Available online at : http://josi.ft.unand.ac.id/

Jurnal Optimasi Sistem Industri

| ISSN (Print) 2088-4842 | ISSN (Online) 2442-8795 |

Article Type

Optimal Order Scheduling for Corrugated Box Production on Flexographic Printing Machines Using a Genetic Algorithm

Irene Karijadi *, Ivan Gunawan, Stephanie Gracia Khangara

Department of Industrial Engineering, Universitas Katolik Widya Mandala Surabaya, Indonesia

*Corresponding Author: irenekarijadi92@gmail.com

DOI: 10.25077/josi.v25.n1.p25-41.2026 **Submitted:** August 27, 2025 **Accepted:** June 12, 2026 **Published:** June 30, 2026

ABSTRACT

In a flexographic printing machine, color changeover actually takes up excessive time and serves as a bottleneck in the corrugated cardboard packaging manufacturer. This causes inefficiency and reduces the output. The objective of this study is to optimize production scheduling using a Genetic Algorithm (GA) to minimize sequence-dependent setup times (SDST). The current manual scheduling method used in the studied flexographic printing environment does not explicitly account for the sequence-dependent color changeover structure considered in this study, creating opportunities for improved scheduling performance through optimized production sequencing. To address this problem, the SDST problem was first formulated as a Mixed-Integer Linear Programming (MILP) model based on systematic observation and historical production data. A customized GA was then developed to generate high-quality scheduling solutions, and a systematic parameter tuning process was conducted, identifying an effective configuration of population size 500, 100 generations, and mutation rate 0.7 to ensure stable convergence. Results show that the proposed GA framework reduced setups times by over 70.78% (equivalent to 5.04 hours per shift) compared to the facility's existing manual scheduling baseline and also outperformed the greedy heuristic benchmark, consistently achieving an average setup time of 8,466.63 seconds across multiple runs with low variability, demonstrating reliable performance. This study shows that GA can serve as a practical approach for optimizing scheduling in flexographic printing and closely related sequence-dependent color changeover production contexts, although the current model is based on deterministic conditions with fixed job sequences, which may limit responsiveness to dynamic production uncertainties such as machine breakdowns or rush orders, suggesting the need for future enhancements using simulation and multi-objective optimization approaches.

Keywords: Production scheduling, sequence-dependent setup times, Genetic Algorithm, color changeover, flexographic printing

INTRODUCTION

Packaging production is an important component of global economic activity, as it supports the transformation of raw materials into products across many industry sectors [1]. As the market develops, packaging functionality not only for product protection from external factors such as dust, contaminants,air, shocks, and sunlight, but also becomes a marketing element that can influence purchasing decisions and build brand image [2], [3]. Among the available packaging options, corrugated cardboard boxes are widely used because of their light weight, strength and cost effectiveness [4], [5], [6], [7], [8]. The increasing demand of the e-commerce industry and international trade causes the demand for packaged products to grow. This situation puts significant pressure on the corrugated box manufacturer to respond to market needs efficiently. Improving production efficiency in corrugated box manufacturing is therefore an important operational objective [11], [12].

The packaging industry has witnessed a significant increase in the technology development that has led to improved production performance and efficiency [10], including the use of flexographic printing machines, which can deliver high quality graphics and branding in corrugated cardboard [11]. A high degree of flexibility in printing different designs and combinations of colors is a crucial factor for product customization and market differentiation, and flexographic printing machines are well suited to this. However, the color switching setup times can be quite time consuming.. This problem is known as the Color Print Scheduling Problem (CPSP), when cleanup and ink change durations are dependent to the sequence order of jobs [12]. The setup time in this case is sequence-dependent, as switching from a darker to a lighter color requires prolonged washing and cleaning of the printing plates and ink chambers to avoid contamination, leading to longer setup times than switching from a light to a dark color. Within flexographic printing operations, color changeovers occur during the printing stage and generate sequence-dependent setup times between jobs. As a result, the order in which print jobs are scheduled directly influences the total setup time incurred during production, making the printing stage a key focus for scheduling optimization in corrugated box manufacturing. Consideration of sequence-dependent setup times (SDST) in scheduling models has been widely recognized as an important concern for operations management research because SDST play a major role in high-setup-time environments and because they can lead to significant savings in total setup times and related production costs when explicitly incorporated into optimization frameworks [13].

Various methods, such as exact and heuristic methods, have been used in order to get optimal or near-optimal schedules [14]. Exact methods, like branch and bound or dynamic programming, usually work only for small problems because of the vast requirement of memory and the long computation time [14]. However, the CPSP with sequence-dependent setup times is a kind of NP-hard combinatorial optimization problem. This condition makes the applicability of exact methods to solve the large instances problem impractical.

Genetic algorithm-based approaches have been applied to a variety of scheduling problems involving setup-time considerations. In the specific context of sequence-dependent setup times, Naderi et al. [13] developed a GA for job-shop scheduling with SDST and preventive maintenance constraints. More broadly, Cheng et al. [25] reviewed scheduling research involving setup times in flowshop environments, providing important background on setup-time-oriented scheduling problems. Related GA applications have also been reported in flexible job-shop and multi-objective scheduling contexts [26], [27]. Within the flexographic printing domain, Hanif et al. [28] applied a swarm-based metaheuristic to job-sequence optimization for setup-cost minimization, providing the closest domain-specific precedent to the present study.

While Hanif et al. [28] applied the Puma Optimizer Algorithm (PUMA), a population-based metaheuristic, to job-sequence optimization in flexographic printing, their cost model imposes a uniform flat penalty for colour transitions of a given direction—a colour-change counting formulation in which the cost is identical regardless of the specific colour pair involved. In contrast, the present study formulates the problem as a sequence-dependent setup time (SDST) scheduling problem in which each colour-pair transition is assigned its own setup cost based on observed production data. This enables the scheduling model to explicitly represent transition-specific setup requirements encountered in flexographic corrugated board production.

To address this modelling challenge, this study develops an optimization framework for minimizing sequence-dependent colour changeover setup times in flexographic printing operations for corrugated box manufacturing. The specific task is to find an order of producing a batch of 20 box jobs that minimizes total setup time needed in this case. In order to reach this goal, firstly the scheduling problem is formulated as a sequence dependent setup time (SDST) optimization model and then solved by using a customized Genetic Algorithm (GA) approach. Then, a systematic full-factorial parameter calibration procedure is applied to find out an efficient GA configuration for the given problem. The study contributes an empirically grounded scheduling framework based on observed production data and demonstrates the potential for substantial setup-time reduction relative to existing scheduling practice.

METHODS

This study developed an optimization framework that combines a mathematical model formulation and a Genetic Algorithm (GA). This combination process aims to achieve a reduction of color changeover setup times.

