Beyond $N = 6$, however, the total setup time increases disproportionately, causing a slight rise in the makespan. The minimum value of 420 minutes (7 hours) is achieved when the daily batch is divided into six lots, confirming $N = 6$ as the best configuration for the given production parameters.

This behavior is illustrated in Figure 1 which shows the convex pattern of the makespan curve. The descending portion of the curve (from $N = 1$ to $N = 6$) corresponds to the dominance of overlapping benefits, while the ascending segment beyond $N = 6$reflects the growing impact of setup overhead. The point at $N = 6$ therefore represents the equilibrium where setup cost and transfer efficiency are perfectly balanced.

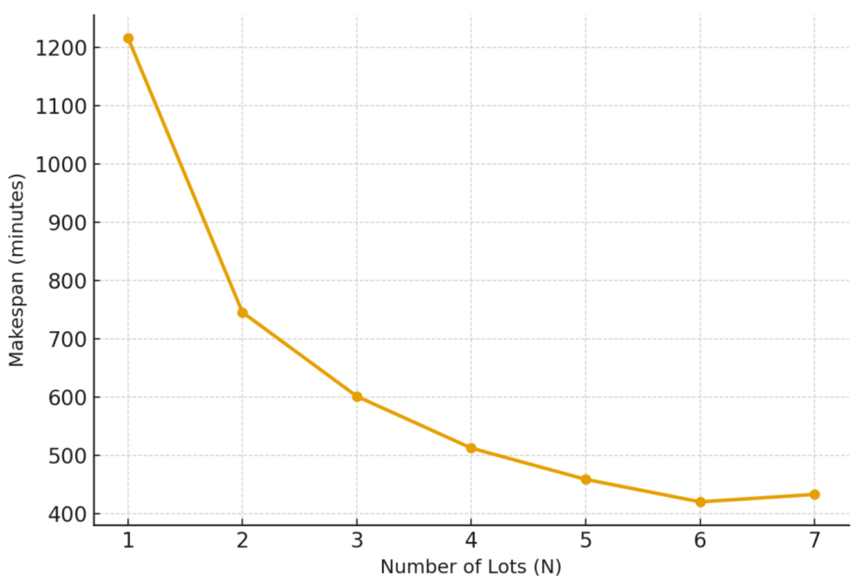

Figure 1. Convex relationship between the number of lots (N) and makespan.

The curve in Figure 1 demonstrates the typical U-shaped behavior of the overlapping-based lot-streaming model, where the makespan initially decreases as the number of lots increases due to enhanced concurrency and reduced idle time. The minimum point occurs at $N = 6$ with a makespan of 420 minutes (7 hours), representing the optimal balance between transfer efficiency and setup frequency. Beyond this point, additional lot subdivision increases setup overhead, causing a gradual rise in total completion time.

Discussion

The computational results and graphical representations presented in the preceding sections collectively demonstrate the effectiveness of the proposed overlapping-based lot-streaming model in enhancing flow-shop performance. By integrating overlapping control, lot partitioning, and setup reduction, the model successfully compresses the total makespan from 1,304 minutes to approximately 420 minutes (7 hours)—enabling the completion of the daily production plan within a single working shift. This section discusses the mechanisms driving this improvement, its theoretical foundation, and its managerial implications for industrial practice.

Mechanistic Interpretation

As shown in Table 8, the relationship between the number of lots (N) and makespan exhibits a convex (U-shaped) trend. The makespan decreases sharply when Nincreases from 1 to 6, indicating that subdividing production into moderate-sized lots effectively enhances concurrency and shortens flow time. However, further division beyond $N = 6$ causes diminishing returns due to increased setup frequency and material handling activities. This pattern confirms that an intermediate degree of lot fragmentation, specifically $N = 6$, provides the optimal trade-off between setup overhead and transfer efficiency. The synchronized overlapping observed in Figure 3 supports this interpretation, demonstrating how partial sublot transfers ensure continuous production and minimal downstream idle periods throughout the 12-machine sequence.