Data Collection and Model Development

To establish a realistic and practical optimisation model on the problem, two types of data were collected in this study: primary and secondary data. The primary data were collected using direct observation from 3-20 September 2024 at a dedicated corrugated box plant serving multiple clients. These observations provided measurements of setup times within the data. A total of 45 different color changeover setup events were recorded in order to estimate the sequence-dependent setup time parameters. For each color-pair transition, the observed setup duration was assigned directly as the corresponding sequence-dependent setup-time parameter. The data collection process covered all color-pair transitions required for the setup-time matrix; therefore, no aggregation of repeated observations and no estimation of unobserved transitions were required. The observation was performed on a single flexographic machine equipped with four printers (ink channels). The secondary data consisted of historical production records from the observed flexographic printing machine covering the period July–September 2024. These records were used to characterise production operations, color-changeover practices, and the existing manual scheduling procedure. A representative production batch containing 20 box orders was selected from the historical records to construct the scheduling instance analysed in this study and to provide the manual baseline schedule for comparison with the GA solution.

After the data collection, a sequence-dependent color setup times formulation of the Production Scheduling Problem was developed. As sequence-dependent setup time scheduling is a permutation-based combinatorial optimization problem, a Genetic Algorithm (GA) was employed to search for high-quality scheduling solutions efficiently [29]. The chromosome representation directly encodes production sequences, making GA operators naturally suited to the problem structure. Tournament selection was used to balance solution quality and population diversity during parent selection. The choice of Order Crossover (OX1) is due to the fact that it does not change the relative job ordering, while still preserving other feasible permutations. Swap mutation introduces local changes in the sequence that do not violate the feasibility of permutations. The parameter configuration was selected by a systematic full factorial tuning test that investigated several independent runs of the system using different values for the population size, number of generations and mutation rate.

The GA and the mathematical model were built in Python 3.9 on Google Colab, using a machine with an Intel Core i3-8130U CPU (2.20 GHz). The custom core of the GA (custom operators and components, as well as encoding and decoding) was built in Python. NumPy (version 1.21.6) and Pandas (version 1.3.5) were used in this study for carrying out numerical and data operations. The GA was set to terminate after 100 generations were completed.

Mathematical Model Formulation

The mathematical model is formulated as a Mixed-Integer Linear Programming (MILP) problem for scheduling 20 products on a single flexographic machine equipped with four printers (ink channels). Each product requires simultaneous allocation of all four printers, and setup times depend on the color changeover between consecutive products. The objective is to minimize the total setup time, with each product processed exactly once.

Indices

P : Set of products (boxes), with $|P| = n = 20$.
R : Set of printers, $R = \{1, 2, 3, 4\}$
C : Set of possible colors, $C = \{0, 1, \ldots, 9\}$
T : Set of discrete time periods, $T = \{1, 2, \ldots, T_{max}\}$. The time index T primarily represents the sequence position and all product processing times are equal.

Parameters

$c_{i,r}$: Color required for product $i \in P$ on printer $r \in R$, $c_{i,r} \in C$
$S_{a,b}$: Setup time for switching from color $a \in C$ to color $b \in C$
$cost_{i,j}$: Transition cost, $cost_{i,j} = \sum_{r \in R} S_{c_{i,r} c_{j,r}}$.

Variables

$x_{i,j,t}$: A binary variable $\{0,1\}$ $\forall i, j \in P, i \neq j, t \in T$; $x_{i,j,t} = 1$ if product i is processed immediately before product j at time t, 0 otherwise.
$y_{i,t}$: A binary variable $\{0,1\}$ $\forall i \in P, t \in T$; $y_{i,t} = 1$ if product i starts processing at time t, 0 otherwise.
$z_{i,k,t}$: A binary variable $\{0,1\}$ $\forall i \in P, k \in R, t \in T$; $z_{i,k,t} = 1$ if product i is processed on printer k at time t, 0 otherwise.

The MILP formulation is presented primarily as a rigorous formal definition of the Sequence-Dependent Setup Time (SDST) problem specific to flexographic printing. It was used to specify the objective function and constraints that led to designs of the GA representation and fitness evaluation. To test whether the GA is consistent with our SDST formulation, a reduced instance of the problem with 5 products was tested. In this case, all the possible sequences (5!= 120 permutations) were then fully enumerated and evaluated using the same SDST objective function defined by the MILP formulation. The optimal sequence obtained through exhaustive enumeration matched the best solution identified by the GA, confirming consistency between the GA implementation and the SDST formulation. For the full 20-product production instance, the GA was employed as the primary optimization method due to the combinatorial complexity of the SDST scheduling problem.

Objective Function

Minimize the total setup time:

$$\text{Min} \sum_{i \in P} \sum_{j \in P, j \neq i} \sum_{t \in T} cost_{i,j} \cdot x_{i,j,t} \tag{1}$$

subject to:

$$\sum_{t \in T} y_{i,t} = 1 \quad \forall i \in P \tag{2}$$

$$\sum_{i \in P} y_{i,t} \leq 1 \quad \forall t \in T \tag{3}$$

$$z_{i,k,t} = y_{i,t} \quad \forall i \in P, k \in R, t \in T \tag{4}$$

$$\sum_{i \in P} z_{i,k,t} \leq 1 \quad \forall k \in R, t \in T \tag{5}$$

$$\sum_{j \in P, j \neq i} x_{i,j,t} \leq y_{i,t-1} \quad \forall i \in P, t = 2, \ldots, T_{max} \tag{6}$$

DOI:10.25077/josi.v25.n1.p25-41.2026

$$\sum_{j \in P, j \neq i} x_{i,j,t} = y_{i,t} \quad \forall i \in P, t = 2, \ldots, T_{max} \tag{7}$$

$$x_{i,j,t}, y_{i,t}, z_{i,k,t} \in \{0,1\} \tag{8}$$

$$cost_{i,j} \geq 0 \tag{9}$$

Equation (1) represents the objective function that minimizes the total setup time across all product transitions. The summation iterates over all products i and j (where $i \neq j$) and all time periods t. The color changeover time of all product transition from one product to another across all four printers is represented as $cost_{i,j}$. The binary variable $x_{i,j,t}$ equals 1 when product i is processed immediately before product j starting at time t, and 0 otherwise. Equation (2) guarantees that each product i will be scheduled for the production exactly only once within the whole schedule horizon. The summation over all time periods t of the binary variable $y_{i,t}$ must equal 1, which indicates that for each product, there is exactly one time period when it starts processing. This constraint is fundamental to the scheduling problem as it prevents products from being processed multiple times or not being processed at all. This constraint is an essential feature of the scheduling problem, since we cannot process a product more than one time or less than one time. Equation (3) ensures that at any given time period t, at most one product is being processed on the machine. This follows from the single machine nature of the problem, that is the machine can process one product at a time. Equation (4) serves as a constraint that links the product scheduling variable $y_{i,t}$ with the printer allocation variable $z_{i,k,t}$. Specifically, it states that if product i starts processing at time t ($y_{i,t} = 1$), then it must be processed on all printers k at that time ($z_{i,k,t} = 1$). Conversely, if product i is not being processed at time t ($y_{i,t} = 0$), then it cannot be processed on any printer k at that time ($z_{i,k,t} = 0$). This enforces the critical operational requirement that all four printers are simultaneously allocated to a single product during its processing, reflecting the integrated nature of flexographic printing where a product uses all available ink channels at once.