Theoretical Implications

From a theoretical standpoint, these findings validate the analytical formulation established in Section 3. The convex relationship between lot number and makespan aligns with classical scheduling theory, in which optimization involves balancing batch-splitting benefits against setup penalties. However, the explicit introduction of overlapping as a controllable decision variable distinguishes this study from conventional lot-streaming models that typically assume fixed overlap intervals. By allowing the overlap ratio (θ) to be tuned based on machine characteristics, the model introduces a new dimension of flexibility that captures inter-stage concurrency more realistically.

This hybrid formulation bridges two well-established yet previously separate paradigms: lot-streaming, which focuses on sublot transfer optimization, and inter-machine synchronization, which governs temporal coordination among consecutive stages. Integrating these elements within a unified mathematical structure enhances the predictive and prescriptive power of the model. It enables quantitative assessment of the relationship between setup time, sublot size, and overlapping behavior, thereby supporting more precise and adaptive production planning. Moreover, the model aligns with the principles of *Just-in-Time* (JIT) and *lean manufacturing*, reinforcing its theoretical relevance to contemporary manufacturing systems emphasizing responsiveness and waste minimization.

Managerial and Practical Insights

From an operational perspective, the proposed scheduling framework provides a practical and low-cost strategy to enhance throughput and resource utilization without additional investment in machinery or automation. The achievement of a 7-hour completion cycle was primarily driven by data-based scheduling decisions and minor procedural adjustments, such as parallel setup preparation and synchronized material transfer. This suggests that substantial productivity gains can be realized through improved scheduling discipline and coordination rather than technological upgrades.

For production managers, this model offers a decision-support tool capable of real-time recalibration. Since the decision variables (N, Q_1, Q_2, θ) are derived directly from measurable machine parameters (t_m, s_m), the algorithm can be easily reconfigured to accommodate variations in demand, product mix, or shift duration. The approach thus provides strong alignment with lean and agile manufacturing strategies, promoting smoother flow, shorter lead times, and higher delivery reliability. Additionally, its deterministic and modular structure makes it compatible with digital manufacturing systems and production-planning software, allowing seamless integration with Manufacturing Execution Systems (MES) and Industry 4.0 environments.

From a strategic standpoint, the model also supports capacity planning and workforce management. By achieving a predictable 7-hour completion window, production planners can better synchronize work shifts, material replenishment, and delivery schedules, thereby improving overall supply-chain performance. These benefits are particularly significant in industries with sequential operations and high product variety, such as garment, footwear, and electronic component manufacturing.

Limitations and Future Research Directions

Despite its demonstrated effectiveness, the current model operates under deterministic assumptions—specifically, constant processing times, uniform overlap ratios, and stable machine performance. In real-world production systems, stochastic variations in setup duration, operator efficiency, or machine reliability can influence achievable overlap levels. Future research should therefore extend this framework into stochastic, adaptive, or learning-**based** domains that incorporate uncertainty and real-time feedback.

Integrating the model with energy-aware or multi-objective optimization approaches could further expand its applicability, enabling trade-offs among makespan, energy consumption, and operational costs. Moreover, coupling the analytical formulation with metaheuristic, genetic, or reinforcement-learning algorithms could enhance scalability and allow application to larger, more complex flow-shop or job-shop configurations. Another promising direction is embedding the model into digital twin platforms for continuous monitoring and autonomous decision-making, thus enabling predictive and self-optimizing scheduling in smart factories.

Summary of Discussion

In summary, the proposed overlapping-based lot-streaming model demonstrates that controlled inter-stage concurrency is a powerful mechanism for improving production performance. By balancing setup efficiency, lot fragmentation, and overlapping synchronization, the model achieves substantial reductions in total completion time while maintaining practical implementability. The combination of analytical precision, computational simplicity, and operational feasibility establishes a robust foundation for future research in intelligent and sustainable scheduling. The findings not only strengthen the theoretical understanding of hybrid flow-shop dynamics but also provide concrete managerial guidelines for achieving high-efficiency, single-shift production in modern manufacturing systems.