Equation (5) ensures that each printer k takes up at most one order in every time period t. This avoids the potential problem of multiple products attempting to share a printer at one time. Equation (6) establishes temporal precedence. If product i is processed immediately before product j starting at time t (i.e., $x_{i,j,t} = 1$), then product i must have started processing at the previous time period $t - 1$ (i.e., $y_{i,t-1} = 1$). This will make sure that predecessor–successor relationships are established between products only if they are consecutive in the schedule. Every product scheduled from the second sequence position onwards only has one predecessor that is scheduled, as guaranteed by equation (7). The first scheduled product ($t = 1$) is not required to have a predecessor arc and is selected at random from the set of products. This explicitly states where the production sequence is going to start and eliminates the need for a dummy predecessor node. Equations (6) and (7) ensure that the production sequence is continuous and that each product is correctly connected to the predecessor and successor of it. Equation (8) defines the decision variables $x_{i,j,t}$, $y_{i,t}$, and $z_{i,k,t}$ as binary, meaning they can only take values of 0 or 1. This is standard for MILP formulations where these variables represent decisions (e.g., whether a product is processed at a certain time or in a certain sequence). Equation (9) ensures that the transition cost $cost_{i,j}$ is non-negative, which is a logical requirement for setup times.

Genetic Algorithm (GA) Implementation

Genetic Algorithm (GA) is widely used to solve scheduling problems, due to their ability to find solutions in large search spaces. Compared to the traditional exact methods, which often become computationally expensive when solving NP-hard scheduling problems, GA has the capability to produce high-quality solutions close to optimal under an acceptable amount of computational time. GA's flexibility enable them to solve various objective functions and constraints, both non-linear, non-differentiable, and multi-objective. The flexibility of this approach has been illustrated in different areas, for example, routing problems [30], [31], flow-shop scheduling [32], and project scheduling [33]. GAs use selection, crossover, and mutation mechanism, which can avoid falling into a local extremum and improve the solution space. We used GA in this work to find a permutation of items such that the total setup time spent on color change in four printers is minimized. The GA implementation had the following stages.

Population Initialization

The GA starts by generating an initial set of candidate solutions, which are called individuals. Every single one of the individuals can be considered as unique permutation of 20 products, or a possible production sequence. These permutations are generated randomly. The number of individuals was defined by a careful parameter tuning in order to trade off the exploration of the solution space with computational efficiency.

Fitness Evaluation

After the population generation, fitness value is being measured for every individual of the population. Fitness value is defined as total setup time needed to process the 20 products in the order specified by the sequence or permutation. The setup time for each printer takes into account the color changeover from the previous product color to the new product color. The Fitness score of an individual is the sum of all such transitions, with lower score indicating better sequence. Such an evaluation mechanism then is performed for each individual in every generation to direct the evolution search.

Selection

After calculating the fitness score for every individual, the tournament selection mechanism was used to produce the next generation of individuals. Tournament selection provides a balanced selection process while helping to maintain population diversity. In each tournament, a subset of five individuals (tournament size = 5) was randomly sampled from the population, and the individual with the lowest setup time was selected as a parent. This process was repeated to select the two parents required for crossover.

To preserve high-quality traits across generations, elitism was incorporated by carrying forward the top two individuals unchanged into the next population. The remaining individuals were generated through tournament selection, crossover, and mutation. This hybrid strategy ensures that superior solutions are retained while still exploring a wide range of promising sequences.

The final GA configuration identified through parameter tuning consisted of a population size of 500, 100 generations, and a mutation rate of 0.7. Order Crossover (OX1) was applied with a crossover probability of 1.0, meaning that crossover was performed for every selected parent pair. These settings were used in the final scheduling experiments.

Crossover

During this crossover stage, the combination of genetic characteristics from the parent was used to produce new offspring. Order Crossover (OX1) utilized in this research since this mechanism is fit for permutation based problems such as production sequencing. In this process, two crossover point was selected randomly and the subsequence between those points is directly inserted at the corresponding positions. The remaining positions are replaced by the products from parent two that are in their original order and without repetition. This approach keeps the relative ordering of the products and ensures the resulting sequence is a valid permutation. This mechanism helps the algorithm to inherit beneficial subsequences, such as sequences that minimize color changeover setup time, without producing duplicate sequences.

Mutation

To add variety and to avoid premature convergence to the local minima, each offspring is selected based on probability for mutation mechanism. The mutation mechanism used is the swap mutation, in which the two random positions in the sequence are chosen and their values swapped. This mutation preserves the possibility of permutation while performing minor modifications on the sequence. The mutation enables the algorithm get out of local minima and potentially find new product groupings that reduce color changeover setup time.

Population Replacement

The next generation is built by combining the individuals retained from the previous iteration with the new offspring. The parents are selected repeatedly through tournament selection and each parent produces two offspring using OX1 operator and then followed by a mutation mechanism. This offspring will fill the population until the desired size is achieved. This strategy allows each new population to consist of newly generated solutions but the best performing individual is still being kept.

Iteration and Termination

The implementation of GA iteratively performs the step of fitness evaluation, selection, crossover, mutation, and population replacement. In each iteration, the best-performing population which is the lowest total setup time is being recorded. The algorithm continues its iteration until a fixed number of generations. After termination, the algorithm produces the best sequence identified throughout the entire process, along with its corresponding total setup time. This final solution represents the optimized production order that minimizes color changeovers across the four printers.

Parameter Optimization

GA parameter configurations affect the GA performance. The contribution of each parameter to searching procedures is different and interaction between the parameters could lead to nonlinear effects on the convergence speed and on the quality of the solutions. Therefore, the selection of the parameters in the system needs to be systematically evaluated and considered for the decision of the parameter configuration. This enables the impact of the different parameters of the algorithm on the algorithm's performance to be tested thoroughly, rather than just picking values that might seem reasonable. A full factorial experimental design was used in this research to assess the parameter configuration. This helps to cover all the combinations of the chosen parameter levels and allows for a comprehensive analysis of the solution set. We proposed the following candidate values for the three major parameters:

- Population Size: Five levels were tested {100, 200, 300, 400, 500}. Larger populations promote genetic diversity but may also increase computational cost; testing a range helps us to find the balance point.
- Number of Generations: Two levels were investigated {50, 100}. This parameter determines the iteration process.
- Mutation Rate: Nine levels were tested, ranging from 0.1 to 0.9 with an increment of 0.1. This wide range was chosen to analyze the trade-off between exploitation (low mutation) and exploration (high mutation).

We examined 90 parameter configurations ($5 \times 2 \times 9$). Because Genetic Algorithm is stochastic, each configuration was run five times independently. The performance in the configuration was thus measured as the average setup time of execution over the five runs. We chose as final GA setting the parameter configuration yielding the lowest mean setup time.

A population of 500 was chosen for 100 generations with a mutation strategy of 0.7 in the selection phase. In order to evaluate the stability of this configuration, 30 independent validation runs were performed. All the other GA settings were a tournament size of 5 and crossover probability of 1.0.

RESULT AND DISCUSSION

This section presents the results from the optimization study, including the GA performance evaluation and a comparison between GA optimized schedules with the manual scheduling currently used in the manufacturer. The results show that the proposed GA-based optimization framework has potential to improve production efficiency in the flexographic printing stage.

Flexographic Machine Characteristics and Initial Observations

During the manufacturing process of corrugated boxes, large sheets of corrugated material are fed into a flexographic printing machine with four independent ink channels to handle customer-specific designs. Color changeover takes lengthy setup times due to each ink channel needing to be cleaned first to avoid ink contamination, especially when transitioning from darker color to lighter color. This order-dependent setup-time creates scheduling complexity and is the focus of optimization in this work.

To obtain the empirical data on actual setup times, initial observation was conducted. Setup time for different color transitions was classified into different categories according to the lightness-darkness value of the colors as described in Rose [34]. This method provides a systematic method to estimate the level of difficulty of a color transformation by breaking the colors into nine levels ranging from brightest (White = Level 1) to darkest (Black = Level 9). Table 1 gives the color matrix in which the elements represent the measured setup times (in seconds) required to change between the different color codes in the same type. There were a number of transitions deemed impractical within the production process: from level 9 (Black) to level 1 (White) and level 2 (Yellow); and from level 1 (White) to level 8 (Purple) and level 9 (Black). This is because of the difficulty of cleaning and the amount of time that it requires, to prevent ink contaminating the system. The optimization model imposed a high setup time penalty on these transitions (5000 seconds). This is a very high penalty that means that if the GA sequence contains such a transition, it will get a bad fitness score. This effectively causes the algorithm to discard these solutions in favor of feasible sequences during the evolutionary process. Detailed color codes and their descriptions are given in Table 2.

From the evaluation of actual production sequence data of 20 box orders, the cumulative setup time under the current manual scheduling methodology was found to be 27,369 seconds, which is 7.6 hours. This lengthy setup time shows that a bottleneck in switching colors is inherent in the current production scheduling process. It shows the

Table 1. Color Transition Setup Times (in seconds)

		To									
		0	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**
From	0	0	0	0	0	0	0	0	0	0	0
	1	0	0	494	575	533	496	785	541		
	2	0	789	0	552	530	900	727	558	559	548
	3	0	871	716	0	552	586	459	469	553	492
	4	0	854	705	548	0	604	494	514	535	588
	5	0	917	875	628	634	0	571	582	558	550
	6	0	1180	883	647	921	719	0	573	619	820
	7	0	1077	996	756	618	602	615	0	535	605
	8	0	1197	1134	744	638	625	876	630	0	467
	9	0			824	793	649	783	605	877	0

Table 2. Color Codes

Code	Description	Code	Description
0	None (No color)	5	Reddish orange and bluish purple
1	White	6	Red and blue
2	Yellow	7	Reddish purple and bluish purple
3	Yellowish orange and bluish green	8	Purple
4	Orange and green	9	Black

need for improvement using a systematic and optimization strategy, instead of simple intuition-based or first-come-first-served scheduling strategies.

Genetic Algorithm Implementation and Parameter Optimization

The GA was used to optimize the production schedule with the goal of minimizing the total setup time. The GA began by encoding potential solutions (production sequences) as chromosomes. After the population was created, the fitness scores for each individual was calculated using the setup time data from color transition matrix. These initial solutions were then iteratively improved by the GA through genetic operations, such as selection, crossover, and mutation mechanism. This iterative process of genetic algorithm is shown in Figure 1

As parameters heavily affect how well the genetic algorithm performs, hence a total of 90 distinct parameter combinations were systematically tested, varying the key control parameters of the GA to identify the optimal configuration for this specific problem. The parameters tested were:

- Population Size: 100, 200, 300, 400, and 500
- Number of Generations: 50 and 100
- Mutation Rate: Ranging from 0.1 to 0.9 (in increments of 0.1)

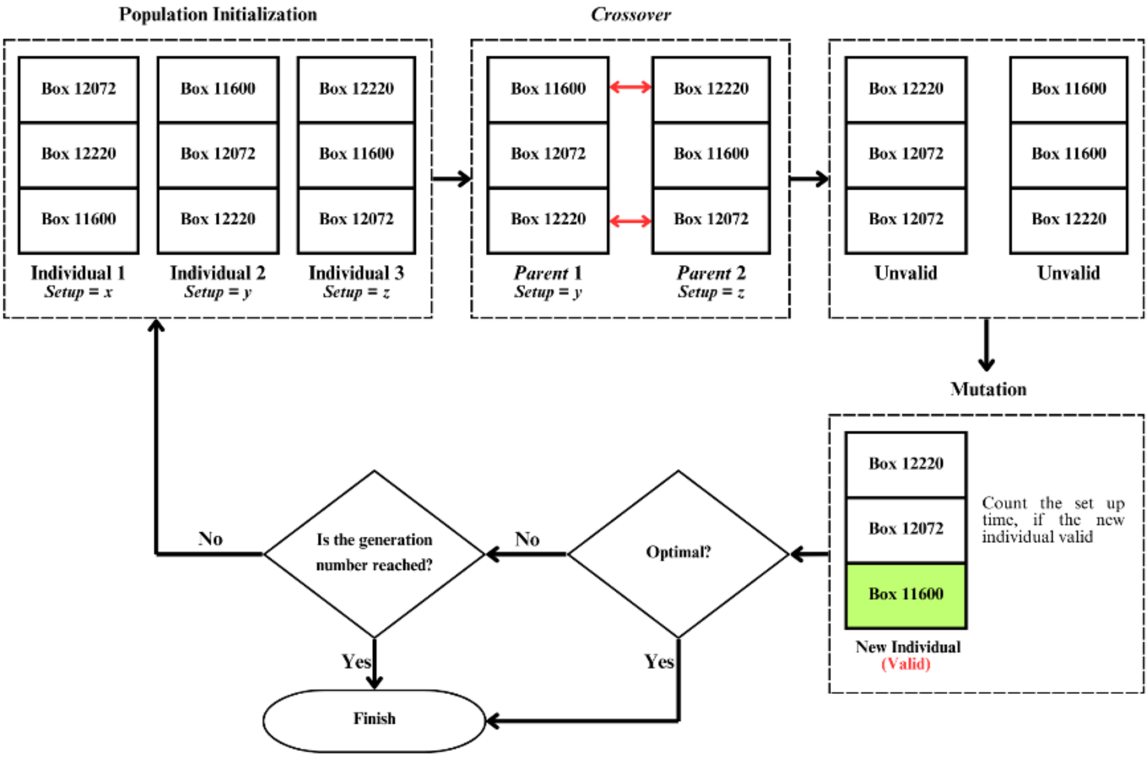

Figure 1. GA Iterative Process

Table 3. Top 10 Parameter Configurations (Ranked by Average Setup Time)