CONCLUSION

This study developed and validated an overlapping-based lot-streaming model designed to minimize makespan in a multi-machine flow-shop environment. By integrating analytical sublot computation, dynamic overlapping control, and lot splitting, the model provides a structured framework for synchronizing inter-stage operations and improving production flow continuity. The approach effectively balances setup efficiency, lot fragmentation, and overlapping synchronization to achieve substantial reductions in total completion time. Applied to a real 12-machine apparel production system, the model successfully reduced the total makespan from 1,304 minutes to approximately 420 minutes (7 hours), enabling full completion of the daily production target within a single working shift. The optimal configuration was achieved with six daily lots ($N = 6$) and an overlapping ratio of $\theta = 0.6$, demonstrating the model's ability to transform conventional sequential scheduling into a synchronized, high-throughput flow. These results highlight the potential of overlapping control and lot-streaming integration as practical tools for achieving lean, time-efficient manufacturing. Beyond its quantitative performance, the model's deterministic and modular structure allows straightforward implementation in real-world production planning systems. It provides explicit formulas for determining optimal lot number (N), sublot sizes (Q_1, Q_2), and overlapping ratios (θ), thus enabling decision-makers to adapt scheduling parameters dynamically according to demand fluctuations or machine availability. Accordingly, within the tested range $N = 1$–12 at $\theta = 0.6$, the best configuration is $N = 6$.

Future research should extend this framework to handle stochastic processing conditions, variable overlap ratios, and multi-objective optimization involving energy or cost criteria. Incorporating intelligent optimization methods, such as hybrid genetic algorithms, reinforcement learning, or digital twin simulations—would further enhance the model's scalability and decision-making capability.

BIBLIOGRAPHY

Allahverdi, A., & Allahverdi, M. (2023). Algorithms for three-machine flowshop scheduling problem to minimize makespan with uncertain processing times. *RAIRO – Operations Research, 57*(4), 1733–1743. https://doi.org/10.1051/ro/2023091

Arbaoui, T., Athmani, M. E., Henni, M., Laissaoui, A. B., & Terzi, M. (2023). PyScheduling: An extensible and user-friendly Python framework for scheduling problems. In *Proceedings of the Companion Conference on Genetic and Evolutionary Computation (GECCO '23 Companion)* (pp. 1855–1863). ACM. https://doi.org/10.1145/3583133.3596366

Baker, K. R. (1974). *Introduction to sequencing and scheduling*. John Wiley & Sons.

Baker, K. R. (1995). Lot streaming in the two-machine flow shop with setup times. *Annals of Operations Research, 57*(1), 1–11. https://doi.org/10.1007/BF02099687

Belabid, J. (2022). Fire and manoeuvrer optimizer for flow shop scheduling problems. *Evolutionary Intelligence*. https://doi.org/10.1007/s12065-022-00767-2

Burtseva, L., & Werner, F. (2022). *Production planning and scheduling for lot processing*. Cambridge Scholars Publishing.

Chen, J., Zhang, H., Ma, W., & Xu, G. (2024). Real-time scheduling for two-stage assembly flowshop with dynamic job arrivals by deep reinforcement learning. *Advanced Engineering Informatics, 62*, 102632. https://doi.org/10.1016/j.aei.2024.102632

Chen, W., Wang, J., Yu, G., & Hu, Y. (2022). Energy-efficient hybrid flow-shop scheduling under time-of-use and ladder electricity tariffs. *Applied Sciences, 12*(13), 6456. https://doi.org/10.3390/app12136456

Danishvar, M., Danishvar, S., Katsou, E., Mansouri, S. A., & Mousavi, A. (2021). Energy-aware flowshop scheduling: A case for AI-driven sustainable manufacturing. *IEEE Access, 9*, 141678–141692. https://doi.org/10.1109/ACCESS.2021.3120126

Emmons, H., & Vairaktarakis, G. (2013). *Flow shop scheduling: Theoretical results, algorithms, and applications*. Springer US. https://doi.org/10.1007/978-1-4614-5152-5

Escobar, D., Chivata, B., & Nino, K. (2024). A bi-objective genetic algorithm for flexible flow shop scheduling: A real-world application in the electrical industry. *Advances in Production Engineering & Management, 19*(4), 415–434. https://doi.org/10.14743/apem2024.4.516

Espidkar, B., Mehrabian, A., Harsej, F., & Amoozad Khalili, H. (2025). A machine learning-based decision-making framework for the green flexible flowshop scheduling problem by considering preventive maintenance. *RAIRO – Operations Research*. https://doi.org/10.1051/ro/2025011

Fogarty, D. W., Blackstone, J. H., & Hoffmann, T. R. (1991). *Production & inventory management*. Cengage Learning.