Population	Generations	Mutation Rate	Repetition	Average Setup Time (s)
500	100	0.7	5	8473.4
400	100	0.9	5	8502.2
500	100	0.8	5	8516.4
500	100	0.9	5	8577
500	100	0.3	5	8595.2
500	50	0.8	5	8613.6
500	50	0.2	5	8617
300	50	0.4	5	8621.2
500	100	0.4	5	8657
300	100	0.8	5	8684.2

The optimal parameter for the Genetic Algorithm was determined using a full-factorial experiment. A total of 90 distinct parameter combinations were tested, by changing the Population Size (100–500), Number of Generations (50, 100), and Mutation Rate (0.1–0.9). To mitigate selecting best parameters based on a single outlier due to stochastic characteristics of the algorithm, 5 independent runs were executed for every combination of the parameter. Then the average setup time of these 5 repetitions were recorded. The average setup time across these five repetitions was recorded and top 10 parameter configurations is summarized in Table 3.

As can be seen from table 3, the optimal parameter was identified as a population size of 500, 100 generations, and a mutation rate of 0.7. Figure 2 illustrates the three-dimensional relationship between population size, mutation rate, and the resulting average setup time, visually highlighting that lower setup times (represented by the lighter data points) are densely clustered around higher mutation rates (0.6–0.8) and larger population sizes.

Convergence and Performance Evaluation

To check the robustness of the identified winner (Pop=500, Gen=100, Mut=0.7) rigorously, we conducted a phase of confirmation based on 30 independent runs. Table 4 shows the results of the independent runs. The best setup time

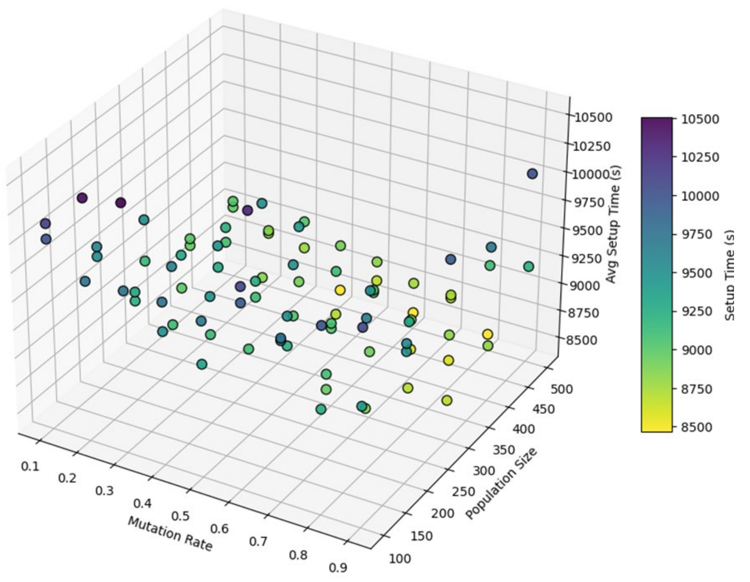

Figure 2. 3D Graph of Parameter Test

Table 4. Summary of Validation Metrics (30 runs)

Metrics	Times
Best Time	7,996 seconds
Average Time:	8,466.63 seconds
Standard Deviation	398.36 seconds

in the 30 runs was 7,996 seconds. The average, with a standard deviation of 398.36 seconds, was 8,466.63 seconds. This shows that, although in some cases it is possible to find the global optimum, the algorithms perform well in a predictable range, even in unfavorable scenarios.

Figure 3 shows a progressive reduction in setup time, with the best-found solution of 7,996 seconds being reached at approximately Generation 40 and remaining stable thereafter. We found that a larger population size yielded better overall performance. This observation is consistent with Naderi et al. [13], who reported similar population-size effects in sequence-dependent job-shop scheduling.

The parameter-tuning results also indicated that a mutation rate of 0.7 produced the lowest average setup time among the configurations tested. A plausible explanation is that the asymmetric setup costs and penalty structure of the SDST problem may benefit from stronger diversification than is commonly reported in generic permutation-scheduling problems. Under this interpretation, a higher mutation rate may help the search process explore a wider range of feasible production sequences. Further investigation would be required to determine the precise mechanisms underlying this behaviour

Comparison of Actual and Optimized Scheduling

The performance of the proposed GA was compared against the existing manual production schedule, a random scheduling baseline, and a greedy nearest-neighbour heuristic. The historical production sequence served as the

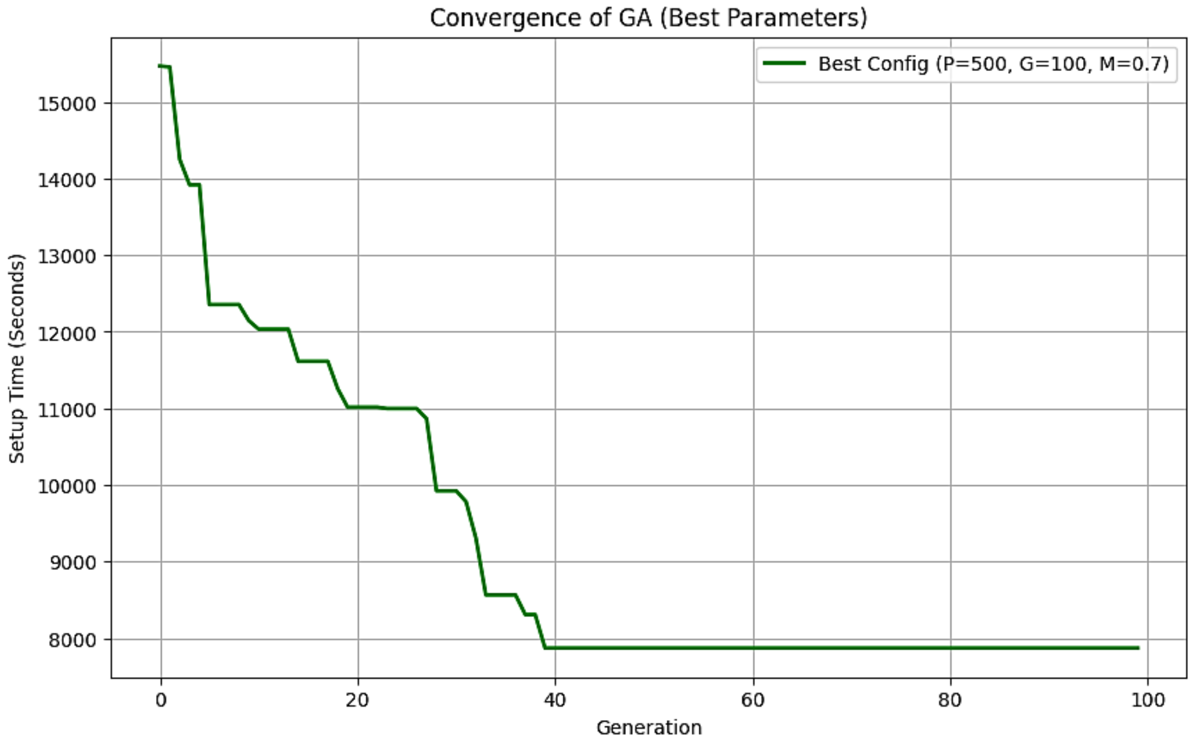

Figure 3. Generation Run Results

Table 5. Comparison of Setup Time (Benchmark vs GA-Optimized Scheduling)