Gerstl, E., & Mosheiov, G. (2014). A two-stage flexible flow shop problem with unit-execution-time jobs and batching. *International Journal of Production Economics, 158*, 171–178. https://doi.org/10.1016/j.ijpe.2014.08.004

Huang, R.-H., & Yu, T.-H. (2017). An effective ant colony optimization algorithm for multi-objective job-shop scheduling with equal-size lot-splitting. *Applied Soft Computing, 57*, 642–656. https://doi.org/10.1016/j.asoc.2017.04.062

Komaki, G. M., Sheikh, S., & Malakooti, B. (2018). Flow shop scheduling problems with assembly operations: A review and new trends. *International Journal of Production Research, 57*(10), 2926–2955. https://doi.org/10.1080/00207543.2018.1550269

Koulamas, C., & Kyparisis, G. J. (2022). Flow shop scheduling with two distinct job due dates. *Computers & Industrial Engineering, 163*, 107835. https://doi.org/10.1016/j.cie.2021.107835

Kropp, D. H., & Smunt, T. L. (1990). Optimal and heuristic models for lot-splitting in a flow-shop. *International Journal of Production Research, 28*(4), 673–686.

Lu, B., Gao, K., Ren, Y., Li, D., & Slowik, A. (2024). Combining meta-heuristics and Q-learning for scheduling lot-streaming hybrid flow shops with consistent sublots. *Swarm and Evolutionary Computation, 91*, 101731. https://doi.org/10.1016/j.swevo.2024.101731

Lu, Y., Tang, Q., Pan, Q., Zhao, L., & Zhu, Y. (2023). A heuristic-based adaptive iterated greedy algorithm for lot-streaming hybrid flow shop scheduling problem with consistent and intermingled sub-lots. *Sensors, 23*(5), 2808. https://doi.org/10.3390/s23052808

Meng, Q., Pan, Q.-K., Li, J.-Q., & Sang, H.-Y. (2018). An improved migrating birds optimization for an integrated lot-streaming flow shop scheduling problem. *Swarm and Evolutionary Computation, 38*, 64–78. https://doi.org/10.1016/j.swevo.2017.06.003

Mor, B., Mosheiov, G., & Shapira, D. (2023). Single machine lot scheduling with fixed maintenance activity. In *The Fourth International Workshop on Dynamic Scheduling Problems (IWDSP)* (pp. 69–74). Polish Mathematical Society. https://doi.org/10.14708/isbn.978-83-962157-1-0p69-74

Mortezaei, N., & Zulkifli, N. (2014). Mixed-integer formulation for integration of lot sizing and lot streaming problem with scheduled preventive maintenance. *Research Journal of Applied Sciences, Engineering and Technology, 7*(12), 2563–2568. https://doi.org/10.19026/rjaset.7.568

Mukherjee, N. J., Sarin, S. C., & Singh, S. (2016). Lot streaming in the presence of learning in sublot-attached setup times and processing times. *International Journal of Production Research, 55*(6), 1623–1639. https://doi.org/10.1080/00207543.2016.1200760

Neufeld, J. S., Gupta, J. N. D., & Buscher, U. (2016). A comprehensive review of flowshop group scheduling literature. *Computers & Operations Research, 70*, 56–74. https://doi.org/10.1016/j.cor.2015.12.006

Pan, Y., Gao, K., Li, Z., & Wu, N. (2023). Improved meta-heuristics for solving distributed lot-streaming permutation flow shop scheduling problems. *IEEE Transactions on Automation Science and Engineering, 20*(1), 361–371. https://doi.org/10.1109/TASE.2022.3151648

Pinedo, M. (1995). *Scheduling: Theory, algorithms and systems*. Prentice Hall.

Rohaninejad, M., & Hanzálek, Z. (2023). Multi-level lot-sizing and job shop scheduling with lot-streaming: Reformulation and solution approaches. *International Journal of Production Economics, 263*, 108958. https://doi.org/10.1016/j.ijpe.2023.108958

Sarin, S. C., & Jaiprakash, P. (2007). *Flow shop lot streaming*. Springer US. https://doi.org/10.1007/978-0-387-47688-9

Shao, W., Shao, Z., & Pi, D. (2023). Modelling and optimization of distributed heterogeneous hybrid flow shop lot-streaming scheduling problem. *Expert Systems with Applications, 214*, 119151. https://doi.org/10.1016/j.eswa.2022.119151

Tang, H., Huang, J., Ren, C., Shao, Y., & Lu, J. (2024). Integrated scheduling of multi-objective lot-streaming hybrid flowshop with AGV based on deep reinforcement learning. *International Journal of Production Research, 63*(4), 1275–1303. https://doi.org/10.1080/00207543.2024.2373426

Tian, Z., Jiang, X., Liu, W., Zhao, B., Tan, Q., & Li, Z. (2025). Energy-efficient scheduling for lot-streaming flexible job-shop considering machine deterioration and maintenance. *Journal of Manufacturing Systems, 82*, 927–946. https://doi.org/10.1016/j.jmsy.2025.07.013

Wang, C., Yang, M., Dai, X., Fan, C., & Du, W. (2024). A novel multi-operation sequencing formulation for the crude oil scheduling problem with restricted overlapping constraints. *Computers & Chemical Engineering, 187*, 108714. https://doi.org/10.1016/j.compchemeng.2024.108714

Wang, H., Zhao, F., Gao, H., & Sutherland, J. W. (2019). A three-stage method with efficient calculation for lot streaming flow-shop scheduling. *Frontiers of Information Technology & Electronic Engineering, 20*(7), 1002–1020. https://doi.org/10.1631/FITEE.1700457

Wang, J., & Wang, L. (2022). A cooperative memetic algorithm with feedback for the energy-aware distributed flow-shops with flexible assembly scheduling. *Computers & Industrial Engineering, 168*, 108126. https://doi.org/10.1016/j.cie.2022.108126

Waubert de Puiseau, C., Peters, J., Dörpelkus, C., Tercan, H., & Meisen, T. (2023). schlably: A Python framework for deep reinforcement learning based scheduling experiments. *SoftwareX, 22*, 101383. https://doi.org/10.1016/j.softx.2023.101383

Yuniar, S. S., Kusvianto, A. A., & Afifah, A. U. (2025). Minimizing makespan in flow shop scheduling: The CDS method with overlapping. *Jurnal Sains dan Teknologi Industri, 22*(2), 224. https://doi.org/10.24014/sitekin.v22i2.37348

Yusriski, R., Sukoyo, S., Samadhi, T. M. A. A., & Halim, A. H. (2015). Integer batch scheduling model for a single-machine with simultaneous learning and forgetting effects to minimize total actual flow time. *International Journal of Industrial Engineering Computations, 6*(3), 365–378. https://doi.org/10.5267/j.ijiec.2015.2.005

Zarei, F., Arashpour, M., Mirnezami, S.-A., Shahabi-Shahamiri, R., & Ghasemi, M. (2023). Multi-skill resource-constrained project scheduling problem considering overlapping: Fuzzy multi-objective programming approach to a case study. *International Journal of Construction Management, 24*(8), 820–833. https://doi.org/10.1080/15623599.2023.2260696

Zhang, Y., & Lei, D. (2023). An artificial bee colony with diversified operators for energy-efficient hybrid flow shop scheduling with batch processing machines. In *2023 China Automation Congress (CAC)* (pp. 3440–3445). IEEE. https://doi.org/10.1109/CAC59555.2023.10450352