Scheduling Method	**Total Setup Time (seconds)**	**Total Setup Time (hours)**	**Improvement over Manual**
Manual Scheduling	27,369	7.60	-
Random Scheduling (Average of 1000 runs)	25,290	7.02	7.60%
Greedy Heuristic	11,854	3.29	56.69%
GA-Optimized (Best)	**7,996**	**2.22**	**70.78%**

manual scheduling baseline and, for this production instance, is equivalent to a FIFO schedule because jobs were processed in their order of appearance in the production record. For the random scheduling baseline, the production sequence was randomly shuffled 1,000 times and the average setup time across all generated schedules was reported. For the greedy nearest-neighbour heuristic, each job was considered as a possible starting point. At each step, the unscheduled job with the lowest setup time from the current job was selected until a complete schedule was obtained. The heuristic was repeated for all possible starting jobs, and the best-performing schedule was reported. Both benchmark methods used the same sequence-dependent setup-time matrix and the same prohibitive penalty structure (5,000 seconds for impractical color transitions) applied in the GA evaluation.

Table 5 shows the comparative results. The intuitive, operator-based, manual scheduling method produced a total setup time of 27,369 seconds (7.60 hours). The random scheduling method produced a total setup time of 25,290 seconds. By contrast, the Greedy Heuristic produced a total setup time of 11,854 seconds, which indicates that even a simple, less sophisticated, rule-based method can exploit a significant amount of structure in the color-transition cost matrix, and can do so more effectively than the manual scheduling method. The proposed GA achieved a best-found setup time of 7,996 seconds and an average setup time of 8,466.63 seconds (SD = 398.36 seconds) across 30 independent runs. Because the GA is a stochastic optimization method, the average result provides a more representative measure of expected performance. On an average-case basis, the GA outperformed the Greedy Heuristic benchmark (11,854 seconds) by approximately 28.6% and reduced setup time by approximately 69.1% relative to the FIFO/manual production schedule (27,369 seconds). The best-found solution of 7,996 seconds is reported separately as an indicator of the algorithm's peak performance.

The reduction from 7.60 hours to 2.22 hours effectively saves 5.38 hours per shift. This saving of time is equal to about 67 percent of a typical operator shift directly freeing up the capacity to do real printing instead of cleaning. This indicates that there is an immediate and significant change in the labor utilization and the general machine throughput. The Gantt Chart of Production Scheduling using GA (Figure 4) illustrates the best sequence of production orders graphically to clearly show how the GA is effective at reducing the amount of idle time between orders by intelligently sequencing the color changesovers. The streamlined schedule will have a more flowing and continuous flow production and directly solve the huge bottlenecks that have been discovered in the existing system. Gantt chart clearly illustrates a less bulky schedule that has a much less amount of non-productive time; this is further translated to high throughput and production capacity.

Research Implications

The parameter tuning results indicate that a population size of 500 produced the best overall performance among the tested levels. A similar observation was reported by Naderi et al. [13], who found that larger population sizes outperformed smaller ones in a sequence-dependent setup time job-shop scheduling problem. This consistency provides additional support for the suitability of the selected parameter setting in scheduling environments involving sequence-dependent setup times.

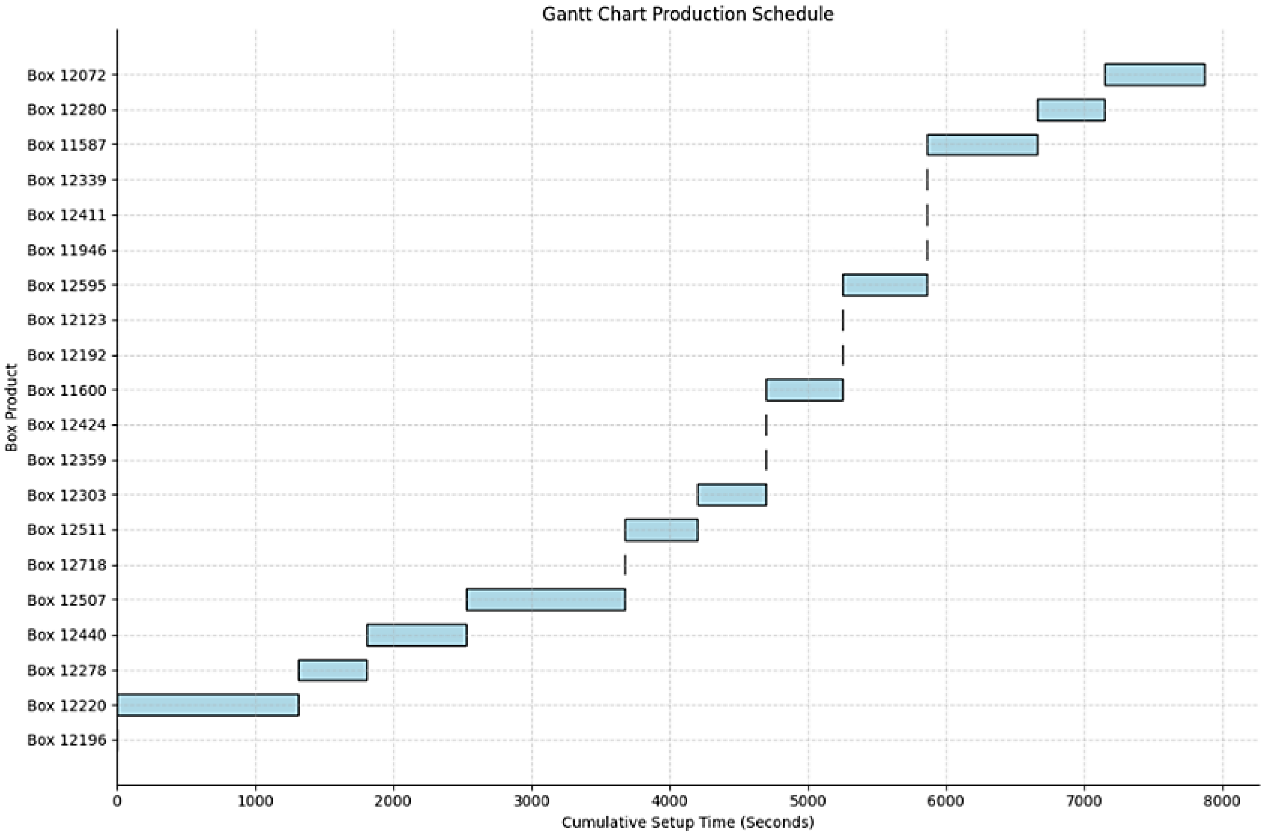

Figure 4. Gantt Chart of Production Scheduling using GA

The benefit of the application in this optimization model is a decrease in the set-up time of the color changeovers. It will save the set up time and the company can increase its productive time without having to incur extra cost to acquire machine or overtime costs. This will result in improved capacity to produce. Effective scheduling may also reduce the amount of ink and substrate consumed during changeovers, since shorter setup sequences involve fewer cleaning cycles; however, this study did not directly measure material consumption or waste, so this remains a plausible downstream benefit to be confirmed in future work.

By implementing an optimized production schedule, it has the potential to create a more stable working environment. The optimized scheduling can minimize the pressure that is regularly imposed on the operators as there will be less rushed changeovers and a more organized timetable. Such a simplified system is likely to have a positive impact on a safer working atmosphere and the number of human errors in the setups because operators will be able to move in a logical sequence of color transitions instead of responding to the non-systematic scheduling.

Although this study exhibits high efficiency gains, it has a number of limitations that characterize the scope of this study. To begin with, the model was tested on a single flexographic printing machine; to extend the framework to a parallel-machine setup would involve more constraints to deal with the workload balancing. Second, the experiment used a fixed set of job 20. In a dynamic real-world environment, orders would come in stochastically, and this would require a rolling-horizon strategy or dynamic rescheduling. Third, the model concerns only with the reduction of setup times, having the hypothesis that processing times are either constant or negligible compared to the setup bottlenecks. Lastly, the Genetic Algorithm is very strict on the input data provided; it does not factor in at the moment outside uncertainties like machine failure, shortage of raw materials, or even the level of skill of the operator, which may affect the performance of the theoretical schedule.

CONCLUSION

This study designed a GA-based optimization framework for minimizing sequence-dependent setup time in flexographic printing, achieving substantial setup-time reduction over the manual scheduling baseline under the tested conditions. The results demonstrate that color-sequence optimization can substantially reduce setup time in flexographic corrugated box production. Such reductions may contribute to improved operational efficiency and increased availability of productive machine time, although these downstream effects were not directly measured in the present study. This study demonstrates the effectiveness of combining a formal sequence-dependent setup time (SDST) scheduling formulation with a systematically calibrated Genetic Algorithm to reduce color-changeover setup times in flexographic corrugated box production. The proposed framework achieved notable setup-time reductions relative to the existing production schedule and benchmark heuristics. The systematic parameter tuning, supported by multi-run averaging, improved solution quality and produced consistent performance across repeated runs. Using the selected parameter configuration, the algorithm converged to high-quality solutions across validation runs, indicating stable run-to-run performance. The proposed Genetic Algorithm outperformed evaluated benchmark methods in this study, achieving lower setup times compared with both heuristic approaches and the existing production schedule. The best-found solution also indicates the capability of the algorithm to identify improved scheduling alternatives under the defined conditions. However, it is important to contextualize these findings. Although the framework shows clear potential improvement compared with manual scheduling conditions, it is constrained by deterministic input data and fixed job orders, which reduces its short-term flexibility in dynamic environments such as machine failures or variable job arrivals. As a result, the dynamics-based performance needs further investigation. Future research should prioritize enhancing the framework's robustness through simulation-based approaches to represent uncertainty such as rush orders and machine downtime, or expanding the model to achieve trade-offs between setup minimization and other operational considerations, since complex industrial trade-offs are better captured in multi-objective models.

ACKNOWLEDGEMENT

The authors would like to express their sincere gratitude to the Editor and reviewers for their valuable comments and constructive suggestions, which helped improve the quality and clarity of this manuscript.

CONFLICT OF INTEREST

The authors declare that there is no conflict of interest regarding the publication of this article.

FUNDING

The authors received no financial support for the research, authorship, and/or publication of this article.

DATA AVAILABILITY STATEMENT

The data that support the findings of this study are available from the corresponding author upon reasonable request.

References

[1] Y. Sun, H. Yu, and H. Wang, “A systems perspective on the approach to industrial chain and supply chain ecology,” Syst Res Behav Sci, 2024.

[2] N. C. Saha, “Food Packaging: Concepts and Its Significance,” 2022. doi: 10.1007/978-981-16-4233-3_1.

[3] RAJKUMAR and V. JAIN, “A Literature Study on the Product Packaging Influences on the Customers Behavior,” Journal of Contemporary Issues in Business and Government| Vol, vol. 27, no. 3, p. 780, 2021.

[4] D. G. Ertaş and B. T. Dönmezer, “The Effect of Industrial Design on Corrugated Cardboard Packaging Optimization,” Archives of Design Research, vol. 38, no. 2, pp. 45–69, 2025.

[5] R. Fitas, H. J. Schaffrath, and S. Schabel, “A review of optimization for corrugated boards,” Sustainability, vol. 15, no. 21, p. 15588, 2023.

[6] T. Garbowski and M. Pośpiech, “The impact of scientific research on corrugated board on the lifecycle of packaging,” Packaging Review, 2024.

[7] T. Gajewski, T. Garbowski, N. Staszak, and M. Kuca, “Crushing of double-walled corrugated board and its influence on the load capacity of various boxes,” Energies (Basel), vol. 14, no. 14, p. 4321, 2021.

[8] T. Garbowski, T. Gajewski, and J. K. Grabski, “Estimation of the compressive strength of corrugated cardboard boxes with various perforations,” Energies (Basel), vol. 14, no. 4, p. 1095, 2021.

[9] K. R. Patel, “Harmonizing sustainability, functionality, and cost: navigating responsible packaging innovations in modern supply chains,” American Journal of Economic and Management Business (AJEMB), vol. 2, no. 8, pp. 287–300, 2023.

[10] T. Fadiji and P. B. Pathare, “Technological advancements in food processing and packaging,” 2023, MDPI.

[11] Z. Abusaq et al., “Improving energy performance in flexographic printing process through lean and AI techniques: a case study,” Energies (Basel), vol. 16, no. 4, p. 1972, 2023.

[12] A. P. Burger, C. G. Jacobs, J. H. van Vuuren, and S. E. Visagie, “Scheduling multi-colour print jobs with sequence-dependent setup times,” Journal of Scheduling, vol. 18, no. 2, 2015, doi: 10.1007/s10951-014-0400-2.

[13] A. Naderi, M. Zandieh, and S. M. T. Fatemi Ghomi, “Scheduling sequence-dependent setup time job shops with preventive maintenance,” International Journal of Advanced Manufacturing Technology, vol. 43, no. 1–2, 2009, doi: 10.1007/s00170-008-1693-0.

[14] Adhi, B. Santosa, and N. Siswanto, “A meta-heuristic method for solving scheduling problem: Crow search algorithm,” in IOP Conference Series: Materials Science and Engineering, 2018. doi: 10.1088/1757-899X/337/1/012003.

[15] Agnetis, J.-C. Billaut, M. Pinedo, and D. Shabtay, “Fifty years of research in scheduling–theory and applications,” Eur J Oper Res, 2025.

[16] A. Bertsimas and J. Tsitsiklis, “Simulated annealing,” Statistical Science, vol. 8, no. 1, 1993, doi: 10.1214/ss/1177011077.

[17] F. Glover, “Tabu Search: A Tutorial,” Interfaces (Providence), vol. 20, no. 4, 1990, doi: 10.1287/inte.20.4.74.

[18] M. Dorigo and C. Blum, “Ant colony optimization theory: A survey,” Theor Comput Sci, vol. 344, no. 2–3, 2005, doi: 10.1016/j.tcs.2005.05.020.

[19] R. Benni, S. R. Umarani, and S. Totad, “A Comprehensive Study of Meta-Heuristic Algorithms for Job Shop Scheduling Optimization,” in 2024 15th International Conference on Computing Communication and Networking Technologies (ICCCNT), IEEE, 2024, pp. 1–10.

[20] C. Blum and A. Roli, “Metaheuristics in Combinatorial Optimization: Overview and Conceptual Comparison,” 2003. doi: 10.1145/937503.937505.

[21] E. Ziegel, “Genetic Algorithms and Engineering Optimization,” Technometrics, vol. 44, no. 1, 2002, doi: 10.1198/tech.2002.s675.

[22] A. E. Smith, “Multi-objective optimization using evolutionary algorithms [Book Review],” IEEE Transactions on Evolutionary Computation, vol. 6, no. 5, pp. 526–526, Apr. 2005, doi: 10.1109/tevc.2002.804322.

[23] M. Andrew, “Introduction to Evolutionary Computing,” 2004. doi: 10.1108/03684920410699216.

[24] Lei, “A genetic algorithm for flexible job shop scheduling with fuzzy processing time,” Int J Prod Res, vol. 48, no. 10, pp. 2995–3013, 2010.

[25] T. C. E. Cheng, J. N. D. Gupta, and G. Wang, “A review of flowshop scheduling research with setup times,” Prod Oper Manag, vol. 9, no. 3, pp. 262–282, 2000.

[26] Rooyani and F. Defersha, “A Two-Stage Multi-Objective Genetic Algorithm for a Flexible Job Shop Scheduling Problem with Lot Streaming,” Algorithms, vol. 15, no. 7, 2022, doi: 10.3390/a15070246.

[27] K. Li, L. Xu, H. Zhang, and J. Chen, “A genetic algorithm for fuzzy identical parallel machine scheduling of minimising total weighted tardiness under resource constraint,” Int J Prod Res, vol. 62, no. 21, 2024, doi: 10.1080/00207543.2024.2323065.

[28] H. Hanif, N. Daud, D. Kurnianingtyas, I. K. Hutomo, and I. Gunawan, “Cost-effective job sequence optimization for minimizing downtime in flexographic printing with puma optimizer algorithm,” in 2024 IEEE International Conference on Communication, Networks and Satellite (COMNETSAT), IEEE, 2024, pp. 15–20.

[29] Alhijawi and A. Awajan, “Genetic algorithms: theory, genetic operators, solutions, and applications,” 2024. doi: 10.1007/s12065-023-00822-6.

[30] P. K. Ariningsih, T. Iswari, K. D. Poetra, and Y. M. K. Aritonang, “Sequential Routing-Loading Algorithm for Optimizing One-Door Container Closed-Loop Logistics Operations,” Jurnal Optimasi Sistem Industri, vol. 19, no. 2, 2020, doi: 10.25077/josi.v19.n2.p122-132.2020.

[31] N. Zhafira, F. Afrinaldi, and Taufik, “An Application of Genetic Algorithm in Determining Salesmen’s Routes: A Case Study,” Jurnal Optimasi Sistem Industri, vol. 17, no. 1, 2018, doi: 10.25077/josi.v17.n1.p26-34.2018.

[32] M. Utama, L. R. Ardiansyah, and A. K. Garside, “Penjadwalan Flow shop Untuk Meminimasi Total Tardiness Menggunakan Algoritma Cross Entropy–Algoritma Genetika,” Jurnal Optimasi Sistem Industri, vol. 18, no. 2, 2019, doi: 10.25077/josi.v18.n2.p133-141.2019.

[33] S. S. Tjandra, F. Setiawan, and H. Salsabila, “Application of Genetic Algorithms to Solve MTSP Problems with Priority (Case Study at the Jakarta Street Lighting Service),” Jurnal Optimasi Sistem Industri, vol. 21, no. 2, 2022, doi: 10.25077/josi.v21.n2.p75-86.2022.

[34] A. A. Said, Dasar Desain Dwimatra, vol. 53, no. 9. 2012.

AUTHORS BIOGRAPHY

Irene Karijadi is a lecturer with a Ph.D. in Industrial Management, specializing in intelligent systems and data-driven decision-making. She earned her Master’s and Doctoral degrees from National Taiwan University, Taiwan, where she was actively involved in research at the Center for Internet of Things Innovation. Her work encompassed the development of smart campus applications, renewable energy forecasting platforms, and national healthcare data analytics. Irene has published her research in several Scopus Q1 indexed journals, including Renewable Energy and Energy and Buildings. Her research interests span machine learning applications, production scheduling optimization, and renewable energy.

Ivan Gunawan is a distinguished lecturer and industry consultant, specializing in supply chain management and optimization modeling. He earned his Bachelor's degree in Industrial Engineering from Widya Mandala Surabaya Catholic University. He then continued his advanced studies at Sepuluh Nopember Institute of Technology (ITS), where he obtained both his Master's and Doctoral degrees in the same field. Ivan's extensive research is reflected in his publications in various Scopus Q1 indexed journals, including the Journal of Food Engineering, Annals of Operations Research, and the Journal of Islamic Marketing, among others.

Stephanie Gracia Khangara is a highly accomplished Industrial Engineering graduate, earning her degree with Cum Laude honors. She's currently furthering her technical expertise by pursuing advanced studies in the Professional Engineer Program at Widya Mandala Surabaya Catholic University. Professionally, Stephanie works at PT Karya Wangsa Investama, a prominent company specializing in the production of corrugated boxes. With a strong academic foundation and practical experience, Stephanie is dedicated to making significant contributions in all aspects of her work.